



ПРОТОКОЛ ВЗАИМОДЕЙСТВИЯ ТЕРМИНАЛЬНОГО ПО И ПРОЦЕССИНГА вер. 4.129

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ
вер. 4.19

МОСКВА
8-495-783-5959

РОССИЯ
8-800-200-0059

ФАКС
8-495-926-4619

WEB
CORP.QIWI.COM

СОДЕРЖАНИЕ

1.	СПИСОК ИЗМЕНЕНИЙ	6
2.	ВВЕДЕНИЕ.....	11
2.1.	ГЛОССАРИЙ	11
3.	ОПИСАНИЕ XML ПРОТОКОЛА.....	13
3.1.	ФОРМАТ ПЕРЕДАЧИ ДАННЫХ	13
3.2.	АВТОРИЗАЦИЯ	13
3.3.	ОРГАНИЗАЦИЯ ПРОТОКОЛА	14
3.3.1.	Интерфейсы и действия.....	14
3.3.2.	Формат запроса и ответа.....	15
3.3.3.	Режимы обработки действий	19
4.	ПРИЕМ ПЛАТЕЖЕЙ	21
4.1.	СЦЕНАРИЙ ПРОВЕДЕНИЯ ПЛАТЕЖЕЙ ОФФЛАЙН	21
4.2.	СЦЕНАРИЙ ПРОВЕДЕНИЯ ПЛАТЕЖЕЙ ОНЛАЙН	22
4.3.	СТАТУСЫ ПЛАТЕЖЕЙ	23
4.4.	ПРЕДОСТАВЛЕНИЕ ДОПОЛНИТЕЛЬНЫХ ДАННЫХ О ПЛАТЕЛЬЩИКЕ	24
4.4.1.	Перечень сведений о плательщике, предоставляемых кредитной организацией	25
4.4.2.	Перечень сведений о плательщике, предоставляемых банковским платежным агентом... 27	
5.	КОНФИГУРАЦИЯ ТЕРМИНАЛОВ	30
5.1.	ПОСТРОЕНИЕ ИНТЕРФЕЙСА ДЛЯ ТЕРМИНАЛОВ	30
5.1.1.	Структура интерфейса.....	30
5.1.2.	Параметры отображения страниц интерфейса	31
5.2.	РЕАЛИЗАЦИЯ СЦЕНАРИЯ ПРОВЕДЕНИЯ ПЛАТЕЖА В ИНТЕРФЕЙСЕ ПРОВАЙДЕРА	33
5.3.	АЛГОРИТМ РАСЧЕТА ТЕРМИНАЛЬНОЙ КОМИССИИ.....	41
5.3.1.	Установка терминальной комиссии	41
5.3.2.	Расчет терминальной комиссии	41
6.	ИНТЕРФЕЙСЫ И ДЕЙСТВИЯ	43
6.1.	SYSTEM	43
6.1.1.	sendMail	43
6.1.2.	getReferencesVersions	44
6.1.3.	getCurrencyRates	45
6.1.4.	getNews	46
6.1.5.	getActions	47
6.1.6.	getResultCodes	48
6.1.7.	getForbiddenBanknotes	49
6.2.	TERMINALS	50
6.2.1.	checkMessages	51
6.2.2.	getMessages	52
6.2.3.	getPlaylist	55

6.2.4.	getLastIds	57
6.2.5.	getRemainingLimit.....	58
6.2.6.	setState	59
6.2.7.	resetReceiptNumber	62
6.2.8.	setFiscalMode	62
6.2.9.	addCashout	63
6.2.10.	setHosts	65
6.2.11.	getTerminalInfo	66
6.2.12.	getTerminalRegionHierarchy	67
6.2.13.	getTerminals.....	68
6.2.14.	getTerminalsWithCoordinates.....	71
6.2.15.	setConfigId	72
6.2.16.	getConfigId.....	73
6.2.17.	getConfig.....	74
6.2.18.	getCommissions	76
6.2.19.	getCommissionProfiles.....	77
6.2.20.	getCommissionsHistory.....	79
6.2.21.	getTerminalCommissionTemplates	80
6.2.22.	bindTerminalTemplate	82
6.2.23.	getForeignCommissions	83
6.2.24.	getTerminalTypes.....	84
6.2.25.	addShiftClosure.....	85
6.2.26.	requestLog	86
6.2.27.	requestProcessList.....	87
6.2.28.	requestZReport	88
6.2.29.	restartClientSoftware	89
6.2.30.	rebootTerminal	90
6.2.31.	disableTerminal.....	91
6.3.	AGENTS	91
6.3.1.	getAgentInfo.....	92
6.3.2.	getAgents	94
6.3.3.	getBalance.....	95
6.3.4.	getBalanceTree	96
6.3.5.	getOffer.....	97
6.3.6.	transferFunds.....	99
6.3.7.	setFiscalMode	100
6.3.8.	setOuterCommission.....	101
6.3.9.	getCommissionTemplates	102
6.3.10.	setCommissionProfile.....	104
6.3.11.	addCommissionProfile.....	106
6.3.12.	setCommissionTemplate	108
6.3.13.	addCommissionTemplate	109

6.4.	PROVIDERS	110
6.4.1.	authorizePayment	110
6.4.2.	authorizePaymentFromTemplate	112
6.4.3.	confirmPayment	114
6.4.4.	addOfflinePayment	115
6.4.5.	getPaymentStatus	117
6.4.6.	getRequiredPersonalData	119
6.4.7.	checkPaymentRequisites	120
6.4.8.	getProviderByPhone	122
6.4.9.	getProviders	123
6.4.10.	getGroups	125
6.4.11.	getUIGroups	126
6.4.12.	getUIProviders	128
6.4.13.	getPhoneRanges	142
6.4.14.	getRegions	143
6.4.15.	cancelPayment	144
6.4.16.	interruptPayment	146
6.5.	PERSONS	147
6.5.1.	getPersonInfo	147
6.5.2.	getPersons	148
6.5.3.	getRoles	150
6.5.4.	changePassword	151
6.5.5.	setPublicKey	152
6.6.	TEMPLATES	153
6.6.1.	createUser	153
6.6.2.	modifyUser	155
6.6.3.	getPaymentTemplates	157
6.6.4.	createPaymentTemplate	159
6.6.5.	blockPaymentTemplate	160
6.6.6.	getTemplateProviders	161
6.6.7.	getTemplateProviderInfo	162
6.7.	REPORTS	164
6.7.1.	getPayments	164
6.7.2.	getTerminalsShiftHistory	168
6.7.3.	getStatistics	170
6.7.4.	getCashouts	172
6.7.5.	getTerminalsStatus	173
6.7.6.	getTerminalsSettings	175
6.7.7.	getTerminalsStatisticalData	177
6.7.8.	getTerminalsCash	178
6.8.	ACCOUNTINGREPORTS	180
6.8.1.	getSubagentAccountStatement	181

6.8.2.	getAccountingTerminal	182
6.8.3.	getAgentTariffs	184
6.8.4.	getAgentTP.....	185
6.8.5.	getExemptionRates	187
ПРИЛОЖЕНИЕ А:	МАСКИ ПОЛЕЙ ВВОДА.....	189
ПРИЛОЖЕНИЕ Б:	РЕГУЛЯРНЫЕ ВЫРАЖЕНИЯ В ПОЛЯХ ВВОДА	191
ПРИЛОЖЕНИЕ В:	КОДЫ ОШИБОК	194
ПРИЛОЖЕНИЕ Г:	СТАТУСЫ ПЛАТЕЖЕЙ	201
ПРИЛОЖЕНИЕ Д:	СТАТУСЫ АСИНХРОННЫХ ЗАПРОСОВ.....	201
ПРИЛОЖЕНИЕ Е:	ТИПЫ ТРАНЗАКЦИЙ	202
ПРИЛОЖЕНИЕ Ж:	АВТОРИЗАЦИЯ ПО ЭЦП	204
	Алгоритм работы с использованием ЭЦП.....	204
ПРИЛОЖЕНИЕ З:	ТИПЫ ИНКАССАЦИЙ.....	205
ПРИЛОЖЕНИЕ И:	РАСШИФРОВКА БИТА СОСТОЯНИЯ СТОРОЖЕВОГО ТАЙМЕРА.....	206
ПРИЛОЖЕНИЕ К:	СЧЕТА УЧЕТА	207
ПРИЛОЖЕНИЕ Л:	РАСШИФРОВКА СТРОКИ ФЛАГОВ СОСТОЯНИЯ АСО	208
ПРИЛОЖЕНИЕ М:	ТИПЫ ТЕРМИНАЛОВ.....	210
ПРИЛОЖЕНИЕ Н:	ТИПЫ ОБЪЕКТОВ КЛАДР	211
ПРИЛОЖЕНИЕ О:	КОДЫ ОПЕРАТОРОВ СОТОВОЙ СВЯЗИ	215
ПРИЛОЖЕНИЕ П:	ТИПЫ ДЕНЕЖНЫХ СРЕДСТВ (MONEYTYPE).....	216
СПИСОК РИСУНКОВ		217
СПИСОК ТАБЛИЦ.....		217
СПИСОК ПРИМЕРОВ.....		217

1. СПИСОК ИЗМЕНЕНИЙ

Версия документа	Изменения
4.19	Глоссарий приведен в соответствие действующим Правилам работы Системы
4.18	Добавлены новые значения атрибута <code>disp_type</code>
4.17	Добавлен новый тип сообщения в запрос <code>getMessages</code>
4.16	Добавлен атрибут <code>lastPaymentTime</code> в ответ на запросы <code>getTerminalsSettings</code> , <code>getTerminalsStatisticalData</code>
4.15	Добавлен атрибут <code>iccid</code> в ответ на запрос <code>getTerminalsSettings</code>
4.14	Добавлены атрибуты комиссий в ответ на запрос <code>getAgentTP</code>
4.13	Добавлено описание обработки параметра <code>real_prv_id</code> в теге <code>constParams</code> ответа на запрос <code>getUIProviders</code>
4.12	Добавлен параметр <code>add-client-bank-com</code> в запрос <code>getPayments</code> Добавлен код ошибки 170
4.11	Добавлен запрос <code>getRemainingLimit</code> в интерфейс <code>terminals</code> Добавлен тег <code>cur_id</code> в ответ на запрос <code>getAgentInfo</code> Добавлен атрибут <code>trm_state</code> в ответ на запрос <code>getTerminals</code>
4.10	Добавлен тег <code>bonus_rate</code> в ответ на запрос <code>getAgentTP</code>
4.9	Добавлен запрос <code>getForbiddenBanknotes</code> в интерфейс <code>system</code>
4.8	Изменены запросы <code>getAgentTP</code> и <code>getTerminalTypes</code> (в ответы добавлены новые атрибуты)
4.7	Изменен запрос <code>getTerminalsSettings</code> (добавлен атрибут <code>CardReaderModel</code> в ответ запроса) Изменен запрос <code>getTerminalsStatisticalData</code> (добавлены атрибуты <code>CardReaderUsedHour</code> , <code>CardReaderUserDay</code> в ответ запроса) Изменен запрос <code>getTerminalsStatus</code> (добавлен атрибут <code>CardReaderStatus</code> в ответ запроса)
4.6	Добавлена процедура идентификации пользователя, изменен запрос <code>getUIProviders</code> (добавлен параметр <code>needPersonalInfo</code> в ответ запроса и возможное значение <code>identification</code> атрибута <code>disp_type</code>)
4.5	Добавлены запросы <code>needIdentifyUser</code> и <code>setIdentifyUser</code>
4.4	Изменен запрос <code>getNews</code> (добавлен параметр <code>cutTags</code>)
4.3	Обновлены списки значений для параметра <code>moneyType</code> для платежей МТС в запросах <code>authorizePayment</code> , <code>addOfflinePayment</code> , <code>checkPaymentRequisites</code>
4.2	Добавлен запрос <code>getProviderByPhone</code> в интерфейс <code>providers</code>
4.1	Изменен запрос <code>cancelPayment</code> (добавлен параметр <code>uid</code> ; добавлен тег <code>payment</code> в ответ запроса) Изменен запрос <code>getCashouts</code> (добавлен параметр <code>ttp-id</code>)

Версия документа	Изменения
3.17	Изменен запрос <code>getPersonInfo</code> (добавлен атрибут <code>right_id</code> в ответ запроса)
3.16	Изменен запрос <code>getPersonInfo</code> (добавлен тег <code>labels</code> в ответ запроса)
3.15	Изменен запрос <code>getAgentInfo</code> (добавлены атрибуты <code>cnt-phone</code> , <code>cnt-email</code> в ответ запроса) Изменен запрос <code>getPayments</code> (добавлен атрибут <code>is-canceled</code> в ответ запроса)
3.14	Изменен запрос <code>getPayments</code> (добавлен параметр <code>debit</code> ; добавлен атрибут <code>extras</code> в ответ запроса)
3.13	Добавлен запрос <code>getTerminalsWithCoordinates</code> в интерфейс <code>terminals</code>
3.12	Изменен запрос <code>getBalanceTree</code>
3.11	Добавлен запрос <code>getExemptionRates</code> в интерфейс <code>accountingReports</code>
3.10	Добавлен запрос <code>getBalanceTree</code> в интерфейс <code>agents</code>
3.9	Изменен запрос <code>getTerminals</code> (добавлен параметр <code>show-label</code> ; добавлен атрибут <code>labels</code> в ответ запроса)
3.8	Добавлен тег <code>languages</code> в ответ запроса <code>getConfig</code> Добавлен атрибут <code>labels</code> в ответ запроса <code>getTerminalInfo</code> Изменены запросы <code>getTerminalsSettings</code> , <code>getTerminalsStatus</code> : выполняются как в синхронном, так и в асинхронном режиме
3.7	Дополнен запрос <code>getUIProviders</code> : добавлен новый тип интерфейсного элемента (<code>type="disp_label"</code>), добавлены атрибуты в типы элементов <code>type="text_input"</code> и <code>type="combobox"</code>
3.6	Добавлен атрибут <code>news</code> в ответ запроса <code>getReferencesVersions</code> Добавлены атрибуты <code>CashInCapacity</code> , <code>MobileOperatorId</code> в ответ запроса <code>getTerminalsSettings</code>
3.5	Изменены запросы: <code>getOffer</code> , <code>getPayments</code> , <code>getProviders</code> Добавлен атрибут <code>value-date</code> в ответ запросов <code>getPaymentStatus</code> , <code>getPayments</code>
3.4	Изменено описание параметра <code>moneyType</code> в запросах <code>authorizePayment</code> , <code>addOfflinePayment</code> , <code>checkPaymentRequisites</code>
3.3	Добавлено описание атрибута <code>date</code> в запросах <code>addOfflinePayment</code> , <code>authorizePayment</code> , <code>confirmPayment</code> , <code>getPaymentStatus</code> , <code>checkPaymentRequisites</code> .
3.2	Изменено описание тега <code>extras</code> в запросах <code>authorizePayment</code> , <code>addOfflinePayment</code> , <code>checkPaymentRequisites</code> , <code>getPayments</code> Изменено описание атрибутов <code>terminal</code> , <code>currency</code> , <code>coins</code> в запросе <code>getTerminalsCash</code>
3.1	Изменен ответ на запрос <code>getConfigId</code>

Версия документа	Изменения
3.0	Добавлено ограничение на длину транзакции на терминале. Изменение см. в разделе 3.3.2.3
2.9	Перенос запроса <code>setCommissionTemplate</code> из интерфейса <code>terminals</code> в интерфейс <code>agents</code>
2.8	Изменены запросы: <code>getAgentTP</code> , <code>getAgentTariffs</code> , <code>getPayments</code> , <code>getTerminalsShiftHistory</code> , <code>getStatistics</code> , <code>getTerminalsStatus</code> , <code>getTerminalsCash</code> , <code>getPersons</code> , <code>getAgents</code> , <code>getBalance</code> , <code>getOffer</code> , <code>getConfig</code> <code>authorizePayment (moneyType)</code> , <code>addOfflinePayment (moneyType)</code> , <code>checkPaymentRequisites (moneyType)</code> , <code>getUIProviders</code> (добавлен атрибут <code>curId</code> в тег <code>provider</code>).
2.7	Добавлены запросы <code>addCommissionProfile</code> , <code>setCommissionProfile</code> , <code>bindTerminalTemplate</code> , <code>getCommissionTemplates</code> , <code>addCommissionTemplate</code> , <code>setCommissionTemplate</code> , <code>getCommissionsHistory</code> , <code>getTerminalCommissionTemplates</code> Изменены запросы <code>getCommissions</code> , <code>getCommissionProfiles</code> , <code>getUIProviders</code> Добавлен параметр <code>language</code> в запросы <code>getGroups</code> , <code>getUIGroups</code>
2.6	Дополнен запрос <code>getUIProviders</code> : добавлены новые типы интерфейсных элементов (<code>type="combobox"</code> и <code>type="disp_combobox"</code>)
2.5	Изменены запросы <code>getCommissionProfiles</code> , <code>getOffer</code> , <code>getConfig</code> Добавлены запросы <code>getAgentTariffs</code> , <code>getAgentTP</code> Изменено описание атрибута <code>visible</code> в запросе <code>getUIProviders</code>
2.4	Изменен запрос <code>getTerminals</code>
2.3	Добавлен новый запрос <code>setOuterCommission</code> в интерфейс <code>agents</code> Добавлены параметры в запрос <code>setState</code> Добавлены параметры в ответ <code>getTerminalsSettings</code> Добавлен тег <code>logos</code> в ответ запроса <code>getUIGroups</code> и <code>getUIProviders</code> Изменен формат времени в ответе <code>getTerminalsStatus</code>
2.2	Для каждого запроса добавлен список ролей, у которых есть права на выполнение запроса Обновлен ответ запроса <code>getTerminalInfo</code> интерфейса <code>terminals</code> Добавлен запрос <code>getTerminalRegionHierarchy</code> в интерфейс <code>terminals</code> Добавлены типы объектов КЛАДР (см. Приложение Н) Обновлен запрос <code>getCashouts</code> интерфейса <code>accountingReports</code> Обновлен запрос <code>getOffer</code> интерфейса <code>agents</code>

Версия документа	Изменения
2.1	<p>Обновлено описание запросов <code>setState</code> и <code>getConfig</code></p> <p>Изменены запросы <code>addCashout</code> и <code>addShiftClosure</code></p> <p>Добавлен справочник типов терминалов (см. Приложение М)</p> <p>Добавлены параметры в запрос <code>getAccountingTerminal</code></p> <p>Изменение в запросе <code>sendMail</code> (увеличен размер пересылаемого сообщения до 640 КБ)</p> <p>Внесены изменения в запрос <code>getUIProviders</code> и <code>getUIGroups</code> интерфейса <code>providers</code></p>
2.0	<p>Добавлен запрос <code>setConfigId</code> в интерфейс <code>terminals</code></p> <p>Добавлена таблица <code>MACHINE_STATUS</code></p>
1.32	Внесены изменения в запрос <code>getUIProviders</code> интерфейса <code>providers</code>
1.31	<p>Внесены изменения в авторизацию по цифровой подписи пакетов данных: логин должен быть закодирован с использованием схемы Base64 (см. раздел 3.2)</p> <p>Дополнены атрибуты и теги в запросы <code>getUIGroups</code> и <code>getUIProviders</code> интерфейса <code>providers</code></p> <p>На запрос <code>sendMail</code> интерфейса <code>system</code> наложено ограничение</p>
1.30	<p>Добавлены запросы в интерфейс <code>terminals</code>: <code>requestLog</code> (получение лога терминала), <code>requestProcessList</code> (получение списка процессов терминала), <code>requestZReport</code> (получение снятого Z-отчета терминала), <code>restartClientSoftware</code> (перезагрузка ПО АСО), <code>rebootTerminal</code> (перезагрузка терминала), <code>disableTerminal</code> (выключение терминала), <code>addShiftClosure</code> (закрытие смены на терминале)</p> <p>Запрос <code>setPassword</code> заменен на <code>changePassword</code></p> <p>Добавлен запрос <code>setFiscalMode</code> (установка фискального режима для агента) в интерфейс <code>agents</code></p> <p>Добавлены запросы <code>interruptPayment</code> (отмена платежа в промежуточном статусе) и <code>getGroups</code> (получение списка групп провайдеров) в интерфейс <code>providers</code></p> <p>Удален запрос <code>getCurrencyRates</code> (получение курсов валют провайдера) из интерфейса <code>providers</code></p> <p>В ответе на любой запрос содержится локализованное описание ошибки (раздел 3.3.2.2)</p>
1.29	Обновлен запрос <code>getUIProviders</code>
1.28	<p>Внесен пример нового механизма отмен в <code>getPayments</code></p> <p>Добавлены ассоциативное поле ввода и ассоциативная кнопка в запрос <code>getUIProviders</code></p>
1.27	<p>Добавлены необязательные параметры в запрос <code>getNews</code></p> <p>Изменено описание ответа на запрос <code>sendMail</code></p>
1.26	Дополнено описание запроса <code>getTerminals</code>
1.24	Реализована возможность указания диапазона строк для пересылаемого ответа (для асинхронного режима)
1.23	Обновлено описание запроса <code>getSubagentAccountStatement</code>

Версия документа	Изменения
1.22	Изменен запрос <code>getPayments</code> : выполняется строго в асинхронном режиме Удалены запросы: <code>getTerminalsCashes</code> (вместо него используйте <code>getTerminalsCash</code>), <code>getTerminalsStuffing</code> (вместо него используйте <code>getTerminalsSettings</code>) Добавлены запросы: <code>getTerminalsStatus</code> , <code>getTerminalsSettings</code> , <code>getTerminalsStatisticalData</code> , <code>getTerminalsCash</code> , <code>getAccountingTerminal</code>
1.21	Изменены запросы <code>getConfig</code> , <code>getUIProviders</code> , <code>getUIGroups</code> , <code>getProviders</code>
1.20	Изменены запросы <code>getPayments</code> , <code>getCashouts</code>
1.19	Удален запрос <code>getTerminalsCashouts</code> Добавлен запрос <code>getCashouts</code> Исправлен запрос <code>getTerminalsShiftHistory</code> Добавлена таблица типов инкассации
1.18	Дополнен запрос <code>getUIProviders</code> Добавлен запрос <code>getTerminalsCashouts</code> Исправления в запросах <code>getCommissions</code> , <code>getCommissionProfiles</code> , <code>getForeignCommissions</code> , <code>getPayments</code> , <code>setPassword</code>
1.17	Изменено описание запроса отправки почты (<code>sendMail</code>): добавлена возможность присоединения файлов к письму
1.16	Исправлено описание запроса <code>getAgents</code>
1.15	Добавлено описание запроса <code>cancelPayment</code>
1.12	К тегу <code>payment</code> добавлен атрибут <code>comment</code>
1.11	Исправлено описание атрибутов запросов <code>getPayments</code> и <code>getPhoneRanges</code>
1.10	Добавлен параметр <code>all</code> в запрос <code>getPayments</code>
1.9	Добавлены параметры в описание действия <code>getCommissionProfile</code>
1.8	Изменено описание запроса <code>setPublicKey</code>
1.7	В запрос <code>getPayments</code> добавлены параметры фильтрации
1.6	В запрос <code>getPayments</code> добавлен параметр <code>txn-type</code>
1.5	Добавлено описание запроса <code>getReport</code>

2. ВВЕДЕНИЕ

Данный документ представляет собой описание технического взаимодействия Агента с Системой в процессе приема и проведения платежей за услуги провайдеров.

2.1. Глоссарий

Термин	Определение
<i>Система</i>	Совокупность программно-аппаратных средств КИВИ, предназначенная для обеспечения приема и проведения платежей/переводов за услуги Провайдеров.
<i>Терминал</i>	Программно-технические комплексы различных типов, в том числе POS-терминалы (специализированные прикассовые устройства типа Point Of Sale), стационарное оборудование, мобильные устройства карманного типа, переносные терминалы и кассовые аппараты, автоматы самообслуживания, банкоматы, каналы дистанционного банковского обслуживания (интернет-банк, мобильный банк), банковский сервис «Автоплатеж».
<i>Агент</i>	Юридическое лицо или индивидуальный предприниматель, осуществляющие предпринимательскую деятельность, осуществляющие техническое взаимодействие с КИВИ в рамках настоящего Протокола, подписавшее Договор о приеме Платежей/договор об осуществлении Переводов/иной вид договора.
<i>Технический агент</i>	Логическая сущность в Системе, созданная для реализации группировки различных объектов Системы: терминалов, персон, платежей
<i>Персона</i>	Сотрудник агента: менеджер, кассир, бухгалтер и т.п. <i>Персона</i> – пользователь, который непосредственно производит какие-либо действия с <i>Системой</i> от имени агента. Персона должна принадлежать какому-либо <i>агенту</i> , поэтому при идентификации персоны <i>Система</i> автоматически определяет, от имени какого агента действует данная персона.
<i>Роль</i>	Набор прав на выполнение действий с Системой. Каждой персоне Системы назначена роль.
<i>Провайдер</i>	Коммерческая организация или индивидуальный предприниматель, предоставляющие потребителям Услуги или продающие Товары от собственного имени, а также благотворительная организация, созданная для осуществления благотворительной деятельности
<i>Версионность</i>	В <i>Системе</i> поддерживается версионность по справочникам, т.е. при изменении каких-либо данных изменяется версия справочника. Версионность позволяет загружать не всю информацию из справочника, а только измененную.
<i>Процессинг</i>	Совокупность аппаратно-программных средств, выполняющая функции приема, обработки и хранения информации о платежах
<i>АСО</i>	Автомат самообслуживания, предназначенный для проведения платежей через <i>Систему</i> процессинга АО «КИВИ»

Термин	Определение
<i>Протокол</i>	Набор правил обмена данными между ПО терминалов для приема платежей и процессингом КИВИ
<i>Интерфейс</i>	Совокупность средств и методов взаимодействия пользователя с <i>Системой КИВИ</i>
<i>Пользовательский интерфейс АСО</i>	Совокупность страниц, отображаемых на экране АСО
<i>Плейлист</i>	Полная совокупность рекламных данных на автомате (включает информацию о месте и времени размещения баннеров, ссылки на файлы, размеры файлов и т.п.). Подробнее о размещении рекламы на терминалах см. в разделе 6.2.3.1 .
<i>Авторизация</i>	Проверка подлинности персоны и предоставление этой персоне некоторых прав
<i>Верхняя комиссия (терминальная)</i>	Сумма, взимаемая агентом с платежа при внесении средств в терминал. Ставка комиссии устанавливается на терминале и указывается в процентах от суммы платежа. (Может быть установлена как дифференцированная комиссия , в этом случае ставка определяется в зависимости от времени суток и суммы платежа.)
<i>Нижняя комиссия (провайдерская)</i>	Вознаграждение, предоставляемое КИВИ агенту за проведение платежа за услуги провайдера. (Устанавливается на основании размера комиссии, предоставляемой провайдером услуг компании КИВИ.)
<i>Тарифный план</i>	Совокупность ставок <i>нижней комиссии</i> агента, рассчитанных согласно единым правилам, по шлюзам провайдеров
<i>Профиль комиссии</i>	Набор правил для вычисления комиссий в зависимости от времени платежа, суммы платежа и других условий
<i>Шаблон комиссии</i>	Набор правил взимания комиссий, заданных для провайдеров или групп провайдеров

3. ОПИСАНИЕ XML ПРОТОКОЛА

3.1. Формат передачи данных

Протокол:	Передача данных между клиентским и серверным ПО построена на основе протокола HTTP (RFC 2616: HTTP/1.1).
Адреса серверов (URL):	http(s)://xml1.qiwi.com/xmlgate/xml.jsp http(s)://xml2.qiwi.com/xmlgate/xml.jsp
Внешние IP-адреса:	Системы агента должны принимать запросы с IP-адресов подсетей: <ul style="list-style-type: none">• 79.142.16.0/20• 195.189.100.0/22• 91.232.230.0/23• 91.213.51.0/24
Метод передачи данных:	Для передачи данных используется метод RAW POST – поток данных передается в теле запроса, отправляемого клиентом на сервер.
Формат данных:	Данные передаются в формате XML (<i>Extensible Markup Language (XML) 1.0</i>). Подробнее о формате запроса и ответа сервера см. раздел 3.3.2 .

3.2. Авторизация

При обработке запроса от клиентского ПО выполняется авторизация (проверка подлинности) и аутентификация (проверка наличия прав на выполнение запроса) клиента.

Поддерживаются два основных способа авторизации:

- **авторизация по цифровой подписи** пакетов данных. Наиболее приоритетный способ авторизации.

XML пакет подписывается электронной цифровой подписью (ЭЦП), при этом в HTTP-запрос помещаются следующие заголовки:

- "X-Digital-Sign" – ЭЦП, закодированная с использованием схемы **Base64**;
- "X-Digital-Sign-Alg" – алгоритм вычисления ЭЦП;
- "X-Digital-Sign-Login" – обычный логин пользователя или закодированный с использованием схемы **Base64**.

Подробнее об авторизации с использованием ЭЦП см. в [Приложении Ж](#).

- **авторизация по данным в теле запроса** (подробнее см. в разделе [3.3.2.1](#)). Менее приоритетный способ авторизации.

Авторизация производится по логину и паролю персоны, а также идентификатору терминала.

ПРИМЕЧАНИЕ



Текущий агент – агент, от имени которого произошла авторизация. Агент определяется по логину персоны, указанному в запросе.

3.3. Организация протокола

Взаимодействие клиентского ПО и *Системы* происходит следующим образом:

1. Клиентское ПО формирует запрос в формате XML, включая в него необходимые действия, и передает его по протоколу HTTP на сервер.
2. Сервер осуществляет [авторизацию клиента](#) и разбор XML.
3. В случае успешной авторизации сервер выполняет действия (или добавляет действия в очередь, в случае [асинхронного режима обработки](#)).
4. Сервер возвращает клиенту [ответ в формате XML](#), в том числе результат выполнения запроса (ОК или код ошибки в случае ошибки).

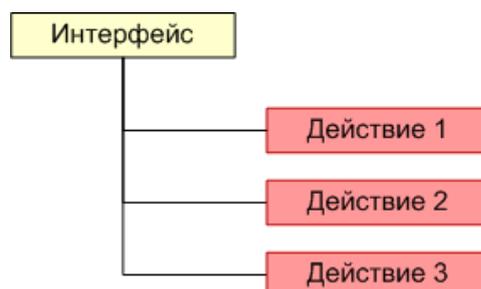
3.3.1. Интерфейсы и действия

Реализация протокола основана на *интерфейсах* и *действиях*.

- *Интерфейсы* (см. [Глоссарий](#)) – типы (группы) возможных действий (аналогичны классам объектов в ООП).
- *Действия* (см. [Глоссарий](#)) – действия, выполняемые в *Системе* (аналогичны методам объектов в ООП).

Каждый интерфейс объединяет несколько действий (см. [Рис. 1](#)).

Рис. 1. Структура интерфейса



В *Системе* существуют следующие интерфейсы:

- [System](#) – системный интерфейс.
Включает действия над *Системой* в целом: получение общих справочников (курсы валют, список действий, список кодов завершения), получение новостей, а также отправка сообщений на терминалы.
- [Terminals](#) – интерфейс управления терминалами для проведения платежей.
Включает действия, относящиеся к терминалам: проверка и загрузка новых сообщений, проведение инкассаций, запрос состояния терминала и т.п.
- [Agents](#) – интерфейс управления агентами.
Включает действия, относящиеся к агентам: получение подробной информации о *текущем* агенте, а также обо всех его субагентах, получение информации о балансе и овердрафте, распределение средств между субагентами и т.п.
- [Providers](#) – интерфейс провайдеров.
Включает действия, относящиеся к провайдерам: авторизация и проведение платежей, получение пользовательского интерфейса, запрос номерных емкостей, получение списка провайдеров и т.п.
- [Persons](#) – интерфейс управления персонами.

Включает действия, относящиеся к персонам: смена пароля, передача публичного ключа, запрос полной информации о персоне, а также запрос списка персон текущего агента.

- [Templates](#) – интерфейс управления шаблонами платежей.

Включает действия, относящиеся к шаблонам: регистрация клиента и изменение клиентских данных, создание платежного шаблона и его блокировка, получение списка всех шаблонов по конкретному клиенту, списка доступных услуг и необходимых реквизитов для определенной услуги.

- [Reports](#) – формирование стандартной отчетности.

Включает действия, позволяющие получить стандартные отчеты по платежам и терминалам: получение списка платежей, информация о состоянии терминалов, информация о проведенных инкассациях и т.п.

- [AccountingReports](#) – отчеты для управленческого учета.

Включает действия, позволяющие получить выписку по счету агента, сальдо по счету агента, список счетов и договоров агента, и т.д.

В описании каждого действия указан набор ролей, имеющих право на выполнение данного действия. Проверка прав персоны (т.е. прав назначенной ей роли) выполняется при авторизации запроса.

3.3.2. Формат запроса и ответа

3.3.2.1. Запрос

В общем виде запрос выглядит следующим образом:

```
<?xml version="1.0" encoding="utf-8"?>
<request>
<auth login="login" sign="password-sign" signAlg="MD5"/>
<client terminal="123" software="Dealer v0" serial="123456" timezone="GMT+23"
language="en"/>
  <interfacel>
    <action1 language="ru">
      Параметры действия
    </action1>
    <action2>
      Параметры действия
    </action2>
  </interfacel>
  <interface2>
    <action3>
      Параметры действия
    </action3>
  </interface2>
</request>
```

Где:

- request – корневой тег запроса;
- auth и client – теги, содержащие авторизационную информацию:
 - ⊕ auth – тег содержит информацию для авторизации пользователя, от имени которого отправляется запрос:

ВНИМАНИЕ



Данный тег не требуется, если используется авторизация по цифровой подписи (см. Приложение Ж).

- ❖ `login` – логин пользователя (персоны);
- ❖ `sign` – подпись, сформированная на основании *неодноразового* пароля персоны по алгоритму MD5;
- ❖ `signAlg` – алгоритм авторизации (только "MD5");
- ⊕ `client` – тег содержит информацию о терминале, с которого отправляется запрос:
 - ❖ `terminal` – идентификатор терминала;
 - ❖ `software` – версия ПО, установленного на терминале (соответствует типу терминала);
 - ❖ `serial` – серийный номер терминала;
 - ❖ `timezone` – используется для получения результатов во временной зоне, отличной от Europe/Moscow;

ПРИМЕЧАНИЕ

Синтаксис смещения времени может задаваться текстовым значением (например, Asia/Vladivostok). Перечень всех зон см. в http://en.wikipedia.org/wiki/List_of_zoneinfo_time_zones. Также в запросе можно использовать формат даты со смещением (например, GMT+4).

- ❖ `language` – язык получаемого ответа на запрос (используется только для локализованных справочников).
- `interface1, interface2` – теги интерфейсов. Содержат теги действий.
- `action1, action2, action3` – теги действий. Содержат параметры действий и, возможно, атрибут:
 - ⊕ `language` – язык получаемого ответа для выбранного действия (используется только для локализованных справочников).

ПРИМЕЧАНИЕ

В одном клиентском запросе может передаваться несколько действий различных интерфейсов.

ВНИМАНИЕ

Выполняются только действия, которые разрешены роли персоны, использованной для авторизации клиентского запроса. Для каждого действия в документе приведен список разрешенных ролей.

3.3.2.2. Ответ

В общем виде ответ сервера на запрос выглядит следующим образом:

```
<?xml version="1.0" encoding="utf-8"?>
<response result="N">
  <interface1>
    <action1 result="N" result-description="описание ошибки">
      Параметры обработки действия
    </action1>
    <action2 result="N" result-description="описание ошибки">
      Параметры обработки действия
    </action2>
```

```

</interface1>
<interface2>
  <action3 result="N" result-description="описание ошибки">
    Параметры обработки действия
  </action3>
</interface2>
</response>

```

где:

- `response` – корневой тег ответа. Содержит атрибуты:
 - `result` – отражает успешность/неуспешность выполнения запроса в целом и содержит код ошибки. Полный список кодов ошибок содержится в [Приложении В](#).
 - `result-description` – содержит описание ошибки.

ПРИМЕЧАНИЕ



Если ошибок при выполнении действия не возникло, атрибут `result` возвращает значение "0", а атрибут `result-description` отсутствует.

ВНИМАНИЕ



Если запрос не прошел первичную проверку (например, ошибка авторизации или сервер загружен), теги интерфейсов и действий (см. ниже) будут отсутствовать.

- `interface1`, `interface2` – теги интерфейсов из запроса. Содержат теги действий.
- `action1`, `action2`, `action3` – теги действий из запроса. Содержат параметры обработки действий:
 - ⊕ `result` – атрибут возвращает код результата выполнения действия (код ошибки). Полный список кодов ошибок содержится в [Приложении В](#).
 - ⊕ `result-description` – атрибут содержит описание ошибки.

ПРИМЕЧАНИЕ



Если ошибок при выполнении действия не возникло, атрибут `result` возвращает значение "0", а атрибут `result-description` отсутствует.

Наиболее часто встречающиеся ошибки выполнения действия и рекомендации по их устранению:

- ❖ `<action1 result="133" ...>` – нет прав на прием платежей.
Роль персоны, использованной для авторизации, не имеет прав на выполнение данного действия.
- ❖ `<action1 result="202" ...>` – ошибка данных запроса.
Проверьте корректность параметров данного действия.
- ❖ `<action1 result="245" ...>` – в [тегах авторизации](#) неверно указан тип терминала.
Для XML-терминала тег `client` должен содержать атрибут `software="Dealer v0"`.

- ❖ `<action1 result="295"...>` – ошибка в названии действия или интерфейса.

Проверьте наименование интерфейса / действия;

- ❖ `<action1 result="300"...>` – техническая ошибка.

Повторите запрос с тем же набором данных (при проведении платежа – с тем же идентификатором платежа) до получения ответа со статусом/[кодом ошибки](#), на основе которого можно сделать вывод о прохождении платежа/выполнении действия.

3.3.2.3. Формат данных

ПРИМЕЧАНИЕ



Клиент имеет возможность передавать запросы в сжатом виде и получать сжатые ответы. Для сжатия используется алгоритм **gzip**.

При передаче сжатых запросов в заголовке HTTP-пакета должна содержаться запись "Content-encoding: gzip".

Для получения ответов в сжатом виде клиент должен указать в заголовке: "Accept-encoding: gzip".

ВНИМАНИЕ



Платежная система может вводить ограничения на размер запроса.

Если значение Content-length в заголовке пакета HTTP больше максимального заданного *Системой*, то она может отказать в обработке данного запроса.

В случае отказа обработки сервер платежной системы посылает ответ с **413** кодом ошибки HTTP (*Объект запроса слишком большой, Request Entity Too Large*), при этом в теле ответа содержится подобный текст:

```
<?xml version="1.0" encoding="utf-8"?>
<response>Request too large. Request length limit is 100
Kb.</response>
```

Система читает из входного потока только указанное в Content-length число байт.

Данные в тегах и атрибутах представляются следующим образом:

- Идентификатор платежа – натуральное число длиной до 18 цифр.
- Целые числа (в основном, идентификаторы) представляются в десятичном виде. Например, 1234.
- Суммы платежей представляются в основных единицах валюты.
Например, в рублях: в десятичном виде с двумя знаками после разделителя. Разделителем является точка.
- Кодировка строк должна соответствовать значению атрибута `encoding` тега `xml`.
- Дата и время представляются в соответствии с **ISO 8601:2004**. (Используется следующий формат: "ГГГГ-ММ-ДДТчч:мм:сс", однако может быть использован и другой, соответствующий стандарту).
- При описании логических значений истине соответствует значение `true`, лжи – `false`.
- Бинарные данные должны кодироваться с использованием схемы **Base64**.
- Коды валют представляются как целые числа. Значения кодов соответствуют: **ISO 4217**. Общий справочник кодов валют можно посмотреть [здесь](#).

Основные коды валют, в соответствии со стандартом:

- российский рубль – 643;
 - доллар США – 840;
 - евро – 978;
 - украинская гривна – 980;
 - грузинский лари – 981;
 - таджикский сомони – 972.
- Номера телефонов представляются в десятизначном формате.

3.3.3. Режимы обработки действий

Протокол поддерживает синхронный и асинхронный режимы обработки действий:

ПРИМЕЧАНИЕ



Некоторые действия могут обрабатываться как в синхронном, так и в асинхронном режимах. По умолчанию действия выполняются в синхронном режиме.

В описании каждого действия отмечено, в каком режиме оно может обрабатываться.

- *Синхронный режим* (`mode="sync"`). В данном режиме ответ на действие возвращается сразу после выполнения, т.е. синхронно.
- *Асинхронный режим* (`mode="async"`). Для выполнения в асинхронном режиме необходимо добавить атрибут `mode="async"` (См. [Пример 1](#)). Данный режим предназначен для запросов, выполнение которых требует длительного времени и/или значительных ресурсов БД.

Порядок использования асинхронного режима следующий:

1. Запрос на добавление действия в очередь (см. [Пример 1](#)).
2. В ответ в параметре `quid` возвращается номер действия в очереди (см. [Пример 2](#)).
3. По номеру действия выполняется запрос статуса выполнения запроса (см. [Пример 3](#)).
4. В ответе возвращается статус обработки асинхронного запроса (см. [Приложение Д](#)) и, если запрос успешно обработан, запрошенные данные (см. [Пример 4](#)).

Пример 1. Асинхронный запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="password-sign" signAlg="MD5"/>
  <client terminal="123" software="Dealer v0" serial="123456"/>
  <providers>
    <getProviders mode="async"/>
  </providers>
</request>
```

Пример 2. Ответ на асинхронный запрос

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <getProviders quid="50" result="0" status="1"/>
  </providers>
</response>
```

Пример 3. Получение статуса обработки асинхронного запроса

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="password-sign" signAlg="MD5"/>
  <client terminal="123" software="Dealer v0" serial="123456"/>
  <providers>
    <getProviders quid="50"/>
  </providers>
</request>
```

Пример 4. Ответ на асинхронный запрос с данными

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <getProviders quid="50" result="0" status="3">
      <row fiscal-name="Телевидение" long-name="Полное название провайдера" max-
amount="14999" min-amount="1" prv-id="1" prv-inn="1111111111" prv-support-
phone="8(495) 111-11-11, 8(495) 111-11-12" receipt-name="Провайдер" regexp="^\d+$"
short-name="Короткое название провайдера">
        ...
      </getProviders>
    </providers>
  </response>
```

ВНИМАНИЕ

При получении промежуточного (не финального) статуса обработки запроса для проверки текущего статуса необходимо повторить запрос получения статуса ([Пример 3](#)).

В асинхронных запросах реализована возможность возврата части данных ответа: вы можете задать интервал номеров возвращаемых строк ответа (см. [Пример 5](#)):

- start – номер первой строки интервала,
- end – номер последней строки интервала.

