НАЦИОНАЛЬНЫЙ ЦЕНТР ПРАВОВОЙ ИНФОРМАЦИИ

**РЕСПУБЛИКИ БЕЛАРУСЬ**

УТВЕРЖДЕН

**ЗЩ. 1530.37477375-0083-ЛУ**

**ИНФОРМАЦИОННО-ПОИСКОВАЯ СИСТЕМА**

**”ЭТАЛОН-ONLINE“**

**Рабочий проект**

**Описание способов оплаты**

**ЗЩ. 1530.37477375-0083**

Листов 52

###### **2013**

СОДЕРЖАНИЕ

[1 Общие сведения 3](#_Toc355781064)

[2 Оплата посредством электронных платежей WebMoney/EasyPay 3](#_Toc355781065)

[3 Модуль приема SMS-платежей 6](#_Toc355781066)

[4 Модуль для взаимодействия с системой ЕРИП 6](#_Toc355781067)

[5 Установка, вызов и загрузка 9](#_Toc355781068)

[ПРИЛОЖЕНИЕ A Взаимодействие Поставщика и EasyPay по протоколу SOAP 18](#_Toc355781069)

[ПРИЛОЖЕНИЕ Б Описание XML-интерфейсов системы WebMoney 30](#_Toc355781070)

[ПРИЛОЖЕНИЕ В Описание структуры XML сообщений для доступа в режиме on-line из центрального узла АИС ЕРИП к биллинговым системам производителей услуг 41](#_Toc355781071)

# 1 Общие сведения

Информационно-поисковая система ”ЭТАЛОН-ONLINE“ предоставляет широкий выбор форм оплаты за использование:

* оплата по безналичному расчету с заключением договора в печатной форме;
* оплата с использованием систем электронных платежей WebMoney и EasyPay без необходимости заключать договор в печатной форме;
* наиболее упрощенная форма подключения на одни сутки без регистрации с помощью SMS-сообщения, отправленного на короткий номер;
* оплата с помощью системы ”Единое расчетное информационное пространство“ (далее – ЕРИП) также без необходимости заключать договор в печатной форме.

# 2 Оплата посредством электронных платежей WebMoney/EasyPay

Модуль взаимодействия с платежными системами WebMoney, EasyPay состоит из двух частей:

1. Подмодуль, подключенный непосредственно к сайту ИПС ”ЭТАЛОН-ONLINE“ и используемый для выписки счета пользователю (далее – подмодуль для выписки счета).
2. Подмодуль, выполняющий обращение к платежным системам по заданному расписанию для получения сведений о поступлении платежей от пользователей (далее – подмодуль для проверки счетов).

Подмодуль выписки счета представляет собой .NET приложение на языке C#, реализованное в виде динамически подключаемой библиотеки AccountLib. На рисунке 2.1 представлено окно Solution Explorer, в котором отображаются все файлы проекта подмодуля.

Данный подмодуль осуществляет:

1. Обращение к системе EasyPay по протоколу SOAP (Simple Object Access Protocol) для выписки счета пользователю. Спецификация протокола обмена данными с системой EasyPay приведена в приложении А (доступ к актуальной версии спецификации можно получить по адресу – <https://ssl.easypay.by/soap>).
2. Обращение к системе WebMoney по протоколу HTTPS методом POST с использованием интерфейса X1. Обмен информацией с системой WebMoney осуществляется в формате XML. Спецификация интерфейса X1, используемого для выписки счета, приведена в приложении Б (доступ к актуальной версии спецификации можно получить по адресу – http://www.webmoney.ru/rus/developers/interfaces/xml).

Подмодуль выписки счета вызывается при регистрации пользователя в ИПС ”ЭТАЛОН-ONLINE“ и выборе в качестве формы оплаты ”Электронные деньги (EasyPay/WebMoney)“ (рисунок 2.2) или при выписке счета через личный кабинет пользователя (рисунок 2.3).

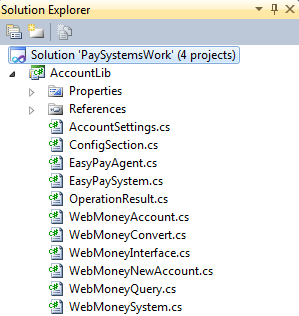


Рисунок 2.1 – Вид окна Solution Explorer

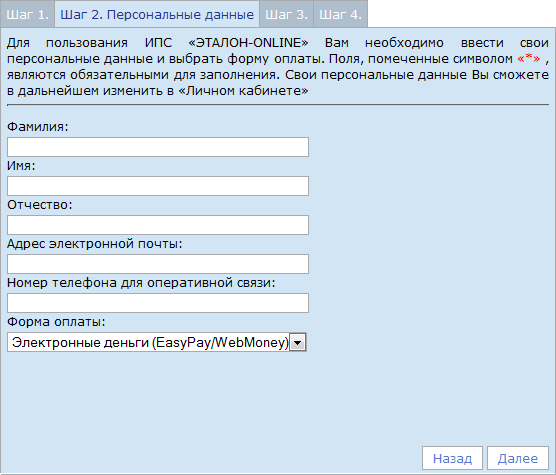


Рисунок 2.2 – Выбор формы оплаты при регистрации пользователя

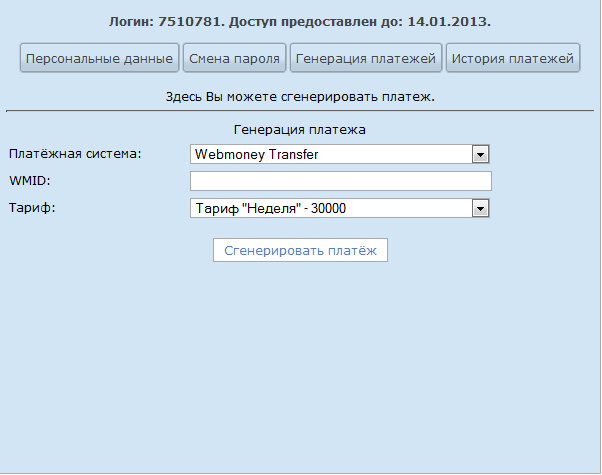


Рисунок 2.3 – Выписка счета пользователем через личный кабинет

Подмодуль для проверки счетов представляет собой .NET приложение, разработанное на языке C#, и состоит из 3 проектов. На рисунке 2.4 представлен вид окна Solution Explorer, в котором отображены файлы проектов подмодуля.