Пример 5. Асинхронный запрос с интервалом номеров

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="password-sign" signAlg="MD5"/>
  <client terminal="123" software="Dealer v0" serial="123456"/>
  <providers>
    <getProviders quid="50" start="277" end="2532"/>
  </providers>
</request>
```

В ответ будут переданы строки из указанного интервала, а также общее число строк ответа (в атрибуте count).

ПРИМЕЧАНИЕ

При описании асинхронно выполняющихся запросов не приводятся ответ сервера, возвращающий номер запроса в очереди ([Пример 2](#)) и запрос на получение результатов от сервера ([Пример 3](#)).

4. ПРИЕМ ПЛАТЕЖЕЙ

При приеме платежа выполняется сбор необходимых сведений о пользователе для отправки провайдеру. Набор данных определяется [интерфейсом](#) провайдера.

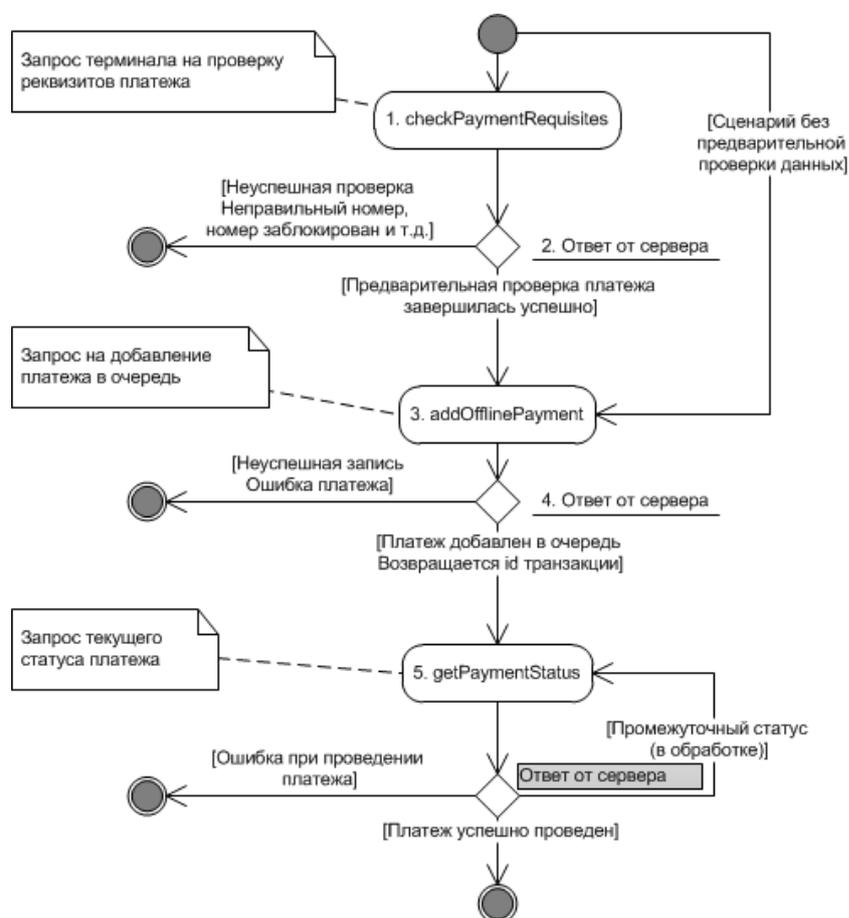
В *Системе* поддерживаются два метода проведения платежей: оффлайн и онлайн.

- [Оффлайн-проведение](#) – платеж отправляется с терминала в Систему, плательщик получает только подтверждение о сохранении платежа в Системе. Вся дальнейшая обработка платежа выполняется без участия плательщика.
- [Онлайн-проведение](#) – платеж обрабатывается Системой, плательщик получает подтверждение, что платеж отправлен провайдеру, и результат обработки платежа у провайдера.

Также при приеме платежа по требованию провайдера может быть проведена **Ошибка! Источник ссылки не найден..**

4.1. Сценарий проведения платежей оффлайн

Рис. 2. Схема проведения оффлайн платежа



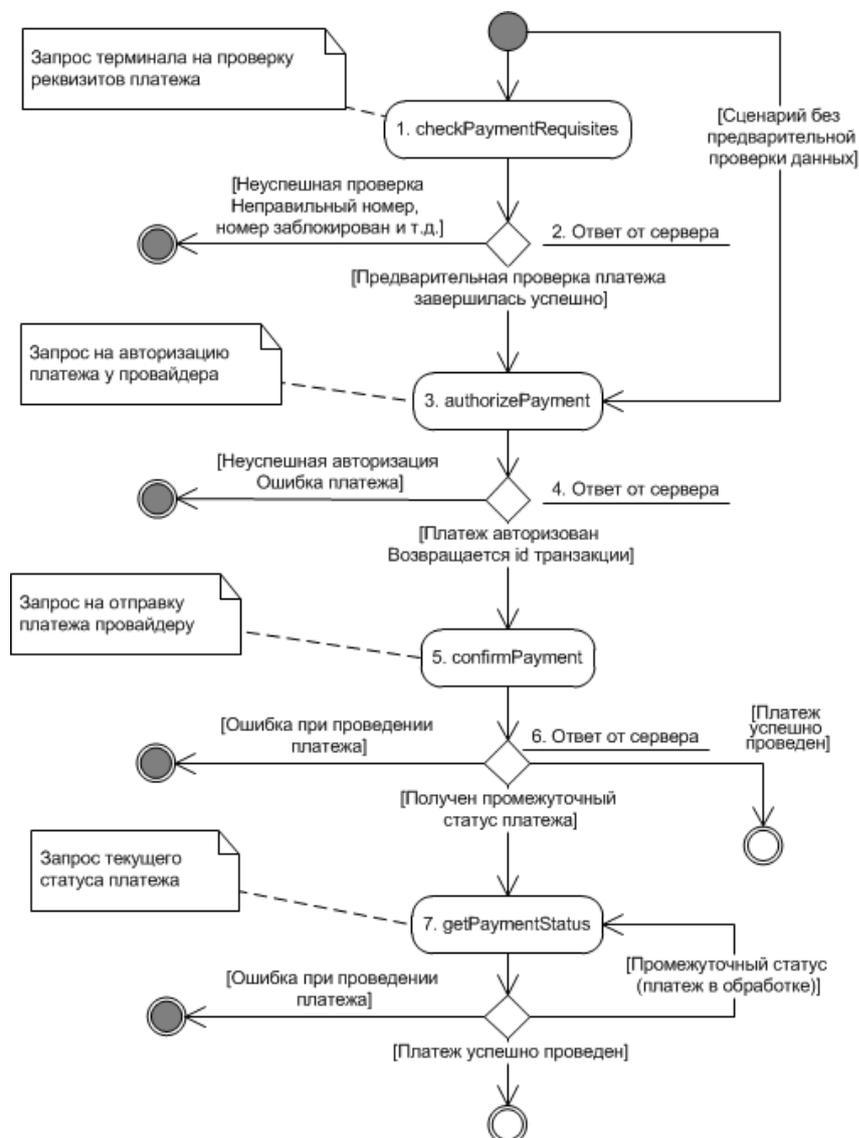
1. Предварительно может быть выполнен запрос проверки данных плательщика и реквизитов платежа `providers.checkPaymentRequisites` (зависит от [интерфейса](#) провайдера).
2. Ответ от сервера включает информацию о результате проверки, [статус транзакции](#), а также уникальный идентификатор транзакции в *Системе* и дополнительные поля для отображения

плательщику. Платеж может пройти предварительную проверку или отклониться в связи с ошибкой.

3. С терминала отправляется запрос `providers.addOfflinePayment`.
4. Ответ от сервера включает информацию о результате записи платежа в базу, [статус проведения платежа](#) (промежуточный или финальный), а также уникальный идентификатор транзакции в *Системе* и дополнительные поля для отображения пользователю.
5. В Системе платеж попадает в очередь обработки для отправки провайдеру. Для проверки статуса платежа необходимо использовать запрос `providers.getPaymentStatus`.

4.2. Сценарий проведения платежей онлайн

Рис. 3. Схема проведения онлайн платежа



1. Предварительно может быть выполнен запрос проверки данных плательщика и реквизитов платежа `providers.checkPaymentRequisites` (зависит от [интерфейса](#) провайдера).

2. Ответ от сервера включает информацию о результате проверки, а также уникальный идентификатор транзакции в *Системе* и дополнительные поля для отображения плателъщику. Платеж может пройти предварительную проверку или отклонен в связи с ошибкой.
3. С терминала отправляется запрос авторизации платежа `providers.authorizePayment`. Выполняется авторизация платежа у Провайдера.
4. Ответ от сервера включает информацию о результате авторизации платежа у провайдера, а также уникальный идентификатор транзакции в *Системе*. Платеж может перейти в статус авторизованного или неуспешного (при неуспешной авторизации).
5. В случае успешной авторизации с терминала отправляется запрос на проведение платежа `providers.confirmPayment`. В Системе выполняется отправка платежа Провайдеру и получение от него ответа.
6. Ответ от сервера включает информацию о результате отправки платежа провайдеру, а также уникальный идентификатор транзакции в *Системе*. При этом платеж может перейти в статус проведенного или отклоненного в связи с ошибкой проведения.
7. При получении промежуточного статуса платежа для проверки текущего статуса платежа необходимо использовать запрос [providers.getPaymentStatus](#).

4.3. Статусы платежей

Платеж может находиться в одном из следующих статусов (см. также [Приложение Г](#)):

- **3** – «авторизован»;
- **1** – «проводится»;
- **2** – «проведен» (финальный статус);
- **0** – «ошибка проведения» (финальный статус).

Статус платежа возвращается в атрибуте `status` в ответе на платежные запросы. Статус платежа изменяется в ходе выполнения сценариев платежей ([Табл. 1](#)).

Табл. 1. Изменение статусов платежей

Платежный запрос	Статус платежа	Описание
addOfflinePayment	1	Платеж в проведении. Необходимо уточнять статус платежа запросом providers.getPaymentStatus
	2	Платеж проведен. Внимание! Если в ответе присутствует код ошибки проведения платежа, необходимо уточнять статус платежа запросом providers.getPaymentStatus , и при отсутствии изменений в течение 24 часов обратиться в техническую поддержку.
	0	Ошибка проведения платежа. Платеж принял окончательный неуспешный статус. Для выяснения причины ошибки необходимо рассмотреть код ошибки проведения платежа. При получении такого же статуса при отправке другого платежа следует обратиться в техническую поддержку
checkPaymentRequisites	0	Ошибка предварительной проверки платежа. Для выяснения причины ошибки необходимо рассмотреть код ошибки проведения платежа.
	3	Платеж прошел предварительную проверку

Платежный запрос	Статус платежа	Описание
authorizePayment	0	Платеж не авторизован. Для выяснения причины ошибки необходимо рассмотреть код ошибки проведения платежа.
	3	Платеж авторизован
confirmPayment	1	Подтверждение авторизованного платежа. Необходимо уточнять статус платежа запросом providers.getPaymentStatus
	2	Платеж проведен
	0	Ошибка подтверждения авторизованного платежа

4.4. Предоставление дополнительных данных о плательщике

Согласно действующему законодательству РФ при совершении платежей в пользу получателей-физических лиц, получателей-нерезидентов, а также платежей на сумму свыше 15 тыс. рублей Агент обязан предоставить дополнительные сведения о плательщике. Предварительно следует обратиться к вашему курирующему менеджеру для подписания Дополнительного соглашения (ДС) на такие платежи. В случае, если в Системе отсутствуют сведения о подписанном ДС, платеж будет отклонен с кодом ошибки 173 - Превышен лимит суммы платежа без идентификации.

Перечень необходимых данных зависит от получателя платежа, типа денежных средств (`moneyType`), суммы платежа, а также от Агента. Например, для кредитных организаций обязанность по предоставлению сведений о плательщике наступает при совершении платежей на сумму свыше 15 тыс. рублей, а для банковских платежных агентов - при платежах на любую сумму в пользу определенных получателей. Данные передаются в экстра-полях совершаемого платежа. Общий перечень экстра-полей, используемых для передачи запрашиваемых сведений, представлен в [Табл. 2](#).

Чтобы определить, какие сведения потребуются предоставить для конкретного платежа, перед его отправкой следует выполнить запрос `getRequiredPersonalData` (см. [6.4.6](#)), подставив в него основные реквизиты этого платежа.

На стороне Системы настроен контроль передачи необходимых данных, при их отсутствии платеж будет отклонен с кодом ошибки 205 - Отсутствуют обязательные данные плательщика.

При совершении платежа банковским платежным агентом будет выполнена упрощенная идентификация плательщика. Если этот процесс завершится неудачно, платеж будет отклонен с кодом ошибки 172 - Данные плательщика не прошли проверку.

Табл. 2. Общий перечень экстра-полей, используемых для передачи сведений о плательщике

Наименование экстра-поля	Описание
<code>payer_last_name</code>	Фамилия плательщика
<code>payer_first_name</code>	Имя плательщика
<code>payer_middle_name</code>	Отчество плательщика
<code>payer_phone</code>	Номер телефона плательщика
<code>payer_birth_date</code>	Дата рождения плательщика

Наименование экстра-поля	Описание
payer_document_type	Тип документа, удостоверяющего личность плательщика
payer_document_series	Серия документа, удостоверяющего личность плательщика
payer_document_number	Номер документа, удостоверяющего личность плательщика
payer_document_issue_date	Дата выдачи документа, удостоверяющего личность плательщика
payer_citizenship	Гражданство плательщика
payer_inn	ИНН плательщика
payer_address	Адрес регистрации плательщика
payer_account	Номер счета плательщика
external_uid	Уникальный номер транзакции во внешней системе
payer_company_name	Наименование плательщика - юридического лица

4.4.1. Перечень сведений о плательщике, предоставляемых кредитной организацией

4.4.1.1. Общие сведения

Правила, описанные ниже, распространяются на все платежи, у которых сумма к зачислению (атрибут `amount` тега `<to>`) превышает 15 тыс. рублей.

Перечень дополнительных сведений о плательщике зависит от источника платежа (атрибут `moneyType` тега `<to>`) и типа поставщика (резидент/нерезидент).

Выделяются три категории источников платежа:

- Карты – источником средств для платежа является банковская карта;
- Счет – платеж осуществляется со счета;
- Иное – другие источники средств, например наличные или электронные деньги.

Соответствующие этим категориям значения `moneyType` представлены в [Приложении П](#).

Если на оплачиваемом провайдере (атрибут `service` тега `<to>`) имеется метка Провайдер-нерезидент (атрибут `tag` тега `<provider>` запроса [getUIProviders](#)), то он является нерезидентом. В противном случае провайдер является резидентом.

Таким образом, для каждого платежа определяется категория источника платежа и тип поставщика, после чего платеж однозначно относится к одному из 6 вариантов:

- Источник платежа - карты, получатель - резидент
- Источник платежа - карты, получатель - нерезидент
- Источник платежа - счет, получатель - резидент
- Источник платежа - счет, получатель - нерезидент
- Иной источник платежа, получатель - резидент
- Иной источник платежа, получатель - нерезидент

Конкретные наборы дополнительных данных о плательщике для каждого из вариантов описаны в пп. 4.4.1.2 - 4.4.1.7

4.4.1.2. Источник платежа - карты, получатель - резидент

Для таких платежей предоставление дополнительных данных о плательщике не требуется.

4.4.1.3. Источник платежа - карты, получатель - нерезидент

Для таких платежей предоставление дополнительных данных о плательщике не требуется.

4.4.1.4. Источник платежа - счет, получатель - резидент

Дополнительно в экстра-полях платежа (тег <extras>) требуется передать:

Наименование экстра-поля	Описание
payer_last_name	Фамилия плательщика
payer_first_name	Имя плательщика
payer_middle_name	Отчество плательщика (при наличии)
payer_account	Номер счета плательщика
payer_inn ИЛИ payer_address	ИНН плательщика Адрес регистрации плательщика

4.4.1.5. Источник платежа - счет, получатель - нерезидент

Дополнительно в экстра-полях платежа (тег <extras>) требуется передать:

Наименование экстра-поля	Описание
payer_last_name	Фамилия плательщика
payer_first_name	Имя плательщика
payer_middle_name	Отчество плательщика (при наличии)
payer_address	Адрес регистрации плательщика

4.4.1.6. Иной источник платежа, получатель - резидент

Дополнительно в экстра-полях платежа (тег <extras>) требуется передать:

Наименование экстра-поля	Описание
payer_last_name	Фамилия плательщика
payer_first_name	Имя плательщика
payer_middle_name	Отчество плательщика (при наличии)

Наименование экстра-поля	Описание
payer_inn ИЛИ payer_address	ИНН плательщика Адрес регистрации плательщика
external_uid	Уникальный номер транзакции во внешней системе (при наличии)

4.4.1.7. Иной источник платежа, получатель - нерезидент

Дополнительно в экстра-полях платежа (тег <extras>) требуется передать:

Наименование экстра-поля	Описание
payer_last_name	Фамилия плательщика
payer_first_name	Имя плательщика
payer_middle_name	Отчество плательщика (при наличии)
payer_address	Адрес регистрации плательщика
external_uid	Уникальный номер транзакции во внешней системе (при наличии)

4.4.2. Перечень сведений о плательщике, предоставляемых банковским платежным агентом

4.4.2.1. Общие сведения

По всем платежам, попадающим под действие правил проверки, будет выполнена упрощенная идентификация плательщика.

ПРИМЕЧАНИЕ



Если процесс идентификации завершится неудачно, платеж будет отклонен с кодом ошибки 172 - Данные плательщика не прошли проверку.

В правилах проверки суммой платежа считается сумма к зачислению (атрибут amount тега <to>).

Принадлежность получателя платежа (провайдера) к нерезидентам определяется наличием на нем метки (атрибут tag) Провайдер-нерезидент.

Категория получателя платежа (провайдера) определяется группой провайдера (атрибут grpId).

4.4.2.2. Получатель - физическое лицо, сумма платежа - любая

Под действие данного правила попадают все платежи в пользу поставщиков из категорий Переводы на карты, Денежные переводы и Погашение кредитов, услуги банков (grpId 111, 485 и 362 соответственно).

Дополнительно в экстра-полях платежа (тег <extras>) требуется передать:

Наименование экстра-поля	Описание
payer_last_name	Фамилия плательщика
payer_first_name	Имя плательщика
payer_middle_name	Отчество плательщика (при наличии)
payer_phone	Номер телефона плательщика
payer_birth_date	Дата рождения плательщика
payer_document_type	Тип документа, удостоверяющего личность плательщика
payer_document_series	Серия документа, удостоверяющего личность плательщика
payer_document_number	Номер документа, удостоверяющего личность плательщика
payer_document_issue_date	Дата выдачи документа, удостоверяющего личность плательщика
payer_citizenship	Гражданство плательщика

4.4.2.3. Получатель - нерезидент, сумма платежа - любая

Дополнительно в экстра-полях платежа (тег <extras>) требуется передать:

Наименование экстра-поля	Описание
payer_last_name	Фамилия плательщика
payer_first_name	Имя плательщика
payer_middle_name	Отчество плательщика (при наличии)
payer_phone	Номер телефона плательщика
payer_birth_date	Дата рождения плательщика
payer_document_type	Тип документа, удостоверяющего личность плательщика
payer_document_series	Серия документа, удостоверяющего личность плательщика
payer_document_number	Номер документа, удостоверяющего личность плательщика
payer_document_issue_date	Дата выдачи документа, удостоверяющего личность плательщика
payer_citizenship	Гражданство плательщика
payer_address	Адрес регистрации плательщика

4.4.2.4. Получатель - юридическое лицо-резидент либо электронные деньги, сумма платежа - не более 15 тыс рублей.

Для таких платежей предоставление дополнительных данных о плательщике не требуется, упрощенная идентификация не проводится.

4.4.2.5. Получатель - юридическое лицо-резидент либо электронные деньги, сумма платежа - свыше 15 тыс рублей

Дополнительно в экстра-полях платежа (тег <extras>) требуется передать:

Наименование экстра-поля	Описание
payer_last_name	Фамилия плательщика
payer_first_name	Имя плательщика
payer_middle_name	Отчество плательщика (при наличии)
payer_phone	Номер телефона плательщика
payer_birth_date	Дата рождения плательщика
payer_document_type	Тип документа, удостоверяющего личность плательщика
payer_document_series	Серия документа, удостоверяющего личность плательщика
payer_document_number	Номер документа, удостоверяющего личность плательщика
payer_document_issue_date	Дата выдачи документа, удостоверяющего личность плательщика
payer_citizenship	Гражданство плательщика

5. КОНФИГУРАЦИЯ ТЕРМИНАЛОВ

Конфигурация – текущее состояние терминала, включающее интерфейс, комиссии, номерные емкости сотовых операторов и др. Все параметры конфигурации терминала задаются в *Системе*.

Конфигурация характеризуется параметром «идентификатор конфигурации». При изменении какого-либо параметра в состоянии терминала данный идентификатор изменяется.

Если терминал получает информацию о том, что идентификатор конфигурации изменился, происходит автоматическое обновление изменившихся данных с сервера приложений Системы.

5.1. Построение интерфейса для терминалов

Интерфейс терминала – это совокупность средств и методов взаимодействия пользователя с *Системой КИВИ* посредством терминалов.

Интерфейс терминала содержит следующие компоненты:

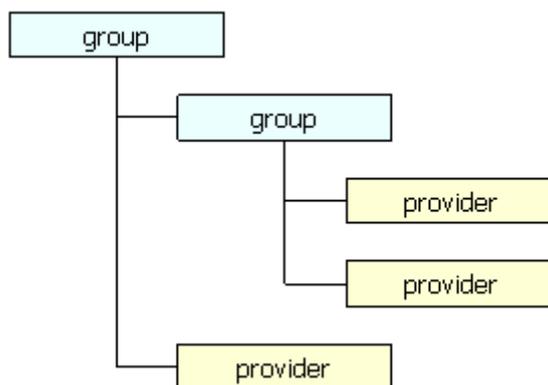
- иерархическая структура групп и содержащихся в них провайдеров (см. раздел [5.1.1](#)),
- параметры отображения страниц интерфейса провайдера (порядок и элементы управления для каждой станицы), см. раздел [5.1.2](#).

5.1.1. Структура интерфейса

Загрузка интерфейса на терминал производится с помощью следующего действия терминального XML-протокола:

- [getUIGroups](#) – получение структуры групп и провайдеров. Каждой группе и провайдеру присваивается порядковый номер, определяющий его положение в интерфейсе терминала ([Рис. 4](#)). Структура интерфейса содержит логотипы для групп.

Рис. 4. Структура XML-интерфейса групп



Рассмотрим пример получения и построения интерфейса:

Пример 6. Интерфейс терминала

```

<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <getUIGroups result="0">
      <group id="20" logo="logos/!anonymous/icon_mobile.gif" logo_crc="CFE03E84"
logo_size="5054" name="Сотовая связь" orderId="1" tag="visible,ranges">
        <logos>
  
```

```
<logo crc="CFE03E84" path="logos/!anonymous/icon_mobile.gif" size="5054"
type="standard"/>
</logos>
<provider id="2" orderId="13" showInTop="5"/>
<provider id="1" orderId="14" showInTop="1"/>
...
</group>
<group id="103" logo="logos/!anonymous/grp103Icon_internet_telephons.gif"
logo_crc="fdf52353" logo_size="10744" name="Интернет и телефония" orderId="16"
tag="visible">
...
</group>
<group id="40" logo="logos/!anonymous/icon_tv.gif" logo_crc="A6B0C447"
logo_size="5515" name="ТВ" orderId="1464" tag="visible">
...
</group>
...
</getUIGroups>
</providers>
</response>
```

Разбор ответа сервера

В соответствии со списком групп, полученным в ответе на запрос `getUIGroups`, перечень категорий (групп) услуг выглядит следующим образом:

- Группа **Сотовая связь** (ID=20).
- Группа **Интернет и телефония** (ID=103).
- Группа **ТВ** (ID=40) и т.д.

Группы и провайдеры внутри каждой группы упорядочены по параметру `orderId`.

Например, внутри группы **Сотовая связь** провайдеры упорядочены следующим образом:

- Билайн (ID=2).
- МТС (ID=1) и т.д.

ПРИМЕЧАНИЕ



Каждая секция XML, соответствующая определенной группе или провайдеру, содержит специальный атрибут `<...tag="visible">`, в котором определяется видимость данного элемента интерфейса. Если данное значение атрибута отсутствует, то соответствующая группа/провайдер должен быть скрыт в интерфейсе. Тем не менее, этот элемент может быть найден функцией поиска (если она реализована на терминале).

5.1.2. Параметры отображения страниц интерфейса

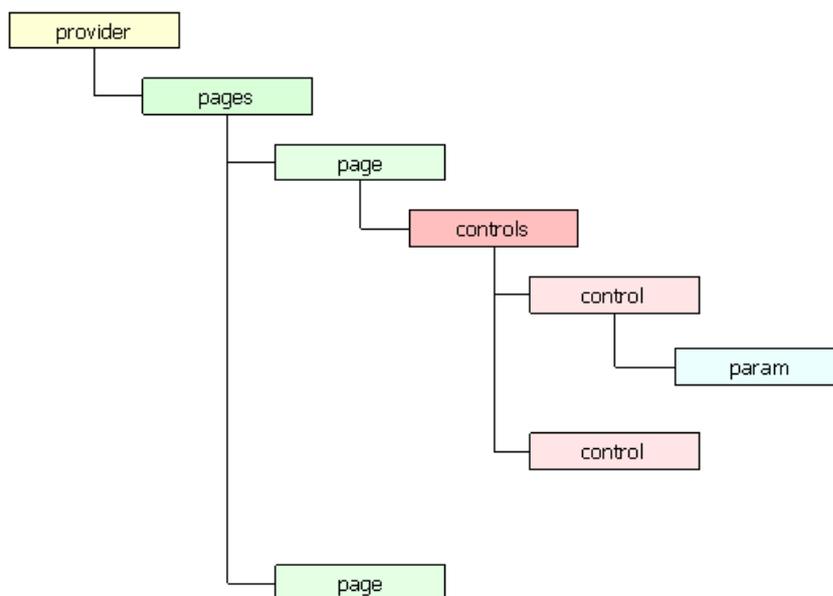
Конфигурация страниц интерфейса каждого провайдера предоставляется с помощью следующего действия терминального XML-протокола:

- [getUIProviders](#) – получение подробной информации об отображении страниц провайдеров, включая:
 - порядковый номер провайдера,
 - логотип,
 - название услуги,
 - поисковые слова,

- другие данные,
- страницы вместе с элементами управления:
 - ⊕ экранные клавиатуры,
 - ⊕ поля ввода,
 - ⊕ параметры каждого элемента: тип клавиатуры, маски полей ввода и др.

Структура XML-секций для каждого провайдера представлена на [Рис. 5](#).

Рис. 5. Структура XML-интерфейса провайдера



Для каждого провайдера определяются следующие элементы:

- `page` – страница сценария оплаты провайдера. Содержит идентификатор страницы и порядковый номер ее отображения в интерфейсе провайдера,
- `control` – специальные элементы управления для страниц интерфейса, например:
 - ⊕ поле ввода номера телефона. Задаются следующие параметры:;
 - ❖ маска ввода номера телефона, видимая на экране терминала (см. [Приложение А](#));
 - ❖ регулярное выражение для проверки введенных данных (см. [Приложение Б](#));
 - ❖ и т.д.

ПРИМЕЧАНИЕ



Страницы для проверки и подтверждения клиентом введенного номера телефона/счета/других данных для оплаты услуги, принятия наличных денег и подтверждения отправки платежа являются стандартными и не описываются в интерфейсе страниц провайдера.

Примеры структуры страниц интерфейса различных провайдеров приведены в следующем разделе.

5.2. Реализация сценария проведения платежа в интерфейсе провайдера

В описании интерфейса каждого провайдера содержится одна или несколько страниц, соответствующих возможным сценариям оплаты провайдера.

Для отправки платежа в Систему необходимо выполнить следующие шаги:

1. Собрать данные о платеже на страницах интерфейса провайдера и проверить их с помощью регулярных выражений, определенных в параметрах страницы.
2. Если в интерфейсе провайдера задана предварительная проверка данных платежа, отправить их на проверку и получить необходимые данные от провайдера (для отображения плательщику).

ПРИМЕЧАНИЕ



Онлайн-проверка данных платежа, указанных на страницах интерфейса провайдера, требуется, если тег соответствующего элемента page содержит атрибут useOnline="true".

3. Выполнить один из возможных сценариев проведения платежа: оффлайн или онлайн (см. раздел 4).

В данном разделе содержатся примеры построения интерфейса различных провайдеров по конфигурации, полученной в ответе на XML-запрос [getUIProviders](#).

Провайдер без онлайн-проверки

Одностраничный интерфейс простого провайдера приведен ниже (см. [Пример 7](#)).

Пример 7. Интерфейс оплаты с одной страницей

```
<provider id="83" fiscalName="Информационные услуги" ... tag="visible">
  <logos>
    <logo type="standard" path="logos!/anonymous/cian.gif" crc="CDCCCB26"
size="5207"/><logo type="charity" path="logos!/anonymous/prv2069help_promo.jpg"
crc="3dd0e81e" size="152799"/>
  </logos>
  <constParams/>
  <pages>
    <page pageId="1141" orderId="1">
      <controls>
        <control type="keyboard" orderId="2" layout="AL"/>
        <control type="text_input" orderId="1" strip="true" nobr="false"
regex="^[_a-z0-9]{3,25}$" header="Введите ваш идентификатор" mask="<!^[_a-z0-
9]+${3,25}>" name="account"/>
      </controls>
    </page>
  </pages>
</provider>
```

Интерфейс содержит только одну страницу с управляющим элементом:

- **Поле ввода** `<control type="text_input" orderId="1" strip="true" nobr="false" regex="^[_a-z0-9]{3,25}$" header="Введите ваш идентификатор" mask="<!^[_a-z0-9]+${3,25}>" name="account"/>`. Наиболее важные атрибуты:
 - Название поля – `account`.
 - Заголовок поля – "Введите ваш идентификатор".
 - Регулярное выражение `regex="^[_a-z0-9]{3,25}$"`, т.е. допустимы буквенно-цифровые значения (и символ подчеркивания) общей длиной от 3 до 25 символов.

Сценарий платежа состоит из двух этапов:

1. Ввод идентификатора персоны и проверка на соответствие регулярному выражению.
2. Отправка запроса на платеж после подтверждения данных платежа и внесения купюр.

Провайдер с онлайн-проверкой

Интерфейс провайдера с предварительной онлайн-проверкой данных платежа приведен ниже (см. [Пример 8](#)).

Пример 8. Интерфейс оплаты с онлайн-проверкой

```
<provider id="990" grpId="124" curId="643" jDocDate="25.09.2008"
supportPhone="8(495) 974-70-90, 8(800) 200-70-90" keywords="a m f" tag="visible" ...>
  <logos>
    <logo type="standard" path="logos/!anonymous/sendflowers.gif" crc="A4B59446"
size="2038"/> </logos>
  <constParams/>
  <pages>
    <page pageId="443" orderId="1" useOnline="true">
      <controls>
        <control type="text_input" orderId="1" regexp="^\d{6,10}$" header="Введите
номер заказа" mask="<!^\d+${6,10}>" name="account"/>
        <control type="keyboard" orderId="2" layout="DG"/>
      </controls>
    </page>
  </pages>
</provider>
```

Интерфейс содержит только одну страницу (теги page) с онлайн-проверкой (атрибут useOnline="true") и управляющим элементом:

- Поле ввода `<control type="text_input" orderId="1" regexp="^\d{6,10}$" header="Введите номер заказа" mask="<!^\d+${6,10}>" name="account"/>`.
Наиболее важные атрибуты:
 - Название поля – account.
 - Заголовок поля – "Введите номер заказа".
 - Регулярное выражение `regexp="^\d{6,10}$"`, т.е. допустимы цифровые значения общей длиной от 6 до 10 символов.

Сценарий платежа состоит из трех этапов:

1. Ввод номера заказа и проверка на соответствие регулярному выражению.
2. Отправка запроса онлайн-проверки собранных данных платежа (XML-запрос [checkPaymentRequisites](#)). Если проверка успешная, можно отправить запрос на платеж.
В ином случае, необходимо отобразить пользователю информацию об ошибке ("Указаны неправильные данные платежа").
3. В случае успешной проверки, выполняется отправка запроса на платеж после подтверждения данных платежа и внесения купюр.

Провайдер с несколькими страницами

Интерфейс провайдера, содержащий несколько страниц и дополнительные (экстра-) поля, приведен ниже (см. [Пример 9](#)).

Пример 9. Интерфейс оплаты с экстра-полями

```

<provider id="9998" grpId="113" ... buttonName="Skype" ... maxSum="15000"
keywords="Skype|skaip|skypescom|скайп">
  <logos>
    <logo type="standard" path="logos/5543810459565143056" crc="F4A8B0F6"
size="3687" />
  </logos>
  <constParams>
    <param name="_extra_sum" value="14999" />
  </constParams>
  <pages>
    <page pageId="53201" orderId="1">
      <controls>
        <control type="text_input" orderId="1" mask="<!^[a-zA-Z0-9, . _]+${6,32}>"
strip="false" header="Введите ваш логин Skype" footer="..." name="account-number"
disp_desc="Skype login" disp_type="extra,receipt" nobr="false" regexp="^[a-zA-Z][a-
zA-Z0-9-_,.]{5,31}$" />
        <control type="keyboard" orderId="2" layout="AL" />
      </controls>
    </page>
    <page pageId="53202" orderId="2" pageType="input_page">
      <controls>
        <control type="keyboard" orderId="2" layout="DG" />
        <control type="text_input" orderId="1" mask="(?!^\\d+${3}><!^d+${3}>-<!^
d+${2}>-<!^d+${2}>" strip="true" header="Введите номер телефона" footer="..."
name="account" nobr="false" regexp="^\\d{10}$" />
      </controls>
    </page>
  </pages>
</provider>

```

Интерфейс содержит две страницы (перечислены в порядке следования в интерфейсе согласно атрибуту `orderId` тега `page`):

1. Страница `pageID="53201"` с управляющими элементами:
 - поле ввода `<control type="text_input" ...>`. Наиболее важные атрибуты:
 - ⊕ название поля – `account-number`,
 - ⊕ заголовок поля – "Введите ваш логин Skype",
 - ⊕ регулярное выражение `regexp="^[a-zA-Z][a-zA-Z0-9-_,.]{5,31}$"`, т.е. допускается текст, содержащий одну букву, от 5 до 31 буквы/цифры и другие символы,
2. Страница `pageID="53202"` с управляющими элементами:
 - поле ввода `<control type="text_input" ...>`:
 - ⊕ название поля – `account`,
 - ⊕ заголовок поля – "Введите номер телефона",
 - ⊕ маска поля ввода номера телефона – соответствует строке "(три цифры)" " три цифры" – " две цифры" – " две цифры",
 - ⊕ регулярное выражение `regexp="^\\d{10}$"`, т.е. допускается текст, содержащий ровно 10 цифр,
 - ⊕ признак удаления статических символов из введенной строки перед проверкой по регулярному выражению `strip="true"`,

Сценарий платежа состоит из трех этапов:

1. Ввод логина на первой странице и проверка на соответствие регулярному выражению.

2. Ввод номера телефона на второй странице и проверка на соответствие регулярному выражению. Перед проверкой из номера удаляются статические символы маски ввода.
3. Отправка запроса на платеж после подтверждения данных платежа и внесения купюр. Запрос должен содержать два экстра-параметра в теге `<extras.../>`:
 - `account-number="value"`, где значение `value` введено в поле ввода на первой странице,
 - `sum="14999"` – постоянный экстра-параметр, определенный в теге `<constParams>`.

Провайдер с экстра-полями и онлайн-проверкой

Интерфейс провайдера с предварительной онлайн-проверкой данных платежа, содержащий дополнительные (экстра-) поля, приведен ниже (см. [Пример 10](#)).

Пример 10. Интерфейс оплаты с онлайн-проверкой и экстра-полями

```
<provider id="1084" grpId="124" curId="643" lName="JSC Faberlic" sName="Faberlic"
supportPhone="(495)384-46-33" keywords="Фаберлик" tag="visible">
<logos><logo type="standard" path="logos/!anonymous/faberlic.gif" crc="D67BA6E5"
size="1954"/></logos>
<constParams/>
<pages>
<page pageId="3223" orderId="1" useOnline="true">
<controls>
<control type="keyboard" orderId="2" layout="DG"/>
<control type="text_input" orderId="1" name="account"
mask="<!^\d+${1,15}>" regexp="^\d{1,15}$" header="Введите ваш регистрационный
номер для пополнения счета" strip="false" nobr="false"/>
</controls>
</page>
<page pageId="23135" orderId="2">
<controls>
<control type="disp_input" orderId="1" visible="true" mask="<!^\.+${1,35}>"
strip="false" header="Проверьте ваши данные:" disp_name="disp1" name="fio"
disp_type="extra" nobr="false" regexp="^\.{1,35}$"/>
</controls>
</page>
</pages>
</provider>
```

Интерфейс содержит две страницы (теги `page`):

1. Страница `pageID="3223"` с онлайн-проверкой (атрибут `useOnline="true"`) и элементами управления:
 - поле ввода `<control type="text_input" ...>`:
 - ⊕ название поля – `account-number`,
 - ⊕ заголовок поля – "Введите регистрационный номер для пополнения счета",
 - ⊕ регулярное выражение `regexp="^\d{1,15}$"`, т.е. допускается текст, содержащий от 1 до 15 цифр,
2. Страница `pageID="23135"` с управляющими элементами:
 - ассоциативное поле ввода `<control type="text_input" ...>` для отображения (не редактирования!) дополнительных параметров (см. [Табл. 10](#)):
 - ⊕ название поля – `fio`;
 - ⊕ ассоциативное поле – `disp1` (атрибут `disp_name="disp1"`). Это экстра-поле, полученное в [ответе](#) на XML-запрос онлайн-проверки платежа (в теге `<extras .../>` должен

присутствовать атрибут `disp1="value"`). Значение (`value`) должно отображаться с заголовком и комментарием, указанным в атрибутах поля;

- ⊕ заголовок поля – "Проверьте ваши данные";
- ⊕ регулярное выражение `regexp="^.{1,35}$"` (произвольные символы от 1 до 35) для проверки значения поля;

Сценарий платежа состоит из следующих этапов:

1. Ввод регистрационного номера и проверка на соответствие регулярному выражению.
2. Отправка запроса онлайн-проверки собранных данных платежа (XML-запрос [checkPaymentRequisites](#)).
3. Отображение результата проверки:
 - 3.1. Если проверка успешная, то на специальной странице подтверждения отображается полученное в ответе на запрос ассоциативное экстра-поле.
 - 3.2. Если проверка неуспешная, то отображается стандартная страница ошибки ("Указаны неправильные данные").
4. В случае успешной проверки, выполняется отправка запроса на платеж после подтверждения данных платежа и внесения купюр. Запрос должен содержать экстра-поле в теге `<extras.../>`:
 - `fio="disp_value"`, где значение `disp_value` получено на шаге 3. Перед отправкой значение проверяется на соответствие регулярному выражению.

Провайдер с ветвлением сценария

Интерфейс провайдера, содержащий несколько страниц, множественные онлайн-проверки и условия перехода между страницами (сценарий с ветвлением), приведен ниже (см. [Пример 11](#)).

Пример 11. Интерфейс оплаты с экстра-полями на нескольких страницах и ветвлением

```
<provider id="330" fiscalName="МосЭнепро" grpId="382" curId="643" inn="3123011520"
jDocDate="18.05.2009" jDocNum="26-34-2/53/15-09"...>
  <logos>
    <logo type="standard" path="logos/!anonymous/prv265_d7158bbd.gif" crc="D7158BBD"
size="4294"/>
  </logos>
  <constParams/>
  <pages>
    <page pageId="86258" orderId="2" title="Проверьте правильность данных
плательщика" >
      <controls>
        <control type="disp_input" orderId="1" visible="true"
mask="&lt;!^.+${1,150}&gt;" strip="false" header="Адрес плательщика"
disp_name="disp2" name="disp2" disp_type="extra" nobr="false" regexp="^.{1,150}$"
/>
        <control type="disp_input" orderId="2" visible="true"
mask="&lt;!^.+${1,10}&gt;" strip="false" header="Текущий баланс" disp_name="disp1"
name="disp1" disp_type="extra" nobr="false" regexp="^.{10}$" />
      </controls>
    </page>
    <page pageId="86259" orderId="1" successPage="86258" useOnline="true" >
      <controls>
        <control type="text_input" orderId="1" mask="&lt;!^\\d+${5}&gt;-
&lt;!^\\d+${3}&gt;-&lt;!^\\d+${2}&gt;" strip="true" header="Введите номер лицевого
счета" name="account" nobr="false" regexp="^\\d{10}$" disp_type="extra,receipt" />
        <control type="keyboard" orderId="2" layout="DG" />
      </controls>
    </page>
    <page pageId="86260" orderId="3" title="Выберите действие" >
      <controls>
```