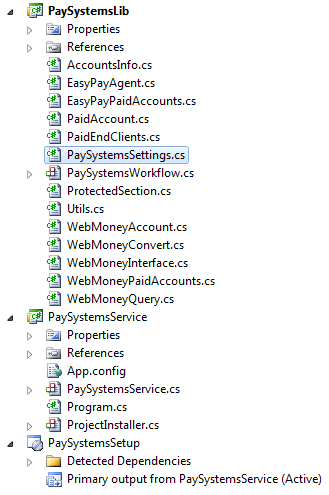


Рисунок 2.4 – Вид окна Solution Explorer

Подмодуль представляет собой windows-службу, благодаря чему он функционирует на сервере в фоновом режиме и самостоятельно перезапускается при перезагрузке сервера. Подмодуль формирует лог-файлы и может отсылать их на почту по настроенному расписанию, что значительно упрощает мониторинг его работы.

Основными функциями подмодуля являются:

1. Обращение к системе EasyPay по заданному расписанию для проверки наличия новых оплаченных и отклоненных счетов. Спецификация протокола обмена данными с системой EasyPay приведена в приложении А.
2. Обращение к системе WebMoney по заданному расписанию для проверки наличия новых оплаченных и отклоненных счетов посредством интерфейса X4. Спецификация интерфейса X4, используемого для проверки счета, приведена в приложении Б.
3. Рассылку электронных писем клиентам за сутки до окончания оплаченного периода и после закрытия доступа.

# 3 Модуль приема SMS-платежей

Модуль приема SMS-платежей реализован в виде ASP.NET http-handler на языке C#. Обращение к данному модулю происходит каждый раз при отправке пользователем SMS на указанный на сайте короткий номер с текстом сообщения ncpi1. Скрипт осуществляет сохранение данных о номере телефона в базу данных и отправляет пользователю логин и пароль для входа в систему. При первом входе в систему пользователю предоставляется договор, доступ к ИПС ”ЭТАЛОН-ONLINE“ открывается сразу после принятия условий договора.

# 4 Модуль для взаимодействия с системой ЕРИП

Модуль для взаимодействия с системой ЕРИП представляет собой ASP.NET http-handler на языке C#. Данный модуль обеспечивает обработку XML сообщений, поступающих в режиме реального времени от центрального узла автоматизированной информационной системы ЕРИП (далее – АИС ЕРИП). Подключение в режиме реального времени (режим on-line) подразумевает наличие постоянного канала связи между АИС ЕРИП и нашим сервером, так как в данном случае обмен информацией должен осуществляться в любое время, когда пользователь ИПС ”ЭТАЛОН-ONLINE“ выполняет оплату. Далее приведен список обрабатываемых сообщений от центрального узла АИС ЕРИП:

|  |  |
| --- | --- |
| **Наименование сообщения** | **Описание** |
| ServiceInfo | Запрос задолженности плательщика |
| TransactionStart | Запрос на выполнение оплаты |
| TransactionResult | Извещение о результате оплаты |

Оплата доступа через систему ЕРИП включает в себя выполнение нескольких шагов:

1. Пользователю предлагается выбрать из дерева услуг подходящий вариант для оплаты (рисунок 2.5).