```

        <control type="button" orderId="2" altName="Ввод показаний" disp_desc="Ввод
показаний возможен только в период с 15 по 26 число каждого месяца" pageId="86262"
/>
        <control type="button" orderId="1" altName="Оплата" pageId="86261" />
    </controls>
</page>
<page pageId="86261" orderId="4" nextPage="-1" >
    <controls>
        <control type="text_input" orderId="1"
mask="&lt;!^\d+${2}&gt;.&lt;!^\d+${4}&gt;" strip="false" header="Укажите период
оплаты " footer="Введите период оплаты в формате ММ.ГГГГ" name="date"
disp_type="extra,receipt" nobr="false" regexp="^\d{2}\.\d{4}$" />
    </controls>
</page>
<page pageId="86262" orderId="5" title="Введите показания счетчиков"
useOnline="true" >
    <controls>
        <control type="text_input" orderId="1" strip="false" header="Т1 (Пик)"
name="t1" nobr="false" regexp="^\d{1,6}$" mask="&lt;!^\d+${1,6}&gt;"
disp_type="extra,receipt" />
        <control type="text_input" orderId="2" strip="false" header="Т2 (Ночь)"
name="t2" nobr="false" regexp="^\d{0,6}$" mask="&lt;!^\d+${0,6}&gt;"
disp_type="extra,receipt" />
        <control type="text_input" orderId="3" strip="false" header="Т3 (Полупик)"
name="t3" nobr="false" regexp="^\d{0,6}$" mask="&lt;!^\d+${0,6}&gt;"
disp_type="extra,receipt" />
        <control type="button" orderId="4" altName="Далее" pageId="86263"
disp_type="extra" >
            <param name="REQUEST" value="PokCheckRequest" />
        </control>
    </controls>
</page>
<page pageId="86263" orderId="6" nextPage="-1" >
    <controls>
        <control type="disp_input" orderId="1" nobr="false" regexp="^{1,10}$"
visible="true" mask="&lt;!^\.+${1,10}&gt;" strip="false" header="Баланс с учетом
введенных показаний" disp_name="advSum " name="advSum " />
    </controls>
</page>
</pages>
</provider>

```

Интерфейс содержит шесть страниц (перечислены в порядке следования в интерфейсе согласно атрибуту `orderId` тега `page`):

1. Страница `pageID="86259"` с онлайн-проверкой (атрибут `useOnline="true"`) и управляющим элементом "поле ввода" `<control type="text_input" ...>`. Наиболее важные атрибуты:
 - название поля – `account`;
 - заголовок поля – "Введите номер лицевого счета";
 - регулярное выражение `regexp="^\d{10}$"`, т.е. строка из 10 цифр;
2. Страница `pageID="86258"` (проверка данных пользователем) с управляющими элементами:
 - ассоциативное поле ввода `<control type="disp_input" ...>` для отображения (не редактирования!) дополнительных параметров (см. [Табл. 10](#)):
 - ⊕ название поля – `disp1`;
 - ⊕ ассоциативное поле – `disp1` (атрибут `disp_name="disp1"`). Это экстр-поле, полученное в [ответе](#) на XML-запрос онлайн-проверки платежа (в теге `<extras.../>` должен присутствовать атрибут `disp1="value"`). Значение `value` должно отображаться с заголовком и комментарием, указанным в атрибутах поля;

- ⊕ заголовок поля – "Текущий баланс ";
- ⊕ регулярное выражение `regexp="^.{10}$"` (произвольные символы, длина строки от 1 до 35) для проверки значения поля;
- ассоциативное поле ввода `<control type="disp_input" ...>` для отображения (не редактирования!) дополнительных параметров (см. [Табл. 10](#)):
 - ⊕ название поля – `disp2`;
 - ⊕ ассоциативное поле – `disp2` (атрибут `disp_name="disp2"`). Это экстра-поле, полученное в [ответе](#) на XML-запрос онлайн-проверки платежа (в теге `<extras.../>` должен присутствовать атрибут `disp2="value"`). Значение `value` должно отображаться с заголовком и комментарием, указанным в атрибутах поля;
 - ⊕ заголовок поля – "Адрес плательщика ";
 - ⊕ регулярное выражение `regexp="^{1,150}$"` (произвольные символы, длина строки от 1 до 150) для проверки значения поля;
- 3. Страница `pageID="86260"` (выбор действия пользователем, ветвление страниц) с управляющими элементами:
 - кнопка `<control type="button" ...>` (два элемента). Каждый элемент содержит признак выбора действия (явно указано в атрибуте `altName`). В зависимости от выбора действия, выбирается следующий блок с параметрами по атрибуту `pageId` в соответствующем теге `<control ... type="button">`:
 - ⊕ `pageId="86261"` – для действия "Оплата";
 - ⊕ `pageId="86262"` – для действия "Ввод показаний";
- 4. Страница `pageID="86261"` (завершающая страница для действия "Оплата", признак `nextPage="-1"`) с управляющим элементом "поле ввода" `<control type="text_input" ...>`:
 - название поля – `date`;
 - признак экстра-поля - `disp_type="extra"`;
 - заголовок поля – "Укажите период оплаты";
 - комментарий к полю в footer;
 - регулярное выражение `regexp="^\d{2}\.\d{4}$"`, т.е. допускается формат периода ММ.ГГГГ;
- 5. Страница `pageID="86262"` (страница для действия "Ввод показаний") с онлайн-проверкой (атрибут `useOnline="true"`) и управляющими элементами:
 - "поле ввода" `<control type="text_input" ...>`:
 - ⊕ название поля – `t1`;
 - ⊕ признак экстра-поля - `disp_type="extra"`;
 - ⊕ заголовок поля – "Т1(Пик)";
 - ⊕ регулярное выражение `regexp="^\d{1,6}$"`, т.е. допускается число длиной от 1 до 6 цифр;
 - "поле ввода" `<control type="text_input" ...>`:
 - ⊕ название поля – `t2`;
 - ⊕ признак экстра-поля - `disp_type="extra"`;
 - ⊕ заголовок поля – "Т2(Ночь)";
 - ⊕ регулярное выражение `regexp="^\d{0,6}$"`, т.е. допускается число длиной от 1 до 6 цифр, или незаполненное поле;

- "поле ввода" `<control type="text_input" ...>`:
 - ⊕ название поля – `t3`;
 - ⊕ признак экстра-поля - `disp_type="extra"`;
 - ⊕ заголовок поля – "ТЗ(Полупик)";
 - ⊕ регулярное выражение `regexp="^\d{0,6}$"`, т.е. допускается число длиной от 1 до 6 цифр, или незаполненное поле;
 - "кнопка" `<control type="button" ...>` для перехода на следующую страницу (см. атрибут `pageId="86263"`) и параметр, передаваемый как экстра-поле (признак экстра-поля - атрибут `disp_type="extra"`):
 - ⊕ название параметра – `REQUEST`;
 - ⊕ значение параметра – `"PokCheckRequest"`;
6. Страница `pageID="86263"` с отображением результата онлайн-проверки введенных показаний (баланс с учетом новых показаний, для информирования пользователя):
- "поле ввода" `<control type="disp_input" ...>`:
 - ⊕ название экстра-поля – `disp_name="advSum"`. Это экстра-поле, полученное в [ответе](#) на XML-запрос онлайн-проверки платежа (в теге `<extras.../>` должен присутствовать атрибут `advSum="value"`). Значение `value` должно отображаться с заголовком и комментарием, указанным в атрибутах поля.

Сценарий платежа состоит из нескольких этапов:

1. Ввод номера счета на первой странице и проверка на соответствие регулярному выражению.
2. Отправка запроса онлайн-проверки номера счета (XML-запрос [checkPaymentRequisites](#)). Если проверка успешная, можно перейти к следующему шагу сценария (п.3). В ином случае, необходимо отобразить пользователю информацию об ошибке ("Указаны неправильные данные платежа").
3. Отображение данных пользователя (баланс, адрес) и подтверждение данных.
4. Выбор действия – ввод показаний (переход к п.6) или оплата на произвольную сумму (переход к п.5).
5. Выбор периода оплаты и переход к внесению денег. Далее отправляется запрос на платеж после внесения купюр. Запрос должен содержать два экстра-параметра в теге `<extras.../>`:
 - `date="MM.PPPP"`, где значение – выбранный период,
 - `sum="14999"` – постоянный экстра-параметр, определенный в теге `<constParams>`.
6. Ввод показаний счетчика и проверка на соответствие регулярному выражению.
7. Отправка запроса онлайн-проверки показаний (XML-запрос [checkPaymentRequisites](#)). Отображение пользователю информации о балансе (значение экстра-поля `advSum` в ответе на запрос).
8. Отправка запроса на платеж после внесения денег. Запрос должен содержать следующие экстра-параметры в теге `<extras.../>`:
 - `REQUEST="PokCheckRequest"`,
 - `t1="value"`, где значение `value` указано пользователем в поле ввода (п.6),

Следующие два значения могут отсутствовать:

 - `t2="value"`, где значение `value` указано пользователем в поле ввода (п.6),
 - `t3="value"`, где значение `value` указано пользователем в поле ввода (п.6).

5.3. Алгоритм расчета терминальной комиссии

В Системе для каждого провайдера может быть установлена *фиксированная ставка комиссии* (в процентах от суммы платежа). Кроме этого, существует возможность установить различные правила взимания комиссии в зависимости от времени суток, суммы платежа и других параметров, с помощью *Профиля комиссии* – набора правил вычисления суммы комиссии. Совокупность правил взимания комиссии (ставок и профилей комиссии) составляет *Шаблон комиссии*.

5.3.1. Установка терминальной комиссии

Фиксированные ставки и профили комиссии можно установить в Личном кабинете на сайте agent.qiwi.com в справочнике «Шаблоны комиссии». Также для установки терминальной комиссии можно использовать XML-запросы:

- [setOuterCommission](#) - установка комиссии или профиля комиссии на провайдера на терминале;
- [setCommissionProfile](#) – изменение профиля комиссии;
- [setCommissionTemplate](#) – изменение шаблона комиссии.

5.3.2. Расчет терминальной комиссии

Для получения комиссий на терминале используются XML-запросы:

- [getCommissions](#) - запрос комиссий и профилей комиссии, установленных на терминале;
- [getCommissionProfiles](#) - запрос параметров профилей комиссии;
- [getTerminalCommissionTemplates](#) - запрос списка шаблонов комиссии на терминале;
- [getCommissionTemplates](#) - запрос параметров шаблонов комиссии агента.

ПРИМЕЧАНИЕ



Если для провайдера задан и профиль, и фиксированная ставка комиссии, то профиль имеет более высокий приоритет.

В случае если несколько правил профиля удовлетворяют условиям, комиссия будет рассчитана в соответствии с правилом, имеющим меньший порядковый номер.

Если не может быть применен ни один из профилей, будет использоваться фиксированная ставка комиссии.

Комиссия с платежа взимается согласно установленным ставке или профилю комиссии при отправке платежа с терминала (см. раздел [Прием платежей](#)). Сумма комиссии должна быть равна разнице суммы, принятой от плательщика (указана в атрибуте `amount` тега `from` платежных запросов), и суммы к зачислению (указана в атрибуте `amount` тега `to` платежных запросов).

ВНИМАНИЕ



Платеж на провайдера, по которому запрещено взимание комиссии, будет проведен на сумму, указанную в атрибуте `amount` тега `from` платежного запроса.

Комиссия с платежа на провайдера, по которому установлена максимальная комиссия, не может превышать этой величины. В ином случае сумма к зачислению (в атрибуте `amount` тега `to`) будет принудительно установлена как разница суммы от плательщика (в атрибуте `amount` тега `from`) и максимальной комиссии.

Пример 12. Расчет комиссии

Пример расчета комиссии при следующих параметрах:

1. Фиксированная ставка комиссии для провайдера: 2,5%.
2. Профиль комиссии:

Правило	Условия	Комиссия %	Абсолютное значение комиссии	Не меньше
1	Сумма платежа < 500 руб	3% от суммы платежа	10 руб	20 руб
2	Сумма < 500 руб Время с 06:00 до 16:00		7 руб	

Платеж на сумму 400 руб. принят в 15:00. В этом случае несколько правил профиля удовлетворяют условиям платежа. Комиссия будет рассчитана в соответствии с правилом, имеющим меньший порядковый номер – в данном случае правило №1.

Так как может быть применен профиль комиссии, то фиксированная ставка комиссии не применяется (2,5% от суммы платежа).

Результат: при платеже на сумму 400 руб. комиссия составит $400 \text{ руб.} \cdot 3\% + 10 \text{ руб.} = 22 \text{ руб.}$

Платежный запрос выглядит следующим образом:

```
<?xml version="1.0" encoding="utf-8"?>
<request>
<auth login="login" sign="sign" signAlg="MD5"/>
<client terminal="111" software="Dealer v0" serial=""/>
<providers>
<addOfflinePayment>
<payment id="701">
<from currency="643" amount="400"/>
<to currency="643" service="3" amount="378" account="9261111111"/>
<receipt id="1" date="2013-08-16T10:38:19"/>
</payment>
</addOfflinePayment>
</providers>
</request>
```

6. ИНТЕРФЕЙСЫ И ДЕЙСТВИЯ

6.1. System

Список действий интерфейса:

Действие	Описание
sendMail	Отправка электронного сообщения на указанный адрес электронной почты
getReferencesVersions	Получение версий справочников
getCurrencyRates	Получение курсов валют по отношению к рублю (для мультивалютных терминалов)
getNews	Получение новостей с сайта КИВИ
getActions	Получение списка действий всех интерфейсов, на выполнение которых у данного пользователя есть права
getResultCodes	Получение справочника кодов завершения
Ошибка! Источник ссылки не найден.	Проверка необходимости произвести идентификацию пользователя.
Ошибка! Источник ссылки не найден.	Установка первичных идентификационных данных.
getForbiddenBanknotes	Получение списка запрещенных номиналов банкнот

ПРИМЕЧАНИЕ



Обязательные параметры запросов отмечены символом *.

6.1.1. sendMail

Описание: Отправление письма на указанный адрес e-mail с заданным текстом и присоединенными файлами.

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.1.1.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <system>
    <sendMail>
```

```

<message address="adress@domen.ru" subject="mail subject">
  <text>text message</text>
  <attachments>
    <attachment name="file.log">0000000000000000000000000000</attachment>
  </attachments>
</message>
</sendMail>
</system>
</request>

```

Для выполнения действия следует задать дополнительные параметры:

- `message` – тег, содержащий информацию о сообщении. Включает атрибуты:
 - `address*` – адрес электронной почты, на который должно быть отправлено сообщение,
 - `subject*` – тема сообщения,
- `text*` – тег, содержащий текст сообщения письма;
- `attachments` – тег, содержащий прикрепляемые к сообщению файлы. Состоит из тегов:
 - `attachment` – содержимое прикрепляемого файла, закодированное с использованием схемы **Base64**. Атрибут тега:

ВНИМАНИЕ



Длина закодированного сообщения не должна превышать 640 килобайт.

⊕ `name` – имя прикрепляемого файла.

6.1.1.2. Ответ

```

<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <system>
    <sendMail result="0"/>
  </system>
</response>

```

Ответ содержит информацию об успешности запроса (см. [Приложение В](#)).

6.1.2. `getReferencesVersions`

Описание: Запрос позволяет получить текущие версии версионных справочников (номерные емкости и провайдеры), а также номер последней новости на сайте КИВИ (по состоянию на момент запроса).

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.1.2.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <system>
    <getReferencesVersions/>
  </system>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.1.2.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <system>
    <getReferencesVersions result="0">
      <phone-ranges>1234568</phone-ranges>
      <providers>11111</providers>
      <news>546429</news>
    </getReferencesVersions>
  </system>
</response>
```

Ответ содержит следующие данные:

- `phone-ranges` – версия справочника номерных емкостей,
- `providers` – версия справочника провайдеров,
- `news` – номер последней новости на сайте КИВИ (по состоянию на момент запроса).

ВНИМАНИЕ



Если новости отсутствуют, в теге `news` передается значение 0.

6.1.3. `getCurrencyRates`

Описание: Запрос позволяет получить курсы USD и EUR по отношению к рублю для **мультивалютных** терминалов.

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.1.3.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <system>
    <getCurrencyRates/>
  </system>
```

```
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.1.3.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <system>
    <getCurrencyRates result="0">
      <usd>24.706</usd>
      <eur>35.6038</eur>
    </getCurrencyRates>
  </system>
</response>
```

Ответ содержит следующие данные:

- usd – курс USD,
- eur – курс EUR.

6.1.4. getNews

Описание: Запрос позволяет получить новости за последние две недели (но не более 10 сообщений) с сайта КИВИ.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.1.4.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <system>
    <getNews/>
  </system>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- id-from – идентификатор сообщения, с номера которого необходимо получить все последующие новости;
- id-to – идентификатор последней новости. Будут получены все новости, опубликованные до указанной;
- date-from – будут получены все новости, опубликованные после указанной в теге даты (формат: «ГГГГ.ММ.ДД»);
- date-to – будут получены все новости, опубликованные до указанной в теге даты (формат: «ГГГГ.ММ.ДД»);
- cutTags – признак формата текста сообщения в новостях:
 - true – получить текст новости без HTML-тегов (сокращенный объем ответа);

- false (или параметр не указан) – включать теги HTML в текст новостей.

6.1.4.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <system>
    <getNews result="0">
      <row news_anons="Приём платежей Мегафон Сибирь восстановлен."
news_cat="МегаФон Сибирь" news_date="2009-01-26T09:49:19" news_id="542379"
news_text=""/>
      <row news_anons="Приём платежей Уралсвязьинформ восстановлен."
news_cat="Уралсвязьинформ" news_date="2009-01-26T09:47:06" news_id="542378"
news_text=""/>
      ...
    </getNews>
  </system>
</response>
```

Ответ содержит следующие данные:

- row – тег, содержащий информацию о новости. Включает атрибуты:
 - news_anons – текст анонса новости,
 - news_cat – категория, к которой относится новость,
 - news_date – дата публикации новости,
 - news_id – идентификатор новости,
 - news_text – полный текст сообщения.

6.1.5. getActions

Описание: Запрос позволяет получить список [действий](#) для всех интерфейсов, на которые есть права у персоны, от имени которой отправляется данный запрос.

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.1.5.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <system>
    <getActions/>
  </system>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.1.5.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <system>
    <getActions result="0">
      <row description="Баланс агента" execution="Возможно асинхронно"
interface="agents" name="getBalance"/>
      <row description="Описание агента" execution="Возможно асинхронно"
interface="agents" name="getAgentInfo"/>
      ...
    </getActions>
  </system>
</response>
```

Ответ содержит следующие данные:

- row – тег, содержащий информацию о действии. Включает атрибуты:
 - description – текстовое описание действия,
 - execution – способ выполнения запроса (синхронно или асинхронно),
 - interface – интерфейс, к которому относится действие,
 - name – название действия.

6.1.6. getResultCodes

Описание: Запрос позволяет получить актуальный список кодов ошибок.

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.1.6.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <system>
    <getResultCodes/>
  </system>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.1.6.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <system>
    <getResultCodes result="0">
      <row err_id="0" err_text="ОК"/>
      <row err_id="1" err_text="Провайдер временно недоступен"/>
    </getResultCodes>
  </system>
</response>
```

```
    ...  
  </getResultCodes>  
</system>  
</response>
```

Ответ содержит следующие данные:

- `row` – тег, содержащий информацию об ошибке. Включает атрибуты:
 - `err_id` – идентификатор (код) ошибки,
 - `err_text` – текстовое описание ошибки.

6.1.7. getForbiddenBanknotes

Описание: Получение списка запрещенных на терминале номиналов банкнот

Режим обработки: Строго синхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.1.7.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>  
<request>  
  <auth login="login" sign="sign" signAlg="MD5"/>  
  <client terminal="111" software="ATL v4.06" serial=""/>  
  <system>  
    <getForbiddenBanknotes/>  
  </system>  
</request>
```

Запрос отправляется без параметров.

6.1.7.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>  
<response result="0">  
  <system>  
    <getForbiddenBanknotes result="0" version="80843">  
      <row currency-id="643" nominal-id="14" value="5000"/>  
      <row currency-id="643" nominal-id="11" value="100"/>  
    </getForbiddenBanknotes>  
  </system>  
</response>
```

Ответ содержит код ошибки, указывающий на результат выполнения запроса (см. [Приложение В](#)), версию данных запроса, а также тег `row`, в атрибутах которого передаются следующие данные:

- `currency-id` – идентификатор валюты в формате **ISO 4217**;
- `nominal-id` – идентификатор номинала банкноты;
- `value` – номинал банкноты.

6.2. Terminals

ПРИМЕЧАНИЕ



Обязательные параметры запросов отмечены символом *.

Список действий интерфейса:

Действие	Описание
checkMessages	Проверка новых сообщений на терминале
getMessages	Получение новых сообщений на терминал
getPlaylist	Загрузка плейлиста на АСО. Подробнее о размещении рекламы на АСО см. в документе «Интерфейс размещения рекламы на АСО»
getLastIds	Получение последнего номера транзакции и номера чека для данного терминала
getRemainingLimit	Получение остатка лимита платежей для данного терминала
setState	Передача информации о состоянии автомата самообслуживания на сервер
resetReceiptNumber	Сброс счетчика номеров чеков
setFiscalMode	Установка фискального режима для терминала
addCashout	Отправка на сервер информации об инкассации
setHosts	Установка адреса сервера (списка адресов) для терминала
getTerminalInfo	Получение подробной информации о терминале
getTerminalRegionHierarchy	Получение иерархии объектов КЛАДР для адреса терминала
getTerminals	Получение списка терминалов
getTerminalsWithCoordinates	Получение списка терминалов с координатами местоположения
setConfigId	Изменение идентификатора конфигурации терминала
getConfigId	Получение идентификатора конфигурации терминала
getConfig	Получение конфигурации терминала
getCommissions	Получение списка терминальных комиссий и профилей комиссий, установленных для различных провайдеров
getCommissionProfiles	Получение профилей комиссий
getCommissionsHistory	Получение истории комиссии по терминалу
getTerminalCommissionTemplates	Получение шаблонов комиссий терминала
bindTerminalTemplate	Привязка шаблона комиссии к терминалу
getForeignCommissions	Получение информационного профиля комиссии по провайдеру (например, комиссии банка)

Действие	Описание
getTerminalTypes	Получение справочника типов терминалов
addShiftClosure	Закрытие смены на терминале
requestLog	Получение лог-файла терминала
requestProcessList	Получение списка процессов, выполняемых на терминале
requestZReport	Получение снятого с терминала Z-отчета
restartClientSoftware	Перезагрузка клиентского ПО АСО (<i>maratl.exe</i>)
rebootTerminal	Перезагрузка терминала
disableTerminal	Выключение терминала

ВНИМАНИЕ

По умолчанию все действия выполняются для *текущего терминала/ текущего агента*.

Текущим терминалом считается терминал, идентификатор которого указан в теге <client>.

Текущим агентом считается агент, которому принадлежит терминал, указанный в теге <client> и персоне, данные которой указаны в теге <auth>.

Если требуется получить информацию по другому терминалу (агенту), необходимо указать идентификаторы терминала (или персоны) в теге <target-terminal> (<target-agent>) внутри тега действия (см. пример запроса).

6.2.1. checkMessages

Описание: Запрос позволяет проверить наличие новых сообщений для терминала.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.1.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <checkMessages>
      <target-terminal>123</target-terminal>
      <last-msg-id>51805112</last-msg-id>
    </checkMessages>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-terminal` – терминал, для которого необходимо проверить наличие новых сообщений (если не указан, выполняется для *текущего терминала*),

- `last-msg-id` – идентификатор последнего сообщения, принятого терминалом.

6.2.1.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <checkMessages result="0">
      <new-msg-count>0</new-msg-count>
    </checkMessages>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `new-msg-count` – число новых сообщений.

6.2.2. `getMessages`

Описание: Запрос позволяет получить новые сообщения, не полученные терминалом.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.2.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getMessages>
      <target-terminal>123</target-terminal>
    </getMessages>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-terminal` – терминал, для которого необходимо получить список новых сообщений (если не указан, выполняется для *текущего* терминала).

6.2.2.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getMessages result="0">
      <row msg_created="2010-11-12T17:52:43+03:00" msg_dst_trm_id="111"
msg_id="51805112" msg_lifetime="3600" msg_src_prs_id="123456789"
msg_src_trm_id="0">
        <head msg-type="prv-msg"/>
        <body>
          <text prv-id="0">Текст отправки сообщения</text>
        </body>
      </row>
    </getMessages>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `row` – тег содержит информацию об одном сообщении. Включает атрибуты:
 - `msg_created` – дата/время создания сообщения;
 - `msg_dst_trm_id` – идентификатор терминала, на который было отправлено сообщение;
 - `msg_id` – идентификатор сообщения;
 - `msg_lifetime` – время жизни сообщения (в секундах). Если значение «0» или не указано, то время жизни сообщения неограничено;
 - `msg_src_prs_id` – идентификатор персоны, отправившей сообщение;
 - `msg_src_trm_id` – идентификатор терминала, отправившего сообщение;
 - `head` – описывает раздел HEAD сообщения. Включает атрибут:
 - ⊕ `msg-type` – тип сообщения. Может принимать следующие значения:
 - ❖ `"prv-msg"` - Позволяет поместить текст на экране терминала (на главной странице или на странице указанного провайдера). См. [Пример 13](#);
 - ❖ `"get-log"` - Позволяет получить на указанный email журнал состояния терминала за указанную дату и интервал времени. См. [Пример 14](#);
 - ❖ `"get-pslist"` - Позволяет получить на указанный email список запущенных процессов со списком драйверов, их версий, производителей, подгруженных библиотек, а также подобную информацию об активных сервисах. См. [Пример 15](#);
 - ❖ `"trm-cmd"` - Служебные сообщения, которые позволяют управлять состоянием терминала. См. [Пример 16](#);
 - ❖ `"del-msg"` – Позволяет удалить ранее полученное сообщение, размещенное на экране терминала (т.е. только сообщение с типом "prv-msg"). См. [Пример 17](#);
 - ❖ `"popup-msg"` – Позволяет получить новостное всплывающее сообщение (pop-up) в КИВИ Кассир. См. [Пример 18](#);
 - `body` – описывает раздел BODY сообщения. Может включать различные параметры в зависимости от типа сообщения в теге `head`:
 - ⊕ если тип сообщения `"prv-msg"` – единственный параметр:
 - ❖ `text` – текстовая строка, которая отображается на экране терминала. Атрибут:
 - `prv-id` – идентификатор провайдера, на странице которого отображается сообщение, или «0» – для размещения сообщения на главной странице.

- ⊕ если тип сообщения "get-log" – параметры:
 - ❖ reply-to – адрес электронной почты получателя;
 - ❖ date – дата, за которую необходимо получить журнал;
 - ❖ hour – тег, задающий промежуток времени, за который необходимо получить журнал:
 - from – с какого часа суток;
 - to – по какой час суток.
- ⊕ если тип сообщения "get-pslist" – единственный параметр:
 - ❖ reply-to – адрес электронной почты получателя;
- ⊕ если тип сообщения "trm-cmd" – единственный параметр:
 - ❖ cmd – команда для терминала. Может быть одной из следующих:
 - restart – перезапуск ПО терминала,
 - reboot – перезагрузка терминала,
 - shutdown – завершение работы терминала.
- ⊕ если тип сообщения "del-msg" – единственный параметр:
 - ❖ msg_id – идентификатор удаляемого сообщения;
- ⊕ если тип сообщения "popup-msg" – параметры:
 - ❖ text – текст сообщения;
 - ❖ number – количество авторизаций в КИВИ Кассире, в течение которых пользователь будет получать сообщение.

Пример 13. Отправленное на терминал текстовое сообщение.

```
<head msg-type="prv-msg"/>
<body>
  <text prv-id="2">Тестовое сообщение на провайдера Билайн</text>
</body>
```

Пример 14. Сообщение для получения журнала работы терминала за указанный период.

```
<head msg-type="get-log"/>
<body>
  <reply-to>email</reply-to>
  <date>20130602</date>
  <hour from="00" to="24"/>
</body>
```

Пример 15. Сообщение для получения списка запущенных процессов.

```
<head msg-type="get-pslist"/>
<body>
  <reply-to>email</reply-to>
</body>
```

Пример 16. Служебное сообщение, отправленное на терминал

```
<head msg-type="trm-cmd"/>
<body>
  <cmd>restart</cmd>
</body>
```

Пример 17. Удаление ранее принятого текстового сообщения

```
<head msg-type="del-msg"/>
<body>
  <msg_id>51805112</msg_id>
</body>
```

Пример 18. Получение новостного сообщения в КИВИ Кассир

```
<head msg-type="popup-msg"/>
<body>
  <text>Please, update Qiwi Cashier. Тестовое сообщение. </text>
  <number>10</number>
</body>
```

6.2.3. getPlaylist

Описание: Запрос позволяет получить на АСО рекламный плейлист.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.3.1. Загрузка рекламных плейлистов

В *Системе* реализована возможность размещения рекламы на экране АСО.

Реклама представляет собой «плавающие» или статичные баннеры, расположенные в определенных местах дисплея автомата – рекламных зонах (список рекламных зон определен в [Правилах работы системы КИВИ](#)).

ПРИМЕЧАНИЕ



Размер баннера определяется рекламной зоной.

При нажатии на баннер открывается страница рекламы. Подробнее о типах рекламных баннеров см. далее.

Рекламные данные могут представлять собой текст, изображения, java-скрипты и др. Полная совокупность рекламных данных, размещенных на АСО, называется *плейлистом* (см. [Глоссарий](#)).

Типы рекламных баннеров

Возможные типы рекламных баннеров:

- баннеры без активной ссылки – представляют собой статичную картинку,
- баннеры с активной ссылкой – представляют собой изображение, щелчок на котором вызывает какое-то действие (например, открытие окна, содержащего информацию об услугах),
- всплывающие страницы – представляют собой мини-сайты, содержащие 1-2 информационные страницы,
- «плавающий» баннер – может быть размещен только на главной странице,
- «ротирующиеся» баннеры – баннеры, расположенные в одной и той же рекламной зоне в один и тот же промежуток времени. Отображение таких баннеров на странице терминала чередуются.

Кроме того, по типу содержащейся в них информации рекламные баннеры можно разделить на:

- **контекстные** – размещаются на страницах определенного провайдера и содержат контекстную информацию об услугах или акциях этого провайдера,
- **общие** – могут размещаться на главной странице, странице выбора услуг и странице выбора провайдера и содержать информацию общего характера.

Формат файлов рекламы

В зависимости от типа баннеров файлы рекламы могут представлять собой:

- изображения (*.gif, *.jpeg);
- java-скрипты (*.js);
- flash (*.swf);
- текст (*.html, *.txt).

6.2.3.2. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getPlaylist/>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-terminal` – терминал, для которого необходимо получить рекламный плейлист (если не указан, выполняется для *текущего* терминала).

6.2.3.3. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getPlaylist result="0">
      <playlist terminal="1111111" ftp_home="interface_magitel" version="174363_3">
        <page id="index_page_2">
          <range stopDate="28022011" startDate="01102010">
            <content type="script" value="megafon10122010_0.js" fileCRC="976A832E"
              fileLength="1932" prv_id="1,2,3,14" />
          </range>
          ...
        </page>
        <page id="receipt_header_text">
          <range startDate="05122007" stopDate="12012008">
            <content fileCRC="C15211111" fileLength="74" type="image"
              value="Подробности акции по телефону (495) 661-49-19 или на www.arctel.ru" />
          </range>
        </page>
      </playlist>
    </getPlaylist>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `playlist` – основной тег плейлиста. Включает атрибуты:
 - `version` – версия используемого плейлиста,
 - `ftp_home` – папка рекламодателя на FTP-сервере КИВИ,
 - `terminal` – идентификатор терминала;
- `page` – тег, задающий размещение рекламного контента на указанной рекламной зоне. Включает атрибут:
 - `id` – идентификатор рекламной зоны,
- `range` – период действия рекламы:
 - `startDate` – дата начала показа,
 - `stopDate` – дата окончания показа,
- `content` – тег рекламного контента. Включает атрибуты:
 - `fileCRC` – циклический избыточный код (CRC) передаваемого файла. CRC используется для проверки правильности переданных данных,
 - `fileLength` – размер файла контента в байтах,
 - `type` – тип файла контента,
 - `value` – имя файла контента,
 - `prv_id` – список идентификаторов провайдеров, на страницах которых должна быть размещена реклама (в случае если рекламная информация должна размещаться на страницах только определенных провайдеров).

6.2.4. getLastIds

Описание: Получение номера последней транзакции и номера чека, поступивших с данного терминала.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.4.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getLastIds/>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-terminal` – терминал, для которого необходимо получить номера транзакции и чека (если не указан, выполняется для *текущего* терминала).

6.2.4.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getLastIds result="0">
      <last-payment id="12345678" receipt-number="4321">
    </getLastIds>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `id` – идентификатор последнего платежа, полученного с терминала;
- `receipt-number` – последний номер чека.

6.2.5. getRemainingLimit

Описание: Получение остатка лимита платежей для данного терминала.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.5.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getRemainingLimit/>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-terminal` – терминал, для которого необходимо получить значение остатка лимита (если не указан, выполняется для *текущего* терминала).

6.2.5.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getRemainingLimit result="0">
      <remaining-limit>150000.00</remaining-limit>
    </getRemainingLimit>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `remaining-limit` – величина остатка лимита терминала (тег отсутствует, если для терминала не определен лимит).

6.2.6. setState

Описание: Пинг для автоматов самообслуживания. Передаваемые теги запроса содержат значения технических параметров оборудования АСО. Данные в тегах должны следовать в строго определенном порядке через разделитель «;» (точка с запятой).

Режим обработки: Строго синхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.6.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <setState>
      <fiscal-info></fiscal-info>
      <status>12;34.00;34;34.00;43;32;88.88;640x480;playlist;CRC;333</status>
      <watchdog></watchdog>
      <hardware></hardware>
      <statistics></statistics>
      <gsm-status>101.0,60,250996610655548</gsm-status>
      <software></software>
      <dispenser>
        <info model = "" serial = "" error = "" recycler = "">
          <cassette></cassette>
          ...
        </info>
        <incass>
          <cassette></cassette>
          ...
        <reject>
          <cassette></cassette>
          ...
        </reject>
      </incass>
      <addcash>
        <cassette></cassette>
        ...
      </addcash>
    </dispenser>
    <extdispenser>
      <info model = "" serial = "" error = "" recycler = "">
        <cassette></cassette>
        ...
      </info>
      <incass>
        <cassette></cassette>
        ...
      <reject>
        <cassette></cassette>
        <cassette></cassette>
        ...
      </reject>
    </extdispenser>
  </setState>
</terminals>
</request>
```

```

</incass>
<addcash>
  <cassette></cassette>
  ...
</addcash>
</extdispenser>
<money>
  <currency id ="643">
    <notes GoBy="0" SumGoBy="0.0" count="9" sum="90.0">
      <nominal count="9" value="10"/>
      <nominal count="0" value="50"/>
      <nominal count="0" value="100"/>
      <nominal count="0" value="500"/>
      <nominal count="0" value="1000"/>
      <nominal count="0" value="5000"/>
    </notes>
    <coins GoBy="0" SumGoBy="0.0" count="0" sum="0.0">
      <nominal count="0" value=".01"/>
      <nominal count="0" value=".05"/>
      <nominal count="0" value=".1"/>
      <nominal count="0" value=".5"/>
      <nominal count="0" value="1"/>
      <nominal count="0" value="2"/>
      <nominal count="0" value="5"/>
      <nominal count="0" value="10"/>
    </coins>
  </currency>
</money>
<interface></interface>
</setState>
</terminals>
</request>

```

Для выполнения запроса можно указать дополнительные параметры:

- `fiscal-info` – информация о фискализации АСО (фискализирован/нет, дата фискализации и прочее);
- `status` – параметры состояния автомата (ошибки купюроприемника, IP ftp, ошибки HDD и прочее);
- `watchdog` – параметры состояния сторожевого таймера (модель, конфигурация, события и прочие характеристики);
- `hardware` – информация об оборудовании АСО (технические данные HDD, модема и прочее);
- `statistics` – статистические данные о работе автомата (время, оставшееся до заполнения купюроприемника, системное время, время до заправки бумаги в чековый принтер и прочее);
- `gsm-status` – параметры состояния сети GSM (уровень сигнала, серийный номер SIM-карты);
- `software` – версия установленного ПО (страна, клиентское ПО, версия flash и прочее);
- `dispenser` – информация о состоянии устройства выдачи наличных;
 - `info` – общая информация о диспенсере:
 - ⊕ `model` – модель диспенсера;
 - ⊕ `serial` – серийный номер диспенсера;
 - ⊕ `error` – ошибки диспенсера;
 - ⊕ `cassette` – данные о кассетах, находящихся в диспенсере;
 - `incass` – информация об инкассированных денежных средствах;

- ⊕ `cassette` – информация об инкассированной кассете;
- ⊕ `reject` – информация о неработающих кассетах:
 - ❖ `cassette` – информация о каждой неправильно функционирующей кассете;
- `addcash` – информация о загрузке денежных средств в диспенсер. Денежные средства загружаются в кассетах. Данные о кассетах должны быть представлены в тегах:
 - ⊕ `cassette` – информация о загружаемой кассете;
- `extdispenser` – информация о дополнительном состоянии устройства выдачи наличных;
- `money` – информация о денежных средствах, находящихся в купюроприемнике:
 - `currency` – тег, содержащий информацию о валюте денежных средств:
 - ⊕ `id` – идентификатор валюты. Значения кодов соответствуют **ISO 4217**;
 - ⊕ `notes` – тег информации о купюрах:
 - ❖ `GoBy` – количество купюр, внесенных, но не учтенных в терминале,
 - ❖ `SumGoBy` – сумма всех купюр, внесенных, но не учтенных в терминале,
 - ❖ `count` – количество учтенных купюр,
 - ❖ `sum` – сумма всех купюр, учтенных в терминале,
 - ❖ `nominal` – информация о количестве купюр одного номинала (10, 50, 100, 500, 1000, 5000, 10000, 50000):
 - `value` – номинал;
 - `count` – количество купюр этого номинала;
 - ⊕ `coins` – тег информации о монетах (в случае если имеется монетоприемник):
 - ❖ `GoBy` – количество монет, внесенных, но не учтенных в терминале;
 - ❖ `SumGoBy` – сумма всех монет, внесенных, но не учтенных в терминале;
 - ❖ `count` – количество монет, учтенных в терминале;
 - ❖ `sum` – сумма всех монет, учтенных в терминале;
 - ❖ `nominal` – информация о количестве монет одного номинала (0.01, 0.05, 0.1, 0.5, 1.0, 2.0, 5.0, 10.0):
 - `value` – номинал;
 - `count` – количество монет этого номинала;
- `interface` – параметры интерфейса АСО (версия, значения для 2 и 3 кнопок).

6.2.6.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <setState result="0"/>
  </terminals>
</response>
```

6.2.7. resetReceiptNumber

Описание: Запрос позволяет обнулить номер чека (для фискализации).

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.2.7.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <resetReceiptNumber>
      <target-terminal>123</target-terminal>
    </resetReceiptNumber>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-terminal` – терминал, для которого необходимо обнулить номер чека (если не указан, выполняется для *текущего* терминала).

6.2.7.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <resetReceiptNumber result="0"/>
  </terminals>
</response>
```

Ответ содержит информацию об [успешности запроса](#).

6.2.8. setFiscalMode

Описание: Установка фискального режима для терминала.

Режим обработки: Строго синхронно.

- Роли:
- Кассир;
 - Дистрибьютор;
 - Главный менеджер;
 - Техподдержка;
 - Продавец;
 - Автомат;
 - Мониторинг;
 - Менеджер.

6.2.8.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <setFiscalMode>
      <target-terminal>123</target-terminal>
      <fiscal-mode>2</fiscal-mode>
      <kkm-reg-num>123456789012</kkm-reg-num>
    </setFiscalMode>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-terminal` – терминал, для которого необходимо установить фискальный режим (если не указан, выполняется для *текущего* терминала);
- `fiscal-mode` – установка (2)/ сброс (0) фискального режима работы терминала;
- `kkm-reg-num` – регистрационный номер ККМ;
- `taxpayer-reg-num` – учетный номер налогоплательщика НДС (заполняется для Казахстана).

6.2.8.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <setFiscalMode result="0"/>
  </terminals>
</response>
```

Ответ содержит [код завершения запроса](#).

6.2.9. addCashout

Описание: Запрос позволяет отправить на сервер информацию об инкассации.

Режим обработки: Строго синхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.9.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <addCashout>
      <cashout id="12345" status="2">
        <currency id="643">
          <notes sum="130.00" count="4">
            <nominal value="10.00" count="3"/>
            <nominal value="100.00" count="1"/>
          </notes>
          <coins sum="25.00" count="3">
            <nominal value="5.00" count="1"/>
            <nominal value="10.00" count="2"/>
          </coins>
        </currency>
      </cashout>
    </addCashout>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `cashout` – тег информации об инкассации:
 - `id` – идентификатор инкассации;
 - `status` – статус инкассации (см. [Приложение 3](#));
- `currency` – тег информации о валюте:
 - `id` – идентификатор валюты платежа. Значения кодов соответствуют **ISO 4217**;
- `notes` – тег информации о купюрах:
 - `sum` – сумма (в купюрах);
 - `count` – количество купюр;
 - `nominal` – информация о количестве купюр одного номинала (10, 50, 100, 500, 1000, 5000, 10000, 50000):
 - ⊕ `value` – номинал;
 - ⊕ `count` – количество купюр этого номинала;
- `coins` – тег информации о монетах (в случае если имеется монетоприемник).
 - `sum` – сумма (в монетах);
 - `count` – количество монет;
 - `nominal` – информация о количестве монет одного номинала (0.01, 0.05, 0.1, 0.5, 1.0, 2.0, 5.0, 10.0):
 - ⊕ `value` – номинал;
 - ⊕ `count` – количество монет этого номинала.

6.2.9.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <addCashout result="0"/>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `result` – результат инкассации.

6.2.10. setHosts

Описание: Установка списка альтернативных хостов для терминала, на которые будут отправляться запросы с терминала.

Режим обработки: Строго синхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.10.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <setHosts>
      <target-url>www.upyachka.ru</target-url>
      <target-terminal>111111</target-terminal>
    </setHosts>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-url` – URL хоста. Можно указать несколько хостов в данном теге, разделяя их ";",
- `target-terminal` – терминал, для которого необходимо установить альтернативные хосты (если не указан, выполняется для *текущего* терминала).

6.2.10.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <setHosts result="0"/>
  </terminals>
</response>
```

Ответ содержит [код завершения запроса](#).

6.2.11. getTerminalInfo

Описание: Запрос позволяет получить подробную информацию о терминале.

Режим обработки: Строго синхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.11.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getTerminalInfo/>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-terminal` – терминал, для которого необходимо получить подробную информацию о терминале (если не указан, выполняется для *текущего* терминала).

6.2.11.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getTerminalInfo result="0">
      <terminal address="Window" agent="1234567" agent-name=" "
configuration="111821814" label="tag1, trml" id="1111111" info=" " phone="8-901-
111-11" receipt-address="Варшавское ш., 125" region-id="26" region-name="г Москва"
type="4"/>
    </getTerminalInfo>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `terminal` – тег, содержащий информацию о терминале. Включает атрибуты:
 - `address` – наименование терминала. Вводится при заведении терминала на сайте agent.qiwi.com в обязательное поле «название терминала»;
 - `agent` – идентификатор агента, которому принадлежит терминал;
 - `agent-name` – название агента, которому принадлежит терминал;
 - `configuration` – идентификатор конфигурации терминала;
 - `id` – идентификатор терминала;
 - `info` – дополнительная информация о терминале;
 - `label` – список меток терминала;

- phone – номер телефона, указанного в качестве контактного для терминала, или (если номер телефона для терминала не указан) контактный номер телефона агента;
- receipt-address – адрес терминала для печати на чеке;
- region-id – идентификатор региона, на основании которого сформирован региональный интерфейс терминала;
- region-name – название региона, на основании которого построен региональный интерфейс терминала;
- type – идентификатор типа терминала (см. [Приложение М](#)).