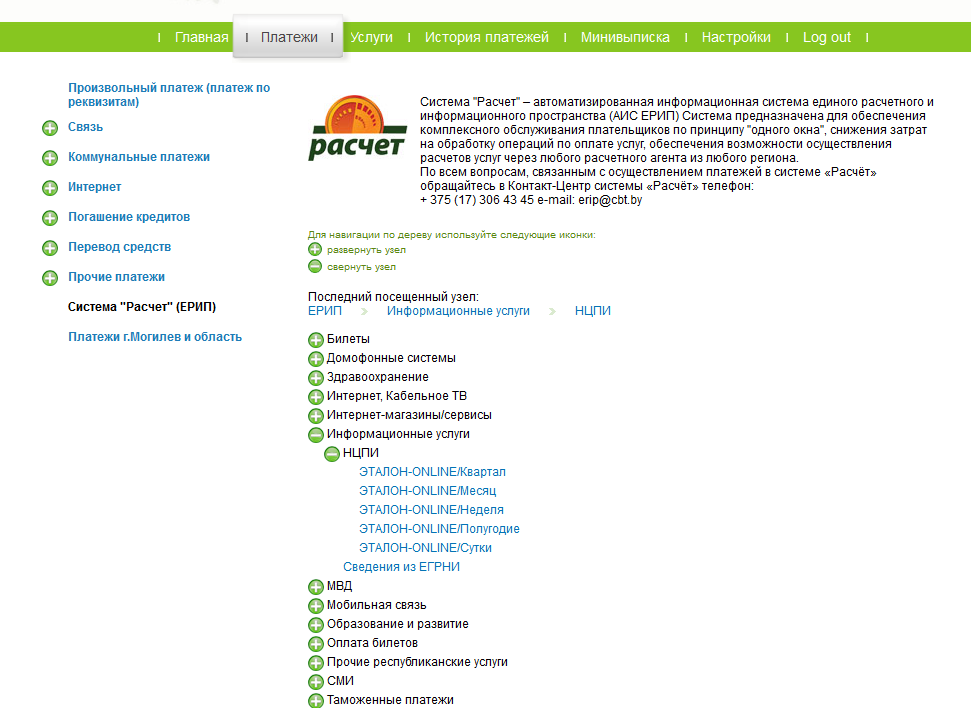


Рисунок 2.5 – Дерево услуг ЕРИП

1. Пользователь вводит номер лицевого счета (логин, полученный при регистрации на сайте [www.etalonline.by](http://www.etalonline.by)) (рисунок 2.6).

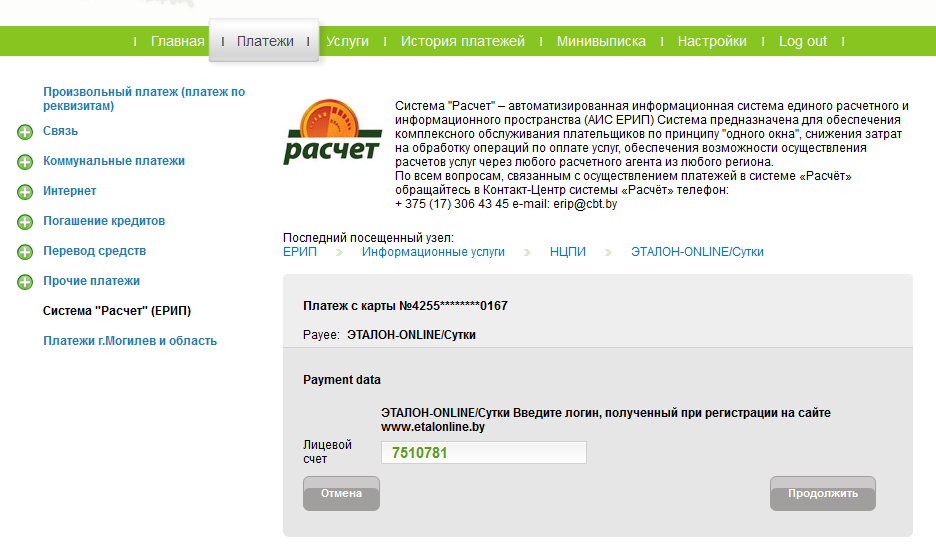


Рисунок 2.6 – Ввод лицевого счета (логина) пользователя

1. При нажатии на кнопку ”Продолжить“ модулю для взаимодействия с системой ЕРИП отправляется сообщение **ServiceInfo** (запрос задолженности плательщика). Модуль проверяет наличие в базе данных пользователя с введенным номером лицевого счета (логином) и возвращает фамилию клиента, а также сумму к оплате в зависимости от выбранного пользователем тарифа (рисунок 2.7).

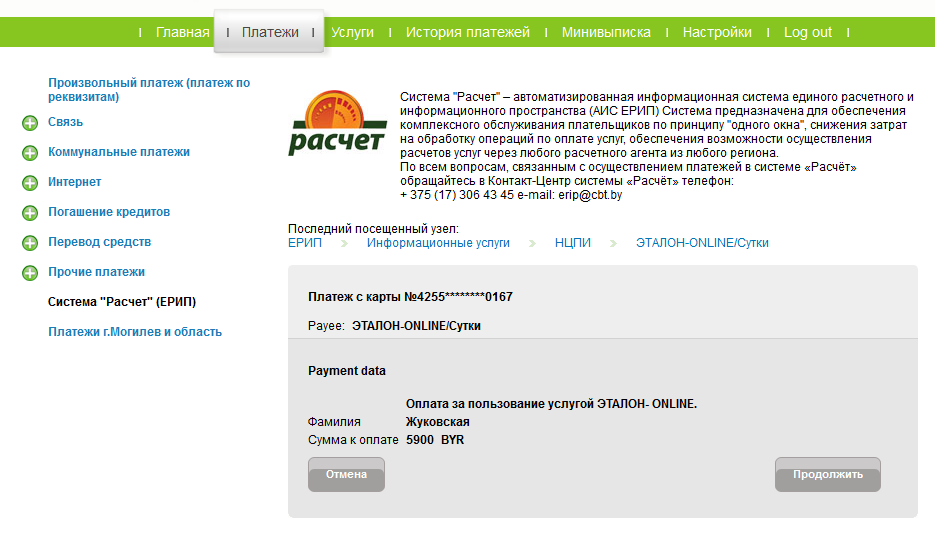


Рисунок 2.7 – Отображение информации о пользователе

1. При нажатии кнопки ”Продолжить“ модулю отправляется сообщение **TransactionStart**, информирующее о начале операции оплаты. На данном этапе в базу данных заносится информация об идентификаторе клиента, сумме оплаты, дате начала операции. Затем модуль отправляет АИС ЕРИП ответ об успешной обработке данного сообщения, содержащий номер транзакции в нашей системе и информационное сообщение для пользователя.