6.2.12. getTerminalRegionHierarchy

Описание: Запрос получения иерархии объектов из КЛАДР для адреса терминала
Применимо только для терминалов типа АСО

Режим обработки: Строго синхронно

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.12.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getTerminalRegionHierarchy/>
  </terminals>
</request>
```

Для выполнения запроса не требуется указывать дополнительные параметры.

6.2.12.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getTerminalRegionHierarchy result="0">
      <region-hierarchy>
        <address id="1" level="1" name="Российская федерация"
          regUICalculatedBy="0" type="1"/>
        <address id="2" level="2" name="Центральный" regUICalculatedBy="0"
          type="2"/>
        <address id="26" level="3" name="Москва" regUICalculatedBy="1"
          type="5"/>
      </region-hierarchy>
    </getTerminalRegionHierarchy>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- address – информация об объекте;

- `id` – идентификатор объекта КЛАДР;
- `name` – название объекта из КАЛДР;
- `type` – тип объекта КЛАДР (см. [Приложение Н](#));
- `level` – уровень вложенности объекта в иерархии КЛАДР;
- `regUICalculatedBy` – признак использования текущего адреса для формирования регионального интерфейса АСО (0 – нет, 1 – да).

6.2.13. getTerminals

Описание: Запрос позволяет получить список терминалов, принадлежащих *текущему (или явно указанному) агенту* и его субагентам.

Режим обработки: Возможно асинхронно.

- Роли:**
- Автомат;
 - Главный менеджер;
 - Менеджер;
 - Кассир;
 - Продавец;
 - Техподдержка;
 - Дистрибьютор;
 - Мониторинг.

6.2.13.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getTerminals/>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-agent` – агент, для которого необходимо получить список терминалов (если не указан, выполняется для *текущего* агента и его субагентов);
- `target-terminal` – терминал, для которого необходимо получить список параметров терминала (если не указан, выполняется для всех терминалов *текущего* агента и его субагентов);
- `trm-state` – признак возвращения списка только незаблокированных терминалов:
 - `null` (не передан параметр) – список содержит все терминалы, принадлежащие агенту (в том числе заблокированные);
 - `1` (или любое другое число) – список содержит только незаблокированные терминалы агента;

- `fiscal-mode` – признак работы терминала в фискальном режиме (если не указан, запрос возвращает весь список терминалов, работающих как в фискальном, так и в нефискальном режимах):
 - 0 – терминал работает в нефискальном режиме;
 - 1 – терминал работает в фискальном режиме;
- `show-label` – признак получения меток терминалов:
 - `true` – получить информацию о метках;
 - `false` (или параметр не указан) – не включать информацию о метках в ответе.

6.2.13.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getTerminals result="0">
      <row address="Москва, Варшавское ш, 125" agt_id="1111111" arenda=""
city="Москва r" remote_fiscal_id="" fiscal_mode="0" full_address="POS"
kkm_registration_number="" main_address="Варшавское ш, 125" priority=""
taxpayer_regnum="" trm_address="125" trm_agt_display="" trm_approved="9"
trm_bee_id="" trm_bee_region="" trm_city_id="26" trm_contact_pers=""
trm_display="POS" trm_district_id="" trm_dup="" trm_id="2222222 trm_info=""
trm_metro_id="" trm_monitor_bills="" trm_monitor_health="" trm_monitor_lastdate=""
trm_phone="" trm_serial="" trm_state="1" trm_street_id="318318" trm_tip_mesta=""
trm_who_added="12345" trm_work_time="" ttp_id="1" union_code="" url=""/>
      ...
    </getTerminals>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `row` – тег, содержащий информацию о терминале. Включает атрибуты:
 - `address` – полный адрес терминала (включает город, улицу и номер дома);
 - `agt_id` – идентификатор агента, которому принадлежит терминал;
 - `arenda` – сумма аренды в месяц;
 - `city` – название города, в котором установлен терминал;
 - `fiscal_mode` – признак работы терминала в фискальном режиме;
 - `remote_fiscal_id` – служебное поле;
 - `full_address` – если название терминала указано, атрибут содержит название терминала, в противном случае атрибут отображает название улицы и номер дома терминала;
 - `kkm_registration_number` – регистрационный номер ККМ (заполняется для Казахстана);
 - `label` – список меток терминала;

ПРИМЕЧАНИЕ



Данный атрибут возвращается, только если в запросе указано значение `true` для параметра `show-label`.

- `main_address` – улица и номер дома адреса терминала;
- `priority` – служебное поле;
- `taxpayer_regnum` – учетный номер налогоплательщика НДС (заполняется для Казахстана);
- `trm_address` – часть адреса терминала (номер дома, строения);
- `trm_agt_display` – альтернативное название субагента;
- `trm_approved` – служебное поле;
- `trm_bee_id` – служебное поле;
- `trm_bee_region` – служебное поле;
- `trm_city_id` – идентификатор города, в котором установлен терминал;
- `trm_contact_pers` – имя контактной персоны;
- `trm_display` – название терминала;
- `trm_district_id` – идентификатор округа;
- `trm_dup` – служебное поле;
- `trm_id` – идентификатор терминала;
- `trm_info` – дополнительная информация о терминале;
- `trm_metro_id` – идентификатор станции метро, на которой расположен терминал;
- `trm_monitor_bills` – служебное поле;
- `trm_monitor_health` – служебное поле;
- `trm_monitor_lastdate` – служебное поле;
- `trm_phone` – телефон контактной персоны;
- `trm_serial` – серийный номер терминала;
- `trm_state` – признак блокировки терминала (0 - терминал заблокирован, 1 - терминал не заблокирован);
- `trm_street_id` – идентификатор улицы, на которой расположен терминал;
- `trm_tip_mesta` – идентификатор типа местоположения терминала;
- `trm_who_added` – идентификатор персоны, создавшей терминал;
- `trm_work_time` – часы работы терминала;
- `ttp_id` – идентификатор типа терминала (см. [Приложение М](#));
- `union_code` – код группировки;
- `url` – альтернативный хост для терминала (один или несколько альтернативных хостов, перечисленных через «;»).

6.2.14. getTerminalsWithCoordinates

Описание: Запрос позволяет получить список терминалов, принадлежащих агенту и его субагентам, с координатами местоположения.

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.2.14.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getTerminalsWithCoordinates/>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-agent` – агент, для которого необходимо получить список терминалов без учета его субагентов;
- `target-terminal` – терминал, для которого необходимо получить координаты местоположения;
- `target-terminal-type` – идентификатор типа терминала (см. [Приложение М](#)). Если указан, то список будет содержать только терминалы указанного типа.

6.2.14.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getTerminalsWithCoordinates result="0">
      <row address="Тула г (Тульская обл), 1-я ул, 1" agt_id="1111111" place="АЗС"
place_description="РосНефть" tc_latitude="55.627285" tc_longitude="37.624283"
tc_precision="8530" trm_id="11221111" ttp_id="4"/>
      <row address="Рязань г (Рязанская обл), 2-я ул, 2" agt_id="1111111"
place="Бизнес центр" place_description="корпус 1а" tc_latitude="55.62698"
tc_longitude="37.627262" tc_precision="1750" trm_id="11221113" ttp_id="4"/>
      ...
    </getTerminals>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `row` – тег, содержащий информацию о терминале. Включает атрибуты:
 - `address` – полный адрес терминала (включает город, улицу и номер дома);
 - `agt_id` – идентификатор агента, которому принадлежит терминал;
 - `trm_id` – идентификатор терминала;
 - `place` – тип местоположения терминала;
 - `place_description` – описание местоположения терминала;
 - `tc_latitude` – широта местоположения терминала;
 - `tc_longitude` – долгота местоположения терминала;
 - `tc_precision` – точность определения местоположения (в метрах);
 - `ttp_id` – идентификатор типа терминала (см. [Приложение М](#)).

6.2.15. setConfigId

Описание: Запрос позволяет изменить идентификатор конфигурации терминала, Текущий экземпляр конфигурации терминала удаляется из кэш.

Режим обработки: Строго синхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.15.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <setConfigId>
      <target-terminal>1234567</target-terminal>
    </setConfigId>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-terminal` – терминал, для которого необходимо изменить конфигурацию (если не указан, выполняется для *текущего* терминала).

6.2.15.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <setConfigId result="0"/>
  </terminals>
</response>
```

Ответ содержит [код завершения запроса](#).

6.2.16. getConfigId

Описание: Запрос позволяет получить идентификатор конфигурации терминала. Если идентификатор конфигурации изменился, терминал загружает с сервера новую конфигурацию.

Режим обработки: Строго синхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.16.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getConfigId/>
  </terminals>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.2.16.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getConfigId result="0">
      <id>123456789</id>
    </getConfigId>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `id` – идентификатор текущей конфигурации терминала.

6.2.17. getConfig

Описание: Запрос позволяет получить конфигурацию терминала.

Режим обработки: Строго синхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.17.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getConfig/>
  </terminals>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.2.17.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getConfig result="0">
      <max-pay-amount>0</max-pay-amount>
      <gmt-time>28.06.2011 07:55:38</gmt-time>
      <osmp-ts-phone>8-800-3330059</osmp-ts-phone>
      <osmp-general-phone>8-800-3330059</osmp-general-phone>
      <ftp-home>/interface_beta_test/</ftp-home>
      <online-auth>1</online-auth>
      <buttons>1,2,3,22,42</buttons>
      <p-height>0</p-height>
      <p-width>0</p-width>
      <receipt-info> </receipt-info>
      <sixth-button>18</sixth-button>
      <third_button_grp>485</third_button_grp>
      <second_button_grp>482</second_button_grp>
    </getConfig>
  </terminals>
</response>
```

```

<third_button_microsite>vmt</third_button_microsite>
<serviceMenuLogin>123456</serviceMenuLogin>
<PasswordFromSite>TRUE</PasswordFromSite>
<second_button_microsite>gibdd</second_button_microsite>
<serviceMenuPasswordMD5>Password</serviceMenuPasswordMD5>
<second_button_prv>3249</second_button_prv>
<max-offline-count>100</max-offline-count>
<serviceMenuSecretCode>123</serviceMenuSecretCode>
<left-margin>3</left-margin>
<languages>
  <language default="1" lng_name="Русский" lng_short="ru"/>
</languages>
</getConfig>
</terminals>
</response>

```

Ответ содержит следующие данные:

- `max-pay-amount` – максимальная сумма платежа на данном терминале (0 – ограничений нет);
- `gmt-time` – время, установленное на сервере;
- `osmp-ts-phone` – телефон технической поддержки КИВИ;
- `osmp-general-phone` – телефон КИВИ;
- `ftp-home` – домашняя папка пользователя на ftp;
- `online-auth` – признак необходимости проведения онлайн-авторизации на терминале;
- `buttons` – список из 5 идентификаторов провайдеров, размещенных на главной странице в верхнем меню (используется для стандартного интерфейса АСО);
- `receipt-info` – строка, которая должна быть напечатана на чеке;
- `p-width` – ширина чека (в символах);
- `p-height` – высота чека (в символах);
- `sixth-button` – название шестой кнопки в стандартном интерфейсе АСО;
- Параметры сервиса, размещенного на 2 кнопке регионального интерфейса АСО.

Если для одной кнопки определено несколько значений, то в интерфейсе отображается с учетом приоритетов: микросайт, группа провайдеров и только потом провайдер.

Если для второй кнопки конфигурации нет (то есть, не заданы параметры `second_button_`), то по умолчанию используется QIWI Кошелек. Параметры конфигурации 2 кнопки:

- `second_button_microsite` – путь к папке с файлами микросайта;
- `second_button_grp` – идентификатор группы провайдеров;
- `second_button_prv` – идентификатор провайдера;

- Параметры сервиса, размещенного на 3 кнопке регионального интерфейса АСО.

Если для одной кнопки определено несколько значений, то в интерфейсе отображается с учетом приоритетов: микросайт, группа провайдеров и только потом провайдер.

Если для третьей кнопки конфигурации нет (то есть, не заданы параметры `third_button_`), то по умолчанию используется группа **Услуги банков**. Параметры конфигурации 3 кнопки:

- `third_button_microsite` – путь к папке с файлами микросайта;
- `third_button_grp` – идентификатор группы провайдеров;

- `PasswordFromSite` – признак получения реквизитов доступа в сервисное меню из процессинга:
 - `true` – данные, задаются пользователем на сайте;
 - `false` – авторизационные данные вводятся непосредственно на терминале;
- `serviceMenuLogin` – логин (при наличии признака включения режима передачи секретных реквизитов);
- `serviceMenuPasswordMD5` – пароль md5 (при наличии признака включения режима передачи секретных реквизитов);
- `serviceMenuSecretCode` – секретный номер телефона для входа в сервисное меню АСО (при наличии признака включения режима передачи секретных реквизитов);
- `max-offline-count` – максимальное число платежей в пакете, которое может передавать терминал;
- `left-margin` – отступ слева на чеке;
- `languages` – тег, содержащий список языков, установленных на терминале. Содержит вложенные теги:
 - `language` – параметры языка:
 - ⊕ `default` – признак языка по умолчанию;
 - ⊕ `lng_name` – название языка;
 - ⊕ `lng_short` – код языка по стандарту **ISO 639-1** (двухбуквенное обозначение; например, `en` – английский, `ru` – русский). Общий справочник языковых кодов можно посмотреть [здесь](#).

6.2.18. getCommissions

Описание: Запрос позволяет получить список комиссий, установленных на терминале для различных провайдеров.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.18.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getCommissions>
      <target-terminal>1234567</target-terminal>
    </getCommissions>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-terminal` – терминал, для которого необходимо получить список комиссий (если не указан, выполняется для *текущего* терминала).

6.2.18.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getCommissions result="0">
      <row fix-com="0" profile-id="12772" prv-id="441" trm-id="8395547"/>
      <row fix-com="10" profile-id="18104" prv-id="136" trm-id="8395547"/>
      <row fix-com="10" profile-id="18104" prv-id="572" trm-id="8395547"/>
      ...
    </getCommissions>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `row` – тег, содержащий информацию о терминальной комиссии, установленной для указанного провайдера. Включает атрибуты:
 - `fix-com` – фиксированная ставка комиссии, указывается в % от суммы платежа;
 - `profile-id` – идентификатор профиля комиссии. Данные о профиле комиссии можно получить запросом [getCommissionProfiles](#). Алгоритм расчета комиссий см. в разделе [5.35.1](#);

ПРИМЕЧАНИЕ



Профиль имеет больший приоритет, чем фиксированная ставка комиссии.

Фиксированная ставка используется, если не указан профиль комиссии или не выполняется ни одно из правил профиля.

- `prv-id` – идентификатор провайдера, к которому применяется данная комиссия.

6.2.19. getCommissionProfiles

Описание: Запрос позволяет получить профили комиссий (см. [5.2](#)).

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.19.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getCommissionProfiles>
      <target-terminal>1111119</target-terminal>
      <target-agent>112233</target-agent>
    </getCommissionProfiles>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-agent` – агент, для терминалов которого необходимо получить список профилей комиссий, в том числе для его субагентов (если не указан, выполняется для *текущего* агента);
- `target-terminal` – терминал, для которого необходимо получить список профилей комиссий (если не указан, выполняется для *текущего* терминала).

6.2.19.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getCommissionProfiles result="0">
      <profile agent-id="112233" id="111111" name="Basel">
        <profile-data order="1" com-percent="0" from-amount="10" to-amount="1000"/>
        <profile-data order="2" com-fix="51" from-amount="1000" from-hour="6:00"
to-amount="1010" to-hour="22:00"/>
        <profile-data order="3" com-fix="52" from-amount="1010" from-hour="22:00"
to-amount="1020" to-hour="6:00"/>
        <nominal-commissions cur-id="643">
          <nominal value="1" type="1" com-fix="0.5"/>
          <nominal value="10" type="1" com-fix="2"/>
          <nominal value="10" type="2" com-fix="0.9"/>
        </nominal-commissions>
        <nominal-commissions cur-id="940">
          <nominal value="0.25" type="1" com-fix="0.1"/>
        </nominal-commissions>
      </profile>
    </getCommissionProfiles>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `profile` – тег профиля комиссий. Атрибуты:
 - `id` – идентификатор профиля;
 - `agent-id` – идентификатор агента или субагента;
 - `name` – название профиля;
- `profile-data` – тег, описывающий правило вычисления размера комиссии (см. раздел [5.2](#)). Включает следующие атрибуты:
 - `order` – порядковый номер *правила*;

ПРИМЕЧАНИЕ



Правило с меньшим порядковым номером имеет больший приоритет.

- `com-fix` – абсолютное значение комиссии (в валюте платежа);
- `com-percent` – ставка комиссии (в процентах);

ПРИМЕЧАНИЕ



Обычно указывается одно из значений: `com-fix` или `com-percent`.
В тех случаях, когда указаны оба значения, комиссия суммируется.

- `from-amount, to-amount` – диапазон сумм платежей, на которых действует данное *правило*;
- `from-hour, to-hour` – время суток, на которое распространяется действия *правила*;
- `com-min` – минимальное значение комиссии;
- `com-max` – максимальное значение комиссии;
- `nominal-commissions` – данные по комиссиям на номиналы купюр и монет;
 - ⊕ `cur-id` – код валюты номиналов (значения кодов соответствуют **ISO 4217**);
 - ⊕ `nominal` – тег, содержит информацию о комиссии на 1 номинал купюры или монеты, атрибуты:
 - ❖ `value` – номинал, для которого установлена комиссия;
 - ❖ `type` – тип денежных средств:
 - 1 – купюры;
 - 2 – монеты;
 - ❖ `com-fix` – взимаемая с купюры/монеты комиссия (в процентах).

6.2.20. getCommissionsHistory

Описание: Получение истории изменений комиссии на терминале (см. [5.2](#)).

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.2.20.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getCommissionsHistory from-date="2010-01-01" to-date="2010-01-31" prv-id="1">
      <target-terminal>112233</target-terminal>
    </getCommissionsHistory>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-terminal` – терминал, для которого необходимо получить историю изменений профилей комиссий (если не указан, запрос выполняется для текущего терминала);
- `from-date` – дата начала периода, за который необходимо получить историю изменений комиссии (если не указан, запрос выполняется, начиная с даты создания первого профиля комиссии). Формат: `YYYY-MM-DD`;
- `to-date` – дата окончания периода, за который необходимо получить историю изменений комиссии (если не указан, выводятся все изменения, начиная с даты в теге `from-date`). Формат: `YYYY-MM-DD`;
- `prv-id` – провайдер, по которому необходимо получить историю изменений комиссии (если не указан, запрос выполняется для всех провайдеров).

6.2.20.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getCommissionsHistory result="0">
      <row change-date="2009-06-18T17:01:01+04:00" fix-com="0" id="12345678"
person-id="1" profile-id="23293" prv-id="441"/>
      ...
    </getCommissionsHistory>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `row` – тег с данными о терминальной комиссии, установленной для указанного провайдера. Включает атрибуты:
 - `change-date` – дата, начиная с которой действует комиссия (дата изменения);
 - `fix-com` – фиксированная ставка комиссии, указывается в % от суммы платежа;
 - `id` – идентификатор комиссии;
 - `person-id` – идентификатор персоны, изменившей комиссию;
 - `profile-id` – идентификатор профиля комиссии. Алгоритм расчета комиссий см. в разделе [5.1](#);

ПРИМЕЧАНИЕ



Профиль имеет больший приоритет, чем фиксированная ставка комиссии.

Фиксированная ставка используется, если не указан профиль комиссии или не выполняется ни одно из правил профиля.

- `prv-id` – идентификатор провайдера, к которому применена комиссия.

6.2.21. `getTerminalCommissionTemplates`

Описание: Запрос позволяет получить шаблоны комиссий терминала.

Режим обработки: Возможно асинхронно.

- Роли:**
- Автомат;
 - Главный менеджер;
 - Менеджер;
 - Кассир;
 - Продавец;
 - Техподдержка;
 - Дистрибьютор;
 - Мониторинг.

6.2.21.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getTerminalCommissionTemplates>
      <target-agent>3</target-agent>
      <target-terminal>123</target-terminal>
    </getTerminalCommissionTemplates>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-agent` – агент, для терминалов которого (и для всех терминалов его субагентов) необходимо получить список шаблонов комиссий (если не указан, выполняется для *текущего* агента);
- `target-terminal` – терминал, для которого необходимо получить список шаблонов комиссий (если не указан, выполняется для *текущего* терминала).

6.2.21.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getTerminalCommissionTemplates result="0">
      <row agent-id="3" id="2" last-modified="2011-07-28T10:38:15+04:00" terminal-
id="123"/>
    </getTerminalCommissionTemplates>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `row` – тег строки шаблона комиссии. Атрибуты:
 - `agent-id` – идентификатор агента или субагента;
 - `id` – идентификатор шаблона;
 - `last-modified` – время последней модификации шаблона;
 - `terminal-id` – идентификатор терминала.

6.2.22. bindTerminalTemplate

Описание: Запрос позволяет привязать шаблон комиссии к терминалу.

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.2.22.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <bindTerminalTemplate>
      <target-template>415</target-template>
      <target-terminal>1112233</target-terminal>
    </bindTerminalTemplate>
  </terminals>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- `target-terminal` – терминал, к которому необходимо привязать шаблон комиссии (если не указан, выполняется для текущего терминала);
- `target-template*` – шаблон, который необходимо привязать к терминалу.

6.2.22.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <bindTerminalTemplate result="0" template-id="415" terminal-id="112233"/>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `template-id` – идентификатор шаблона;
- `terminal-id` – идентификатор терминала.

6.2.23. getForeignCommissions

Описание: Получение информационного профиля комиссии по провайдеру (например, комиссии банка)

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.23.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getForeignCommissions/>
  </terminals>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.2.23.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getForeignCommissions result="0">
      <provider profile-id="85985" prv-id="112"/>
      <provider profile-id="85987" prv-id="113"/>
      <provider profile-id="85988" prv-id="114"/>
      <provider profile-id="85984" prv-id="115"/>
      <provider profile-id="85990" prv-id="117"/>
      <provider profile-id="85990" prv-id="118"/>
      <provider profile-id="85990" prv-id="119"/>
      <provider profile-id="85989" prv-id="129"/>
      <provider profile-id="85990" prv-id="149"/>
      <profile id="85984">
        <profile-data com-min="12.0" com-percent="1.5" order="1"/>
      </profile>
      <profile id="85985">
        <profile-data com-min="60.0" com-percent="2.0" order="1"/>
      </profile>
      <profile id="85987">
        <profile-data order="1"/>
      </profile>
      <profile id="85988">
        <profile-data com-min="50.0" com-percent="1.5" order="1"/>
      </profile>
      <profile id="85989">
        <profile-data com-min="50.0" com-percent="1.5" order="1"/>
      </profile>
      <profile id="85990">
        <profile-data com-min="50.0" com-percent="1.5" order="1"/>
      </profile>
    </getForeignCommissions>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- `provider` – связь провайдера и профиля;
- `profile` – тег, описывающий профиль комиссий. Описание параметров тега см. в разделе [6.2.19.2](#).

6.2.24. getTerminalTypes

Описание: Получение справочника типов терминалов.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.2.24.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <getTerminalTypes/>
  </terminals>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.2.24.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <getTerminalTypes result="0">
      <row can_pay="1" client_software_regexp="^Dealer\s+v" id="6" long_name="XML
      протокол" short_name="XML протокол" tpt_id="40"/>
      ...
    </getTerminalTypes>
  </terminals>
</response>
```

Ответ содержит следующие данные:

ПРИМЕЧАНИЕ



Ответ содержит информацию только о тех типах терминалов, которые есть у текущего агента.

- `row` – тег, содержащий информацию о терминале:
 - `can_pay` – признак возможности осуществления платежей с данного типа терминала;
 - `client_software_regexp` – регулярное выражение, которому должна соответствовать строка с описанием типа клиентского ПО в запросе;
 - `id` – идентификатор типа терминала (см. [Приложение М](#));
 - `long_name` – полное название терминала;
 - `short_name` – короткое название терминала;

- `tpt_id` – идентификатор дополнительного типа терминала (см. [Приложение М](#)).

6.2.25. addShiftClosure

- Описание:** Запрос позволяет закрыть смену на терминале.
- Режим обработки:** Строго синхронно.
- Роли:** Выполнение запроса доступно персоне с любой ролью.

6.2.25.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <addShiftClosure>
      <shift id="101" status="10">
        <currency id="643">
          <notes sum="10.00" count="1">
            <nominal value="10.00" count="1"/>
          </notes>
          <coins sum="10.00" count="10">
            <nominal value="1.00" count="10"/>
          </coins>
        </currency>
      </shift>
    </addShiftClosure>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `shift` – тег информации о смене:
 - `id` – идентификатор смены;
 - `status` – статус закрытия смены (всегда "10");
- `currency` – тег информации о валюте:
 - `id` – идентификатор валюты платежа. Значения кодов соответствуют **ISO 4217**;
- `notes` – тег информации о купюрах:
 - `sum` – сумма (в купюрах);
 - `count` – количество купюр;
 - `nominal` – информация о количестве купюр одного номинала (10, 50, 100, 500, 1000, 5000, 10000, 50000):
 - ⊕ `value` – номинал;
 - ⊕ `count` – количество купюр этого номинала;
- `coins` – тег информации о монетах (в случае если имеется монетоприемник):
 - `sum` – сумма (в монетах);

- count – количество монет;
- nominal – информация о количестве монет одного номинала (0.01, 0.05, 0.1, 0.5, 1.0, 2.0, 5.0, 10.0):
 - ⊕ value – номинал;
 - ⊕ count – количество монет этого номинала.

6.2.25.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <addShiftClosure result="0"/>
  </terminals>
</response>
```

Ответ содержит следующие данные:

- result – результат закрытия смены.

6.2.26. requestLog

Описание: Получение лог-файла терминала.

Режим обработки: Строго синхронно.

- Роли:**
- Автомат;
 - Главный менеджер;
 - Менеджер;
 - Кассир;
 - Продавец;
 - Техподдержка;
 - Дистрибьютор;
 - Мониторинг.

6.2.26.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <requestLog>
      <target-terminal>111111</target-terminal>
      <e-mail>mail@qiwi.ru</e-mail>
      <date-from>2010-08-22T04:04:00</date-from>
      <date-to>2010-08-22T04:04:00</date-to>
    </requestLog>
  </terminals>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- `target-terminal` – терминал, для которого необходимо получить лог-файл (если не указан, выполняется для *текущего* терминала);
- `e-mail*` – адрес электронной почты, на который должен быть отправлен файл;
- `date-from*` и `date-to*` – теги задают период, за который необходимо получить лог-файл.

6.2.26.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <requestLog result="0"/>
  </terminals>
</response>
```

Ответ содержит информацию об успешности запроса (см. [Приложение В](#)).

6.2.27. requestProcessList

Описание: Получение списка процессов, выполняемых на терминале.

Режим обработки: Строго синхронно.

- Роли:**
- Автомат;
 - Главный менеджер;
 - Менеджер;
 - Кассир;
 - Продавец;
 - Техподдержка;
 - Дистрибьютор;
 - Мониторинг.

6.2.27.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <requestProcessList>
      <target-terminal>111111</target-terminal>
      <e-mail>mail@qiwi.ru</e-mail>
    </requestProcessList>
  </terminals>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- `target-terminal` – терминал, для которого необходимо получить список выполняемых процессов (если не указан, выполняется для *текущего* терминала);
- `e-mail*` – адрес электронной почты, на который должен быть отправлен список процессов.

6.2.27.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <requestProcessList result="0"/>
  </terminals>
</response>
```

Ответ содержит информацию об успешности запроса (см. [Приложение В](#)).

6.2.28. requestZReport

Описание: Получение снятого с терминала Z-отчета.

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.2.28.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <requestZReport>
      <target-terminal>1111111</target-terminal>
      <id>123</id>
      <e-mail>mail@qiwi.ru</e-mail>
    </requestZReport>
  </terminals>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- `target-terminal` – терминал, для которого необходимо получить Z-отчет (если не указан, выполняется для *текущего* терминала);
- `id*` – идентификатор Z-отчета, который необходимо получить;
- `e-mail*` – адрес электронной почты, на который должен быть отправлен Z-отчет.

6.2.28.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
```

```
<requestZReport result="0"/>
</terminals>
</response>
```

Ответ содержит информацию об успешности запроса (см. [Приложение В](#)).

6.2.29. restartClientSoftware

Описание: Перегрузка клиентского ПО АСО (*maratl.exe*)

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.2.29.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <restartClientSoftware>
      <target-terminal>1111111</target-terminal>
    </restartClientSoftware>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-terminal` – терминал, клиентское ПО (*maratl.exe*) которого необходимо перезапустить (если не указан, выполняется для *текущего* терминала).

6.2.29.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <restartClientSoftware result="0"/>
  </terminals>
</response>
```

Ответ содержит информацию об успешности запроса (см. [Приложение В](#)).

6.2.30. rebootTerminal

Описание: Перегрузка терминала.

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.2.30.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <rebootTerminal>
      <target-terminal>1111111</target-terminal>
    </rebootTerminal>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-terminal` – терминал, который необходимо перезагрузить (если не указан, выполняется для *текущего* терминала).

6.2.30.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <rebootTerminal result="0"/>
  </terminals>
</response>
```

Ответ содержит информацию об успешности запроса (см. [Приложение В](#)).

6.2.31. disableTerminal

Описание: Выключение терминала

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.2.31.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <terminals>
    <disableTerminal>
      <target-terminal>1111111</target-terminal>
    </disableTerminal>
  </terminals>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-terminal` – терминал, который необходимо выключить (если не указан, выполняется для *текущего* терминала).

6.2.31.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <terminals>
    <disableTerminal result="0"/>
  </terminals>
</response>
```

Ответ содержит информацию об успешности запроса (см. [Приложение В](#)).

6.3. Agents

ПРИМЕЧАНИЕ



Обязательные параметры запросов отмечены символом *.

Список действий интерфейса:

Действие	Описание
getAgentInfo	Получение информации об агенте
getAgents	Получение списка субагентов
getBalance	Получение информации о балансе агента
getBalanceTree	Получение информации о балансе агента и его субагентов
getOffer	Получение реквизитов договора-оферты
transferFunds	Перевод денег от текущего агента субагенту (или наоборот)
setFiscalMode	Установка фискального режима для агента
setOuterCommission	Установка внешней комиссии на терминалы агента
getCommissionTemplates	Получение шаблонов комиссий
setCommissionProfile	Изменение профиля комиссии
addCommissioProfile	Добавление нового профиля комиссии
setCommissionTemplate	Изменение шаблона комиссии
addCommissionTemplate	Добавление нового шаблона комиссии

ВНИМАНИЕ

По умолчанию все действия выполняются для *текущего агента*.

Текущим агентом считается агент, которому принадлежит терминал, указанный в теге `<client>` и персона, данные которой указаны в теге `<auth>`.

Если требуется получить информацию по другому агенту, то его идентификатор необходимо указать в теге `<target-agent>` внутри тега действия (см. пример запроса).

6.3.1. [getAgentInfo](#)

Описание: Запрос позволяет получить подробную информацию об агенте.

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.3.1.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <agents>
    <getAgentInfo>
      <target-agent>2</target-agent>
    </getAgentInfo>
  </agents>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-agent` – идентификатор агента, по которому вы хотите получить информацию (если не указан, действие выполняется для *текущего агента*).

6.3.1.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <agents>
    <getAgentInfo result="0">
      <agent address="Улица, дом, корпус" fio="Горшкова Оксана Сергеевна" id="2"
inn="123" name="2 000 Валсент Ладога" phone="89010000000" cnt-phone="9121111111,
1632132" cnt-email="email1, email2" cur-id="643" />
    </getAgentInfo>
  </agents>
</response>
```

Ответ содержит следующие данные:

- `agent` – тег, содержащий информацию об агенте. Включает атрибуты:
 - `address` – адрес расположения агента;
 - `fio` – ФИО контактной персоны;
 - `id` – идентификатор агента;
 - `info` – дополнительная информация об агенте;
 - `inn` – ИНН агента;
 - `name` – название агента;
 - `phone` – контактный телефон;
 - `www` – адрес веб-сайта;
 - `cnt-phone` – список контактных телефонов технической поддержки агента (разделитель ",");
 - `cnt-email` – список контактных e-mail технической поддержки агента (разделитель ",");
 - `cur-id` – код валюты, в которой агент ведет прием платежей. Значения кодов соответствуют **ISO 4217**.

6.3.2. getAgents

Описание: Запрос позволяет получить список субагентов.

Режим обработки: Возможно асинхронно.

Роли:

- Главный менеджер;
- Менеджер;
- Мониторинг;
- Бухгалтер;
- Техподдержка;
- Дистрибьютор (новый агентский сайт);
- Координатор Дистрибьютора;
- Дистрибьютор.

6.3.2.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <agents>
    <getAgents>
      <target-agent>2</target-agent>
    </getAgents>
  </agents>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-agent` – идентификатор агента, по которому запрашивается информация (если не указан, действие выполняется для *текущего агента*).

6.3.2.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <agents>
    <getAgents result="0">
      <row agt_distributor_id="1" agt_id="2" agt_inn="0111" agt_jur_address="Новое
Москва, Иванова ул, 52" agt_name="" agt_phys_address=" Москва, Иванова ул, 52"
city="Москва" fiscal_mode="0" kkm_registration_number="" taxpayer_regnum=""
vat_payer="1"/>
      ...
    </getAgents>
  </agents>
</response>
```

Ответ содержит следующие данные:

- `row` – тег, содержащий информацию об агенте. Включает атрибуты:
 - `agt_distributor_id` – идентификатор вышестоящего агента;

- agt_id – идентификатор агента;
- agt_inn – ИНН;
- agt_jur_address – юридический адрес агента;
- agt_name – название агента;
- agt_phys_address – физический адрес агента;
- city – название города;
- fiscal_mode – установка (2)/ сброс (0) фискального режима работы агента;
- kkm_registration_number – регистрационный номер ККМ;
- taxpayer_regnum – учетный номер налогоплательщика НДС (заполняется для Казахстана);
- vat-payer – признак плательщика налогов:
 - 0 – плательщик;
 - 1 – неплательщик.

6.3.3. getBalance

Описание: Запрос позволяет получить текущий баланс агента.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.3.3.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <agents>
    <getBalance>
      <target-agent>3</target-agent>
    </getBalance>
  </agents>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- target-agent – идентификатор агента, по которому вы хотите получить информацию (если не указан, действие выполняется для *текущего агента*).

6.3.3.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <agents>
    <getBalance result="0">
      <agent-id>3</agent-id>
      <balance>7782.99</balance>
      <tree-balance>7782.99</tree-balance>
      <overdraft>0.00</overdraft>
    </getBalance>
  </agents>
</response>
```

Ответ содержит следующие данные:

- `agent-id` – идентификатор агента, информацию о балансе которого возвращает процессинг;
- `balance` – текущий баланс агента;
- `tree-balance` – баланс по ветке агента;
- `overdraft` – величина овердрафта агента.

6.3.4. `getBalanceTree`

Описание: Запрос позволяет получить текущий баланс агента и всех его субагентов.

Режим обработки: Возможно асинхронно.

Роли:

- Главный менеджер;
- Менеджер;
- Бухгалтер;
- Техподдержка;
- Дистрибьютор.

6.3.4.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <agents>
    <getBalanceTree/>
  </agents>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-agent` – идентификатор агента, по которому необходимо получить информацию без учета его субагентов.

6.3.4.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <agents>
    <getBalanceTree result="0">
      <row agent_name="Агент1" agent_id="11111467" balance="55.29"
        overdraft="0" tree_balance="55.29"/>
      <row agent_name="Агент2" agent_id="11111469" balance="-280"
        overdraft="-280" tree_balance="-170.35"/>
    </getBalanceTree>
  </agents>
</response>
```

Ответ содержит следующие данные:

- `row` – данные о балансе агента. Включает атрибуты:
 - `agent_name` – название агента;
 - `agent_id` – идентификатор агента;
 - `balance` – текущий баланс агента;
 - `overdraft` – величина овердрафта агента;
 - `tree_balance` – баланс по ветке агента (сумма балансов субагентов, включая *текущего* агента).

6.3.5. getOffer

Описание: Запрос позволяет получить реквизиты агентского договора-оферты.

Режим обработки: Строго синхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.3.5.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <agents>
    <getOffer>
      <target-agent>2</target-agent>
    </getOffer>
  </agents>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `target-agent` – идентификатор агента, по которому требуется получить информацию (если не указан, действие выполняется для *текущего агента*).

6.3.5.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <agents>
    <getOffer result="0">
      <agent-id>2</agent-id>
      <agent-name>Название</agent-name>
      <agent-inn>0111</agent-inn>
      <agent-phone>89100000000</agent-phone>
      <agent-www>www.testirovanie.ru</agent-www>
      <agent-info>Проверка ввода дополнительной информации</agent-info>
      <agent-fio>Иванов Петр Сидорович</agent-fio>
      <agent-address>Варшавское ш, 125</agent-address>
      <vat-payer>0</vat-payer>
      <vat-tax>0.18</vat-tax>
      <agt-contract>реквизиты договора</agt-contract>
      <aof-addr>Москва, Героев Панфиловцев 5-4-35</aof-addr>
      <aof-contact-phone>offer phone</aof-contact-phone>
      <aof-gendir-fio>director_agent</aof-gendir-fio>
      <offer-type>0</offer-type>
    </getOffer>
  </agents>
</response>
```

Ответ содержит следующие данные:

- agent-id – идентификатор агента;
- agent-name – название агента;
- agent-inn – ИНН агента;
- agent-phone – телефон агента;
- agent-www – адрес веб-сайта агента;
- agent-info – дополнительная информация об агенте;
- agent-fio – ФИО контактной персоны агента;
- agent-address – юридический адрес агента;
- vat-payer – признак плательщика налогов:
 - 0 – плательщик;
 - 1 – неплательщик;
- vat-tax – процентная ставка налога;
- agt-contract – реквизиты договора агента (для технических субагентов возвращаются реквизиты договора вышестоящего агента);
- aof-addr – контактный адрес агента;
- aof-contact-phone – контактный телефон, указанный в оферте;
- aof-gendir-fio – ФИО генерального директора, указанного в оферте;
- offer-type – тип оферты:
 - 0 – обычная;
 - 1 – для кредитных организаций.

6.3.6. transferFunds

Описание: Запрос позволяет перевести деньги со счета *текущего* агента на счет его субагента или наоборот (списать деньги со счета субагента на счет *текущего* агента).

Режим обработки: Строго синхронно.

Роли:

- Главный менеджер;
- Продавец;
- Дистрибьютор;
- Дистрибьютор (новый агентский сайт);
- Координатор Дистрибьютора.

6.3.6.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <agents>
    <transferFunds>
      <target-agent>2</target-agent>
      <payment id="1234567" agent="123" enroll="10.0" currency="643"
comment="my internet"/>
      <payment id="1234568" agent="124" charge="10.0" currency="643" comment="my
internet"/>
    </transferFunds>
  </agents>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-agent` – идентификатор агента, для которого выполняется запрос (если не указан, действие выполняется для *текущего агента*).
- `payment` – тег, содержащий информацию о переводе. Включает атрибуты:
 - `id` – идентификатор платежа;
 - `agent` – идентификатор агента – получателя платежа;

ВНИМАНИЕ



Получатель платежа должен быть субагентом *текущего/указанного агента*.

- `enroll` – сумма, которая будет списана с целевого агента на счет его субагента;
- `charge` – сумма, которая будет начислена целевому агенту со счета субагента;
- `currency` – валюта перевода;
- `comment` – комментарий.

ПРИМЕЧАНИЕ



В случае если счета агента и субагента в разной валюте, Система автоматически пересчитает переводимую сумму.

6.3.6.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <agents>
    <transferFunds result="0">
      <payment id="1234567" result="0" status="2" uid="16625592616002"/>
      <payment id="1234568" result="0" status="2" uid="16625592616117"/>
    </transferFunds>
  </agents>
</response>
```

Ответ содержит следующие данные:

- `payment` – тег, содержащий информацию о переводе. Включает атрибуты:
 - `id` – идентификатор платежа;
 - `status` – текущий статус платежа (подробнее см. в [Приложении Г](#));

ПРИМЕЧАНИЕ



При получении нефинального статус платежа следует уточнять статус запросом [providers.getPaymentStatus](#).

- `result` – код ошибки проведения платежа (подробнее см. в [Приложении В](#));
- `uid` – идентификатор транзакции в КИВИ.

6.3.7. setFiscalMode

Описание: Установка фискального режима для агента.

Режим обработки: Строго синхронно.

Роли:

- Кассир;
- Дистрибьютор;
- Главный менеджер;
- Техподдержка;
- Продавец;
- Автомат;
- Мониторинг;
- Менеджер.

6.3.7.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <agents>
    <setFiscalMode>
      <fiscal-mode>2</fiscal-mode>
    </setFiscalMode>
  </agents>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `fiscal-mode` – установка (2)/ сброс (0) фискального режима работы агента.

6.3.7.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?><response result="0">
  <agents>
    <setFiscalMode result="0"/>
  </agents>
</response>
```

Ответ содержит [код завершения запроса](#).

6.3.8. setOuterCommission

Описание: Установка внешней комиссии на терминалы агента

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.3.8.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <agents>
    <setOuterCommission>
      <target-agent>3</target-agent>
      <target-terminal>123</target-terminal>
      <prv-id>2</prv-id>
      <commission-rate>5.6</commission-rate>
      <commission-profile>6880</commission-profile>
      <date-from>01.10.2010</date-from>
    </setOuterCommission>
  </agents>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- `target-agent*` – идентификатор агента, для всех терминалов которого необходимо установить внешнюю комиссию;
- `target-terminal` – терминал, для которого необходимо установить внешнюю комиссию (если не указан, выполняется для текущего терминала);
- `prv-id*` – идентификатор провайдера, на платежи в пользу которого устанавливается внешняя комиссия;
- `commission-rate` – ставка комиссии;
- `commission-profile` – устанавливаемый профиль комиссии;

ПРИМЕЧАНИЕ

Один из параметров: `commission-rate` или `commission-profile` – должен обязательно присутствовать в запросе.

- `date-from` – дата начала действия устанавливаемой комиссии. Формат: `dd.mm.yyyy hh:mm:ss`.

ПРИМЕЧАНИЕ

1. В системе встроена проверка: комиссия не может задаваться за прошедший период. Таким образом, если время и дата обработки запроса в процессинге позже установленных в параметре `date-from` – обработка запроса будет завершена с ошибкой.
2. Рекомендуется задавать параметр `date-from` датой без указания времени начала действия – в этом случае время считается как 00:00:00.

6.3.8.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?><response result="0">
  <agents>
    <setOuterCommission result="0"/>
  </agents>
</response>
```

Ответ содержит [код завершения запроса](#).

6.3.9. `getCommissionTemplates`

Описание: Получение шаблонов комиссий агента и его субагентов.

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.3.9.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <agents>
    <getCommissionTemplates/>
  </agents>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.3.9.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <agents>
    <getCommissionTemplates result="0">
      <template agent-id="123" id="1" name=" test">
        <template-data grp-id="2" last-modified="2011-08-02T00:54:04+04:00"
profile-id="1"/>
        <template-data cms-rate="10.0" last-modified="2011-08-02T00:54:04+04:00"
prv-id="1"/>
      </template>
      <template agent-id="123" id="3" name="test1">
        <template-data cms-rate="1.0" last-modified="2011-06-20T18:16:58+04:00"
prv-id="1"/>
      </template>
      ...
    </getCommissionTemplates>
  </agents>
</response>
```

Ответ содержит следующие данные:

- `template` – тег строки шаблона комиссии. Включает:
 - `agent-id` – идентификатор агента;
 - `id` – идентификатор шаблона;
 - `name` – название шаблона;
 - `template-data` – тег содержит данные о комиссии на провайдера/группу провайдеров в шаблоне. Атрибуты:
 - ⊕ `grp-id` – идентификатор группы;
 - ⊕ `prv-id` – идентификатор провайдера;
 - ⊕ `profile-id` – идентификатор профиля комиссии, установленного для группы/провайдера;
 - ⊕ `cms-rate` – фиксированная ставка комиссии;
 - ⊕ `last-modified` – время последней модификации шаблона.

6.3.10. setCommissionProfile

Описание: Изменение профиля комиссии (см. [5.2](#)).

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.3.10.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <agents>
    <setCommissionProfile>
      <target-agent>123</target-agent>
      <profile id="12106511" name="xmltest-new">
        <profile-data order="1" from-hour="00:00" to-hour="23:59" com-fix="12.12"/>
        <profile-data order="2" from-hour="23:00" to-hour="23:59" com-min="12.12"/>
      </profile>
    </setCommissionProfile>
  </agents>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- `target-agent` – агент или субагент, для которого необходимо обновить профиль комиссии (если не указан, выполняется для *текущего* агента);
- `profile*` – изменяемый профиль комиссии. Атрибуты:
 - `id*` – идентификатор профиля;
 - `name` – название профиля (если не указан, название не будет изменено);
- `profile-data` – правило вычисления размера комиссии (если не указан, параметры комиссий профиля не изменяются). Атрибуты:

ПРИМЕЧАНИЕ



В запросе может присутствовать несколько таких тегов.

- `order` – порядковый номер *правила*;

ПРИМЕЧАНИЕ



Правило с меньшим порядковым номером имеет больший приоритет.

- com-fix – абсолютное значение комиссии (в валюте платежа);
- com-percent – ставка комиссии (в процентах);

ПРИМЕЧАНИЕ



Обычно указывается одно из значений: com-fix или com-percent.
В тех случаях, когда указаны оба значения, комиссия суммируется.

- from-amount, to-amount – диапазон сумм платежей, на которых действует данное *правило*;
- from-hour, to-hour – время суток, на которое распространяется действия *правила*;
- com-min – минимальное значение комиссии;
- com-max – максимальное значение комиссии;
- nominal-commissions – данные по комиссиям на номиналы купюр и монет;
- ⊕ cur-id – код валюты номиналов (значения кодов соответствуют **ISO 4217**);
- ⊕ nominal – тег, содержит информацию о комиссии на 1 номинал купюры или монеты, атрибуты:
 - ❖ value – номинал, для которого установлена комиссия;
 - ❖ type – тип денежных средств:
 - 1 – купюры;
 - 2 – монеты;
 - ❖ com-fix – взимаемая с купюры/монеты комиссия (в процентах).

6.3.10.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <agents>
    <setCommissionProfile profile-id="12106511" result="0"/>
  </agents>
</response>
```

Ответ содержит следующие данные:

- profile-id – идентификатор профиля, в который пытались внести изменения;
- result – код ошибки изменения профиля (см. [Приложение В](#)).

6.3.11. addCommissionProfile

Описание: Добавление нового профиля комиссии (см. [5.2](#)).

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.3.11.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <agents>
    <addCommissionProfile>
      <target-agent>123</target-agent>
      <profile name="xml-test">
        <profile-data order="1" from-hour="00:00" to-hour="23:59" com-fix="10"/>
        <profile-data order="2" from-hour="22:00" to-hour="23:59" com-min="5" com-
max="7.99"/>
      </profile>
    </addCommissionProfile>
  </agents>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- `target-agent` – агент, для терминалов которого необходимо добавить профиль комиссии (если не указан, выполняется для *текущего* агента);
- `profile*` – создаваемый профиль комиссии. Атрибут:
 - `name` – название нового профиля (если не указан, создается профиль с незаполненным именем);
- `profile-data` – правило вычисления размера комиссии (если не указан, создается пустой профиль). Включает следующие атрибуты:

ВНИМАНИЕ



Теги `profile-data` должны иметь уникальный атрибут `order`.

- `order*` – порядковый номер *правила*;

ПРИМЕЧАНИЕ



Правило с меньшим порядковым номером имеет больший приоритет.

- `com-fix` – абсолютное значение комиссии (в валюте платежа);
- `com-percent` – ставка комиссии (в процентах);

ПРИМЕЧАНИЕ



Обычно указывается одно из значений: `com-fix` или `com-percent`.
В тех случаях, когда указаны оба значения, комиссия суммируется.

- `from-amount, to-amount` – диапазон сумм платежей, на которых действует данное *правило*;
- `from-hour, to-hour` – время суток, на которое распространяется действия *правила*;
- `com-min` – минимальное значение комиссии;
- `com-max` – максимальное значение комиссии;
- `nominal-commissions` – данные по комиссиям на номиналы купюр и монет;
- `cur-id` – код валюты номиналов (значения кодов соответствуют **ISO 4217**);
- ⊕ `nominal` – тег, содержит информацию о комиссии на 1 номинал купюры или монеты, атрибуты:
 - ❖ `value` – номинал, для которого установлена комиссия;
 - ❖ `type` – тип денежных средств:
 - 1 – купюры;
 - 2 – монеты;
 - ❖ `com-fix` – взимаемая с купюры/монеты комиссия (в процентах).

6.3.11.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <agents>
    <addCommissionProfiles profile-id="12106511" result="0"/>
  </agents>
</response>
```

Ответ содержит следующие данные:

- `profile-id` – идентификатор создаваемого профиля;
- `result` – код ошибки создания профиля (см. [Приложение В](#)).

6.3.12. setCommissionTemplate

Описание: Запрос позволяет установить изменения в шаблоне комиссии.

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.3.12.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <agents>
    <setCommissionTemplate>
      <target-agent>112233</target-agent>
      <template id="873" name="updated-xml-test">
        <template-data prv-id="1" cms-rate="11" clear="1"/>
        <template-data grp-id="2" profile-id="1" clear="1"/>
      </template>
    </setCommissionTemplate>
  </agents>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- `target-agent` – агент или субагент, для которого необходимо обновить шаблон комиссии (если не указан, выполняется для *текущего* агента);
- `template*` – тег изменяемого шаблона комиссии. Атрибут:
 - `id` – идентификатор шаблона;
 - `name` – название шаблона;
- `template-data` – тег, описывающий шаблон комиссии (см. описание атрибутов тега в разделе [6.3.9.2](#)). В запросе может присутствовать несколько таких тегов (если не указан, создается пустой шаблон).

6.3.12.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <agents>
    <setCommissionTemplate result="0" profile-id="12106511"/>
  </agents>
</response>
```

Ответ содержит следующие данные:

- `profile-id` – идентификатор шаблона.

6.3.13. addCommissionTemplate

Описание: Запрос позволяет добавить шаблон комиссии.

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.3.13.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <agents>
    <addCommissionTemplate>
      <target-agent>123</target-agent>
      <template name="xml-test">
        <template-data prv-id="1" cms-rate="10"/>
        <template-data grp-id="2" profile-id="1"/>
      </template>
    </addCommissionTemplate>
  </agents>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- `target-agent` – агент или субагент, для терминалов которого необходимо добавить шаблон комиссии (если не указан, выполняется для *текущего* агента);
- `template*` – тег шаблона комиссии. Атрибут:
 - `name` – название шаблона;
- `template-data` – тег, описывающий шаблон комиссии (см. описание атрибутов тега в разделе [6.3.9.2](#)). В запросе может присутствовать несколько таких тегов (если не указан, создается пустой шаблон).

6.3.13.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <agents>
    <addCommissionTemplate result="0" profile-id="1"/>
  </agents>
</response>
```

Ответ содержит следующие данные:

- `profile-id` – идентификатор профиля.

6.4. Providers

ПРИМЕЧАНИЕ



Обязательные параметры запросов отмечены символом *.

Список действий интерфейса:

Действие	Описание
authorizePayment	Авторизация платежа
authorizePaymentFromTemplate	Авторизация платежа по шаблону
confirmPayment	Подтверждение платежа
addOfflinePayment	Отправка платежа в режиме offline
getPaymentStatus	Получение статуса платежа
getRequiredPersonalData	Получение списка данных плательщика, необходимых для совершения платежа
checkPaymentRequisites	Проверка реквизитов платежа. Авторизация платежа без записи информации в базу.
getProviderByPhone	Определение сотового оператора по номеру телефона
getProviders	Получение списка доступных для агента провайдеров (провайдеров, в пользу которых агент может принимать платежи)
getGroups	Получение списка групп провайдеров
getUIGroups	Получение групп провайдеров (для формирования структуры пользовательского интерфейса терминалов)
getUIProviders	Получение файлов пользовательского интерфейса для терминала
getPhoneRanges	Получение номерных емкостей
getRegions	Получение списка номерных емкостей (регионы)
cancelPayment	Отмена платежа
interruptPayment	Отмена платежа в промежуточном статусе

6.4.1. authorizePayment

Описание: Запрос на авторизацию платежа.

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.4.1.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <providers>
    <authorizePayment>
      <payment id="301">
        <extras ev_paytype="5" PAY_TYPE="5" .../>
        <from amount="0.01" currency="643"/>
        <to service="2" account="1919" amount="0.01" currency="643" moneyType="1"/>
        <receipt id="132" date="2010-09-01T15:54:16"/>
      </payment>
    </authorizePayment>
  </providers>
</request>
```

Для выполнения запроса необходимо указать дополнительные параметры (формат параметров см. в п. [3.3.2.3](#)):

- **payment*** – тег, содержащий информацию о платеже. Включает атрибуты:
 - **id** – идентификатор платежа;
 - **comment** – комментарий (необязательный атрибут);
- **extras** – тег, в атрибутах которого указываются экстра-поля платежа. В одном теге может быть указано несколько экстра-полей в соответствии со спецификацией [интерфейса провайдера](#);
- **from*** – тег содержит информацию о сумме от клиента. Включает атрибуты:
 - **amount** – сумма;
 - **currency** – код валюты (значения кодов соответствуют **ISO 4217**);
- **to*** – тег содержит информацию о назначении платежа и сумме, которая должна быть перечислена на счет. Включает атрибуты:
 - **service** – идентификатор сервиса, услуги которого оплачиваются;
 - **account** – номер счета;
 - **amount** – сумма к зачислению;
 - **currency** – валюта;
 - **moneyType** – тип денежных средств, с помощью которых клиент совершил оплату. Возможные значения см. в [Приложении П](#);
- **receipt*** – тег, содержит информацию о чеке. Включает атрибуты:
 - **id** – идентификатор чека;
 - **date** – дата и время печати чека на терминале.

6.4.1.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <authorizePayment result="0">
      <payment date="2010-09-01T15:54:38+04:00" id="301" result="0"
        fatal="false" status="3" uid="7777777777777777">
        <extras AUTH_HOST="app8.osmp.ru" PARSER_HOST="app8.osmp.ru"/>
      </payment>
    </authorizePayment>
  </providers>
</response>
```

Ответ содержит следующие данные:

- `payment` – тег, содержащий информацию о платеже. Включает атрибуты:
 - `date` – дата и время (московские) приема платежа в процессинг;
 - `id` – идентификатор платежа;
 - `status` – текущий статус платежа (подробнее см. в разделе [4.3](#));
 - `result` – код ошибки проведения платежа (подробнее см. в [Приложении В](#));
 - `fatal` – признак фатальной ошибки:
 - ⊕ `true` - отрицательный ответ на платежный запрос;
 - ⊕ `false` - промежуточный статус обработки платежа;
 - `uid` – идентификатор транзакции в КИВИ.

ПРИМЕЧАНИЕ



Внутри тега `<payment>` может быть тег `<extras>`, содержащий в качестве атрибутов экстра-поля ответа.

В одном теге `<extras>` может быть указано несколько экстра-полей. Экстра-поля провайдера передаются агенту в соответствии со спецификацией интерфейса провайдера.

6.4.2. authorizePaymentFromTemplate

Описание: Авторизация платежа по шаблону.

Режим обработки: Строго синхронно.

Роли:

- Автомат
- Kassир
- Продавец

6.4.2.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <templates>
    <createPaymentTemplate result="0">
      <template createdDate="2022-12-16T18:20:47+03:00" id="110001175"
service="40">
        <fields account="1111111116" extra="extra_value2"
firstName="Иван" middleName="Иванович" secondName="Иванов"/>
      </template>
    </createPaymentTemplate>
  </templates>
</response>
```

- **payment*** – тег, содержащий данные платежа. Включает атрибуты:
 - **id*** – идентификатор платежа;
 - **to*** – тег содержит информацию о назначении платежа и сумме, которая должна быть перечислена на счет. Включает атрибуты:
 - ⊕ **amount** – сумма к зачислению;
 - ⊕ **currency** – код валюты (значения кодов соответствуют **ISO 4217**);
 - **from*** – тег содержит информацию о сумме от клиента. Включает атрибуты:
 - ⊕ **amount** – сумма;
 - ⊕ **currency** – код валюты (значения кодов соответствуют **ISO 4217**);
 - **template** – тег, содержащий данные шаблона. Включает атрибуты:
 - ⊕ **id*** – идентификатор шаблона.

6.4.2.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <authorizePaymentFromTemplate result="0">
      <payment date="2022-12-16T18:22:20+03:00" fatal="false"
id="2022121603" result="0" saved="true" status="3" uid="26774877907008">
        <extras extra="extra_value3"/>
      </payment>
    </authorizePaymentFromTemplate>
  </providers>
</response>
```

Ответ содержит следующие данные:

- **result** – код ошибки обработки запроса (подробнее см. в [Приложении В](#));
- **payment** – тег, содержащий информацию о платеже. Включает атрибуты:
 - **date** – дата и время (московские) приема платежа в процессинг;
 - **id** – идентификатор платежа;
 - **status** – текущий статус платежа (подробнее см. в разделе [4.3](#));
 - **result** – код ошибки проведения платежа (подробнее см. в [Приложении В](#));

- fatal – признак фатальной ошибки:
 - ⊕ true - отрицательный ответ на платежный запрос;
 - ⊕ false - промежуточный статус обработки платежа;
- uid – идентификатор транзакции в процессинге.

ПРИМЕЧАНИЕ

Внутри тега <payment> может быть тег <extras>, содержащий в качестве атрибутов экстра-поля ответа.

В одном теге <extras> может быть указано несколько экстра-полей. Экстра-поля провайдера передаются агенту в соответствии со спецификацией интерфейса провайдера.

6.4.3. confirmPayment

Описание: Запрос позволяет подтвердить платеж, авторизованный в результате выполнения запроса [authorizePayment](#).

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.4.3.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <providers>
    <confirmPayment>
      <payment id="301"/>
    </confirmPayment>
  </providers>
</request>
```

Для выполнения запроса необходимо указать дополнительные параметры:

- payment* – тег описывает один платеж. Включает атрибут:
 - id – идентификатор платежа (должен совпадать с id в предшествующем запросе [authorizePayment](#)).

ПРИМЕЧАНИЕ

Можно использовать несколько тегов payment внутри confirmPayment.

6.4.3.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <confirmPayment result="0">
      <payment date="2010-09-01T15:57:34+04:00" id="301"
        result="0" fatal="false" status="1" uid="777777777777">
        <extras AUTH_HOST="app8.osmp.ru"/>
      </payment>
    </confirmPayment>
  </providers>
</response>
```

Ответ содержит следующие данные:

- `payment` – тег, содержащий информацию о платеже. Включает атрибуты:
 - `date` – дата и время (московские) приема платежа в процессинг;
 - `id` – идентификатор платежа;
 - `status` – текущий статус платежа (подробнее см. в разделе [4.3](#));
 - `result` – код ошибки проведения платежа (подробнее см. в [Приложении В](#));
 - `fatal` – признак фатальной ошибки:
 - ⊕ `true` - отрицательный ответ на платежный запрос;
 - ⊕ `false` - промежуточный статус обработки платежа;
 - `uid` – идентификатор транзакции в КИВИ.

ПРИМЕЧАНИЕ



Внутри тега `<payment>` может быть тег `<extras>`, содержащий в качестве атрибутов экстра-поля ответа.

В одном теге `<extras>` может быть указано несколько экстра-полей. Экстра-поля провайдера передаются агенту в соответствии со спецификацией интерфейса провайдера.

6.4.4. addOfflinePayment

Описание: Запрос на добавление платежа (в режиме оффлайн).

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.4.4.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
```

```
<client terminal="111" software="Dealer v0" serial="" />
<providers>
  <addOfflinePayment>
    <payment id="301">
      <extras ev_paytype="5" PAY_TYPE="5".../>
      <from currency="643" amount="0.01"/>
      <to currency="643" service="3" amount="0.01" account="1111111111"
moneyType="1"/>
      <receipt id="1" date="2010-08-16T10:38:19"/>
    </payment>
  </addOfflinePayment>
</providers>
</request>
```

Для выполнения запроса необходимо указать дополнительные параметры (формат параметров см. в п. [3.3.2.3](#)):

- **payment*** – тег, содержащий информацию о платеже. Включает атрибуты:
 - **id** – идентификатор платежа;
 - **comment** – комментарий (необязательный атрибут);

ПРИМЕЧАНИЕ

Можно использовать несколько тегов **payment** внутри **addOfflinePayment**.

- **extras** – тег, в атрибутах которого указываются экстр-поля платежа. В одном теге может быть указано несколько экстр-полей в соответствии со спецификацией [интерфейса провайдера](#);
- **from*** – тег содержит информацию о сумме от клиента. Включает атрибуты:
 - **amount** – сумма;
 - **currency** – код валюты денежных средств (значения кодов соответствуют **ISO 4217**);
- **to*** – тег содержит информацию о назначении платежа и сумме, которая должна быть перечислена на счет. Включает атрибуты:
 - **service** – идентификатор сервиса, услуги которого оплачиваются;
 - **account** – номер счета;
 - **amount** – сумма к зачислению;
 - **currency** – код валюты суммы к зачислению (значения кодов соответствуют **ISO 4217**);
 - **moneyType** – тип денежных средств, с помощью которых клиент совершил оплату. Возможные значения см. в [Приложении П](#);
 - **receipt*** – тег, содержит информацию о чеке. Включает атрибуты:
 - **id** – идентификатор чека;
 - **date** – дата и время печати чека на терминале.

6.4.4.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <addOfflinePayment result="0">
      <payment date="2010-08-16T10:38:21+04:00" fatal="false" id="301"
        result="0" status="1" uid="7777777777777777"/>
    </addOfflinePayment>
  </providers>
</response>
```

Ответ содержит следующие данные:

- `payment` – тег, содержащий информацию о платеже. Включает атрибуты:
 - `date` – дата и время (московские) приема платежа в процессинг;
 - `id` – идентификатор платежа;
 - `status` – текущий статус платежа (подробнее см. в разделе [4.3](#));
 - `result` – код ошибки проведения платежа (подробнее см. в [Приложении В](#));
 - `fatal` – признак фатальной ошибки:
 - ⊕ `true` - отрицательный ответ на платежный запрос;
 - ⊕ `false` - промежуточный статус обработки платежа (дальнейшие действия зависят от конкретной ошибки);
 - `status` – статус проведения платежа (подробнее см. в разделе [4.3](#));
 - `uid` – идентификатор транзакции в КИВИ.

ПРИМЕЧАНИЕ



Внутри тега `<payment>` может быть тег `<extras>`, содержащий в качестве атрибутов экстра-поля ответа.

В одном теге `<extras>` может быть указано несколько экстра-полей. Экстра-поля провайдера передаются агенту в соответствии со спецификацией интерфейса провайдера.

6.4.5. `getPaymentStatus`

Описание: Запрос позволяет получить информацию о текущем статусе платежа.

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.4.5.1. Запрос

ВНИМАНИЕ

Запрос используется для получения статуса платежа во время его проведения, при этом в запросе должен быть указан тот же идентификатор терминала, который был указан в запросе на проведение этого платежа.

Информация о статусе платежа хранится не более 28 дней. Для получения статуса ранее проведенного платежа необходимо использовать запрос [getPayments](#).

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <providers>
    <getPaymentStatus>
      <payment id="301"/>
    </getPaymentStatus>
  </providers>
</request>
```

Для выполнения запроса необходимо указать дополнительные параметры (формат параметров см. в п. [3.3.2.3](#)):

- `payment*` – тег, содержащий информацию о платеже. Включает атрибут:
 - `id` – идентификатор платежа.

ПРИМЕЧАНИЕ

Для получения информации о нескольких платежах можно использовать несколько тегов `payment` внутри `getPaymentStatus`.

6.4.5.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <getPaymentStatus result="0">
      <payment date="2011-12-04T19:23:10+04:00" fatal="false" id="301"
        result="0" status="2" uid="10625592616002">
        <extras AUTH_HOST="app2.osmp.ru" PARSER_HOST="app5.osmp.ru"/>
      </payment>
    </getPaymentStatus>
  </providers>
</response>
```

Ответ содержит следующие данные:

- `payment` – тег, содержащий информацию о платеже. Включает атрибуты:
 - `date` – дата и время (московские) приема платежа в процессинг;
 - `id` – идентификатор платежа;

- status – текущий статус платежа (подробнее см. в разделе [4.3](#));
- result – код ошибки проведения платежа (подробнее см. в [Приложении В](#));
- fatal – признак фатальной ошибки (true - отрицательный ответ на платежный запрос; false - промежуточный статус обработки платежа);
- uid – идентификатор транзакции в КИВИ;
- value-date - дата и время (московские) списания суммы платежа с баланса агента.

6.4.6. getRequiredPersonalData

Описание: Получение списка данных плательщика, необходимых для совершения платежа.

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.4.6.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="****" sign="****" signAlg="MD5"/>
  <client terminal="****" software="****" serial="****"/>
  <providers>
    <getRequiredPersonalData>
      <payment id="1234567894">
        <from currency="643" amount="20000.00"/>
        <to currency="643" service="****" amount="20000.00" account="****"
moneyType="71"/>
      </payment>
    </getRequiredPersonalData>
  </providers>
</request>
```

Для выполнения запроса необходимо указать дополнительные параметры (формат параметров см. в п. [3.3.2.3](#)):

- payment* – тег, содержащий информацию о платеже. Включает атрибут:
 - id – идентификатор платежа;
- from* – тег содержит информацию о сумме от клиента. Включает атрибуты:
 - amount – сумма;
 - currency – код валюты денежных средств (значения кодов соответствуют **ISO 4217**);
- to* – тег содержит информацию о назначении платежа и сумме, которая должна быть перечислена на счет. Включает атрибуты:
 - currency – код валюты суммы к зачислению (значения кодов соответствуют **ISO 4217**);
 - service – идентификатор сервиса, услуги которого оплачиваются;
 - amount – сумма к зачислению;
 - account – номер счета;

- moneyType – тип денежных средств, с помощью которых клиент совершает оплату. Возможные значения см. в [Приложении П](#).

6.4.6.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <getRequiredPersonalData result="0">
      <field description="Отчество плательщика" name="payer_middle_name" />
      <field description="Номер счета плательщика" name="payer_account" />
      <field description="Фамилия плательщика" name="payer_last_name" />
      <field description="Имя плательщика" name="payer_first_name" />
      <choice>
        <field description="Адрес регистрации плательщика" name="payer_address"
      />
        <field description="ИНН плательщика" name="payer_inn" />
      </choice>
    </getRequiredPersonalData>
  </providers>
</response>
```

Ответ содержит следующие данные:

- result – код ошибки обработки запроса (подробнее см. в [Приложении В](#));
- field – тег, содержащий описание элемента списка данных плательщика, необходимых для совершения платежа. Включает атрибуты:
 - description – описание элемента списка;
 - name – наименование экстра-поля платежа, в котором следует передать указанные в description сведения;
 - choice – групповое поле, содержащее описание двух и более элементов списка данных плательщика в атрибутах field, при этом для совершения платежа достаточно передать один из этих элементов (любой).

ПРИМЕЧАНИЕ



Если для совершения платежа данные плательщика передавать не требуется, тег field в ответе отсутствует.

6.4.7. checkPaymentRequisites

Описание: Авторизация платежа без записи информации в базу.

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.4.7.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <providers>
    <checkPaymentRequisites>
      <payment id="501">
        <from currency="643" amount="11.00"/>
        <to currency="643" service="2" amount="11.00" account="111" moneyType="1" />
        <receipt id="132" date="2010-08-16T12:43:01"/>
      </payment>
    </checkPaymentRequisites>
  </providers>
</request>
```

Для выполнения запроса необходимо указать дополнительные параметры (формат параметров см. в п. [3.3.2.3](#)):

- **payment*** – тег, содержащий информацию о платеже. Включает атрибут:
 - **id** – идентификатор платежа;
- **extras** – тег, в атрибутах которого указываются экстр-поля платежа. В одном теге может быть указано несколько экстр-полей в соответствии со спецификацией [интерфейса провайдера](#);
- **from*** – тег содержит информацию о сумме от клиента. Включает атрибуты:
 - **amount** – сумма;
 - **currency** – код валюты денежных средств (значения кодов соответствуют **ISO 4217**);
- **to*** – тег содержит информацию о назначении платежа и сумме, которая должна быть перечислена на счет. Включает атрибуты:
 - **service** – идентификатор сервиса, услуги которого оплачиваются;
 - **account** – номер счета;
 - **amount** – сумма к зачислению;
 - **currency** – код валюты (значения кодов соответствуют **ISO 4217**);
 - **moneyType** – тип денежных средств, с помощью которых клиент совершил оплату. Возможные значения см. в [Приложении П](#);
 - **receipt*** – тег, содержит информацию о чеке. Включает атрибуты:
 - **id** – идентификатор чека;
 - **date** – дата и время печати чека на терминале.

6.4.7.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <checkPaymentRequisites result="0">
      <payment date="2010-08-16T12:43:02+04:00" id="501" fatal="true"
        result="5" status="0" uid="7777777777777777">
        <extras AUTH_HOST="app9.osmp.ru" disp1="[param1]"
          disp2="[param2]"/>
      </payment>
    </checkPaymentRequisites>
  </providers>
</response>
```

Ответ содержит следующие данные:

- `payment` – тег, содержащий информацию о платеже. Включает атрибуты:
 - `date` – дата и время (московские) приема платежа в процессинг;
 - `id` – идентификатор платежа;
 - `status` – текущий статус платежа (подробнее см. в разделе [4.3](#));
 - `result` – код ошибки проведения платежа (подробнее см. в [Приложении В](#));
 - `fatal` – признак фатальной ошибки:
 - ⊕ `true` - отрицательный ответ на платежный запрос;
 - ⊕ `false` - промежуточный статус обработки платежа;
 - `uid` – идентификатор транзакции в КИВИ.

ПРИМЕЧАНИЕ



Внутри тега `<payment>` может быть тег `<extras>`, содержащий в качестве атрибутов экстра-поля ответа.

В одном теге `<extras>` может быть указано несколько экстра-полей. Экстра-поля провайдера передаются агенту в соответствии со спецификацией интерфейса провайдера.

6.4.8. `getProviderByPhone`

- Описание:** Запрос позволяет определить оператора сотовой связи абонента по номеру телефона.
- Режим обработки:** Строго синхронно.
- Роли:** Выполнение запроса доступно персоне с любой ролью.

6.4.8.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <providers>
    <getProviderByPhone>
      <phone>9081112233</phone>
    </getProviderByPhone>
  </providers>
</request>
```

Для выполнения запроса необходимо указать параметр:

- `phone` – номер телефона для определения принадлежности оператору, в 10-значном формате.

6.4.8.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <getProviderByPhone result="0">
      <providerId>2</providerId>
      <regionId>67</regionId>
      <isPorted>true</isPorted>
    </getProviderByPhone>
  </providers>
</response>
```

Ответ содержит следующие данные:

- `providerId` – идентификатор оператора сотовой связи (провайдера), которому принадлежит номер абонента;
- `regionId` – идентификатор региона деятельности оператора (если есть);
- `isPorted` – признак переноса номера (`true` – номер был перенесен от другого оператора, `false` – номер не перенесен).

6.4.9. getProviders

<u>Описание:</u>	Запрос позволяет получить список провайдеров, в пользу которых <i>текущий</i> агент может принимать платежи.
<u>Режим обработки:</u>	Возможно асинхронно.
<u>Роли:</u>	Выполнение запроса доступно персоне с любой ролью.

6.4.9.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <providers>
    <getProviders>
      <target-agent>4416631</target-agent>
    </getProviders>
  </providers>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-terminal` – терминал, для которого необходимо получить список разрешенных провайдеров (если не указан, выполняется для *текущего* терминала);
- `target-agent` – агент, для которого необходимо получить список разрешенных провайдеров.

ВНИМАНИЕ



Не допускается в одном запросе использовать оба параметра (`target-terminal` и `target-agent`).

6.4.9.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <getProviders result="0">
      <row fiscal-name="ТВ" long-name="" max-amount="15000" min-amount="1"
        prv-id="1" prv-inn="1111111111" prv-support-phone="8(495)111-11-11"
        receipt-name="" short-name="Короткое название"/>
    </getProviders>
  </providers>
</response>
```

Ответ содержит следующие данные:

- `row` – тег, содержащий информацию о провайдере. Тег содержит атрибуты:
 - `fiscal-name` – наименование предоставляемого сервиса, которое печатается на чеке фискальным регистратором;
 - `long-name` – полное название провайдера;
 - `max-amount` – ограничение на максимальную сумму платежа в пользу провайдера;
 - `min-amount` – ограничение на минимальную сумму платежа в пользу провайдера;
 - `prv-id` – идентификатор провайдера;
 - `prv-inn` – ИНН провайдера;
 - `prv-support-phone` – телефон технической поддержки провайдера;
 - `receipt-name` – наименование предоставляемого сервиса, которое печатается на чеке;
 - `short-name` – короткое название провайдера.

6.4.10. getGroups

Описание: Получение списка групп провайдеров.

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.4.10.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <providers>
    <getGroups language="ru"/>
  </providers>
</request>
```

Для выполнения запроса можно указать дополнительный атрибут в теге действия:

- `language` – язык пользовательского интерфейса (если не указан, подставляется язык по умолчанию для страны, к которой относится *текущий* агент).
- Значение параметра соответствует коду языка по стандарту **ISO 639-1** (двухбуквенное обозначение; например, `en` – английский, `ru` – русский). Общий справочник языковых кодов можно посмотреть [здесь](#).

6.4.10.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <getGroups result="0">
      <group id="20" logo="cellular.gif" name="Сотовая связь" orderId="1"
parentId="1"/>
      <group id="21" logo="megafon.gif" name="Мегафон" orderId="1" parentId="20"/>
      <group id="30" name="Банки" orderId="1" parentId="1"/>
      ...
    </getGroups>
  </providers>
</response>
```

Ответ содержит следующие данные:

- `result` – результат выполнения запроса;
- `group` – тег, содержащий описание группы провайдеров. Атрибуты тега:
 - `id` – идентификатор группы;
 - `logo` – логотип группы;
 - `name` – название группы;
 - `orderId` – порядок сортировки групп при отображении на терминале;

- `parentId` – идентификатор группы, в которую вложена текущая группа.

6.4.11. getUIGroups

Описание: Запрос позволяет получить список групп (организационную структуру) для формирования пользовательского интерфейса терминалов (подробнее читайте в разделе [5.1](#)).
Запрос возвращает только разрешенных для отображения на текущем терминале провайдеров.

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

ПРИМЕЧАНИЕ



В стандартной ситуации данный запрос необходимо выполнять одновременно с запросом [getUIProviders](#).

6.4.11.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <providers>
    <getUIGroups/>
  </providers>
</request>
```

Для выполнения запроса можно указать дополнительный атрибут в теге действия:

- `language` – язык пользовательского интерфейса (если не указан, подставляется язык для *текущего* терминала). Принимает значения:
 - `en` – английский;
 - `ru` – русский.

6.4.11.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <getUIGroups result="0">
      <group id="103" logo="logos/!anonymous/grp103Icon_internet_telephons.gif"
        logo_crc="fdf52353" logo_size="10744" name="Интернет и телефония" orderId="16"
        tag="visible">
        <logos>
          <logo crc="fdf52353"
            path="logos/!anonymous/grp103Icon_internet_telephons.gif" size="10744"
            type="standard"/>
        </logos>
      </provider id="2" orderId="2" showInTop="2" tag="visible"/>
    </getUIGroups>
  </providers>
</response>
```

```

    <group id="21" logo="logos!/anonymous/grp21megafon.gif" logo_crc="22"
logo_size="2502" name="Мегафон" orderId="3" tag="visible,ranges">
      <logos>
        <logo crc="ed13c21f" path="logos!/anonymous/grp113IP.gif" size="6496"
type="standard"/>
      </logos>
      <provider id="3" orderId="4" showInTop="1" tag="visible"/>
      <provider id="552" orderId="5" tag="visible"/>
      <provider id="29" orderId="6" tag="visible"/>
    </group>
    <provider id="507" orderId="16" tag="visible"/>
    <provider id="1" orderId="17" showInTop="3" tag="visible"/>
  </group>
</getUIGroups>
</providers>
</response>

```

Ответ содержит следующие данные:

- `group` – тег, содержащий информацию о группе провайдеров в интерфейсе.

ПРИМЕЧАНИЕ



Данный тег может включать вложенные теги `group` или `provider`.

Тег включает атрибуты:

- `id` – уникальный (в рамках интерфейса) идентификатор группы;
- `logo` – название файла логотипа группы провайдеров;

ПРИМЕЧАНИЕ



Поле `logo` содержит путь относительно каталога логотипов. Путь к каталогу логотипов задается в клиентском приложении.

- `logo_crc` – контрольная сумма файла (алгоритм `crc32`);
- `logo_size` – размер файла логотипа;
- `name` – название группы;
- `orderId` – идентификатор, указывающий на место группы в интерфейсе (внутри другой группы);
- `charity_logo` – название файла с логотипом благотворительного фонда;
- `charity_logo_crc` – контрольная сумма файла (алгоритм `crc32`);
- `charity_logo_size` – размер файла;
- `tag` – свойство группы провайдера:
 - ⊕ `"ranges"` – группа сотовых операторов,
 - ⊕ `"visible"` – признак отображения группы провайдеров в региональном интерфейсе терминала,
 - ⊕ `"commissions"` – с платежа в пользу провайдеров данной группы взимается 2 комиссии (например, при оплате кредита с платежа взимается комиссия агента, а затем – комиссия банка),

- ⊕ "charity" – данная группа объединяет благотворительные фонды,
- ⊕ "promo" – данная группа провайдеров входит в группу QIWI-promo интерфейс,
- ⊕ "empty" – провайдеры, относящиеся к текущей группе, в интерфейсе отображены не будут,
- logos – тег, содержащий данные по логотипам, привязанным к группе (логотипов может быть несколько):
 - logo – данные о каждом логотипе группы:
 - ⊕ crc – контрольная сумма файла (алгоритм crc32);
 - ⊕ path – полное название файла с логотипом;
 - ⊕ size – размер файла;
 - ⊕ type – тип логотипа, определяющий его использование при построении интерфейса:
 - ❖ "standard" – стандартный логотип;
 - ❖ "top8" – для ТОП-8;
 - ❖ "charity" – кнопка для благотворительности;
 - ❖ "index" – кнопка главного экрана (реализована возможности управления кнопками главного экрана приложения);
- provider – тег, содержащий информацию о провайдере. Включает атрибуты:
 - id – идентификатор провайдера,
 - orderId – идентификатор, указывающий на место провайдера в группе,

ПРИМЕЧАНИЕ

Более подробная информация о провайдерах передается в запросе [getUIProviders](#).

- showInTop – идентификатор провайдера в списке TOP 8,
- tag – свойство провайдера:
 - ⊕ "ranges" – данный провайдер входит в группу сотовых операторов;
 - ⊕ "visible" – признак отображения провайдера в региональном интерфейсе терминала.
 - ⊕ "promo" – данный провайдер входит в группу QIWI-promo интерфейс;
 - ⊕ "empty" – провайдер входит в группу, которая отображается в интерфейсе как пустая, и в интерфейсе он показан не будет;
 - ⊕ "charity" – данный провайдер является благотворительным фондом;
 - ⊕ "hideInTop8" – провайдер, отмеченный данным признаком, не будет отображен в списке ТОП8 вне зависимости от его рейтинга.

6.4.12. getUIProviders

Описание:

Запрос позволяет получить файлы пользовательского интерфейса для терминалов (подробнее читайте в разделе [5.1](#)).

Запрос возвращает только разрешенных для отображения на терминале провайдеров.

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.4.12.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <providers>
    <getUIProviders/>
  </providers>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.4.12.2. Ответ