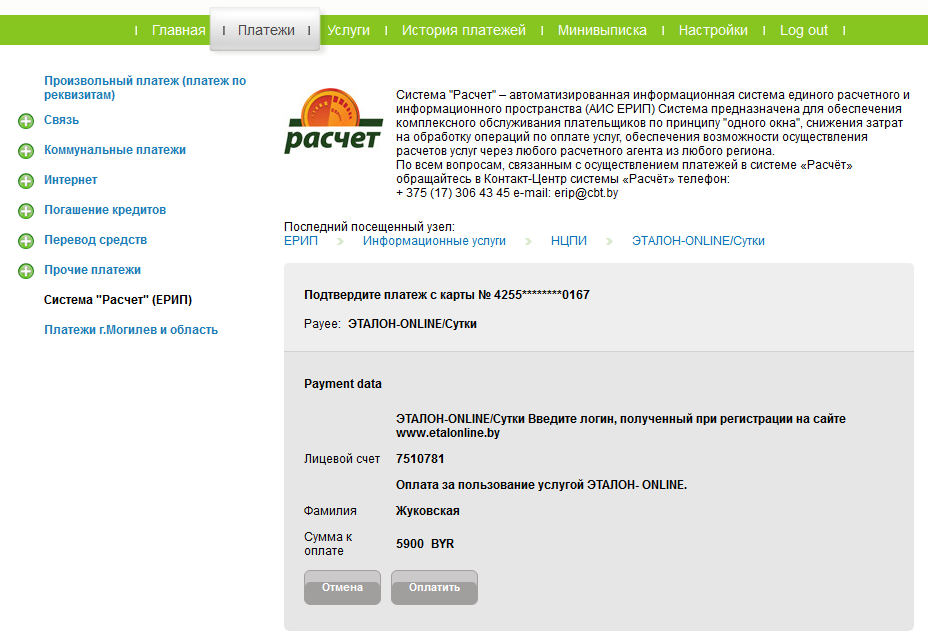


Рисунок 2.8 – Оплата счета

1. При нажатии кнопки ”Оплатить“ модуль получает сообщение **TransactionResult** и выполняет действия для подтверждения оплаты. Затем модуль отправляет АИС ЕРИП ответ об успешной обработке данного сообщения.
2. Подробное описание формата сообщений приведено в приложении 3.

# 5 Установка, вызов и загрузка

В комплект исходных файлов подмодуля для выписки счета в системах EasyPay и WebMoney входят:

|  |  |
| --- | --- |
| **Имя файла** | **Описание** |
| ConfigSection.cs | Содержит класс для работы с пользовательской секцией файла конфигурации. |
| EasyPayAgent.cs | Содержит класс, сгенерированный из WSDL описания web-сервиса EasyPay. |
| EasyPaySystem.cs | Содержит класс, используемый для выписки счета в системе EasyPay. |
| OperationResult.cs | Содержит класс для хранения результата обращения к платежной системе EasyPay. |
| WebMoneyAccount.cs | Содержит класс, описывающий поля запроса на выписку счета в системе WebMoney. |

|  |  |
| --- | --- |
| **Имя файла** | **Описание** |
| WebMoneyConvert.cs | Содержит методы преобразований типов, необходимых для отправки и получения сообщений в систему WebMoney. |
| WebMoneyInterface.cs | Содержит методы для обращения к системе WebMoney. |
| WebMoneyNewAccount.cs | Содержит описание XML-сообщения для выписки счета. |
| WebMoneyQuery.cs | Содержит методы отправки XML-сообщений. |
| WebMoneySystem.cs | Содержит интерфейсный класс для взаимодействия с WebMoney. |

В комплект исходных файлов подмодуля для проверки счетов в системах EasyPay и WebMoney входят:

Проект **PaySystemsLib**

|  |  |
| --- | --- |
| **Имя файла** | **Описание** |
| EasyPayAgent.cs | Содержит класс, сгенерированный из WSDL описания web-сервиса EasyPay. |
| EasyPayPaidAccounts.cs | Содержит класс, используемый для проверки наличия оплаченных счетов в EasyPay. |
| PaidAccount.cs | Содержит класс для работы с уже оплаченными счетами. |
| PaidEndClients.cs | Содержит класс для работы с пользователями, у которых оканчивается период доступа к услуге. |
| PaySystemsSettings.cs | Содержит класс для получения данных из файла конфигурации. |
| PaySystemsWorkflow.cs | Содержит класс, описывающий рабочий процесс. |
| ProtectedSection.cs | Содержит класс для работы с пользовательской секцией файла конфигурации. |
| Utils.cs | Содержит вспомогательные методы. |
| WebMoneyAccount.cs | Содержит класс, описывающий счет в системе WebMoney. |
| WebMoneyConvert.cs | Содержит методы преобразований типов, необходимых для отправки и получения сообщений в систему WebMoney. |

|  |  |
| --- | --- |
| **Имя файла** | **Описание** |
| WebMoneyInterface.cs | Содержит методы для обращения к системе WebMoney. |
| WebMoneyPaidAccounts.cs | Содержит описание XML-сообщения для работы со счетами. |
| WebMoneyQuery.cs | Содержит методы отправки XML-сообщений. |

Проект **PaySystemsService** представляет собой реализацию подмодуля в виде windows-сервиса.

Проект **PaySystemsSetup** представляет собой проект инсталляции подмодуля.

В комплект исходных файлов модуля приема SMS-платежей входят:

|  |  |
| --- | --- |
| **Имя файла** | **Описание** |
| sms\_payment.ashx | Http-handler для приема SMS-платежей. |
| Web.config | Файл конфигурации с указанием строки подключения к базе данных. |

В комплект исходных файлов модуля взаимодействия с ЕРИП входят:

|  |  |
| --- | --- |
| **Имя файла** | **Описание** |
| erip\_payment.ashx | Http-handler для взаимодействия с ЕРИП. |
| Web.config | Файл конфигурации с указанием строки подключения к базе данных. |

Для установки на сервер модуля приема SMS-платежей и модуля для взаимодействия с ЕРИП достаточно использовать команду Publish в Visual Studio (рисунок 3.1).