```

<?xml version="1.0" encoding="utf-8"?>
<response result="0">
<providers>
  <getUIProviders result="0">
    <provider charity_logo="logos/!anonymous/prv2069help_promo.jpg"
charity_logo_crc="3dd0e81e" charity_logo_size="152799" fiscalName="Сотовая св."
grpId="20" id="1" inn="7740000076" jDocDate="18.07.2003" jDocNum="2599/03"
jName="ОАО &quot;Мобильные ТелеСистемы&quot;" keywords="МТС" lName="ОАО
&quot;Мобильные ТелеСистемы&quot;" logo="logos/!anonymous/prv11.gif"
logo_crc="b9410bc4" logo_size="2505" receiptName="Сотовая связь" sName="МТС "
supportPhone="8(800) 333-08-90" tag="visible,овал,ranges" curId="643" >
    <logos>
      <logo crc="b9410bc4" path="logos/!anonymous/prv11.gif" size="2505"
type="standard"/>
      <logo crc="3dd0e81e" path="logos/!anonymous/prv2069help_promo.jpg"
size="152799" type="charity"/>
    </logos>
    <constParams>
      <param name="CITY" value="asdfsdfdf"/>
    </constParams>
    <pages>
      <page pageId="1111" orderId="2">
        <controls>
          <control type="keyboard" orderId="1" layout="DGT"/>
        </controls>
      </page>
      <page pageId="23" orderId="1" nextPage="-1" successPage="12" failPage="12"
progressPage="12" pageType="input_page" pageFile="12" title="12" useOnline="true">
        <controls>
          <control type="keyboard" orderId="1" layout="DGT"/>
          <control type="text_input" orderId="2" header="12" name="12"
regex="12"/>
        </controls>
      </page>
    </pages>
  </provider>
</getUIProviders>
</providers>
</response>

```

Ответ содержит следующие данные:

- `provider` – интерфейс провайдера. Атрибуты тега перечислены в [Табл. 3](#).

Табл. 3. Атрибуты тега `provider`

Атрибут	Назначение
<code>contryId</code>	идентификатор страны, где предоставляется сервис;
<code>buttonName</code>	короткое название провайдера;
<code>charity_logo</code>	название файла с изображением для работы сервиса;
<code>charity_logo_crc</code>	контрольная сумма файла с изображением (алгоритм <code>crc32</code>);
<code>charity_logo_size</code>	размер файла;
<code>fiscalName</code>	наименование предоставляемого сервиса, которое печатается на чеке фискальным регистратором;

Атрибут	Назначение
grpId	идентификатор группы провайдеров, в которую входит данный провайдер;
id	идентификатор сервиса;
inn	ИНН провайдера;
jDocDate	дата заключения договора с провайдером;
jDocNum	номер договора с провайдером;
jName	юридическое название провайдера;
keywords	список ключевых слов через " " для поиска провайдера;
lName	коммерческое (полное) наименование провайдера;
logo	название файла с логотипом сервиса. Поле logo содержит путь относительно каталога логотипов. Путь к каталогу логотипов задается в клиентском приложении.
logo_crc	контрольная сумма файла (алгоритм crc32);
logo_size	размер файла;
maxSum	максимальная сумма платежа в целом по провайдеру;
minSum	минимальная сумма платежа в целом по провайдеру;
prvPage	страница, на которую необходимо перейти по запросам для данного провайдера;
receiptName	наименование предоставляемого сервиса, которое печатается на чеке;
sName	торговая марка провайдера;
small_logo	название файла с уменьшенным логотипом провайдера;
small_logo_crc	контрольная сумма файла с уменьшенным логотипом (алгоритм crc32);
small_logo_size	размер файла с уменьшенным логотипом провайдера;
supportPhone	телефон поддержки провайдера;

Атрибут	Назначение
tag	<p>свойство провайдера:</p> <ul style="list-style-type: none"> "ranges" – данный провайдер входит в группу сотовых операторов; "visible" – признак присутствия провайдера в регионе терминала. Если тег visible для провайдера не указан, то провайдер не должен отображаться в региональном интерфейсе. Однако его можно найти через функцию поиска (см. «Описание регионального интерфейса АСО») "commissions" – с платежа в пользу данного провайдера взимаются две комиссии (например, при оплате кредита с платежа взимается комиссия агента, а затем – комиссия банка); "change" – признак отображения страницы перечисления сдачи с платежа в пользу данного провайдера (платеж с фиксированной суммой); "charity" – данный провайдер является благотворительным фондом; "promo" – данный провайдер входит в группу QIWI-promo интерфейс; "empty" – провайдер входит в группу, которая отображается в интерфейсе как пустая, и в интерфейсе он показан не будет; "hideInTop8" – провайдер, отмеченный данным признаком, не будет отображен в списке ТОП8 вне зависимости от его рейтинга;
useSpecifyAccount	атрибут предназначен для реализации сложных проверок данных провайдера; Подавляющее большинство проверок реализуется с использованием регулярных выражений.
curId	код валюты провайдера

- logos – тег, содержащий данные по логотипам, привязанным к провайдеру (логотипов может быть несколько):

⊕ logo – данные об 1 логотипе. Атрибуты тега см. в [Табл. 4](#).

Табл. 4. Атрибуты тега logo

Атрибут	Назначение
crc	контрольная сумма файла (алгоритм crc32);
path	полное название файла с логотипом;
size	размер файла;
type	<p>тип логотипа, определяющий его использование при построении интерфейса:</p> <ul style="list-style-type: none"> "standard" – стандартный логотип; "top8" – для ТОП-8; "charity" – кнопка для благотворительности; "index" – кнопка главного экрана (реализована возможности управления кнопками главного экрана приложения);

- `constParams` – тег, содержащий описание экстра-полей, которые необходимо передавать в запросах на авторизацию и проведение платежей в пользу данного провайдера. Содержит теги `param`:
 - ⊕ `param` – тег, содержащий описание экстра-поля. Атрибуты тега:

Табл. 5. Атрибуты тега `param`

Атрибут	Назначение
<code>name</code>	<p>название экстра-поля для передачи в процессинг.</p> <p>Параметр с <code>name="EmbedParams"</code> изменяет поведение ПО. Значение такого параметра (атрибут <code>value</code>) передается странице, указанной в атрибуте <code>prvPage</code> тега <code>provider</code>.</p> <p>Параметр с <code>name="real_prv_id"</code> определяет идентификатор сервиса провайдера для платежных запросов addOfflinePayment, authorizePayment, checkPaymentRequisites. Значение такого параметра (атрибут <code>value</code>) передается в атрибуте <code>service</code> указанных запросов для платежей в пользу данного провайдера;</p>
<code>value</code>	значение экстра-поля;

- `pages` – тег, содержащий информацию о страницах интерфейса данного провайдера. Содержит теги:
 - ⊕ `page` – тег, описывающий страницу интерфейса. Атрибуты тега:

Табл. 6. Атрибуты тега `page`

Атрибут	Назначение
<code>pageId</code>	идентификатор страницы интерфейса;
<code>orderId</code>	идентификатор, задающий порядок отображения страниц интерфейса в соответствии со сценарием платежа;
<code>nextPage</code>	идентификатор следующей страницы (<code>nextPage="-1"</code> – признак последней страницы);
<code>successPage</code>	название страницы в случае успешной авторизации;
<code>failPage</code>	название страницы в случае неуспешной авторизации;
<code>progressPage</code>	признак отображения страницы с progress bar;
<code>pageType</code>	<p>шаблон страницы:</p> <ul style="list-style-type: none"> • <code>"input_page"</code> – страница с клавиатурой и одним полем ввода; • <code>"buttons_page"</code> – страница с 2 кнопками; • <code>"info_page"</code> – страница с дополнительной информацией (отображение ассоциированных полей);
<code>pageFile</code>	название файла страницы;

Атрибут	Назначение
<code>title</code>	подсказка для пользователя с краткими указаниями того, что необходимо сделать;
<code>useOnline</code>	включение механизма онлайн проверки данных страницы. В сценарии платежа после данной страницы необходимо отправить запрос checkPaymentRequisites с параметрами платежа, собранными на всех предшествующих страницах интерфейса (т.е. где <code>orderId ≤ orderId</code> текущей страницы);

Каждый тег `page` содержит вложенные теги:

- ❖ `controls` – тег, содержащий информацию об элементах управления, расположенных на странице. Содержит теги `control`:
- `control` – тег, описывающий элемент управления. Тип управляющего элемента определяется атрибутом `type`:

Табл. 7. Типы управляющих элементов

Тип контрола	Назначение
<code>type="keyboard"</code>	Виртуальная клавиатура (см. Табл. 8)
<code>type="text_input"</code>	Поле ввода (см. Табл. 9)
<code>type="disp_input"</code>	Ассоциативное поле ввода (см. Табл. 10). Ассоциативное поле отличается от обычного поля ввода (" <code>text_input</code> ") тем, что оно нередактируемое и имеет ассоциацию с экстра-полем, которое, как правило, возвращается в ответе при онлайн проверке реквизитов платежа
<code>type="button"</code>	Кнопка (см. Табл. 11)
<code>type="disp_button"</code>	Ассоциативная кнопка (см. Табл. 12). Ассоциативная кнопка может принимать значения экстра-полей (в отличие от " <code>button</code> "), которые, как правило, возвращаются в ответе при онлайн проверке реквизитов платежа
<code>type="combobox"</code>	Поле со списком (см. Табл. 13)
<code>type="disp_combobox"</code>	Ассоциативное поле со списком (см. Табл. 14). Ассоциативное поле может принимать значения экстра-полей (в отличие от " <code>combobox</code> "), которые, как правило, возвращаются в ответе при онлайн проверке реквизитов платежа
<code>type="disp_label"</code>	Ассоциативное текстовое поле (см. Табл. 15). Ассоциативное поле принимает значения экстра-полей, которые, как правило, возвращаются в ответе при онлайн проверке реквизитов платежа

Набор атрибутов тега `control` зависит от типа управляющего элемента.

Табл. 8. Атрибуты виртуальной клавиатуры "keyboard"

Атрибут	Назначение
<code>orderId</code>	Идентификатор, задающий порядок отображения элемента на странице

Атрибут	Назначение
<code>layout</code>	<p>Тип виртуальной клавиатуры:</p> <ul style="list-style-type: none"> • <code>"DG"</code> – цифровая, • <code>"DGT"</code> – цифровая с кнопкой Tab, • <code>"DGD"</code> – цифровая с точкой, • <code>"AL"</code> – буквенная клавиатура с английской раскладкой (маленькие буквы), • <code>"ALR"</code> – буквенная клавиатура с русской раскладкой (маленькие буквы), • <code>"ALC"</code> – буквенная клавиатура с английской раскладкой (прописные буквы), • <code>"ALCR"</code> – буквенная клавиатура с русской раскладкой (прописные буквы).

Табл. 9. Атрибуты поля ввода "text_input"

Атрибут	Назначение
<code>orderId</code>	Идентификатор, задающий порядок отображения элемента на странице
<code>errMess</code>	Текст ошибки, отображаемой в случае, если введенное в поле значение не прошло проверку по регулярному выражению
<code>footer</code>	Текст под полем ввода
<code>header</code>	Текст над полем ввода
<code>mask</code>	Маска поля (подробнее см. в Приложении А)
<code>name</code>	Название поля для передачи в процессинг (на латинице)
<code>nobr</code>	<p>Способ отображения полей ввода:</p> <ul style="list-style-type: none"> • <code>"true"</code> – с переводом на следующую строку, • <code>"false"</code> – расположить элементы на одной строке вне зависимости от размера монитора (для многосоставных полей). Многосоставное поле – совокупность связанных полей ввода, текст из которых объединяется и обрабатывается в процессинге как единая строка.
<code>regexp</code>	Регулярное выражение (подробнее см. в Приложении Б)
<code>strip</code>	Признак передачи в процессинг введенного в поле значения без разделителей (<code>true/false</code>). Например, при передаче номера телефона, записанного как (903) 123-45-67, строка, передаваемая в процессинг, должна быть: 9031234567
<code>disp_desc</code>	<p>Символьная константа, выводимая на чеке перед значением экстра-поля (может отсутствовать).</p> <p>«Двоеточие + пробел» после значения атрибута проставляется автоматически на чеке. Если атрибут отсутствует, «двоеточие+пробел» не выводится на печать.</p>

Атрибут	Назначение
<code>disp_type</code>	<p>Атрибут указывает на то, как использовать введенное в поле ввода значение и значение атрибута <code>disp_desc</code>. Возможные значения <code>disp_type</code>:</p> <ul style="list-style-type: none"> <code>"extra"</code> – передавать в процессинг экстра-поле с названием из <code>name</code> и введенным значением. Значение обрабатывается в соответствии с атрибутом <code>strip</code> и затем проверяется по регулярному выражению из <code>regexp</code>, если выражение задано. При несоответствии регулярному выражению переход к следующему шагу сценария оплаты невозможен; <code>"receipt"</code> – печать данных на чеке; <code>"fixedsum"</code> – фиксированная сумма платежа; <code>"private"</code> – признак того, что данный элемент управления не должен быть отображен на экране / напечатан на чеке; <code>"sensitive"</code> – признак того, что введенное в поле значение относится к конфиденциальным данным и при печати на чеке должно быть маскировано согласно требованиям PA-DSS. <p>Если страница <code>page</code> содержит единственный <code>control</code> с параметром <code>"private"</code> и тип терминала не ACO, то данная страница пропускается, выполняется следующий шаг сценария оплаты.</p>
<code>group_id</code>	<p>Идентификатор элемента <code>"combobox"</code> или <code>"disp_combobox"</code>, с которым связано данное поле.</p> <p>Если атрибут <code>group_id</code> с тем же значением указан в элементе <code>"combobox"</code> / <code>"disp_combobox"</code> на той же странице интерфейса (Табл. 6), то поле ввода относится к соответствующему списку и должно отображаться на странице рядом с этим списком.</p>
<code>show_cond</code>	<p>Номер элемента списка в связанном элементе <code>"combobox"</code> / <code>"disp_combobox"</code> (по атрибуту <code>group_id</code>), которому соответствует данное поле ввода. Т.е. поле ввода отображается только при выборе указанного элемента в списке (если атрибут отсутствует, то поле отображается всегда).</p>

Табл. 10. Атрибуты ассоциативного поля ввода `"disp_input"`

Атрибут	Назначение
<code>orderId</code>	Идентификатор, задающий порядок отображения элемента на странице
<code>errMess</code>	Текст ошибки, отображаемой в случае, если значение поля не прошло проверку по регулярному выражению
<code>footer</code>	Текст под полем ввода
<code>header</code>	Текст над полем ввода
<code>mask</code>	Маска поля (подробнее см. в Приложении А)
<code>name</code>	Название поля для передачи в процессинг (на латинице)

Атрибут	Назначение
<code>nobr</code>	Способ отображения полей ввода: <ul style="list-style-type: none"> <code>"true"</code> – с переводом на следующую строку, <code>"false"</code> – расположить элементы на одной строке вне зависимости от размера монитора (для многосоставных полей); Многосоставное поле – совокупность связанных полей ввода, текст из которых объединяется и обрабатывается в процессинге как единая строка.
<code>regexp</code>	Регулярное выражение (подробнее см. в Приложении Б)
<code>strip</code>	Признак передачи в процессинг отображенного в поле значения без разделителей (<code>true/false</code>). Например, при передаче номера телефона, записанного как (903) 123-45-67, строка, передаваемая в процессинг, должна быть: 9031234567.
<code>disp_desc</code>	Символьная константа, выводимая на чеке перед значением экстра-поля (может отсутствовать). «Двоеточие + пробел» после значения атрибута проставляется автоматически на чеке. Если атрибут отсутствует, «двоеточие+пробел» не выводится на печать.
<code>disp_type</code>	Атрибут указывает на то, как использовать отображенное в поле ввода значение и значение атрибута <code>disp_desc</code> . Возможные значения <code>disp_type</code> : <ul style="list-style-type: none"> <code>"extra"</code> – передавать в процессинг экстра-поле с названием из <code>name</code> и значением из ассоциированной экстра в <code>disp_name</code>; <code>"receipt"</code> – печать данных на чеке; <code>"fixedsum"</code> – фиксированная сумма платежа; <code>"minsum"</code> – минимальная сумма платежа; <code>"maxsum"</code> – максимальная сумма платежа; <code>"private"</code> – признак того, что данный элемент управления не должен быть отображен на экране / напечатан на чеке. Атрибут игнорируется только ACO. Другие типы терминалов должны проверять наличие данного поля. Если страница <code>page</code> содержит единственный <code>control</code> с параметром <code>"private"</code> и тип терминала не ACO, то данная страница пропускается, выполняется следующий шаг сценария оплаты.
<code>disp_name</code>	Содержит название ассоциированного экстра-поля. Значение экстра-поля проверяется по регулярному выражению из <code>regexp</code> , если выражение задано. При несоответствии регулярному выражению переход к следующему шагу сценария оплаты невозможен. Обработывается в соответствии с <code>disp_type</code> .
<code>visible</code>	Видимость элемента в интерфейсе (<code>visible="false"</code> – элемент не должен быть отображен в интерфейсе; <code>visible="true"</code> или отсутствует – элемент должен быть отображен в интерфейсе)

Табл. 11. Атрибуты кнопки "button"

Атрибут	Назначение
<code>orderId</code>	Идентификатор, задающий порядок отображения элемента на странице
<code>altName</code>	Текст, отображаемый на кнопке

Атрибут	Назначение
<code>img</code>	Название файла изображения, отображаемого на кнопке
<code>pageId</code>	Идентификатор страницы, на которую выполняется переход после нажатия на кнопку
<code>disp_desc</code>	Символьная константа, выводимая на чеке перед значением экстра-поля (может отсутствовать). «Двоеточие + пробел» после значения атрибута проставляется автоматически на чеке. Если атрибут отсутствует, «двоеточие + пробел» не выводится на печать.
<code>disp_type</code>	Атрибут указывает на то, как использовать кнопку и значение атрибута <code>disp_desc</code> . Возможные значения <code>disp_type</code> : <ul style="list-style-type: none"> • <code>"extra"</code> – передавать в процессинг экстра-поля из вложенного тега <code>param</code> с названием из <code>name</code> и значением из <code>value</code> (см. ниже); • <code>"receipt"</code> – печать данных на чеке; • <code>"fixedsum"</code> – фиксированная сумма платежа; • <code>"private"</code> – признак того, что данный элемент управления не должен быть отображен на экране / напечатан на чеке. Атрибут игнорируется только АСО. Другие типы терминалов должны проверять наличие данного поля. Если страница <code>page</code> содержит единственный <code>control</code> с параметром <code>"private"</code> и тип терминала не АСО, то данная страница пропускается, выполняется следующий шаг сценария оплаты.
<code>visible</code>	Видимость элемента в интерфейсе (<code>visible="false"</code> – элемент не должен быть отображен в интерфейсе; <code>visible="true"</code> или отсутствует – элемент должен быть отображен в интерфейсе)

У элемента кнопки также могут быть вложенные теги:

- `param` – тег, описывающий параметры, передаваемые в клиентское приложение при нажатии на кнопку. Атрибуты:
 - `name` – название параметра;
 - `value` – значение параметра;

Табл. 12. Атрибуты ассоциативной кнопки "disp_button"

Атрибут	Назначение
<code>orderId</code>	Идентификатор, задающий порядок отображения элемента на странице
<code>altName</code>	Отображаемый на кнопке текст
<code>img</code>	Название файла изображения, отображаемого на кнопке
<code>pageId</code>	Идентификатор страницы, на которую осуществляется переход после нажатия на кнопку
<code>disp_desc</code>	Символьная константа, выводимая на чеке перед значением экстра-поля, ассоциированного с элементом (может отсутствовать). «Двоеточие + пробел» после значения атрибута проставляется автоматически на чеке. Если атрибут отсутствует, «двоеточие + пробел» не выводится на печать

Атрибут	Назначение
<code>disp_type</code>	<p>Атрибут указывает на то, как использовать кнопку и значение атрибута <code>disp_desc</code>. Возможные значения <code>disp_type</code>:</p> <ul style="list-style-type: none"> <code>"extra"</code> – передавать в процессинг экстра-поле с названием из <code>name</code> и значением из ассоциированной экстра в <code>disp_name</code>; <code>"receipt"</code> – печать данных на чеке; <code>"fixedsum"</code> – фиксированная сумма; <code>"private"</code> – признак того, что данный элемент управления не должен быть отображен на экране / напечатан на чеке; <p>Атрибут игнорируется только АСО. Другие типы терминалов должны проверять наличие данного поля.</p> <p>Если страница <code>page</code> содержит единственный <code>control</code> с параметром <code>"private"</code> и тип терминала не АСО, то данная страница пропускается, выполняется следующий шаг сценария оплаты.</p>
<code>disp_name</code>	<p>Название экстра-поля, ассоциированного с кнопкой.</p> <p>Значение экстра-поля, полученное из процессинга, отображается в качестве названия кнопки (если не указан атрибут <code>altName</code>). Данное поле обязательно. Обработывается в соответствии с <code>disp_type</code>.</p>
<code>name</code>	<ul style="list-style-type: none"> Название кнопки для передачи в процессинг (на латинице). Используется в качестве имени экстра-поля со значением из ассоциированного экстра-поля в <code>disp_name</code>.
<code>visible</code>	<p>Видимость элемента в интерфейсе (<code>visible="false"</code> – элемент не должен быть отображен в интерфейсе; <code>visible="true"</code> или отсутствует – элемент должен быть отображен в интерфейсе);</p>

У элемента ассоциативной кнопки также могут быть вложенные теги:

- `param` – тег, описывающий параметры, передаваемые в клиентское приложение при нажатии на кнопку. Атрибуты:
 - `name` – название параметра;
 - `value` – значение параметра;

Табл. 13. Атрибуты поля со списком "combobox"

Атрибут	Назначение
<code>disp_desc</code>	<p>Список для передачи в терминал. Формат списка:</p> <pre>"<элемент_списка_1>[\;ID_элемента_1[\;сумма_платежа_1]] <элемент_списка_2>[\;ID_элемента_2[\;сумма_платежа_2]]..."</pre>

Атрибут	Назначение
<code>disp_type</code>	<p>Атрибут указывает на то, как использовать выбранное из списка значение. Возможные значения <code>disp_type</code>:</p> <ul style="list-style-type: none"> • <code>"extra"</code> – передавать в процессинг экстра-поле с названием из <code>name</code> и выбранным значением из списка. Значение обрабатывается в соответствии с атрибутом <code>strip</code> и затем проверяется по регулярному выражению из <code>regex</code>, если выражение задано. При несоответствии регулярному выражению переход к следующему шагу сценария оплаты невозможен; • <code>"receipt"</code> – печать данных на чеке; • <code>"fixedsum"</code> – фиксированная сумма; • <code>"private"</code> – признак того, что данный элемент управления не должен быть отображен на экране / напечатан на чеке. <p>Атрибут игнорируется только АСО. Другие типы терминалов должны проверять наличие данного поля.</p> <p>Если страница <code>page</code> содержит единственный <code>control</code> с параметром <code>"private"</code> и тип терминала не АСО, то данная страница пропускается, выполняется следующий шаг сценария оплаты.</p>
<code>footer</code>	Текст под полем со списком
<code>header</code>	Текст над полем со списком
<code>mask</code>	Маска поля (подробнее см. в Приложении А)
<code>name</code>	Название поля для передачи в процессинг
<code>nobr</code>	<p>Способ отображения полей со списком:</p> <ul style="list-style-type: none"> • <code>"true"</code> – с переводом на следующую строку, • <code>"false"</code> – расположить элементы на одной строке.
<code>orderId</code>	Идентификатор, задающий порядок отображения элемента на странице
<code>regex</code>	Регулярное выражение (подробнее см. в Приложении Б)
<code>strip</code>	<p>Признак передачи значения без разделителей (<code>true/false</code>).</p> <p>Например, при передаче номера телефона, записанного как (903) 123-45-67, строка, передаваемая в процессинг, должна быть: 9031234567.</p>
<code>group_id</code>	<p>Идентификатор для привязки к другим управляющим элементам типа <code>"disp_label"</code>, <code>"text_input"</code>, <code>"combobox"</code> или <code>"disp_combobox"</code>.</p> <p>Если несколько элементов типа <code>"combobox"</code> (или <code>"disp_combobox"</code>) связаны между собой (содержат одинаковый атрибут <code>group_id</code>) и находятся на одной странице интерфейса (Табл. 6), то при выборе значения из любого списка в остальных должен автоматически выбираться элемент с тем же порядковым номером.</p>
<code>visible</code>	Видимость элемента в интерфейсе (<code>visible="false"</code> – элемент не должен быть отображен в интерфейсе; <code>visible="true"</code> или отсутствует – элемент должен быть отображен в интерфейсе).

Табл. 14. Атрибуты ассоциативного поля со списком "disp_combobox"

Атрибут	Назначение
disp_desc	Символьная константа, выводимая на чеке перед значением, выбранным из списка (может отсутствовать). «Двоеточие + пробел» после значения поля <code>disp_desc</code> проставляется автоматически на чеке. Если поле <code>disp_desc</code> отсутствует, «двоеточие+пробел» не выводится на печать.
disp_type	Атрибут указывает на то, как использовать выбранное из списка значение и значение атрибута <code>disp_desc</code> . Возможные значения <code>disp_type</code> : <ul style="list-style-type: none"> "extra" – передавать в процессинг экстра-поле с названием из <code>name</code> и значением из ассоциированной экстры в <code>disp_name</code>; "receipt" – печать данных на чеке; "fixedsum" – фиксированная сумма; "private" – признак того, что данный элемент управления не должен быть отображен на экране / напечатан на чеке. Атрибут игнорируется только АСО. Другие типы терминалов должны проверять наличие данного поля. Если страница <code>page</code> содержит единственный <code>control</code> с параметром "private" и тип терминала не АСО, то данная страница пропускается, выполняется следующий шаг сценария оплаты.
footer	Текст под полем со списком
header	Текст над полем со списком
mask	Маска поля (подробнее см. в Приложении А)
name	Название поля для передачи в терминал
nobr	Способ отображения полей со списком: <ul style="list-style-type: none"> "true" – с переводом на следующую строку, "false" – расположить элементы на одной строке.
orderId	Идентификатор, задающий порядок отображения элемента на странице
regexp	Регулярное выражение (подробнее см. в Приложении Б)
strip	Признак передачи значения без разделителей (<code>true/false</code>). Например, при передаче номера телефона, записанного как (903) 123-45-67, строка, передаваемая в процессинг, должна быть: 9031234567.
disp_name	Название экстра-поля, ассоциированного со списком. Значение, полученное из процессинга, отображается в качестве списка. Данное поле обязательно. Обращается в соответствии с <code>disp_type</code> .
group_id	Идентификатор для привязки к другим управляющим элементам типа "disp_label", "text_input", "combobox" или "disp_combobox". Если несколько элементов типа "combobox" (или "disp_combobox") связаны между собой (содержат одинаковый атрибут <code>group_id</code>) и находятся на одной странице интерфейса (Табл. 6), то при выборе значения из любого списка в остальных должен автоматически выбираться элемент с тем же порядковым номером.
visible	Видимость элемента в интерфейсе (<code>visible="false"</code> – элемент не должен быть отображен в интерфейсе; <code>visible="true"</code> или отсутствует – элемент должен быть отображен в интерфейсе).

Табл. 15. Атрибуты ассоциативного текстового поля "disp_label"

Атрибут	Назначение
disp_type	<p>Атрибут указывает на то, как использовать значение поля и значение атрибута <code>disp_desc</code>. Возможные значения <code>disp_type</code>:</p> <ul style="list-style-type: none"> "extra" – передавать в процессинг экстра-поле с названием из <code>name</code> и значением из <code>disp_name</code>; "receipt" – печать данных на чеке; "fixedsum" – фиксированная сумма; "private" – признак того, что данный элемент управления не должен быть отображен на экране / напечатан на чеке. <p>Атрибут игнорируется только АСО. Другие типы терминалов должны проверять наличие данного поля.</p> <p>Если страница <code>page</code> содержит единственный <code>control</code> с параметром "private" и тип терминала не АСО, то данная страница пропускается, выполняется следующий шаг сценария оплаты.</p>
disp_desc	<p>Символьная константа, выводимая на чеке перед значением, выбранным из списка (может отсутствовать).</p> <p>«Двоеточие + пробел» после значения атрибута проставляется автоматически на чеке. Если атрибут отсутствует, «двоеточие+пробел» не выводится на печать.</p>
footer	Текст под полем
header	Текст над полем
name	Название поля для передачи в процессинг (латиница)
nobr	<p>Способ отображения текста:</p> <ul style="list-style-type: none"> "true" – с переводом на следующую строку, "false" – расположить элементы на одной строке.
orderId	Идентификатор, задающий порядок отображения элемента на странице
disp_name	Название экстра-поля, ассоциированного с текстом. Значение, полученное из процессинга, отображается "как есть" (в виде текста, возможно с HTML разметкой). Данное поле обязательно. Обработывается в соответствии с <code>disp_type</code> .
group_id	<p>Идентификатор элемента "combobox" или "disp_combobox", с которым связано данное поле.</p> <p>Если атрибут <code>group_id</code> с тем же значением указан в элементе "combobox" / "disp_combobox" на той же странице интерфейса (Табл. 6), то текстовое поле относится к соответствующему списку и должно отображаться на странице рядом с этим списком.</p>
visible	Видимость элемента в интерфейсе (<code>visible="false"</code> – элемент не должен быть отображен в интерфейсе; <code>visible="true"</code> или отсутствует – элемент должен быть отображен в интерфейсе).

6.4.13. getPhoneRanges

Описание:

Запрос позволяет получить номерные емкости для всех провайдеров.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.4.13.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <providers>
    <getPhoneRanges/>
  </providers>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.4.13.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <getPhoneRanges result="0" version="3299315">
      <row from="1278400000" priority="1" prv-id="1234" range-id="123" region-
id="3" to="1278699999"/>
      ...
    </getPhoneRanges>
  </providers>
</response>
```

Ответ содержит следующие данные:

- `version` – версия возвращаемого справочника;
- `row` – тег, содержащий информацию о емкости. Включает атрибуты:
 - `from` – начальный номер емкости,
 - `priority` – приоритет,
 - `prv-id` – идентификатор провайдера, к которому относится емкость,
 - `range-id` – номерная емкость,
 - `region-id` – идентификатор региона, в котором действует данная емкость.
 - `to` – конечный номер емкости.

6.4.14. getRegions

Описание: Запрос позволяет получить список регионов.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.4.14.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <providers>
    <getRegions/>
  </providers>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.4.14.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <getRegions result="0">
      <row ADDR_AREA="Белгородская область" REGION_ID="1"/>
      <row ADDR_AREA="Кировская область" REGION_ID="2"/>
      <row ADDR_AREA="Курганская область" REGION_ID="3"/>
      ...
    </getRegions>
  </providers>
</response>
```

Ответ содержит следующие данные:

- `row` – тег, содержащий информацию о регионе. Включает атрибуты:
 - `ADDR_AREA` – название региона,
 - `REGION_ID` – идентификатор региона.

6.4.15. cancelPayment

Описание: Запрос на отмену платежа.

Режим обработки: Строго синхронно.

Роли:

- Бухгалтер;
- Главный менеджер.

6.4.15.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <providers>
    <cancelPayment>
      <payment id="11111112717"/>
    </cancelPayment>
  </providers>
```

```
</request>
```

Для выполнения запроса необходимо указать дополнительный параметр:

- `payment*` – тег, описывающий отменяемый платеж. Атрибуты тега (достаточно указать один из атрибутов):
 - `id` – идентификатор отменяемого платежа;
 - `uid` – идентификатор отменяемой транзакции в КИВИ.

6.4.15.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <cancelPayment result="0">
      <payment cancel-status="1" date="2013-08-12T15:27:07+04:00"
id="11111112717" result="0" status="2" uid="11136145075" uid2="11136186340"/>
    </cancelPayment>
  </providers>
</response>
```

Ответ содержит следующие данные:

- `payment` – тег, содержащий информацию об *отменяющем* платеже. Включает атрибуты:
 - `cancel-status` – результат выполнения процедуры отмены платежа. Возможные значения:
 - ⊕ 0 – отмена платежа невозможна (финальный статус);
 - ⊕ 1 – выполняется отмена платежа;
 - ⊕ 2 – платеж отменен (финальный статус);
 - `date` – дата и время (московские) регистрации отменяемого платежа в процессинге;
 - `id` – идентификатор отменяемого платежа;
 - `status` – статус отменяемого платежа (подробнее см. в [Приложении Г](#));
 - `result` – код ошибки проведения отменяемого платежа (подробнее см. в [Приложении В](#));
 - `uid` – идентификатор отменяемой транзакции в КИВИ;
 - `uid2` – идентификатор *отменяющей* транзакции в КИВИ (присутствует в ответе только в случае, если статус выполнения отмены промежуточный, т.е. когда атрибут `cancel-status="1"`).

ПРИМЕЧАНИЕ



Необходимо повторять запросы `cancelPayment` до получения финального статуса выполнения процедуры отмены платежа (см. атрибут `cancel-status`).

6.4.16. interruptPayment

Описание: Запрос на отмену платежа в промежуточном статусе.

Режим обработки: Строго синхронно.

Роли: Продавец.

6.4.16.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <providers>
    <interruptPayment>
      <payment id="12345"/>
    </interruptPayment>
  </providers>
</request>
```

Для выполнения запроса необходимо указать дополнительный параметр:

- `payment*` – тег, описывающий отменяемый платеж. Атрибуты тега:
 - `id` – идентификатор отменяемого платежа, находящегося в промежуточном статусе.

6.4.16.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <providers>
    <interruptPayment result="0"/>
  </providers>
</response>
```

Ответ содержит следующие данные:

- `result` – результат выполнения запроса.

ПРИМЕЧАНИЕ



Если результат неизвестен, то возвращается код результата 170. Необходимо выполнить повторный запрос до получения ответа с кодом ошибки, на основе которого можно сделать вывод о выполнении запроса.

6.5. Persons

ПРИМЕЧАНИЕ



Обязательные параметры запросов отмечены символом *.

Список действий интерфейса:

Действие	Описание
getPersonInfo	Получение информации о персоне
getPersons	Получение списка персон агента
getRoles	Получение справочника ролей пользователей Системы
changePassword	Смена пароля
setPublicKey	Установка открытого ключа

6.5.1. getPersonInfo

Описание: Запрос позволяет получить информацию о персоне, от имени которой был отправлен запрос.

Режим обработки: Строго синхронно.

Роли:

- Главный менеджер;
- Менеджер;
- Продавец;
- Техподдержка;
- Мониторинг;
- Дистрибьютор;
- Дистрибьютор (новый агентский сайт);
- Координатор Дистрибьютора.

6.5.1.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <persons>
    <getPersonInfo/>
  </persons>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.5.1.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <persons>
    <getPersonInfo result="0">
      <person agent="2" enabled="true" id="12" login="login" name="Имя">
        <roles>
          <role right_id="100, 104">4</role>
        </roles>
        <labels>
          <label>2203</label>
        </labels>
      </person>
    </getPersonInfo>
  </persons>
</response>
```

Ответ содержит следующие данные:

- `person` – тег содержит информацию о персоне. Включает атрибуты:
 - `agent` – идентификатор агента,
 - `enabled` – флаг блокировки персоны:
 - ⊕ `true` – персона активна;
 - ⊕ `false` – персона заблокирована;
 - `id` – идентификатор персоны,
 - `login` – логин персоны,
 - `name` – имя персоны,
- `roles` – тег, содержащий список идентификаторов ролей персоны:
 - `role` – идентификатор роли (название ролей персон см. в разделе [6.5.3](#)). Включает атрибут:
 - ⊕ `right_id` – список идентификаторов операций, доступных данной роли;
- `labels` – тег, содержащий список меток персоны:
 - `label` – идентификатор метки (в справочнике **метки** на сайте agent.giwi.com).

6.5.2. getPersons

Описание: Запрос позволяет получить список персон текущего агента.

Режим обработки: Возможно асинхронно.

Роли:

- Главный менеджер;
- Менеджер;
- Продавец;
- Техподдержка;
- Мониторинг;
- Дистрибьютор;

- Дистрибьютор (новый агентский сайт);
- Координатор Дистрибьютора.

6.5.2.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <persons>
    <getPersons/>
  </persons>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.5.2.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <persons>
    <getPersons result="0">
      <row agt_id="112233" created_by_prs_id="125689" prs_agt_id="112233"
prs_creation_date="2011-05-30T09:19:12+04:00" prs_email="adr@mail.ru"
prs_fname="Иван_ИТ" prs_icq="" prs_id="1135956" prs_login="ivanitm" prs_midname=""
prs_mobile="79111234568" prs_name="" prs_pass_date="2013-08-13T09:41:09+04:00"
prs_phone="" prs_rights="9" prs_role="21" prs_send_how="" prs_send_what=""
prs_state="1" trm_id=""/>
      ...
    </getPersons>
  </persons>
</response>
```

Ответ содержит следующие данные:

- `row` – тег содержит информацию об одной персоне, доступной текущей (использованной для [авторизации](#) запроса) персоне. Включает атрибуты:
 - `agt_id` – идентификатор агента, к которому привязана персона (в инструменте редактирования персон на сайте [agent.qiwi.com](#));
 - `created_by_prs_id` – идентификатор персоны, которая создала описываемую в данном теге `row` персону;
 - `prs_agt_id` – идентификатор агента, под которым заведена персона;
 - `prs_creation_date` – дата и время создания персоны;
 - `prs_email` – адрес электронной почты персоны;
 - `prs_fname` – фамилия персоны;
 - `prs_icq` – номер ICQ персоны;
 - `prs_id` – идентификатор персоны;
 - `prs_login` – логин персоны;
 - `prs_midname` – отчество персоны;

- prs_mobile – номер мобильного телефона персоны;
- prs_name – имя персоны;
- prs_pass_date – срок действия пароля персоны (для одноразового пароля);
- prs_phone – контактный телефон персоны;
- prs_rights – идентификаторы операций, доступных персоне;
- prs_role – идентификаторы роли персоны;
- prs_send_how, prs_send_what – служебные поля;
- prs_state – статус персоны:
 - ⊕ 1 – активна,
 - ⊕ 0 – заблокирована,
- trm_id – идентификатор терминала, к которому привязана персона (в инструменте редактирования персон на сайте agent.qiwi.com).

6.5.3. getRoles

Описание: Запрос позволяет получить справочник ролей Системы.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.5.3.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <persons>
    <getRoles/>
  </persons>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.5.3.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <persons>
    <getRoles result="0">
      <row role_id="4" role_name="Главный менеджер"/>
      <row role_id="21" role_name="Бухгалтер"/>
      ...
    </getRoles>
  </persons>
</response>
```

Ответ содержит следующие данные:

- row – тег, содержащий информацию о роли:

- `id` – идентификатор роли;
- `role_name` – название роли.

6.5.4. changePassword

Описание: Запрос позволяет сменить пароль для персоны, логин которой был указан при авторизации.

Режим обработки: Строго синхронно.

- Роли:**
- Главный менеджер;
 - Дистрибьютор;
 - Дистрибьютор (новый агентский сайт);
 - Координатор Дистрибьютора.

Запрос `changePassword` используется в ПО АСО для формирования постоянного (нового) пароля на основе введенного одноразового.

Схема работы при использовании запроса:

1. Техник вводит логин и одноразовый пароль для авторизации в системе.
2. ПО АСО (*maratl.exe*) формирует новый постоянный пароль для данного пользователя.
3. Новый пароль сохраняется на АСО для дальнейшего его использования в запросах, отправляемых с АСО в процессинг КИВИ.
4. ПО АСО формирует запрос `changePassword` и отправляет его на сервер. В параметре `new-password` передается результат логической операции между старым и новым паролями.
5. Сервер извлекает новый пароль из параметра `new-password` и сохраняет информацию о новом пароле персоны.

В дальнейшем новый пароль персоны используется для авторизации.

6.5.4.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <persons>
    <changePassword>
      <new-password>00000000000000000000000000000001</new-password>
    </changePassword>
  </persons>
</request>
```

Для выполнения запроса можно указать дополнительный параметр:

- `new-password` – результат логической операции между старым и новым паролями.

6.5.4.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <persons>
    <changePassword result="0"/>
  </persons>
</response>
```

Ответ содержит информацию об успешности запроса.

6.5.5. setPublicKey

Описание: Запрос позволяет установить для персоны открытый ключ, используемый для авторизации и проверки целостности передаваемых пакетов данных (Подробнее об алгоритмах формирования цифровой подписи см. [Приложение Ж](#)).

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Продавец;
- Кассир.

6.5.5.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <persons>
    <setPublicKey>
      <store-type>1</store-type>
      <pubkey>PUBLICKEY</pubkey>
    </setPublicKey>
  </persons>
</request>
```

ПРИМЕЧАНИЕ



Параметр терминала `terminal` и `serial` в теге `<client>` указывать не обязательно.

ВНИМАНИЕ



Для авторизации должна использоваться персона с одноразовым паролем.

Для выполнения запроса указываются параметры:

- `store-type*` – тип локального хранилища сертификатов,

- 1 – файл,
- 2 – системное хранилище,
- 4 – eToken,
- pubkey* – публичный ключ.

6.5.5.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <persons>
    <setPublicKey result="0"/>
  </persons>
</response>
```

Ответ содержит информацию об успешности запроса.

6.6. Templates

ПРИМЕЧАНИЕ



Обязательные параметры запросов отмечены символом *.

Список действий интерфейса:

Действие	Описание
createUser	Регистрация клиента
modifyUser	Изменение данных клиента
getPaymentTemplates	Получение всех активных шаблонов клиента
createPaymentTemplate	Создание платежного шаблона
blockPaymentTemplate	Блокировка платежного шаблона
getTemplateProviders	Получение списка доступных провайдеров в сервисе
getTemplateProviderInfo	Получение платежных реквизитов по провайдеру

6.6.1. createUser

Описание: Регистрация нового клиента.

Режим обработки: Строго синхронно.

Роли:

- Автомат
- Кассир
- Продавец
- Сервис шаблонов

6.6.1.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="terminal" software="ATL v5.4.1" serial="serial"/>
  <templates>
    <createUser>
      <user phone="79031234567" firstName="Иван" lastName="Иванов"
middleName="Сергеевич" documentType="0" documentSerial="2020"
documentNumber="202026" birthDate="1980-06-30" issueDate="2020-06-30"
citizenship="RU" address="г. Москва, ул. Пушкина, д. 5А, кв. 1" />
    </createUser>
  </templates>
</request>
```

Для выполнения запроса необходимо указать дополнительные параметры (формат параметров см. в п. [3.3.2.3](#)):

- `user*` – тег, содержащий информацию о клиенте. Включает атрибуты:
 - `phone` – номер телефона клиента в международном формате без символа «+» (от 11 до 15 цифр);
 - `firstName` – имя клиента;
 - `lastName` – фамилия клиента;
 - `middleName` – отчество клиента (указывается при наличии);
 - `documentType` – идентификатор типа документа, удостоверяющего личность. Возможные значения:
 - ⊕ 0 – внутренний паспорт РФ;
 - ⊕ 1 – документ, удостоверяющий личность иностранного гражданина, и выданный его государством;
 - ⊕ 2 – вид на жительство;
 - ⊕ 3 – разрешение на временное проживание;
 - ⊕ 4 – удостоверение беженца;
 - ⊕ 5 – иной документ;
 - ⊕ 6 – внутренний документ, удостоверяющий личность;
 - ⊕ 7 – заграничный паспорт;
 - `documentNumber` – номер документа, удостоверяющего личность;
 - `documentSerial` – серия документа, удостоверяющего личность;
 - `issueDate` – дата выдачи документа, удостоверяющего личность, в формате ГГГГ-ММ-ДД;
 - `citizenship` – гражданство клиента (буквенная аббревиатура);
 - `birthDate` – дата рождения клиента в формате ГГГГ-ММ-ДД;
 - `address` – адрес регистрации клиента;
 - `expireDocumentDate` – дата прекращения срока действия документа, удостоверяющего личность, в формате ГГГГ-ММ-ДД (не является обязательным для внутреннего паспорта РФ).

6.6.1.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <templates>
    <createUser result="0">
      <user firstName="Иван" lastName="И." middleName="Сергеевич"
modifyDate="2022-11-21T13:29:26+03:00" phone="79031234567" regDate="2022-11-
21T13:29:26+03:00" />
    </createUser>
  </templates>
</response>
```

Ответ содержит следующие данные:

- `result` – код ошибки обработки запроса (подробнее см. в [Приложении В](#));
- `user` – тег, содержащий информацию о клиенте. Включает атрибуты:
 - `phone` – номер телефона клиента в международном формате без символа «+» (от 11 до 15 цифр);
 - `firstName` – имя клиента;
 - `lastName` – первая буква фамилии клиента;
 - `middleName` – отчество клиента (указывается при наличии);
 - `modifyDate` – дата и время изменения данных клиента, время московское (+3:00 GMT). При первоначальной регистрации клиента совпадает со значением `regDate`;
 - `regDate` – дата и время регистрации клиента, время московское (+3:00 GMT).

6.6.2. modifyUser

Описание: Изменение данных клиента.

Режим обработки: Строго синхронно.

Роли:

- Автомат
- Кассир
- Продавец
- Сервис шаблонов

6.6.2.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <templates>
    <modifyUser>
      <user phone="79031234567" firstName="Иван" lastName="Иванов"
middleName="Сергеевич" documentType="0" documentSerial="2020"
documentNumber="202026" birthDate="1980-06-30" issueDate="2020-06-30"
citizenship="RU" address="г. Москва, ул. Пушкина, д. 5А, кв. 2" />
    </modifyUser>
  </templates>
</request>
```

```
</templates>
</request>
```

Для выполнения запроса необходимо указать дополнительные параметры (формат параметров см. в п. [3.3.2.3](#)):

- **user*** – тег, содержащий информацию о клиенте. Включает атрибуты:
 - **phone** – номер телефона клиента в международном формате без символа «+» (от 11 до 15 цифр);
 - **firstName** – имя клиента;
 - **lastName** – фамилия клиента;
 - **middleName** – отчество клиента (указывается при наличии);
 - **documentType** – идентификатор типа документа, удостоверяющего личность. Возможные значения:
 - ⊕ 0 – внутренний паспорт РФ;
 - ⊕ 1 – документ, удостоверяющий личность иностранного гражданина, и выданный его государством;
 - ⊕ 2 – вид на жительство;
 - ⊕ 3 – разрешение на временное проживание;
 - ⊕ 4 – удостоверение беженца;
 - ⊕ 5 – иной документ;
 - ⊕ 6 – внутренний документ, удостоверяющий личность;
 - ⊕ 7 – заграничный паспорт;
 - **documentNumber** – номер документа, удостоверяющего личность;
 - **documentSerial** – серия документа, удостоверяющего личность;
 - **issueDate** – дата выдачи документа, удостоверяющего личность, в формате ГГГГ-ММ-ДД;
 - **citizenship** – гражданство клиента (буквенная аббревиатура);
 - **birthDate** – дата рождения клиента в формате ГГГГ-ММ-ДД;
 - **address** – адрес регистрации клиента;
 - **expireDocumentDate** – дата прекращения срока действия документа, удостоверяющего личность, в формате ГГГГ-ММ-ДД (не является обязательным для внутреннего паспорта РФ).

6.6.2.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <templates>
    <modifyUser result="0">
      <user modifyDate="2022-12-02T16:51:55+03:00" phone="79031234567"
regDate="2022-07-08T16:51:55+03:00" firstName="Иван" lastName="И."
middleName="Сергеевич" />
    </modifyUser>
  </templates>
</response>
```

Ответ содержит следующие данные:

- `result` – код ошибки обработки запроса (подробнее см. в [Приложении В](#));
- `user` – тег, содержащий информацию о клиенте. Включает атрибуты:
 - `modifyDate` – дата и время изменения данных, время московское (+3:00 GMT);
 - `phone` – номер телефона клиента в международном формате без символа «+» (от 11 до 15 цифр);
 - `regDate` – дата и время регистрации клиента, время московское (+3:00 GMT);
 - `firstName` – имя клиента;
 - `lastName` – первая буква фамилии клиента;
 - `middleName` – отчество клиента (указывается при наличии).

6.6.3. `getPaymentTemplates`

Описание: Получение всех активных шаблонов клиента.

Режим обработки: Строго синхронно.

Роли:

- Автомат
- Кассир
- Продавец
- Сервис шаблонов

6.6.3.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="***" signAlg="MD5"/>
  <client terminal="111" software="ATL v5.4.1" serial=""/>
  <templates>
    <getPaymentTemplates>
      <user phone="79178246189"/>
    </getPaymentTemplates>
  </templates>
</request>
```

Для выполнения запроса необходимо указать дополнительные параметры (формат параметров см. в п. [3.3.2.3](#)):

- `user*` – тег, содержащий информацию о клиенте. Включает атрибуты:
 - `phone` – номер телефона клиента в международном формате без символа «+» (от 11 до 15 цифр).

6.6.3.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <templates>
    <getPaymentTemplates result="0">
      <user firstName="Иван" lastName="И." middleName="Иванович"
modifyDate="2022-12-01T17:34:23+03:00" phone="79178246189" regDate="2022-10-
12T13:41:02+03:00"/>
      <template createdDate="2022-12-01T17:34:25+03:00" id="110001097"
service="1">
        <fields account="7511645603" dop="dop" extra="extra_value"
firstName="Иван" middleName="Иванович" secondName="Иванов"/>
      </template>
      <template createdDate="2022-12-16T13:19:30+03:00" id="110001120"
service="40">
        <fields account="1111111112" extra="extra_value2"
firstName="Иван" middleName="Иванович" secondName="Иванов"/>
      </template>
      <template createdDate="2022-12-16T16:56:19+03:00" id="110001173"
service="40">
        <fields account="1111111114" extra="extra_value2"
firstName="Иван" middleName="Иванович" secondName="Иванов"/>
      </template>
      <template createdDate="2022-12-16T17:04:31+03:00" id="110001174"
service="40">
        <fields account="1111111115" extra="extra_value2"
firstName="Иван" middleName="Иванович" secondName="Иванов"/>
      </template>
    </getPaymentTemplates>
  </templates>
</response>
```

Ответ содержит следующие данные:

- `result` – код ошибки обработки запроса (подробнее см. в [Приложении В](#));
- `user` – тег, содержащий информацию о клиенте. Включает атрибуты:
 - `phone*` – номер телефона клиента в международном формате без символа «+» (от 11 до 15 цифр);
 - `firstName` – имя клиента;
 - `lastName` – первая буква фамилии клиента с точкой;
 - `middleName` – отчество клиента (указывается при наличии);
 - `modifyDate` – дата и время изменения данных клиента, время московское (+3:00 GMT);
 - `regDate` – дата и время регистрации клиента, время московское (+3:00 GMT);
- `template` – тег, содержащий информацию о шаблоне. Включает атрибуты:
 - `createdDate` – дата и время создания, время московское (+3:00 GMT);
 - `id` – идентификатор шаблона;
 - `service` – идентификатор провайдера;
 - `fields` – тег, содержащий реквизиты шаблона, необходимые для совершения платежа. Обязательно содержит атрибут `account` – идентификатор клиента в системе провайдера, напр. номер лицевого счета или телефона; может содержать иные атрибуты вида наименование экстра-поля платежа = значение экстра-поля платежа.

6.6.4. createPaymentTemplate

Описание: Создание платежного шаблона.

Режим обработки: Строго синхронно.

Роли:

- Автомат
- Кассир
- Продавец
- Сервис шаблонов

6.6.4.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request trace = "true">
  <auth login="login" sign="****" signAlg="MD5"/>
  <client terminal="111" software="ATL v5.4.1" serial=""/>
  <templates>
    <createPaymentTemplate>
      <payment id="1234">
        <from amount="0.01" currency="643"/>
        <to service="40" account="1111111116" amount="0.01"
currency="643"/>
        <extras firstName="Иван" secondName="Иванов"
middleName="Иванович" extra="extra_value2"/>
        <user phone="79176775667"/>
      </payment>
    </createPaymentTemplate>
  </templates>
</request>
```

Для выполнения запроса необходимо указать дополнительные параметры (формат параметров см. в п. [3.3.2.3](#)):

- **payment*** – тег, содержащий данные шаблона платежа. Включает атрибуты:
 - **id** – идентификатор платежа;
 - **from** – тег содержит информацию о сумме от клиента. Включает атрибуты:
 - ⊕ **amount** – сумма;
 - ⊕ **currency** – код валюты (значения кодов соответствуют **ISO 4217**);
 - **to** – тег содержит информацию о назначении платежа и сумме, которая должна быть перечислена на счет. Включает атрибуты:
 - ⊕ **service** – идентификатор провайдера, услуги которого оплачиваются. Провайдер должен быть из списка доступных в сервисе (см. [6.6.6](#));
 - ⊕ **account** – идентификатор клиента в системе провайдера;
 - ⊕ **amount** – сумма к зачислению;
 - ⊕ **currency** – код валюты (значения кодов соответствуют **ISO 4217**);
 - **user** – тег, содержащий информацию о клиенте. Включает атрибуты:
 - ⊕ **phone*** – номер телефона клиента в международном формате без символа «+» (от 11 до 15 цифр);

- `extras` – тег, в атрибутах которого указываются экстра-поля платежа. В одном теге может быть указано несколько экстра-полей, которые заполняются в соответствии с ответом на запрос `getTemplateProviderInfo` (см. [6.6.7](#))

6.6.4.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <templates>
    <createPaymentTemplate result="0">
      <template createdAt="2022-12-16T18:20:47+03:00" id="110001175"
service="40">
        <fields account="1111111116" extra="extra_value2"
firstName="Иван" middleName="Иванович" secondName="Иванов"/>
      </template>
    </createPaymentTemplate>
  </templates>
</response>
```

Ответ содержит следующие данные:

- `result` – код ошибки обработки запроса (подробнее см. в [Приложении В](#));
- `template` – тег, содержащий информацию о шаблоне. Включает атрибуты:
 - `createdAt` – дата и время создания шаблона, время московское (+3:00 GMT);
 - `id` – идентификатор шаблона;
 - `service` – идентификатор провайдера;
 - `fields` – тег, содержащий реквизиты шаблона, необходимые для совершения платежа. Обязательно содержит атрибут `account` – идентификатор клиента в системе провайдера, напр. номер лицевого счета или телефона; может содержать иные атрибуты вида `наименование экстра-поля платежа = значение экстра-поля платежа`.

6.6.5. `blockPaymentTemplate`

Описание: Блокировка платежного шаблона.

Режим обработки: Строго синхронно.

Роли:

- Автомат
- Кассир
- Продавец
- Сервис шаблонов

6.6.5.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="****" signAlg="MD5"/>
  <client terminal="111" software="ATL v5.4.1" serial=""/>
  <templates>
    <blockPaymentTemplate>
      <template id="110001175"/>
      <user phone="79176775667" documentSerial="2020"
documentNumber="202026"/>
    </blockPaymentTemplate>
  </templates>
</request>
```

Для выполнения запроса необходимо указать дополнительные параметры (формат параметров см. в п. [3.3.2.3](#)):

- `template*` – тег, содержащий данные шаблона. Включает атрибуты:
 - `id` – идентификатор шаблона;
 - `user` – тег, содержащий информацию о клиенте. Включает атрибуты:
 - ⊕ `phone` – номер телефона клиента в международном формате без символа «+» (от 11 до 15 цифр);
 - ⊕ `documentSerial` – серия документа, удостоверяющего личность;
 - ⊕ `documentNumber` – номер документа, удостоверяющего личность.

6.6.5.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <templates>
    <blockPaymentTemplate result="0"/>
  </templates>
</response>
```

Ответ содержит следующие данные:

- `result` – код ошибки обработки запроса (подробнее см. в [Приложении В](#)).

6.6.6. `getTemplateProviders`

Описание: Получение списка доступных провайдеров в сервисе.

Режим обработки: Строго синхронно.

Роли:

- Автомат
- Кассир
- Продавец
- Сервис шаблонов

6.6.6.1. Запрос

```
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <templates>
    <getTemplateProviders/>
  </templates>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.6.6.2. Ответ

```
<response result="0">
  <templates>
    <getTemplateProviders result="0">
      <provider service="1" shortName="Короткое название провайдера"/>
      <provider service="2" shortName="Короткое название провайдера"/>
      <provider service="3" shortName="Короткое название провайдера"/>
      . . . .
    </getTemplateProviders>
  </templates>
</response>
```

Ответ содержит следующие данные:

- `result` – код ошибки обработки запроса (подробнее см. в [Приложении В](#));
- `provider` – тег, содержащий информацию о доступных в сервисе провайдерах. Включает атрибуты:
 - `service` – идентификатор провайдера;
 - `shortName` – короткое название провайдера.

6.6.7. `getTemplateProviderInfo`

Описание: Получение платежных реквизитов по провайдеру.

Режим обработки: Строго синхронно.

Роли:

- Автомат
- Кассир
- Продавец
- Сервис шаблонов

6.6.7.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="stat_test" sign="****" signAlg="MD5"/>
  <client terminal="10653347" software="ATL v5.4.1" serial=""/>
  <templates>
    <getTemplateProviderInfo>
      <provider service="40"/>
    </getTemplateProviderInfo>
  </templates>
</request>
```

Для выполнения запроса необходимо указать дополнительные параметры (формат параметров см. в п. [3.3.2.3](#)):

- **provider*** – тег, содержащий данные провайдера, для которого необходимо получить платежные реквизиты. Включает атрибуты:
 - **service** – идентификатор провайдера.

6.6.7.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <templates>
    <getTemplateProviderInfo result="0">
      <provider longName="ПАО Мегафон" maxAmount="15000" service="40"
shortName="Мегафон">
        <field description="Главный реквизит платежа" name="account"
title="Номер карты, счета или договора"/>
        <field description="Указать при оплате по номеру договора"
name="accountClientId" title="Проверочный код"/>
        <field description="При оплате по номеру карты указать значение
1, по номеру счета - 2, по номеру договора - 3" name="accountType" title="Вид
платежа"/>
        <field description="Указать при оплате по номеру счета"
name="firstName" title="Имя получателя"/>
        <field description="Указать при оплате по номеру счета"
name="lastName" title="Фамилия получателя"/>
        <field description="При оплате по номеру счета указать значение
045004763" name="mfo" title="БИК"/>
        <field description="Указать при оплате по номеру счета"
name="middleName" title="Отчество получателя"/>
        <field description="При оплате по номеру карты указать значение
4, по номеру счета - 2" name="operationType" title="Тип операции"/>
      </provider>
    </getTemplateProviderInfo>
  </templates>
</response>
```

Ответ содержит следующие данные:

- **result** – код ошибки обработки запроса (подробнее см. в [Приложении В](#));
- **provider** – тег, содержащий информацию о провайдере. Включает атрибуты:
 - **service** – идентификатор провайдера;
 - **shortName** – короткое название провайдера;
 - **longName** – полное название провайдера;

- `field` – тег, содержащий платежные реквизиты. Включает в себя набор атрибутов, который зависит от спецификации [интерфейса провайдера](#):
 - ⊕ `description` – описание платежного реквизита: описывает возможные значения и особенности использования реквизита в платеже;
 - ⊕ `name` – наименование атрибута для подстановки в запрос создания платежного шаблона `createPaymentTemplate`: в теге `to`, если `name = account`, и в теге `extras` во всех остальных случаях (см. [6.6.4](#));
 - ⊕ `title` – название реквизита, как оно отображается для клиента.

6.7. Reports

Список действий интерфейса:

Действие	Описание
getPayments	Получение списка платежей за период
getTerminalsShiftHistory	Получение истории закрытия смен (для фискализации)
getStatistics	Получение статистики по агентам в разрезе провайдеров на основе шаблона
getCashouts	Получение списка инкассаций АСО
getTerminalsStatus	Состояние терминала (питание, закрытие двери, состоянии батареи UPS и пр.)
getTerminalsSettings	Настройки терминала (включая номер SIM-карты)
getTerminalsStatisticalData	Получение статистики терминала
getTerminalsCash	Получение данных о купюрности терминала

ВНИМАНИЕ



По умолчанию все действия выполняются для *текущего агента* и его субагентов. *Текущим агентом* считается агент, которому принадлежит терминал, указанный в теге `<client>` и персона, данные которой указаны в теге `<auth>`.

Если требуется получить информацию по конкретному терминалу (агенту), необходимо указать идентификатор терминала (или персоны) в теге `<target-terminal>` (`<target-agent>`) внутри тега действия (см. пример [запроса](#))

Обязательные параметры запросов отмечены символом *****.

6.7.1. getPayments

- Описание:** Получение списка платежей за период
- Режим обработки:** Строго асинхронно (см. раздел [3.3.3](#))
- Роли:** Выполнение запроса доступно персоне с любой ролью.

6.7.1.1. Запрос

ВНИМАНИЕ

Для пользователей РФ настоятельно рекомендуется запускать поиск платежа за период не более 1 часа.

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <reports>
    <getPayments mode="async">
      <date-from>2010-05-01T10:00:00</date-from>
      <date-to>2010-05-01T11:00:00</date-to>
      <terminal>111</terminal>
      <txn-type>0</txn-type>
      <account-number>9130000000</account-number>
      <provider>1</provider>
      <trm-txn-id>0123</trm-txn-id>
      <add-client-bank-com>1</add-client-bank-com>
      <extras>PAYTYPE, ev_paytype</extras>
    </getPayments>
  </reports>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- `terminal` – терминал, для которого необходимо выполнить действие (если не указан, выполняется для всех терминалов *текущего* агента);
- `target-agent` – идентификатор агента, для которого необходимо выполнить действие. Запрос выполняется только для указанного субагента;
- `date-from*` ... `date-to*` – теги задают период, за который необходимо получить список платежей;
- `txn-type` – тип транзакции (см. [справочник типов транзакций](#));

ПРИМЕЧАНИЕ

Для поиска отмен платежей установите `txn-type=2`.

- `debit` – дополнительный признак для получения платежных транзакций «Сервис Касса» (распределение средств с баланса агента) или «Проценты за коммерческий кредит» (списание процентов с баланса агента):
 - `true` – включать в ответ на запрос платежи такого типа (в параметре `txn-type` должен быть указан соответствующий тип транзакции),
 - `false` – не возвращать такие платежи (по умолчанию),
- `status-group` – группа статусов платежей:
 - 1 – платежи, авторизованные у провайдера (финансовый контроль пройден), а также те, которые уже проведены,

- 2 – платежи, пока не авторизованные у провайдера (платеж не прошел финансовый контроль), а также те платежи, у которых обработка завершена с ошибкой,
- 3 – платежи ожидают ручного подтверждения,
- 5 – платежи в проведении;
- `provider` – идентификатор провайдера;
- `add-client-bank-com` – признак добавления информации о сумме комиссии Расчетного банка (КИВИ Банк) по платежу в ответ на запрос:
 - 0 – не добавлять (по умолчанию);
 - 1 – добавлять;
- `extras` – список экстр-полей платежа (в формате `extra_name1, extra_name2, ...`) для добавления в ответ на запрос. Указываются в соответствии со спецификацией [интерфейса провайдеров](#);
- `account-number` – номер телефона/счета, на который переведены средства;
- `trm-txn-id` – идентификатор платежа на терминале;
- `max-row-count` – максимальное число возвращаемых записей.

6.7.1.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <reports>
    <getPayments count="1" quid="4" result="0" status="3">
      <row booking-date="2011-09-05T10:03:17+04:00" cancel-uid="" comment=""
        error-code="0" from-account="" from-agt-id="3" from-amount="50.00"
        from-curr="643" from-prv-id="7" id="0123" is-canceled="0"
        operator-id="2" pay-id="5" payment-date="2011-09-05T10:03:17+04:00"
        receipt-date="2011-09-05T13:03:19+04:00" receipt-id="5"
        serial="0123456789" status="2" to-account="9130000000" to-agt-id=""
        to-amount="45.05" to-curr="643" to-prv-id="1" to-prv-short-name="MTC "
        trm-id="111" txn-date="2011-09-05T10:03:16+04:00" txn-type="0"
        uid="012345678" value-date="2011-09-05T10:04:00+04:00" ext_value_1="1"
        ext_value_2="1" client-bank-com="10.00"/>
        <extras PAYTYPE="1" ev_paytype="1"/>
      </row>
    </getPayments>
  </reports>
</response>
```

Ответ содержит следующие данные:

- `count` – число платежей, удовлетворяющих условиям запроса;
- `quid` – номер асинхронного запроса в очереди на сервере (см. раздел [3.3.3](#));
- `row` – данные платежа. Включает атрибуты:
 - `booking-date` – время и дата перехода транзакции в текущий статус;
 - `cancel-uid` – идентификатор отменяемого платежа (заполнено только для транзакций-отмен платежей);

Для отмены платежа формируется транзакция с теми же реквизитами, что у отменяемого платежа, на отрицательную сумму платежа (см. [Пример 19](#) и [Пример 20](#)).

- `comment` – комментарий к транзакции;
- `error-code` – код ошибки обработки транзакции (см. [Приложение В](#));
- `from-account` – счет, с которого осуществляется перевод средств (в случае распределения средств субагентам);
- `from-agt-id` – идентификатор агента, со счета которого происходит списание;
- `from-amount` – сумма, полученная от клиента;
- `from-curr` – валюта платежа;
- `from-prv-id` – идентификатор провайдера;
- `id` – идентификатор транзакции на терминале;
- `is-canceled` – признак отмены платежа (1 – платеж отменен, 0 – платеж не отменен);
- `operator-id` – идентификатор персоны, совершившей платеж;
- `pay-id` – идентификатор шлюза провайдера;
- `payment-date` – дата проведения платежа у провайдера;
- `receipt-date` – дата и время печати чека на терминале;
- `receipt-id` – идентификатор чека;
- `serial` – серийный номер терминала, с которого был отправлен платеж;
- `status` – текущий статус платежа (см. [справочник статусов платежей](#));
- `to-account` – счет, на который зачислены средства;
- `to-agt-id` – идентификатор агента, на счет которого происходит зачисление (в случае распределения средств);
- `to-amount` – сумма, поступившая на счет клиента;
- `to-curr` – валюта зачисляемых на счет клиента средств;
- `to-prv-id` – идентификатор провайдера, в пользу которого был осуществлен платеж;
- `to-prv-short-name` – короткое название провайдера, в пользу которого был осуществлен платеж;
- `trm-id` – идентификатор терминала, с которого был проведен платеж;
- `txn-date` – дата поступления платежа в процессинг;
- `ext_value_1, ext_value_2, ...` – значения запрашиваемых экстра-полей.

ПРИМЕЧАНИЕ

Данные атрибуты включаются в ответ, только если в теге `<extras>` исходного запроса были указаны экстра-поля. Порядок перечисления значений соответствует списку названий в запросе.

Также в ответ добавляется вложенный тег:

- ✦ `extras` – содержит в качестве атрибутов список запрашиваемых экстра-полей платежа и их значений, в формате `extra_name="extra_value"`;
- `txn-type` – тип транзакции (см. [справочник типов транзакций](#));
- `uid` – идентификатор транзакции в процессинге;
- `value-date` – дата и время (московские) списания суммы платежа с баланса агента.

ВНИМАНИЕ

Если дата списания для данного платежа еще не известна, значение атрибута `value-date` отсутствует.

- `client-bank-com` – размер комиссии расчетного банка (КИВИ Банк) в валюте платежа (добавляется в ответ, только если в запросе указан параметр `add-client-bank-com` со значением 1).

Пример 19. Отменяемый платеж

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <reports>
    <getPayments count="1" quid="0000012" result="0" status="3">
      <row booking-date="2010-08-02T15:52:41+04:00" cancel-uid="" comment="" error-
code="0" from-account="" from-agt-id="12345" from-amount="10.00" from-curr="643"
from-prv-id="7" id="1111111" operator-id="1111111" pay-id="1111" payment-
date="2010-08-02T15:52:35+04:00" receipt-date="2010-08-02T15:52:35+04:00" receipt-
id="222" serial="" status="2" to-account="999" to-agt-id="" to-amount="10.00" to-
curr="643" to-prv-id="8888" to-prv-short-name="Яуза Телеком" trm-id="1111111" txn-
date="2010-08-02T15:52:35+04:00" txn-type="0" uid="1000000000001"/>
    </getPayments>
  </reports>
</response>
```

Пример 20. Отменяющий платеж

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <reports>
    <getPayments count="1" quid="0000013" result="0" status="3">
      <row booking-date="2010-08-05T11:58:17+04:00" cancel-uid="1000000000001"
comment="Постановка на отмену платежа за телефон 999 ; заявка № 000001" error-
code="0" from-account="" from-agt-id="12345" from-amount="-10.00" from-curr="643"
from-prv-id="7" id="1000000000002" operator-id="321" pay-id="1111" payment-date=""
receipt-date="2010-08-02T15:52:35+04:00" receipt-id="222" serial="" status="3" to-
account="999" to-agt-id="" to-amount="-10.00" to-curr="643" to-prv-id="8888" to-
prv-short-name="Яуза Телеком" trm-id="1111111" txn-date="2010-08-05T11:58:17+04:00"
txn-type="2" uid="10000000030001"/>
    </getPayments>
  </reports>
</response>
```

6.7.2. getTerminalsShiftHistoryОписание:

Запрос позволяет получить историю закрытия смен (для фискализации).

Режим обработки: Возможно асинхронно.

- Роли:**
- Автомат;
 - Главный менеджер;
 - Менеджер;
 - Кассир;
 - Продавец;
 - Техподдержка;
 - Дистрибьютор;
 - Мониторинг.

6.7.2.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <reports>
    <getTerminalsShiftHistory>
      <target-terminal>123</target-terminal>
      <date-from>2009-07-17T00:00:00</date-from>
      <date-to>2009-07-22T15:01:59</date-to>
    </getTerminalsShiftHistory>
  </reports>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- `target-terminal` – терминал, для которого необходимо получить список инкассаций (если не указан, выполняется для *всех* терминалов *текущего* агента и его субагентов),
- `date-from*` и `date-to*` – теги задают промежуток времени, за который необходимо получить историю закрытия смен.

6.7.2.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?><response result="0">
  <reports>
    <getTerminalsShiftHistory result="0">
      <row cash-sum="13850" date="2011-07-17T08:10:16+04:00" id="1" nv1="30"
nv2="35" nv3="78" nv4="4" nv5="2" nv6="0" nv7="0" nv8="0" nv9="0" trm-id="123"/>
      <row cash-sum="11880" date="2011-07-17T08:57:51+04:00" id="2" nv1="8"
nv2="30" nv3="48" nv4="7" nv5="2" nv6="0" nv7="0" nv8="0" nv9="0" trm-id="123"/>
      <row cash-sum="44490" date="2011-07-17T11:33:44+04:00" id="3" nv1="39"
nv2="112" nv3="195" nv4="24" nv5="7" nv6="0" nv7="0" nv8="0" nv9="0" trm-id="123"/>
    </getTerminalsShiftHistory>
  </reports>
</response>
```

Ответ содержит следующие данные:

- `cash-sum` – сумма инкассации,
- `date` – дата инкассации,
- `id` – идентификатор инкассации,

- `nv1 - nv9` – количество купюр различного номинала,

Для РФ:

- `nv1` – 10-рублевые купюры;
- `nv2` – 50-рублевые купюры;
- `nv3` – 100-рублевые купюры;
- `nv4` – 500-рублевые купюры;
- `nv5` – 1000-рублевые купюры;
- `nv6` – 5000-рублевые купюры;
- `trm-id` – идентификатор терминала.

6.7.3. getStatistics

Описание:

Запрос позволяет получить статистику по платежам агента (в разрезе провайдеров) по указанному шаблону статистики на сайте agent.qiwi.com (раздел **отчеты** → **шаблоны**) за определенный интервал времени.

Режим обработки: Возможно асинхронно.

Роли:

- Главный менеджер;
- Менеджер;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.7.3.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <reports>
    <getStatistics>
      <report-id>1</report-id>
      <date-from>2011-07-01T00:00:00</date-from>
      <date-to>2011-07-27T15:01:59</date-to>
      <with-sub-agents>1</with-sub-agents>
    </getStatistics>
  </reports>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- `report-id*` – идентификатор шаблона статистики (шаблон создается на сайте agent.qiwi.com; название шаблона содержит идентификатор шаблона, например "1708 Учет платежей и верхней комиссии по терминалам");

- `date-from / date-to*` – период, за который необходимо на основании шаблона выгрузить отчет;
- `with-sub-agents` – включить субагентов в отчет (если не указан, данные выгружаются без учета субагентов):
 - 0 – без учета субагентов;
 - 1 – с учетом субагентов.

6.7.3.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <reports>
    <getStatistics result="0">
      <row name="text">
        <column name="2011-07-01">
          <cell accountType="5" dataType="INCOME" value="0.0000"/>
          <cell accountType="5" dataType="EXPENSE" value="0.0000"/>
          <cell accountType="13" dataType="INCOME" value="0.0000"/>
          <cell accountType="13" dataType="EXPENSE" value="3.6400"/>
        </column>
      </row>
      <row name="text">
        <column name="2011-07-02">
          <cell accountType="29" dataType="INCOME" value="97.0000"/>
          <cell accountType="1" dataType="EXPENSE" value="97.0000"/>
          <cell accountType="29" dataType="COUNTINCOME" value="12"/>
          <cell accountType="25" dataType="INCOME" value="10.0000"/>
          <cell accountType="13" dataType="EXPENSE" value="10.0000"/>
        </column>
      </row>
      ...
    </getStatistics>
  </reports>
</response>
```

Ответ содержит следующие данные:

- `row` – тег содержит сгруппированные данные (блок отчета):
 - `name` – содержит название, по которому выполняется группировка;
 - `column` – тег описывает содержимое столбцов за 1 отчетный день:
 - ⊕ `name` – отчетная дата;
 - ⊕ `cell` – тег описывает содержимое ячейки:
 - ❖ `accountType` – идентификатор типа счета (см. [Приложение К](#));
 - ❖ `dataType` – тип данных. Возможные значения:
 - INCOME – приход,
 - OUTCOME – расход,
 - BALANCE – баланс,
 - COUNTINCOME – число операций прихода,
 - COUNTOUTCOME – число операций расхода,
 - ❖ `value` – значение типа данных (сумма прихода/сумма расхода/сумма баланса/число операций прихода/число операций расхода).

6.7.4. getCashouts

- Описание:** Запрос позволяет получить информацию обо всех инкассациях автомата за указанный период.
- Режим обработки:** Строго синхронно.
- Роли:** Выполнение запроса доступно персоне с любой ролью.

6.7.4.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <reports>
    <getCashouts>
      <date-from>2011-04-01T00:00:00</date-from>
      <date-to>2011-04-05T00:00:00</date-to>
    </getCashouts>
  </reports>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- `terminal-id` – терминал, для которого необходимо получить список инкассаций (если не указан, выполняется для всех терминалов *текущего* агента и его субагентов);
- `target-agent` – идентификатор агента, список инкассаций для терминалов которого требуется получить;
- `date-from*` и `date-to*` – промежуток времени, за который необходимо получить список инкассаций;
- `ttp-id` – идентификатор типа терминала (см. [Приложение М](#)). Если указан, то список инкассаций в ответе будет содержать только данные для терминалов указанного типа. Если не указан, то список инкассаций будет содержать данные по терминалам самообслуживания (типы терминалов "Автомат самообслуживания", "Дилерский автомат самообслуживания", "Банкомат", "АСО РИ Light").

6.7.4.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <reports>
    <getCashouts result="0">
      <cashout date="2011-03-23T13:52:20+03:00" id="0123456789" status="1"
        terminal="123">
        <currency id="643">
          <notes count="134" sum="8890.0">
            <nominal count="0" value="1000.0"/>
            <nominal count="39" value="10.0"/>
            <nominal count="3" value="500.0"/>
          </notes>
        </currency>
      </cashout>
    </getCashouts>
  </reports>
</response>
```

```

        <coins count="0" sum="0.0">
            <nominal count="0" value="0.01"/>
            ...
        </coins>
    </currency>
</cashout>
...
</getCashouts>
</reports>
</response>

```

Ответ содержит следующие данные:

- `cashout` – информация об отдельной инкассации:
 - `date` – дата инкассации;
 - `id` – идентификатор инкассации на терминале;
 - `status` – тип инкассации (см. Приложение 3);
 - `terminal` – идентификатор инкассируемого терминала;
 - `currency` – тег, содержащий информацию о валюте денежных средств:
 - ⊕ `id` – идентификатор валюты. Значения кодов соответствуют **ISO 4217**;
 - ⊕ `notes` – тег информации о купюрах:
 - ❖ `count` – количество учтенных купюр;
 - ❖ `sum` – сумма всех купюр, учтенных в терминале;
 - ❖ `nominal` – информация о количестве купюр одного номинала (10, 50,100, 500, 1000, 5000, 10000, 50000):
 - `value` – номинал;
 - `count` – количество купюр одного номинала;
 - ⊕ `coins` – тег информации о монетах (в случае если имеется монетоприемник):
 - ❖ `count` – количество монет, учтенных в терминале;
 - ❖ `sum` – сумма всех монет, учтенных в терминале;
 - ❖ `nominal` – информация о количестве монет одного номинала (0.01, 0.05, 0.1, 0.5, 1.0, 2.0, 5.0, 10.0):
 - `value` – номинал;
 - `count` – количество монет этого номинала.

6.7.5. `getTerminalsStatus`

Описание:

Запрос позволяет получить информацию в зашифрованном побитовом виде о состоянии терминалов (электропитании, закрытии двери, состоянии батареи UPS и пр.).

Режим обработки:

Возможно асинхронно.

Роли:

- Автомат;
- Главный менеджер;

- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.7.5.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <reports>
    <getTerminalsStatus>
      <target-agent>112123</target-agent>
    </getTerminalsStatus>
  </reports>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-terminal` – терминал, для которого необходимо получить статус;
- `target-agent` – агент, для терминалов которого необходимо получить статусы. Запрос выполняется только для указанного агента (без его субагентов).

6.7.5.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <reports>
    <getTerminalsStatus result="0">
      <row agtId="1" lastActivityTime="2011-08-27T11:12:30+04:00"
        lastPaymentTime="2011-08-27T10:32:59+04:00"
        machineStatus="000000010001000000000000 " noteErrorId="OK"
        printerErrorId="OK" CardReaderStatus="1" signalLevel="25"
        simProviderBalance="-1.0" trmId="123" wdtDoorAlarmCount="0"
        wdtDoorOpenCount="0" wdtEvent="0"/>
    </getTerminalsStatus>
  </reports>
</response>
```

Ответ содержит следующие данные:

- `row` – тег, содержащий информацию о терминале. Включает атрибуты:
 - `agtId` – идентификатор агента;
 - `lastActivityTime` – время последней активности (время последнего ring, отправленного с данного терминала);
 - `lastPaymentTime` – время и дата последнего отправленного с терминала платежа;
 - `machineStatus` – набор флагов состояния терминала типа АСО. Подробнее о флагах см. [Приложение Л](#);
 - `noteErrorId` – текстовое описание ошибки купюроприемника;

- `printerErrorId` – текстовое описание ошибки чекового принтера;
- `CardReaderStatus` – состояние работы картридера;
- `signalLevel` – уровень сигнала;
- `simProviderBalance` – баланс на SIM-карте;
- `trmId` – идентификатор терминала;
- `wdtDoorAlarmCount` – счетчик тревог двери;
- `wdtDoorOpenCount` – счетчик открытий двери;
- `wdtEvent` – атрибут содержит в зашифрованном побитовом виде информацию о состоянии сторожевого таймера АСО (питании, закрытии двери, состоянии батареи UPS и пр.);
- `wdtEventText` – описание бита состояния сторожевого таймера АСО (атрибут возвращается, если `wdtEvent` не равен 0). См. [Приложение И](#).

ПРИМЕЧАНИЕ

Если битов несколько, текстовые описания битов перечисляются в одной строке, разделённой пробелами.

6.7.6. getTerminalsSettings

Описание: Запрос позволяет получить информацию о настройках терминала (включая номер SIM-карты)

Режим обработки: Возможно асинхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.7.6.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <reports>
    <getTerminalsSettings/>
  </reports>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-terminal` – терминал, для которого необходимо получить настройки (если не указан, выполняется для всех терминалов *текущего* агента и его субагентов);
- `target-agent` – агент, для которого необходимо выполнить действие (если не указан, выполняется для *текущего* агента).

6.7.6.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
<reports>
  <getTerminalsSettings result="0">
    <row BrowserVersion="7.0.5730.11" CashInModel="CashCode CCNET SM-RU1617(5000)"
CashInCapacity="50" CardReaderModel="CRT 288 K001 (USB)" ClientSoftware="ATL v3.48"
FlashVersion="10.0.42.34" HddFactoryNumber="123456" HddModel="MAXTOR STM380215AS"
HddSmartAttributeId="0" HddSmartAttributeThresholdValue="0" HddStatus="0"
HddSmartAttributeValue="0" HddTemperature="40" InterfaceVersion="ADV v2.23.2"
ModemModel="Model" modemModel="" lastPaymentTime="2012-01-24T15:18:40+04:00"
simProviderId="24" OsVersion="Microsoft Windows XP Embedded Service Pack 2 (Build
2600)" PrinterModel="Citizen PPU 700" ScreenResolution="1280x1024" agtId="1111111"
dataReceived="2919" dataSent="768" trmId="1111111" wdtVersion="v1.00"
simSerial="007902111111" iccid="132148892" cashRegisterSerial="" />
    .....
  </getTerminalsSettings>
</reports>
</response>
```

Ответ содержит следующие данные:

- `row` – тег содержит информацию о программном и аппаратном обеспечении терминала. Включает атрибуты:
 - `BrowserVersion` – версия браузера;
 - `CashInModel` – модель купюроприемника;
 - `CashInCapacity` – емкость стекера купюроприемника;
 - `CardReaderModel` – модель картридера;
 - `trmId` – идентификатор терминала;
 - `agtId` – идентификатор агента;
 - `ClientSoftware` – версия ПО терминала;
 - `PrinterModel` – модель чекового принтера;
 - `HddModel` – модель жесткого диска;
 - `HddFactoryNumber` – серийный номер жесткого диска;
 - `HddTemperature` – температура жесткого диска;
 - `HddStatus` – статус жесткого диска;
 - `HddSmartAttributeId` – идентификатор SMART-атрибута жесткого диска;
 - `HddSmartAttributeValue` – значение SMART-атрибута жесткого диска;
 - `HddSmartAttributeThresholdValue` – предельно допустимое значение SMART-атрибута жесткого диска;
 - `dataReceived` – объем полученных данных сетевым адаптером, подключенным к терминалу, в КБ;

- `dataSent` – объем отправленных данных сетевым адаптером, подключенным к терминалу, в КБ;
- `lastPaymentTime` – время последнего принятого платежа на терминале;
- `OsVersion` – версия операционной системы;
- `InterfaceVersion` – версия интерфейса терминала;
- `ScreenResolution` – разрешение экрана на терминале;
- `wdtVersion` – версия сторожевого таймера;
- `FlashVersion` – версия flash-player;
- `simSerial` – серийный номер SIM-карты, назначенный оператором сотовой связи (IMSI);
- `iccid` – серийный номер SIM-карты (ICCID);
- `ModemModel` – модель модема, зарегистрированная в ОС;
- `modemModel` – модель модема (по версии производителя);
- `simProviderId` – внутренний идентификатор оператора сотовой связи SIM-карты (см. [справочник сотовых операторов](#));
- `cashRegisterSerial` – заводской серийный номер ФР.

6.7.7. getTerminalsStatisticalData

Описание: Запрос позволяет получить статистику работы терминалов

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Главный менеджер;
- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.7.7.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <reports>
    <getTerminalsStatisticalData/>
  </reports>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-terminal` – терминал, для которого необходимо получить статистику (если не указан, выполняется для всех терминалов *текущего* агента и его субагентов);
- `target-agent` – агент, для которого необходимо выполнить действие (если не указан, выполняется для *текущего* агента).

6.7.7.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
<reports>
  <getTerminalsStatisticalData result="0">
    <row agtId="4504425" billsPerPay="1" CardReaderUsedHour="3"
CardReaderUsedDay="20" lastPaymentTime="2012-01-24T15:18:40+04:00"
paysPerHour="3" progUpTime="20140" systemUpTime="20140" timeToCashinFull="-1"
timeToCashinService="108" timeToPrinterPaperOut="-1" timeToPrinterService="0"
trmId="8988212"/>
    ...
  </getTerminalsStatisticalData>
</reports>
</response>
```

Ответ содержит следующие данные:

- `row` – тег, содержащий информацию о терминале. Включает атрибуты:
 - `trmId` – идентификатор терминала;
 - `agtId` – идентификатор агента;
 - `lastPaymentTime` – время последнего принятого платежа на терминале;
 - `systemUpTime` – время работы ОС терминала;
 - `progUpTime` – время работы ПО терминала;
 - `paysPerHour` – среднее число запросов на проведение платежей в час;
 - `billsPerPay` – среднее число платежей, передаваемых в одном запросе;
 - `CardReaderUsedHour` – количество успешных считываний карт в картридере за текущий час;
 - `CardReaderUsedDay` – количество успешных считываний карт в картридере за текущие сутки;
 - `timeToCashinFull` – оставшееся время до заполнения купюроприемника;
 - `timeToCashinService` – оставшееся время до обслуживания купюроприемника;
 - `timeToPrinterPaperOut` – оставшееся время до окончания бумаги в чековом принтере;
 - `timeToPrinterService` – оставшееся время до обслуживания принтера.

6.7.8. `getTerminalsCash`

Описание: Запрос позволяет получить информацию о купюрности терминала.

Режим обработки: Строго синхронно.

Роли:

- Автомат;
- Главный менеджер;

- Менеджер;
- Кассир;
- Продавец;
- Техподдержка;
- Дистрибьютор;
- Мониторинг.

6.7.8.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <reports>
    <getTerminalsCash>
      <target-terminal>123</target-terminal>
    </getTerminalsCash>
  </reports>
</request>
```

Для выполнения запроса можно указать дополнительные параметры:

- `target-terminal` – терминал, для которого необходимо получить информацию о купюренности (если не указан, выполняется для всех терминалов *текущего* агента и его субагентов);
- `target-agent` – агент, для которого необходимо выполнить действие (если не указан, выполняется для *текущего* агента).

6.7.8.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <reports>
    <getTerminalsCash result="0">
      <terminal agent="2" id="123">
        <currency id="643">
          <notes GoBy="0" SumGoBy="0.0" count="43" sum="4370.00">
            <nominal count="2" value="10.00"/>
            <nominal count="19" value="50.00"/>
            <nominal count="19" value="100.00"/>
            <nominal count="3" value="500.00"/>
            <nominal count="0" value="1000.00"/>
            <nominal count="0" value="5000.00"/>
          </notes>
          <coins GoBy="0" count="0" sum="0.000">
            <nominal count="0" value="0.01"/>
            <nominal count="0" value="0.05"/>
            <nominal count="0" value="0.10"/>
            <nominal count="0" value="0.50"/>
            <nominal count="0" value="1.00"/>
            <nominal count="0" value="2.00"/>
            <nominal count="0" value="5.00"/>
            <nominal count="0" value="10.00"/>
          </coins>
        </currency>
      </terminal>
    </getTerminalsCash>
  </reports>
</response>
```

Ответ содержит следующие данные:

- `terminal` – тег терминала. Содержит:
 - `agent` – идентификатор агента, котрому принадлежит описываемый терминал,
 - `id` – идентификатор теминала;
- `currency` – тег , содержащий информацию о валюте денежных средств:
 - `id` – код валюты. Значения кодов соответствуют **ISO 4217**;
 - `notes` – тег информации о купюрах данной валюты:
 - ⊕ `GoBy` – количество купюр, внесенных, но не учтенных в теминале,
 - ⊕ `SumGoBy` – сумма всех купюр, внесенных, но не учтенных в теминале,
 - ⊕ `count` – количество учтенных купюр,
 - ⊕ `sum` – сумма купюр, учтенных в терминале,
 - ⊕ `nominal` – информация о количестве купюр одного номинала (10, 50,100, 500, 1000, 5000, 10000, 50000):
 - ❖ `value` – номинал,
 - ❖ `count` – количество купюр этого номинала;
 - `coins` – тег информации о монетах данной валюты (в случае если имеется монетоприемник):
 - ⊕ `GoBy` – количество монет, внесенных, но не учтенных в теминале,
 - ⊕ `count` – количество учтенных монет,
 - ⊕ `sum` – сумма монет, учтенных в терминале,
 - ⊕ `nominal` – информация о количестве монет одного номинала (0.01, 0.05, 0.1, 0.5, 1.0, 2.0, 5.0, 10.0):
 - ❖ `value` – номинал,
 - ❖ `count` – количество монет этого номинала.

6.8. AccountingReports

Список действий интерфейса:

Действие	Описание
getSubagentAccountStatement	Выписка по счетам субагентов
getAccountingTerminal	Выгрузка движений по терминальным счетам
getAgentTariffs	Выгрузка истории установки тарифных планов для агента и его субагентов
getAgentTP	Выгрузка детализации тарифных планов, установленных агенту и его субагентам

Действие	Описание
getExemptionRates	Выгрузка исключений из ставок тарифных планов, установленных агенту и его субагентам в особом порядке по определенным провайдерам

ПРИМЕЧАНИЕ

Обязательные параметры запросов отмечены символом *

6.8.1. getSubagentAccountStatement

Описание: Получение выписки по счетам субагентов.

Режим обработки: Возможно асинхронно.

Роли:

- Главный менеджер;
- Менеджер;
- Техподдержка;
- Дистрибьютор;
- Бухгалтер.

6.8.1.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <accountingReports>
    <getSubagentAccountStatement>
      <target-date>2011-06-30</target-date>
      <target-agent>3</target-agent>
    </getSubagentAccountStatement>
  </accountingReports>
</request>
```

Для выполнения запроса необходимо указать дополнительные параметры:

- `target-agent*` – агент, для которого необходимо получить выписку;

ПРИМЕЧАНИЕ

Если у агента в `target-agent` есть субагенты, выписка будет содержать движения по счетам не только текущего агента, но и всех его субагентов.

- `target-date*` – отчетный период.

6.8.1.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <accountingReports>
    <getSubagentAccountStatement result="0">
      <row ap_card_type_id="" ap_cms_type_id="1" ap_date="2011-06-
30T00:00:00+04:00" ap_period_date="2011-06-30T00:00:00+04:00" from_acc_type_bu="20"
from_agt_id="1" money_in="10" money_out="0" period_date="2011-06-30T00:00:00+04:00"
to_acc_type_bu="60.К" to_agt_id="3"/>
      <row ap_card_type_id="" ap_cms_type_id="" ap_date="2011-06-30T00:00:00+04:00"
ap_period_date="" from_acc_type_bu="76.К2" from_agt_id="10" money_in="0"
money_out="3630" period_date="2011-06-30T00:00:00+04:00" to_acc_type_bu="76.КА"
to_agt_id="10"/>
    </getSubagentAccountStatement>
  </accountingReports>
</response>
```

Ответ содержит следующие данные:

- row – тег, содержащий данные о проводках по указанным счетам, субагентам и периоду:
 - ap_card_type_id – тип списания по картотеке;
 - ap_cms_type_id – тип комиссии;
 - ap_date – дата и время формирования проводки в процессинге;
 - ap_period_date – дата отнесения к периоду;
 - from_acc_type_bu – счет в плане счетов учета, с которого списаны средства (см. [Приложение К](#));
 - from_agt_id – идентификатор агента, со счета которого списаны средства;
 - money_in – поступило средств на счет, по которому формируется выписка;
 - money_out – списано средств со счета, по которому формируется выписка;
 - period_date – дата отнесения к периоду;
 - ⊕ если поле ap_period_date не пусто, то period_date – дата отнесения к периоду,
 - ⊕ если поле ap_period_date пусто, то period_date – дата формирования проводки в процессинге;
 - to_acc_type_bu – счет в плане счетов учета, на который начислены средства (см. [Приложение К](#));
 - to_agt_id – номер агента, на счет которого начислены средства.

6.8.2. getAccountingTerminal

- Описание:** Выгрузка движений по терминальным счетам всех терминалов, привязанных к агенту и его субагентам.
- Режим обработки:** Возможно асинхронно.
- Роли:** Выполнение запроса доступно персоне с любой ролью.



Агент определяется по логину авторизованной в запросе персоны.

6.8.2.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <accountingReports>
    <getAccountingTerminal>
      <day>2009-04-23</day>
    </getAccountingTerminal>
  </accountingReports>
</request>
```

Для выполнения запроса следует указать обязательный параметр:

- **day*** – дата, за которую необходимо получить движения по счетам.

6.8.2.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <accountingReports>
    <getAccountingTerminal result="0">
      <row amnt="111.33" cnt="20" from_acc="10" pay_id="2" prv_id="1" to_acc="25"
trm_id="111"/>
      ...
      <row amnt="200" cnt="1" from_acc="30" pay_id="3" prv_id="2" to_acc="29"
trm_id="123"/>
    </getAccountingTerminal>
  </accountingReports>
</response>
```

Ответ содержит следующие данные:

- **row** – тег, содержащий данные о движениях по указанным счетам, терминалам и провайдерам:
 - **amnt** – общая сумма движений по указанным в атрибутах счетам для выбранного провайдера на данном терминале;
 - **cnt** – число движений по выбранному провайдеру и указанным терминальным счетам;
 - **from_acc** – идентификатор терминального счета, с которого списаны средства (см. [Приложение К](#));
 - **pay_id** – шлюз провайдера;
 - **prv_id** – идентификатор провайдера;
 - **to_acc** – идентификатор терминального счета, на который зачислены средства (см. [Приложение К](#));
 - **trm_id** – идентификатор терминала.

6.8.3. getAgentTariffs

Описание: Выгрузка истории установки тарифных планов (см. [Глоссарий](#)) для агента и его субагентов.

Режим обработки: Возможно асинхронно.

Роли:

- Главный менеджер;
- Продавец;
- Дистрибьютор;
- Дистрибьютор (новый агентский сайт);
- Координатор Дистрибьютора.

6.8.3.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <accountingReports>
    <getAgentTariffs/>
  </accountingReports>
</request>
```

Для выполнения запроса дополнительные параметры не указываются.

6.8.3.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <accountingReports>
    <getAgentTariffs result="0">
      <row jur_agt_id="3" tp_date_from="2010-09-01T00:00:00+04:00"
tp_date_till="2010-12-01T00:00:00+03:00" tp_id="10" tp_prs_id="1"/>
      <row jur_agt_id="3" tp_date_from="2010-12-01T00:00:00+03:00"
tp_date_till="4000-01-01T00:00:00+03:00" tp_id="10" tp_prs_id="1"/>
      <row jur_agt_id="4" tp_date_from="2011-08-01T15:35:38+04:00"
tp_date_till="4000-01-01T00:00:00+03:00" tp_id="10" tp_prs_id="2"/>
      <row jur_agt_id="5" tp_date_from="2011-08-01T15:36:13+04:00"
tp_date_till="4000-01-01T00:00:00+03:00" tp_id="10" tp_prs_id="2"/>
      <row jur_agt_id="6" tp_date_from="2011-08-01T15:36:31+04:00"
tp_date_till="4000-01-01T00:00:00+03:00" tp_id="10" tp_prs_id="2"/>
      <row jur_agt_id="7" tp_date_from="2011-08-01T15:36:50+04:00"
tp_date_till="4000-01-01T00:00:00+03:00" tp_id="10" tp_prs_id="2"/>
    </getAgentTariffs>
  </accountingReports>
</response>
```

Ответ содержит следующие данные:

- row – тег, содержащий данные о тарифном плане:
 - jur_agt_id – идентификатор агента;
 - tp_date_from – дата начала действия тарифного плана агента;
 - tp_date_till – дата окончания действия тарифного плана агента;

- tp_id – идентификатор тарифного плана;
- tp_prs_id – персона, установившая тарифный план агенту.

6.8.4. getAgentTP

Описание: Выгрузка детализации тарифных планов (см. [Глоссарий](#)) агента и его субагентов.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.8.4.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" signAlg="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <accountingReports>
    <getAgentTP>
      <target-agent>3</target-agent>
      <tp-id>2</tp-id>
      <tp_num>1</tp_num>
    </getAgentTP>
  </accountingReports>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- target-agent* – агент, для которого необходимо получить тарифные планы;
- tp-num – номер тарифного плана, ставку по которому требуется получить;
- tp-id – идентификатор тарифного плана.