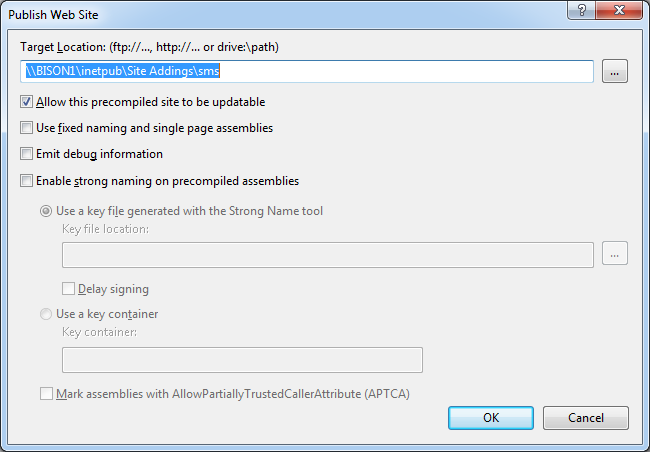


Рисунок 3.1 – Публикация модуля приема SMS-платежей

Также процесс установки можно осуществить простым копированием файлов исходного кода в папку на сервере. После копирования или публикации необходимо настроить web-приложение через ”Панель управления“ → ”Администрирование“ → ”Диспетчер служб IIS“.

Подмодуль для выписки счета в системах EasyPay и WebMoney поставляется в виде динамически подключаемой библиотеки AccountLib.dll и подключается непосредственно к сайту ИПС ”ЭТАЛОН-ONLINE“.

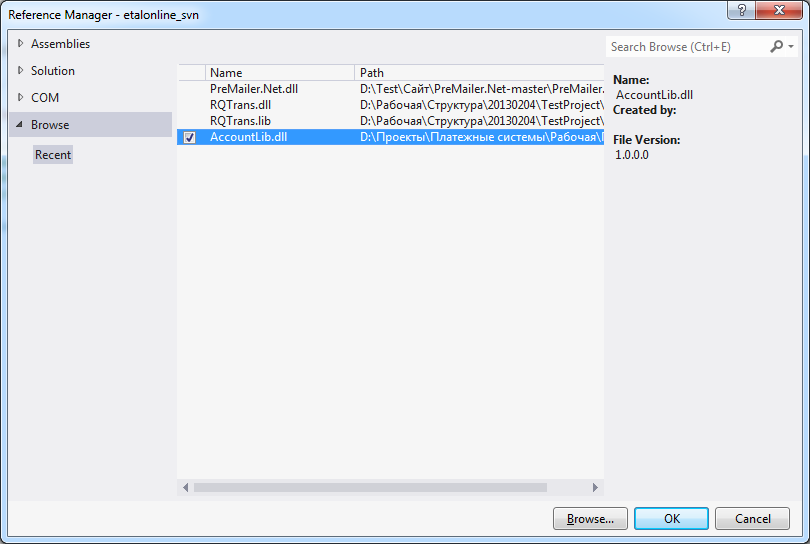


Рисунок 3.2 – Подключение подмодуля для выписки счета через диалоговое окно ” Reference Manager“

Подмодуль для проверки счетов в системах EasyPay и WebMoney поставляется в виде пакета установки, содержащего файлы PaySystemsSetup.msi и setup.exe.

Рассмотрим процесс инсталляции подмодуля для проверки счетов:

1. Запускаем setup.exe. После этого должно появиться следующее окно:

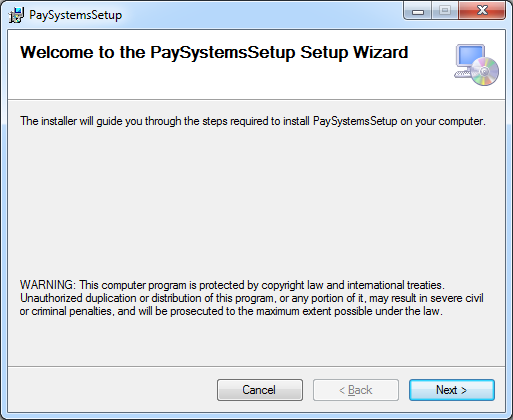


Рисунок 3.3 – Начало установки подмодуля

1. Нажимаем ”Next“. В появившемся диалоговом окне выбираем папку, куда будет выполнена установка программы.

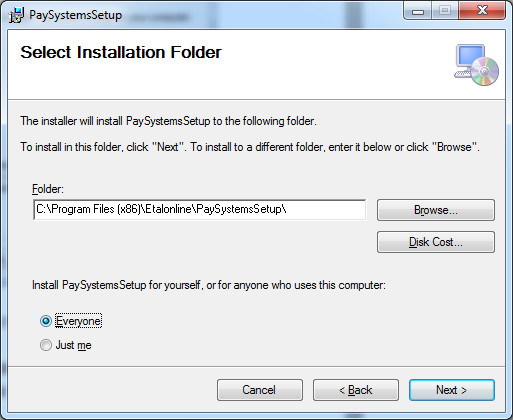


Рисунок 3.4 – Выбор папки на диске, в которую будет выполнена установка программы

1. Нажимаем ”Next“. В результате должно появиться следующее диалоговое окно:

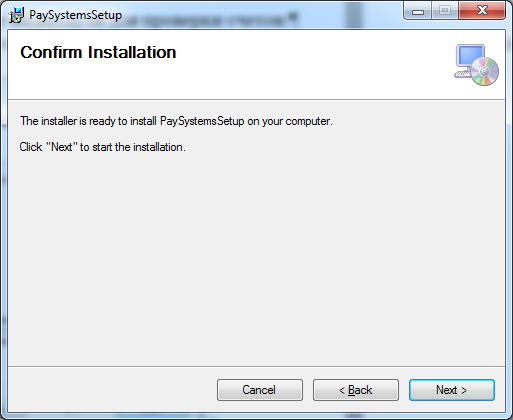


Рисунок 3.5 – Переход к непосредственной установке подмодуля

1. Нажимаем ” Next“. В результате будет выполнена установка.

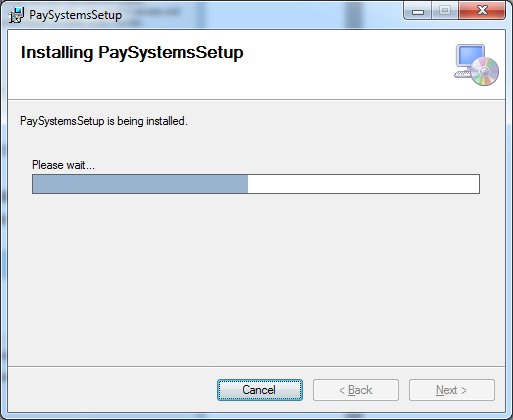
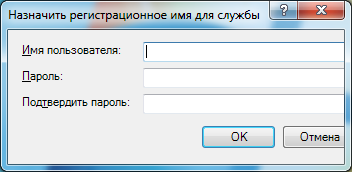


Рисунок 3.6 – Непосредственная установка службы-инициализатора

В процессе установки на экране должно появиться дополнительное диалоговое окно. В данном окне необходимо указать имя пользователя и пароль учетной записи, под которой служба будет функционировать, при этом указанный пользователь должен обладать правами на запись в файловую систему и на выполнение запросов через Интернет.



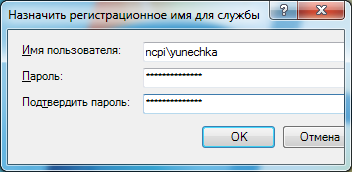


Рисунок 3.7 – Диалоговое окно для указания пользователя, под которым будет выполняться windows-служба

1. После указания пользователя и окончания процесса установки должно появиться следующее окно:

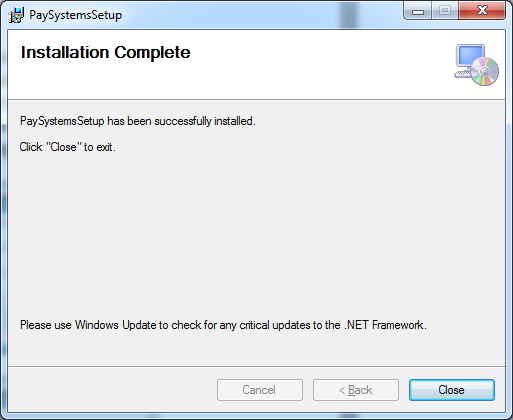


Рисунок 3.8 – Окончание установки подмодуля

1. Нажимаем кнопку ”Close“.

Установленную службу можно увидеть с помощью утилиты ” Панель управления“ → ”Администрирование“ → ”Службы“:

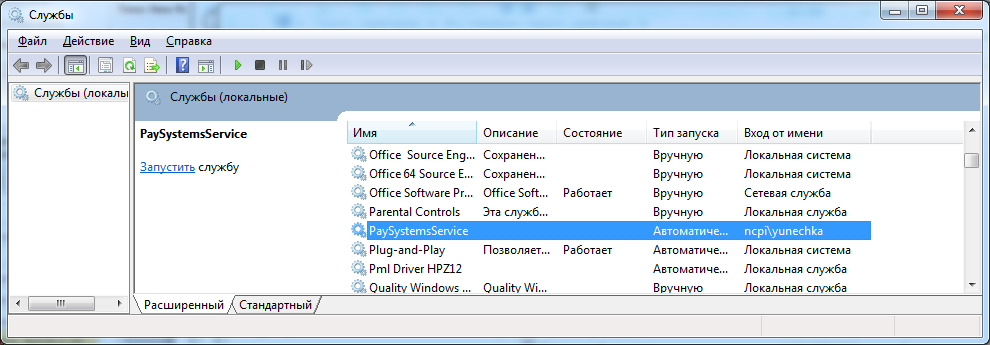


Рисунок 3.9 – Окно ”Службы“

Здесь же можно настроить тип запуска службы и действия, которые необходимо выполнять при сбое.

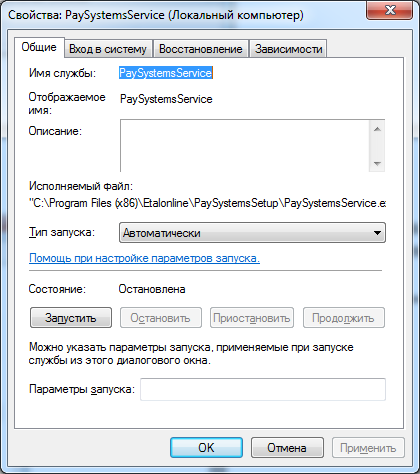


Рисунок 3.10 – Вкладка ”Общие“ окна свойств установленной службы

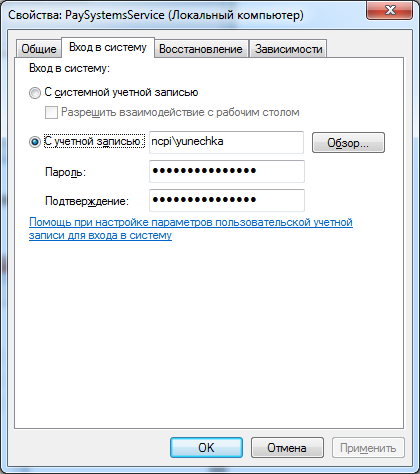


Рисунок 3.11. – Вкладка ”Восстановление“ окна свойств установленной службы

Удалить сервис можно в ”Панель управления“ → ”Программы и компоненты“.