6.8.4.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <accountingReports>
    <getAgentTP result="0">
      <row cms_rate="-.01" date_from="2012-04-23T10:09:36+04:00" date_till="4000-01-01T01:00:00+04:00" pay_id="2" pay_name="Карты" prv_id="1" prv_short_name="МТС Карты" tp_id="2" tp_num="1" software_cms_rate=".001"/>
      <row cms_rate="-1.1" date_from="2012-04-23T10:09:36+04:00" date_till="4000-01-01T01:00:00+04:00" pay_id="3" pay_name="ЗАО ЛАНК" prv_id="3" prv_short_name="ЛАНК" tp_id="2" tp_num="1" software_cms_rate=".001">
        <add_cms_rate tpt_id="10" cms_rate="-.59" date_from="2013-07-01T00:00:00+04:00" date_till="4000-01-01T00:00:00+04:00"/>
        ...
        <bonus_rate tpt_id="10" service_level="1" cms_rate="-.25" date_from="2014-10-01T00:00:00+04:00" date_till="4000-01-01T00:00:00+04:00"/>
        ...
      </row>
      ...
    </getAgentTP>
  </accountingReports>
</response>
```

Ответ содержит следующие данные:

- `row` – тег, содержащий данные о ставках нижней комиссии и других комиссий тарифного плана по шлюзу в следующих атрибутах:
 - `pay_id` – идентификатор шлюза;
 - `pay_name` – наименование шлюза;
 - `prv_id` – идентификатор провайдера;
 - `prv_short_name` – короткое наименование провайдера;
 - `tp_id` – идентификатор тарифного плана;
 - `tp_num` – номер тарифного плана;
 - `date_from` – дата начала действия ставки тарифного плана субагента;
 - `date_till` – дата окончания действия ставки тарифного плана субагента;

ВНИМАНИЕ

Если ставка комиссии не установлена, соответствующий атрибут отсутствует в ответе.

- `cms_rate` – ставка вознаграждения по шлюзу (в процентах);
- `cms_rate_fix` – ставка вознаграждения по шлюзу (фиксированная);
- `register_cms_rate` – ставка платы за КИВИ Регистратор (в процентах);
- `register_cms_rate_fix` – ставка платы за КИВИ Регистратор (фиксированная);
- `opp_cms_rate` – ставка дополнительного вознаграждения КИВИ (в процентах);
- `opp_cms_rate_fix` – ставка дополнительного вознаграждения КИВИ (фиксированная);
- `software_cms_rate` – ставка платы за ПО;
- `kb_cms_fix` – комиссия Расчетного банка (фиксированная);
- `kb_cms_percent` – комиссия Расчетного банка (в процентах);
- `kb_cms_min` – минимальная величина комиссии Расчетного банка;
- `kb_cms_max` – максимальная величина комиссии Расчетного банка;
- `add_cms_rate` – вложенный тег, содержащий данные о дифференцированной ставке нижней комиссии тарифного плана по шлюзу из тега `row` (присутствует для некоторых шлюзов):
 - `tpt_id` – идентификатор дополнительного типа терминала, по которому определяется ставка (см. [Приложение М](#));
 - `cms_rate` – дифференцированная ставка КИВИ по шлюзу (в процентах);
 - `date_from` – дата начала действия дифференцированной ставки;
 - `date_till` – дата окончания действия дифференцированной ставки;
- `bonus_rate` – вложенный тег, содержащий данные о ставке переменного вознаграждения за выполнение уровня сервиса по шлюзу из тега `row` (присутствует для некоторых шлюзов):
 - `tpt_id` – идентификатор дополнительного типа терминала, по которому определяется ставка (см. [Приложение М](#));
 - `cms_rate` – ставка вознаграждения субагента за выполнение уровня сервиса по шлюзу (в процентах);

- date_from – дата начала действия ставки вознаграждения;
- date_till – дата окончания действия ставки вознаграждения;
- service_level – номер уровня сервиса, согласно Правилам работы системы АО "КИВИ".

6.8.5. getExemptionRates

Описание: Выгрузка исключений из ставок тарифных планов агента и его субагентов, установленных в особом порядке по определенным провайдерам.

Режим обработки: Возможно асинхронно.

Роли: Выполнение запроса доступно персоне с любой ролью.

6.8.5.1. Запрос

```
<?xml version="1.0" encoding="utf-8"?>
<request>
  <auth login="login" sign="sign" 187signal="MD5"/>
  <client terminal="111" software="Dealer v0" serial=""/>
  <accountingReports>
    <getExemptionRates>
      <target-agent>3</target-agent>
    </getExemptionRates>
  </accountingReports>
</request>
```

Для выполнения запроса указываются дополнительные параметры:

- target-agent* – агент, для которого необходимо получить ставки исключений из тарифного плана;
- date-from ... date-till – даты начала и окончания периода действия исключений из тарифного плана. Если не указаны, то выгружаются исключения, действующие на текущие сутки.

6.8.5.2. Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<response result="0">
  <accountingReports>
    <getExemptionRates result="0">
      <row agt_id="11112" cms_rate="-1.57" cms_rate_fix="0" date_from="2012-04-23T10:09:36+04:00" date_till="4000-01-01T01:00:00+04:00" pay_id="1157" pay_name="Провайдер1" prv_id="3" prv_short_name="Провайдер1 Столичный"/>
      <row agt_id="11112" cms_rate="-.26" cms_rate_fix="0" date_from="2012-04-23T10:09:36+04:00" date_till="4000-01-01T01:00:00+04:00" pay_id="341" pay_name="ТВ2" prv_id="21" prv_short_name="ТВ2"/>
      ...
      <row agt_id="11112" cms_rate="-1.57" cms_rate_fix="0" date_from="2012-04-23T10:09:36+04:00" date_till="4000-01-01T01:00:00+04:00" pay_id="5754" pay_name="Провайдер2" prv_id="26" prv_short_name="Провайдер2 Сибирь"/>
    </getExemptionRates>
  </accountingReports>
</response>
```

Ответ содержит следующие данные:

- row – тег, содержащий данные об исключениях из тарифного плана субагента:
 - agt_id – идентификатор субагента;
 - cms_rate – входящая ставка КИВИ по шлюзу (в процентах);
 - cms_rate_fix – входящая фиксированная ставка КИВИ по шлюзу;
 - date_from – дата начала действия исключения из тарифного плана субагента;
 - date_till – дата окончания действия исключения из тарифного плана субагента;
 - pay_id – идентификатор шлюза;
 - pay_name – наименование шлюза;
 - prv_id – идентификатор провайдера;
 - prv_short_name – короткое наименование провайдера.

ПРИЛОЖЕНИЕ А: Маски полей ввода

[Интерфейс терминала](#) формируется динамически. Для управления данными, вводимыми пользователем в поле, используются маски ввода (см. формат ответа на запрос [getUIProviders](#)). Маска ввода представляет собой набор текстовых констант и знаков, определяющий, какие данные и каким образом можно вводить.

Стандартное представление маски ввода (в одном поле ввода может задаваться более одной маски):

```
<!^регулярное_выражение${N}>
```

В маске ввода используются следующие символы:

- <!^ ... > – символы, ограничивающие поле ввода,
- регулярное_выражение – специальное выражение, позволяющее задать тип разрешенных для ввода символов,

ПРИМЕЧАНИЕ



Регулярное выражение всегда должно начинаться с символа ^ и заканчиваться символом \$.

В регулярном выражении могут быть использованы *только* следующие управляющие конструкции:

- \d+ – число,
- \w+ – любой символ,
- .* – строка, состоящая из любых символов (в том числе пустая). Длина строки варьируется от 0 до 255 символов, т.е. поле может быть пустым, имеет максимальную длину 255 символов,
- .+ – строка, состоящая из любых символов (длина строки не менее 1 символа). Длина строки варьируется от 1 до 255 символов, поле должно быть длиной минимум – 1 символ, максимум – 255 символов.
- [a-zA-Z_@]+ – позволяет задать определенный набор символов, которые могут быть использованы в поле ввода.

ПРИМЕЧАНИЕ



Символ "-" в управляющих конструкциях вида [a-zA-Z_@] задается непосредственно перед закрывающей скобкой "]" (например, [a-zA-Z_@-]).

ПРИМЕЧАНИЕ



Для управляющих конструкций вида [a-zA-Z_@] не допускается использование логических выражений:

- [A|B]

или

- [!A]

- \$ – специально зарезервированный символ, разделяющий регулярное выражение и число символов в регулярном выражении,
- {N} ({N, M}) – количество символов в поле ввода:
 - N – количество обязательных символов,
 - N, M – длина поля от N до M, т.е. N – минимальное число символов в поле, M – максимальное.

Пример 21. Примеры задания маски полей ввода

1. Для отображения поля ввода следующего вида (номер телефона):

(3 цифры) 3 цифры – 2 цифры – 2 цифры

используется следующая маска:

(<!\d+\${3}>) <!\d+\${3}> – <!\d+\${2}> – <!\d+\${2}>

2. Для отображения поля ввода следующего вида (номер счета):

3 – 6 символов / 2 – 5 цифр

используется следующая маска:

<!.+\${3,6}> / <!\d+\${2,5}>

3. Для отображения поля ввода следующего вида (логин):

3 – 6 букв латинского алфавита

используется следующая маска:

<![a-zA-Z]+\${3,6}>

4. Для отображения поля ввода следующего вида (email):

1 – 255 символов @ 1 – 255 строчных букв латинского алфавита, арабские цифры, дефисы, подчеркивания, точки
 . 2 – 10 строчных букв латинского алфавита

используется следующая маска:

<!.+\${1,255}> @ <![a-z0-9._]+\${1,255}> . <![a-z]+\${2,10}>

5. Для отображения поля ввода следующего вида (фамилия):

1 – 255 букв русского алфавита и дефисы

используется следующая маска:

<![а-яА-Я\ -]+\${1,255}>

ПРИЛОЖЕНИЕ Б: Регулярные выражения в полях ввода

Для проверки значений, введенных в поле ввода, используются регулярные выражения (см. раздел [6.4.12](#)). Регулярное выражение – шаблон, с помощью которого настраивается ограничение на поле ввода.

Регулярное выражение может состоять из простых символов (букв, цифр и прочих символов, не являющихся зарезервированными), а также служебных символов – спецсимволов (см. [Табл. 16](#)) и операторов (см. [Табл. 17](#)).

Табл. 16. Спецсимволы

Спецсимвол	Назначение
^	Начало строки
\$	Конец строки
.	Любой символ, кроме перевода строки
[...]	Любой символ из набора, перечисленного в скобках, диапазоны символов могут указываться через тире (например, [0-3] – любая цифра от 0 до 3)
\s	Пробел
\S	Любой символ, кроме пробела и знаков табуляции
\w	Символ слова (буквы, цифры, знак подчеркивания)
\W	Любой символ, кроме символов слова
\d	Любая цифра
\D	Любой символ, кроме цифр

Табл. 17. Операторы

Оператор	Назначение
(...)	Группировка символов, перечисленных внутри скобок
	Логическое ИЛИ (предыдущий или следующий символ)
*	Повтор предыдущего символа ноль или более раз
+	Повтор предыдущего символа один или более раз
?	Предыдущий символ ноль или один раз
{n}	Повторить предыдущий символ n раз

Оператор	Назначение
{n,}	Повторить предыдущий символ n или более раз
{n,m}	Повторить предыдущий символ от n до m раз
{,m}	Повторить предыдущий символ m или менее раз

Стандартное представление регулярного выражения:

```
^регулярное_выражение$
```

В регулярном выражении используются следующие символы:

- `^` – символ начала строки регулярного выражения,
- `регулярное_выражение` – специальное выражение, позволяющее задать тип символов. Регулярное выражение может быть одним из:
 - простые символы;
 - спецсимволы;
 - операторы;
- `$` – зарезервированный символ конца строки регулярного выражения.
-

Пример 22. Использование спецсимволов и операторов

```
^\d{1,10}(A|B|G)$ – строка, состоящая из любых цифр, в количестве от 1 до 10 и любая буква из A, B, G.
^\d{2}-\d{3}$ – строка, которая состоит ровно из шести символов: две цифры, тире, три цифры
^[0-9a-z]{10}$ – строка, состоящая из цифр и латинских букв в нижнем регистре, расположенных в произвольном порядке, длина строки: ровно 10 цифр
^(32){1}[АБ]?\d{4,8}$ - строка, состоящая из цифр 3, 2 (один раз), любой буквы из А,Б (один раз или ни разу) и от 4 до 8 цифр
^a-\d{3}$ – строка, начинающаяся с символов "a-", после которых идет ровно три цифры.
```

Пример 23. Использование зарезервированных символов

```
^\$\$ – строка, состоящая из знака доллара.
```

Пример 24. Примеры регулярных выражений для проверки

1. Для проверки значения в поле ввода следующего вида:

(3 цифры) 3 цифры – 2 цифры – 2 цифры

используется следующее выражение:

$^{\wedge}(\{3\}\{3\}\{2\}\{2\})\{2\}$

2. Для проверки поля ввода следующего вида:

3 – 6 символов / 2 – 5 цифр

используется следующее выражение:

$^{\wedge}\{s\{3,6\}\{d\{2,5\}\}$

3. Для проверки поля ввода следующего вида:

3 – 6 букв латинского алфавита

используется следующее выражение:

$^{\wedge}[a-zA-Z]\{3,6\}$

ПРИЛОЖЕНИЕ В: Коды ошибок

ПРИМЕЧАНИЕ



Существуют фатальные и нефатальные ошибки (см. столбец **Фатальность** в таблице ниже):

- Фатальная ошибка – отрицательный ответ на клиентский запрос.
- Нефатальная ошибка – промежуточный статус обработки клиентского запроса, дальнейшие действия зависят от конкретной ошибки и [статуса платежа](#) (финальный/нефинальный).

Табл. 18. Коды ошибок процессинга

Код	Текст	Комментарий	Фатальность
0	ОК	Платеж успешно проведен	0
1	Провайдер временно недоступен	Провайдер не успевает обработать ваш запрос в указанное время	0
2	Превышено число попыток	Техническая ошибка сервера. Временно прекратите прием платежей по данному провайдеру. Сообщите о возникновении ошибки в службу поддержки. В течение 5-10 минут проблема будет устранена.	1
3	Техническая ошибка, нельзя отправить запрос провайдеру	Техническая ошибка сервера. Временно прекратите прием платежей по данному провайдеру. Сообщите о возникновении ошибки в службу поддержки. В течение 5-10 минут проблема будет устранена.	0
4	Неверный формат счета/телефона	Вы неверно ввели номер счета или номер телефона. Попробуйте снова внимательно ввести номер счета или телефона. Если ошибка повторяется – обратитесь в службу поддержки.	1
5	Номер не принадлежит оператору	Номер счета или телефона, который клиент Вам сообщил, не существует в базе данных провайдера, или Вы неверно его ввели. Проверьте, правильно ли Вы выбрали провайдера, правильно ли ввели номер телефона. Попробуйте ввести данные еще раз. Возможно, клиент сообщил Вам ошибочный номер.	1
6	Прием платежа запрещен, обратитесь в банк	Проблема в финансовых взаиморасчетах между Вами и платежной системой, либо между платежной системой и провайдером. Проверьте баланс Вашего дилерского счета в платежной системе.	1
7	Прием платежа запрещен, обратитесь к оператору	Проблема во взаимоотношениях клиента с провайдером. Провайдер запретил прием платежа для этого клиента. Сообщите клиенту об этой проблеме, попросите его самостоятельно связаться с провайдером и решить этот вопрос.	1

Код	Текст	Комментарий	Фатальность
8	Прием платежа запрещен по техническим причинам	Проблема платежной системы или провайдера. Возможно, провайдер производит технические работы с оборудованием и временно прекратил прием платежей. Временно прекратите прием платежей по данному провайдеру. Сообщите о возникновении ошибки в службу поддержки. Время устранения зависит от серьезности проблемы.	1
9	Timeout от провайдера	Проблема коммуникации платежной системы с провайдером. Временно прекратите прием платежей по данному провайдеру. Сообщите о возникновении ошибки в службу поддержки. Время устранения зависит от серьезности проблемы.	0
10	Дублирование платежа	Найдены платежи на тот же номер телефона (и возможно ту же сумму – завистит от режима проверки дубликатов)	1
11	Устаревшая версия клиентского приложения	Обновите клиентское приложение до последней актуальной версии	1
12	Невозможно записать сертификат на данное хранилище	Невозможно записать сертификат на данное хранилище	1
13	Сервер занят, повторите запрос через минуту	Временная проблема с обработкой запроса на стороне QIWI, повторите отправку запроса с теми же реквизитами через минуту.	0
15	Истекло время ожидания платежа в очереди	Истекло время ожидания платежа в очереди на отправку провайдеру	1
16	Превышен суточный лимит на сумму операций	Превышен суточный лимит на сумму операций	1
18	Ошибка при регистрации терминала	Ошибка при регистрации терминала	0
19	Транзакция не подтверждена в течение 24 часов	Истекло время ожидания запроса на подтверждение платежа	0
20	Ошибка допинформации платежа	Ошибка в экстра-полях платежа	1
23	Не удалось выставить шлюз	Не удалось назначить платежу шлюз провайдера	0
42	Провайдер не определен	Невозможно определить провайдера сотовой связи автоматически. Укажите провайдера вручную.	1
71	Домашний оператор не может принять платеж	Домашний оператор абонента не может принять платеж. Попросите клиента обратиться к своему домашнему оператору.	1
72	ЛС платежной системы не доступен	Проблема взаимодействия платежной системы с провайдером. Временно прекратите принимать платежи по данному провайдеру. Обратитесь в службу поддержки.	0

Код	Текст	Комментарий	Фатальность
73	Домашний оператор не доступен	Внутренняя проблема МТС при взаимодействии с домашним оператором абонента. Обратитесь в службу поддержки.	0
75	Ошибка взаимодействия с провайдером	Проблема взаимодействия процессингового центра с провайдером. Временно прекратите прием платежей для данного провайдера. Обратитесь в службу поддержки.	1
77	Платеж запрещен провайдером	Прием данного платежа запрещен провайдером	1
78	Провайдер не может провести платеж	В информационной системе провайдера возникла техническая ошибка, в результате которой прием данного платежа в настоящее время невозможен. В случае финального статуса платежа обратитесь в службу поддержки.	0
79	Счет абонента не активен	Счет клиента не активен. Попросите клиента обратиться к своему домашнему оператору для активации счета.	1
81	Не найдено описание клиента	Не найдено описание клиента в информационной системе провайдера. Клиент должен обратиться к провайдеру.	1
82	Провайдер запретил прием платежа ПЦ	Провайдер запретил прием платежа процессинговому центру из-за состояния счета или других административных причин. Временно прекратите прием платежей для данного провайдера. Обратитесь в службу поддержки.	0
85	Отмена платежа невозможна	Запрет на отмену платежа	1
86	Не определено состояние платежа	Статус платежа не определен	0
90	Проведение платежа не окончено	Проведение платежа не окончено. Результат проведения будет известен позже.	0
100	Ошибка сервера	Техническая ошибка сервера. Временно прекратите прием платежей по данному провайдеру. Сообщите о возникновении ошибки в службу поддержки. В течение 5-10 минут проблема будет устранена.	0
101	Нет такого файла для загрузки	Ошибка ПО терминала. Временно прекратите прием платежей по данному провайдеру. Сообщите о возникновении ошибки в службу поддержки. Время устранения зависит от серьезности проблемы.	0
130	Работа с данным провайдером не реализована	В настоящий момент прием платежей для выбранного провайдера невозможен. Сообщите клиенту о невозможности приема платежа в данный момент.	1
133	Нет прав на прием платежей	Персона, от имени которой посылается запрос, не имеет права на выполнение данного действия	1

Код	Текст	Комментарий	Фатальность
135	Нельзя списывать с чужого счета	Попытка списания средств со счета другого агента	0
150	Неверный пароль или нет прав на этот терминал	Ошибочно введен номер терминала, имя пользователя или пароль. Проверьте ваши регистрационные данные.	1
151	Невозможно выполнить операцию. Одноразовый пароль.	Невозможно выполнить операцию. Требуется одноразовый пароль.	1
152	Невозможно выполнить операцию. Неодноразовый пароль.	Невозможно выполнить операцию. Требуется одноразовый пароль.	1
153	Персона временно заблокирована. 10 неудачных попыток пароля.	Персона временно заблокирована (на час, после 10 неудачных авторизаций в течение часа)	1
154	Проведение платежа запрещено СБ	Проведение платежа запрещено службой безопасности	1
155	Прием платежа для данного провайдера запрещен	Прием платежа для данного провайдера запрещен для данного терминала или агента	1
158	Невозможно выполнить операцию. Запрещено использование постоянного пароля.	Использование авторизации по постоянному паролю для этой операции запрещено. Используйте авторизацию по цифровой подписи.	1
170	Запрос принят	Запрос принят сервером	0
171	Запрос не актуален	Запрос выполнен слишком поздно	1
172	Данные плательщика не прошли проверку	Данные плательщика не прошли проверку, платеж отклонен	1
173	Превышен лимит суммы платежа без идентификации	Превышен лимит суммы платежа без идентификации, для совершения таких платежей обратитесь к своему курирующему менеджеру.	1
201	Слишком много запросов с терминала	Превышено число одновременных запросов с терминала	0
202	Ошибка данных запроса	Ошибка параметров запроса – возможно, не соблюден формат протокола. Исправьте причину и повторно отправьте запрос. Возможные причины возникновения ошибки: <ul style="list-style-type: none"> • отсутствие логина и/или пароля; • отсутствие открывающего и/или закрывающего тега; • отсутствуют обязательные параметры. 	1
203	Транзакция не найдена в базе данных	Транзакция не найдена в базе данных	1
204	Переданы не все обязательные поля	В платеже отсутствуют обязательные поля, заданные в конфигурации провайдера	1

Код	Текст	Комментарий	Фатальность
205	Отсутствуют обязательные данные плательщика	В платеже отсутствуют данные плательщика, требуемые законодательством	1
209	Ошибка авторизации	Ошибка SQL авторизации, вероятно транзакция заблокирована (locked) в базе	1
210	Нет такой транзакции в базе	Ошибка ПО терминала. Временно прекратите прием платежей по данному провайдеру. Сообщите о возникновении ошибки в службу поддержки. Время устранения зависит от серьезности проблемы.	1
211	Неверный статус транзакции	Ошибка ПО терминала. Временно прекратите прием платежей по данному провайдеру. Сообщите о возникновении ошибки в службу поддержки. Время устранения зависит от серьезности проблемы.	1
212	Не задана сумма платежа (или лимит терминала)	Вы не ввели сумму платежа или ввели сумму, превышающую лимит платежа. Попробуйте снова ввести данные и повторить операцию. Если ошибка повторяется, обратитесь в службу поддержки.	1
213	Не задана сумма списания	Сумма, полученная от клиента, не указана в данных платежа	1
214	Запросы с данного IP-адреса запрещены	IP-адрес, с которого выполняется запрос, не находится в списке разрешенных для терминала или агента	1
215	Транзакция с таким номером уже есть в базе	Нарушение логики работы. Ошибка возникает, когда под одним и тем же номером терминала работают несколько пользователей. Платеж, на котором возникла эта ошибка, нужно провести снова.	0
216	Ошибка при сохранении транзакции на сервере	Ошибка при сохранении данных о транзакции в БД	0
217	Одинаковые номера терминальной транзакции в пакете	Одинаковые номера терминальной транзакции в одном пакете платежей	0
220	Недостаток средств у агента	Проблема в финансовых взаимоотношениях между Вами и платежной системой. На Вашем ЛС недостаточно средств для приема этого платежа. Произведите пополнение ЛС.	0
240	Не проведена	Платеж не проведен или истек лимит времени от авторизации до платежа	1
241	Сумма слишком мала	Сумма слишком мала	1
242	Сумма слишком велика	Сумма слишком велика	1
244	Терминал не зарегистрирован у оператора	Терминал не зарегистрирован у оператора	0

Код	Текст	Комментарий	Фатальность
245	Неверный тип терминала	В тегах авторизации неверно указан тип терминала. Проверьте, соответствует ли тип терминала, указанный в запросе, тому терминалу, который вы реально используете.	1
246	Терминал привязан к другому компьютеру	Серийный номер терминала указан неверно	1
252	Превышена максимальная сумма оплаты для терминала	Сумма превышает максимально установленную для терминала	1
255	Недопустимая дополнительная комиссия	Недопустимая дополнительная комиссия для провайдера	1
270	Ошибка от провайдера	Ошибка, сгенерированная провайдером	0
271	Ошибка в ответе провайдера	Проблема платежной системы или провайдера. Возможно, провайдер производит технические работы с оборудованием и временно прекратил прием платежей. Временно прекратите прием платежей по данному провайдеру. Сообщите о возникновении ошибки в службу поддержки. Время устранения зависит от серьезности проблемы.	0
272	Временно нет связи с провайдером	Временно нет связи с провайдером. Приостановите прием платежей по данному провайдеру. Сообщите о возникновении ошибки в службу поддержки.	0
275	Некорректная сумма платежа	Сумма не принадлежит фиксированному перечню разрешенных платежей на данный тип абонента	0
280	Недостаток средств у агента выше	Проблема в финансовых взаимоотношениях между вышестоящим агентом и системой. На ЛС вышестоящего агента недостаточно средств для приема этого платежа.	0
290	Ошибка блокировки баланса для списания средств	При прохождении финансового контроля не может быть заблокирован баланс агента	0
291	Ошибка при блокировке баланса	Ошибка при блокировке баланса агента	0
295	Ошибка в названии интерфейса или действия	Проверьте корректность наименования указанного в запросе интерфейса или действия	1
300	Другая (неизвестная) ошибка провайдера	Повторите запрос с тем же набором данных (при проведении платежа – с тем же идентификатором платежа) до получения ответа со статусом/кодом ошибки, на основе которого можно сделать вывод о прохождении платежа/выполнении действия	0
302	Проведение для пользовательского провайдера	Проведение платежа для пользовательского провайдера	1
304	Тариф недоступен для пополнения	Тариф недоступен для пополнения	1
305	Тарифов для пополнения не найдено	По указанным реквизитам тарифов для пополнения не найдено	1

Код	Текст	Комментарий	Фатальность
358	Начислений не найдено	По указанным реквизитам неоплаченных начислений не найдено	1
507	Платёж отменён	Платёж отменён в системе	0
508	Платёж отменён через отдел отмен	Платёж отменён вручную	0
515	Прием платежей вне режима работы терминала запрещен	Прием платежей вне установленного времени работы терминала запрещен	1
518	Данный тип терминала заблокирован	Работа с терминалами данного типа невозможна	1
564	Карта заблокирована	Карта заблокирована	1
565	Истек срок действия карты	Истек срок действия карты	1
570	Процессинг банка-эмитента карты недоступен	Процессинг банка-эмитента карты недоступен. Попробуйте повторить операцию позднее	1
571	Платеж отклонен банком-эмитентом	Платеж на данную карту отклонен банком-эмитентом. Обратитесь в банк-эмитент	1

ПРИЛОЖЕНИЕ Г: Статусы платежей

ПРИМЕЧАНИЕ



Статус платежа возвращается в атрибуте `status` в ответе на платежные запросы.

Табл. 19. Статусы платежей в процессинге

Код	Описание	Финальный
0	Ошибка проведения	Да
1	Проводится	Нет
2	Проведен	Да
3	Авторизован	Нет

ПРИЛОЖЕНИЕ Д: Статусы асинхронных запросов

Табл. 20. Статусы обработки асинхронных запросов

Статус	Описание	Финальный
1	Ожидание обработки	Нет
2	Обрабатывается	Нет
3	Успешно обработан	Да
4	Ошибка обработки	Да
5	Таймаут	Да
6	Результат запроса был удален из БД	Да

ПРИЛОЖЕНИЕ Е: Типы транзакций

Табл. 21. Справочник типов транзакций

Тип	Описание
0	Обычный платеж, не связанный с другими
1	Платеж – перепроведение другого платежа
2	Платеж – отмена другого платежа
5	Банковское поступление
6	Движение по кассе
8	Транзакция, не вызывающая движения средств в системе
9	Списание комиссии по плательщикам НДС
10	Списание комиссии по неплательщикам НДС
11	Корректировка счёта КА
12	Комиссия за рекламу и дистрибуцию
14	Списание средств за использование коммерческого кредита
15	Начисление комиссии на счёт прочей комиссии
17	Отмена платежа/группы платежей по провайдеру без участия счёта агента
18	Комиссия за программное обеспечение
20	Транзакция инкассации терминала
22	Начисление НДС на комиссию головным агентам
23	Начисление НДС на комиссию дистрибьютора
24	Начисление комиссии дистрибьютора
25	Зачисление комиссии дистрибьютора на баланс
26	Зачисление комиссии дистрибьютора на расчетный счет
27	Списание комиссии по дистрибьютору в доход КИВИ
28	Списание комиссии за рекламу по агенту в доход КИВИ
29	Платеж - перепроведение за счет КИВИ
30	Фиксированная комиссия за ИТО
31	Зачисление комиссии КО на расчетный счет
32	Зачисление комиссии КО на баланс
33	Списание комиссии по КО в доход КИВИ
34	Зачисление фиксированной комиссии КО на расчетный счет
35	Зачисление фиксированной комиссии КО на баланс
36	Списание фиксированной комиссии по КО в доход КИВИ
37	Платёж для формирования проводки вручную, баланс меняется у агента-адресата
38	Платёж для формирования проводки вручную, баланс не меняется ни у кого
39	Поступления на ОГФ

Тип	Описание
40	Поступления на счет для расчетов с КО
41	Выплата задолженности по плате ПО на расчетный счет
42	Комиссия за программное обеспечение для агентов с ролью КО
43	Начисление депозита для головного агента по счету КАС
44	Вознаграждение за качественную дистрибьюцию
45	Вознаграждение за качественный сервис
48	Вознаграждение субагенту за качественную дистрибьюцию
49	Вознаграждение субагенту за качественный сервис

ПРИЛОЖЕНИЕ Ж: Авторизация по ЭЦП

В системе поддерживаются способы авторизации клиента по следующим алгоритмам:

Табл. 22. Алгоритмы авторизации по ЭЦП

Алгоритм	Описание
SHA1withRSA	Для авторизации используется ЭЦП передаваемых XML пакетов с использованием алгоритма хеширования SHA1. Алгоритм работы с использованием электронной подписи см. ниже .
SHA256withRSA	Для авторизации используется ЭЦП передаваемых XML пакетов с использованием алгоритма хеширования SHA256. Алгоритм работы с использованием электронной подписи см. ниже .

Алгоритм работы с использованием ЭЦП

Подготовительные действия:

1. Создание пары RSA ключей с помощью клиентского ПО (например, с помощью утилиты OpenSSL). Длина ключа может быть 1024, 2048 или 4096 бит. Ключи должны быть закодированы в BASE64.
2. Открытая часть ключа сохраняется на сервере запросом `setPublicKey` интерфейса `persons`.
Для доступа к серверу необходимо использовать персону с одноразовым паролем и авторизацию в теле запроса. В параметре запроса `store-type` необходимо указать значение 1 (файл).
3. Закрытая часть ключа сохраняется в локальном хранилище (например, в системном хранилище или на eToken).

Отправка запросов с ЭЦП:

1. Для каждого XML пакета вычисляется хэш от текста XML-запроса с использованием выбранного алгоритма SHA1/SHA256, подписывается сохраненным закрытым ключом и кодируется по схеме **Base64**. Таким образом формируется ЭЦП пакета.
2. В заголовке HTTP запроса передаются следующие параметры для выполнения авторизации и проверки целостности данных пакета:
 - "X-Digital-Sign" – ЭЦП пакета;
 - "X-Digital-Sign-Alg" – алгоритм вычисления ЭЦП (см. поле **Алгоритм** в [Табл. 22](#));
 - "X-Digital-Sign-Login" – обычный логин пользователя или закодированный по схеме **Base64**.

ПРИМЕЧАНИЕ



При авторизации по ЭЦП `ter auth` в XML-запросе не используется.

ПРИЛОЖЕНИЕ 3: Типы инкассаций

Табл. 23. Типы инкассаций

Тип	Описание
1	Инкассация
2	Инкассация подменена
3	Инкассация отредактирована
4	Инкассация удалена
10	Закрытие смены
20	Внесение денег в диспенсер
21	Внесение денег в дополнительный диспенсер

ПРИЛОЖЕНИЕ И: Расшифровка бита состояния сторожевого таймера

Номер бита	Расшифровка	Описание
1	sensorAlarm	Сработала сигнализация от сенсора
2	doorAlarm	Сработала сигнализация от датчика двери
4	doorOpened	Открыта дверь автомата
8	lockOpened	Открыт электрозамок автомата
16	no220Power	Отсутствует напряжение 220В
32	220PowerResumed	Напряжение 220В было восстановлено
64	scheduledPowerOff	Автомат выключается по расписанию
128	scheduledPowerOn	Автомат включился по расписанию
256	UpsLowBattarey	Батарея UPS разряжена
512	no12VoltsFromComputer	12 вольт от компьютера отсутствуют
1024	SecurutyAlarmMode	Сигнализация находится в режиме охраны

ПРИЛОЖЕНИЕ К: Счета учета

Табл. 24. Описание счетов

Идентификатор счета	Описание	Счет
1	Счёт агента	76.09
4	Картотека (штрафы/проценты)	76.К
5	Комиссия агента	60.К
10	Счет терминала	76.АТ
12	Сумка инкассации	99.9
13	Комиссия агента в пользу системы	62.А
18	Прочая комиссия	60.П
22	Счет купюроприемника	76.СК
23	Счет успешных платежей	76.УП
24	Счет неуспешных платежей	76.НУП
25	Верхняя комиссия	98.В
29	Счет купюроприемника №2	76.СК2
30	Карман абонента	76.КА
40	Счет КО	76.09 КО
41	Гарантийный фонд	76.09.ГФ

ПРИЛОЖЕНИЕ Л: Расшифровка строки флагов состояния АСО

Параметр `machineStatus` (см. раздел [6.7.5.2](#)) содержит 24-битную строку флагов состояния АСО. Каждый флаг характеризуется номером (в порядке слева направо) и может принимать следующие значения:

- 0 – отсутствие флага состояния.
- 1 – флаг состояния.
- Пробел (" ") – нет данных.

Все флаги состояния перечислены в [Табл. 25](#). Также см. [Пример 25](#) расшифровки флагов состояния.

Пример 25. Набор флагов состояния терминала

```
... machineStatus="000101100000010000000000 "
```

Атрибут содержит следующий набор флагов состояния:

- Флаг 4 – Автомат остановлен из-за ошибки в конфигурации интерфейса
- Флаг 6 – В чековом принтере скоро закончится бумага
- Флаг 7 – С автомата был снят купюроприемник
- Флаг 14 – Автомат работает через прокси-сервер
- Флаг 24 – Нет данных

Табл. 25. Флаги состояния терминала

Номер флага	Битовая маска	Состояние
1	1000 0000 0000 0000 0000 0000	Автомат остановлен из-за ошибок купюроприемника или принтера
2	0100 0000 0000 0000 0000 0000	Автомат остановлен из-за ошибки в конфигурации интерфейса
3	0010 0000 0000 0000 0000 0000	Автомат загружает с сервера обновление приложения
4	0001 0000 0000 0000 0000 0000	Автомат остановлен из-за того, что при старте не обнаружено оборудование (купюроприемник или принтер)
5	0000 1000 0000 0000 0000 0000	Работает сторожевой таймер
6	0000 0100 0000 0000 0000 0000	В чековом принтере скоро закончится бумага
7	0000 0010 0000 0000 0000 0000	С автомата был снят купюроприемник
8	0000 0001 0000 0000 0000 0000	Отсутствуют или неверно заполнены один или несколько реквизитов для терминала
9	0000 0000 1000 0000 0000 0000	Проблемы с жестким диском
10	0000 0000 0100 0000 0000 0000	Остановлен по сигналу сервера или из-за отсутствия денег на счету агента

Номер флага	Битовая маска	Состояние
11	0000 0000 0010 0000 0000 0000	Автомат оснащен вторым монитором
12	0000 0000 0001 0000 0000 0000	Автомат работает в локальной сети
13	0000 0000 0000 1000 0000 0000	Обнаружено стороннее ПО, которое может вызвать сбой модемного соединения
14	0000 0000 0000 0100 0000 0000	Автомат работает через прокси-сервер
16	0000 0000 0000 0001 0000 0000	Терминал обновляет конфигурацию
17	0000 0000 0000 0000 1000 0000	Терминал обновляет номерные емкости
18	0000 0000 0000 0000 0100 0000	Терминал обновляет список провайдеров
19	0000 0000 0000 0000 0010 0000	Терминал проверяет и обновляет рекламный плейлист
20	0000 0000 0000 0000 0001 0000	Терминал проверяет и обновляет файлы
22	0000 0000 0000 0000 0000 0100	Модифицировано приложение АСО
24	0000 0000 0000 0000 0000 0001	Монитор АСО выключен

ПРИЛОЖЕНИЕ М: Типы терминалов

Табл. 26. Типы терминалов

Идентификатор типа терминала	Полное название терминала	Дополнительный тип терминала	Идентификатор дополнительного типа терминала
3	Windows терминал "ОСМП Дилер"	Терминал QIWI	30
4	Автомат самообслуживания	АСО QIWI	10
6	XML протокол	Терминал XML	40
7	QIWI Кассир для 1С:Предприятия	Терминал QIWI	30
13	QIWI Защита	Терминал QIWI	30
19	АСО РИ Light	АСО Light	20
33	QIWI Кассир	Терминал QIWI	30
102	Адаптер OpenWay	Терминал XML	40
1540	QIWI Кассир Android	Терминал QIWI	30
1600	XML Касса	Терминал XML	40
1601	XML Банкомат	Терминал XML	40
1602	XML Терминал самообслуживания	Терминал XML	40
1603	XML Банкомат Cash-In	Терминал XML	40
1604	XML POS-терминал (оплата картой)	Терминал XML	40
1605	XML Мобильный банкинг	Терминал XML	40
1606	XML Интернет банкинг	Терминал XML	40
1607	XML POS-терминал (оплата наличными)	Терминал XML	40
1608	XML Автоплатежи	Терминал XML	40
1609	XML Интернет-портал	Терминал XML	40

ПРИЛОЖЕНИЕ Н: Типы объектов КЛАДР

ID типа	Название
0	Улица
1	Государство
2	Федеральный округ
3	Автономный округ
4	Автономная область
5	Город
6	Край
7	Область
8	Республика
9	Округ
10	Район
11	Улус
12	Территория
13	Поселок городского типа
14	Рабочий поселок
15	Курортный поселок
16	Дачный поселок
17	Сельсовет
18	Сельская администрация
19	Сельский округ
20	Волость
21	Почтовое отделение
22	Сельское поселение
23	Аал
24	Аул
25	Выселки(ок)
26	Деревня
27	Железнодорожная будка
28	Железнодорожная казарма
29	ж/д останов. (обгонный) пункт
30	Железнодорожный пост
31	Железнодорожный разъезд
32	Железнодорожная станция
33	Заимка

ID типа	Название
34	Казарма
35	Местечко
36	Микрорайон
37	Населенный пункт
38	Остров
39	Поселок
40	Планировочный район
41	Поселок и(при) станция(и)
42	Починок
43	Промышленная зона
44	Разъезд
45	Село
46	Слобода
47	Станция
48	Станица
49	Хутор
50	Городок
51	Железнодорожная платформа
52	Квартал
53	Арбан
54	Садовое неком-е товарищество
55	Леспромхоз
56	Погост
57	Кордон
58	Аллея
59	Бульвар
60	Въезд
61	Дорога
62	Животноводческая точка
63	Заезд
64	Километр
65	Кольцо
66	Линия
67	Набережная
68	Парк
69	Переулок

ID типа	Название
70	Переезд
71	Площадь
72	Площадка
73	Проезд
74	Проспект
75	Просек
76	Проселок
77	Проулок
78	Сад
79	Сквер
80	Строение
81	Тракт
82	Тупик
83	Участок
84	Шоссе
85	Платформа
86	Полустанок
87	Спуск
88	Канал
89	Гаражно-строительный кооперат
90	Проток
91	Коса
92	Вал
93	Ферма
94	Мост
95	Автономная республика
96	Пляж
97	Бухта
98	Кооператив
122	Зона
123	Жилой район
124	Автодорога
126	Берег
128	Абонентский ящик
129	Бугор
130	Жилая зона

ID типа	Название
132	Массив
133	Просека
135	Ряды
136	Сельское муниципальное образ-е
168	Дачное некоммерческое партнерство
169	Некоммерческое хозяйство
170	Фермерское хозяйство
171	местность
172	балка
173	горка
174	маяк
175	тоннель

ПРИЛОЖЕНИЕ О: Коды операторов сотовой связи

Внутренний код провайдера	Название провайдера
0	Провайдер не определен
1	Билайн
2	МТС
3	Мегафон
5	NCC
6	SIBCHALLENGE
7	SMARTS
10	БайкалВестКом
11	SMARTS
14	TELE2
16	Uralsvyazinform
18	UMC
20	KyivStar
22	Life(DCC-Astelit)
23	Velcom
24	MTS-BY
25	BeST (Life)

ПРИЛОЖЕНИЕ П: Типы денежных средств (moneyType)

ПРИМЕЧАНИЕ

При отсутствии атрибута источником платежа будут считаться наличные

moneyType	Описание	Категория
0	Наличный платеж без комиссии	Иное
10	Наличный платеж с комиссией	Иное
1	Карта своего банка без комиссии	Карты
11	Карта своего банка с комиссией	Карты
2	Карта другого банка без комиссии	Карты
21	Карта другого банка с комиссией	Карты
3	Интернет-банк без комиссии	Иное
31	Интернет-банк с комиссией	Иное
4	Мобильный банк без комиссии	Иное
41	Мобильный банк с комиссией	Иное
5	Электронные деньги без комиссии	Иное
51	Электронные деньги с комиссией	Иное
7	Платеж со счета без комиссии	Счет
71	Платеж со счета с комиссией	Счет

СПИСОК РИСУНКОВ

Рис. 1. Структура интерфейса	14
Рис. 2. Схема проведения оффлайн платежа	21
Рис. 3. Схема проведения онлайн платежа	22
Рис. 4. Структура XML-интерфейса групп	30
Рис. 5. Структура XML-интерфейса провайдера.....	32

СПИСОК ТАБЛИЦ

Табл. 1. Изменение статусов платежей	23
Табл. 2. Общий перечень экстра-полей, используемых для передачи сведений о плательщике	24
Табл. 3. Атрибуты тега provider.....	130
Табл. 4. Атрибуты тега logo	132
Табл. 5. Атрибуты тега param	133
Табл. 6. Атрибуты тега page.....	133
Табл. 7. Типы управляющих элементов.....	134
Табл. 8. Атрибуты виртуальной клавиатуры "keyboard"	134
Табл. 9. Атрибуты поля ввода "text_input"	135
Табл. 10. Атрибуты ассоциативного поля ввода "disp_input"	136
Табл. 11. Атрибуты кнопки "button"	137
Табл. 12. Атрибуты ассоциативной кнопки "disp_button".....	138
Табл. 13. Атрибуты поля со списком "combobox"	139
Табл. 14. Атрибуты ассоциативного поля со списком "disp_combobox"	141
Табл. 15. Атрибуты ассоциативного текстового поля "disp_label"	142
Табл. 16. Спецсимволы	191
Табл. 17. Операторы	191
Табл. 18. Коды ошибок процессинга	194
Табл. 19. Статусы платежей в процессинге.....	201
Табл. 20. Статусы обработки асинхронных запросов.....	201
Табл. 21. Справочник типов транзакций	202
Табл. 22. Алгоритмы авторизации по ЭЦП	204
Табл. 23. Типы инкассаций	205
Табл. 24. Описание счетов	207
Табл. 25. Флаги состояния терминала.....	208
Табл. 26. Типы терминалов.....	210

СПИСОК ПРИМЕРОВ

Пример 1. Асинхронный запрос.....	19
Пример 2. Ответ на асинхронный запрос	19
Пример 3. Получение статуса обработки асинхронного запроса	20
Пример 4. Ответ на асинхронный запрос с данными	20
Пример 5. Асинхронный запрос с интервалом номеров	20
Пример 6. Интерфейс терминала	30
Пример 7. Интерфейс оплаты с одной страницей.....	33
Пример 8. Интерфейс оплаты с онлайн-проверкой	34

Пример 9. Интерфейс оплаты с экстра-полями	35
Пример 10. Интерфейс оплаты с онлайн-проверкой и экстра-полями	36
Пример 11. Интерфейс оплаты с экстра-полями на нескольких страницах и ветвлением.....	37
Пример 12. Расчет комиссии	42
Пример 13. Отправленное на терминал текстовое сообщение.	54
Пример 14. Сообщение для получения журнала работы терминала за указанный период.	54
Пример 15. Сообщение для получения списка запущенных процессов.	54
Пример 16. Служебное сообщение, отправленное на терминал	54
Пример 17. Удаление ранее принятого текстового сообщения.....	55
Пример 18. Получение новостного сообщения в КИВИ Кассир	55
Пример 19. Отменяемый платеж	168
Пример 20. Отменяющий платеж	168
Пример 21. Примеры задания маски полей ввода	190
Пример 22. Использование спецсимволов и операторов	192
Пример 23. Использование зарезервированных символов	192
Пример 24. Примеры регулярных выражений для проверки	193
Пример 25. Набор флагов состояния терминала	208