# ПРИЛОЖЕНИЕ A

**Взаимодействие Поставщика и EasyPay по протоколу SOAP**

[Общие сведения](https://ssl.easypay.by/soap/#inout3)

[Описание входных и выходных параметров функций](https://ssl.easypay.by/soap/#inout)

[Выписка счета в кошелек Покупателя](https://ssl.easypay.by/soap/#inout1)

[Оплата заказа](https://ssl.easypay.by/soap/#oplata)

[Проверка оплаты счета Покупателем](https://ssl.easypay.by/soap/#proverka)

[Отмена платежа](https://ssl.easypay.by/soap/#otmena)

[Отправка сообщения Покупателю](https://ssl.easypay.by/soap/#soobschenie)

[Перевод денег в кошелек Покупателя от Поставщика (для Поставщиков, которые выполняют роль Агента по распространению электронных денег)](https://ssl.easypay.by/soap/#perevod)

[Проверка наличия счета в системе](https://ssl.easypay.by/soap/#vsisteme)

[Уменьшение суммы оплаченного счета](https://ssl.easypay.by/soap/#izmenenie)

[Описание полученных кодов](https://ssl.easypay.by/soap/#kod)

[Схема взаимодействия Поставщика с Процессинговым центром](https://ssl.easypay.by/soap/#reglament)

[Пример xml-пакетов](https://ssl.easypay.by/soap/#xml)

[Пример клиента на PHP5 с использованием библиотеки SOAP](https://ssl.easypay.by/soap/#php)

**Общие сведения**

**SOAP** (аббревиатура обозначает - Simple Object Access Protocol — простой протокол доступа к объектам) - используется для вызова функций и обмена произвольными сообщениями в формате XML.

Адрес к SOAP-серверу EasyPay и пароли для работы можно получить отправив заявку на e-mail **admin@easypay.by** или связаться по телефону **(017) 21-101-21**.

Для отладки и **тестирования** взаимодействия можно воспользоваться [тестовыми интерфейсами](https://ssl.easypay.by/sandbox/).

##### Операции, используемые для работы с EasyPay:

1. Выставление счета Покупателю

2. Получение списка оплаченных Покупателями счетов

3. Вспомогательные операции (отправка сообщения Покупателю, отмена платежа и др.)

4. Перечисление средств с аккаунта Поставщика в кошелек Покупателя

После заключения договоров Поставщик для интеграции интернет-ресурса в систему EasyPay проводит предварительную настройку всех свойств магазина.

Процессинговый центр предоставляет следующие данные для идентификации:

|  |  |  |
| --- | --- | --- |
| **Название полей** | **Формат полей** | **Описание полей** |
| mer\_no | 6 алфавитно-цифровых символов вида okXXXX, где X – число от 0 до 9 | номер Поставщика в EasyPay. Выдается Администратором EasyPay при регистрации |
| pass | от 8 до 25 алфавитно-цифровых символов | пароль Поставщика в EasyPay. Выдается Администратором EasyPay при регистрации |

**Описание входных и выходных параметров функций**

|  |  |  |
| --- | --- | --- |
| **Название полей** | **Формат полей** | **Описание полей** |
| mer\_no | 6 алфавитно-цифровых символов вида okXXXX, где X – число от 0 до 9 | номер Поставщика в EasyPay. Выдается Администратором EasyPay при регистрации |
| pass | от 8 до 25 алфавитно-цифровых символов | пароль Поставщика в EasyPay. Выдается Администратором EasyPay при регистрации |
| order (order\_mer\_code) | от 1 до 20 алфавитно-цифровых символов.  Допускаются буквы, цифры, а также символы .-\_ | уникальный номер электронного счета. Формируется Поставщиком. Номер должен быть уникальным в течение всего периода работы с EasyPay |
| sum | целое число больше нуля | сумма в белорусских рублях |
| exp | число от 1 до 30, если период задан в днях или от 600 до 3600\*24, если период задан в секундах | время действительности счета в днях |
| card | 8 цифр | идентификатор (номер) электронного кошелька EasyPay |
| comment | произвольное число (но не более 50) алфавитно-цифровых символов кроме "<" и ">". Допускаются буквы, цифры, а также символы .,-\_()+=;:?!@#№ | краткий комментарий счета. Формируется Поставщиком |
| **Название полей** | **Формат полей** | **Описание полей** |
| cancel\_comment | произвольное число (но не более 150) алфавитно-цифровых символов кроме "<" и ">".  Допускаются буквы, цифры, а также символы .,-\_()+=;:?@#№$&\*[]"'`/|\ | комментарий к отмененному счету. Формируется Поставщиком |
| info | произвольное число, (но не более 2000) алфавитно-цифровых символов кроме "<" и ">", возможны переводы строк.  Допускаются буквы, цифры, а также символы .,-\_()+=;:?!@#№$&\*[]"'`/|\ | подробный комментарий счета. Формируется Поставщиком |
| subject | произвольное число (но не более 250) алфавитно-цифровых символов кроме "<" и ">".  Допускаются буквы, цифры, а также символы .,-\_()+=;:?!@#№ | заголовок сообщения Покупателю. Формируется Поставщиком |
| text | произвольное число (но не более 10000) алфавитно-цифровых символов кроме "<" и ">", возможны переводы строк.  Допускаются буквы, цифры, а также символы .,-\_()+=;:?!@#№$&\*[]"'`/|\ | текст сообщения Покупателю. Формируется Поставщиком |
| code | целое число | код ответа от сервера EasyPay |
| message | произвольное число алфавитно-цифровых символов кроме "<" и ">" | комментарий на code ошибки от сервера на русском языке. Формируется сервером EasyPay, поясняет код ошибки |
| xml | От 0 до 64 килобайта текстовой информации в произвольной форме | дополнительный XML или другие текстовые данные |
| **Название полей** | **Формат полей** | **Описание полей** |
| pay\_id | от 1 до 32 алфавитно-цифровых символов кроме "<" и ">". Допускаются буквы, цифры, а также символы .-\_ | уникальный идентификатор платежа. Формируется Поставщиком |
| invoices |  | структура, в которой содержатся поля, описывающие счета: **order\_mer\_code** - номер счета, **purch\_date** - дата и время оплаты счета |
| invoices\_ext |  | **Invoices\_ext** – структура описания счетов с дополнительными полями: **order** – номер счета, **purch\_date** – дата и время оплаты счета, **sum** – сумма в белорусских рублях, **exp** – дата окончания времени жизни счета, **card** – идентификатор (номер) электронного кошелька EasyPay, **comment** – краткий комментарий счета, **info** – полный комментарий счета, **xml** – дополнительное описание счета в произвольной форме |

Во всех функциях, в случае успешного выполнения операции, функция возвращает code = 200 или code = 211, в противном случае код отличается и сопровождается сообщением об ошибке.

Рабочая кодировка в системе Windows-1251. Все принимаемые/передаваемые данные должны быть в этой кодировке, комментарии на ошибки система отправляет тоже в этой кодировке. Внешний xml-обмен происходит в кодировке utf-8.

**Выписка счета Покупателю**

(code, message) = EP\_CreateInvoice(mer\_no, pass, order, sum, exp, card, comment, info, xml)

**Оплата заказа**

Производится на сайте EasyPay. Здесь Покупатель совершает операцию оплаты на основании счета.

**Проверка оплаты счета Покупателем**

Для проверки каждого счета на предмет оплаты/неоплаты используйте одну из функций проверки или получения списка оплаченных счетов. Интервал между проверками оплаченных счетов рекомендуется установить 5 минут.

При получении списка оплаченных счетов (функции **EP\_GetPaidInvoices**, **EP\_GetPaidInvoicesExt**) выдаются **не более 100** счетов, оплаченных после даты оплаты счета order, указанного в качестве параметра. Если такого счета нет в системе, то выдаются **не более 100 первых** оплаченных счетов.

Таким образом, для получения списка оплаченных счетов необходимо при первом вызове функции передавать в качестве параметра order='' или несуществующий номер счета, а при последующих вызовах последний номер счета order, полученный на предыдущем шаге.

Получение списка оплаченных счетов

(code, message, invoices) = EP\_GetPaidInvoices(mer\_no, pass, order)

Получение списка оплаченных счетов с подробной информацией

(code, message, invoices\_ext) = EP\_GetPaidInvoicesExt(mer\_no, pass, order)

Проверка оплаты определенного (одного) счета

Рекомендуется использовать эту функцию как дополнительную к списку: для ускорения получения оплаты счета по запросу Покупателя на сайте Поставщика.

(code, message) = EP\_IsInvoicePaid(mer\_no, pass, order)

Code = 200 – счет оплачен, code = 211 – счет не оплачен, code = 503 – такой счет не существует в системе (Покупатель может самостоятельно удалить неоплаченный счет из системы), code = 506 – счет просрочен, code = 517 – счет отменен.

**Отмена платежа**

(code, message) = EP\_CancelInvoice (mer\_no, pass, order, cancel\_comment)

Предназначен для отмены операции по инициативе Поставщика в случае невозможности ее завершения.

**Отправка сообщения Покупателю**

(code, message) = EP\_SendMessage(mer\_no, pass, order, card, subject, text)

Отправка сообщения Покупателю с указанием номера оплаченного счета - укажите order. Если вы хотите отправить сообщение, не привязанное к определенному счету, то установите order пустым (order='').

**Перевод денег в кошелек Покупателя от Поставщика**

Данная функция используется, если Поставщик, оказывающий услуги, выполняет также роль Агента по распространению электронных денег, т.е. осуществляет перевод денег в кошелек Покупателя (настраивается Администратором EasyPay). Например, выплата выигрышей (букмекерские конторы, лотереи).

**Проверка наличия счета в системе**

(code, message) = EP\_IsInvoicePaid(mer\_no, pass, order)

Данная функция используется как для проверки оплаты счета, так и для проверки наличия счета в системе.

**Уменьшение суммы оплаченного счета**

(code, message) = EP\_DecreaseInvoice(mer\_no, pass, order, sum)

Для некоторых Поставщиков бывает необходимо изменить сумму счета после оплаты его Покупателем. В EasyPay существует возможность уменьшения суммы оплаченного счета. Для включения возможности уменьшения суммы оплаченных счетов Поставщику необходимо обратиться в [службу поддержки](https://ssl.easypay.by/feedback/) EasyPay.

**Описание полученных кодов**

В случае успешно проведенной операции возвращается code = 200 или code = 211 - счет не оплачен. Ниже приведен список некоторых встречающихся кодов:

code = 200 - успешное выполнение операции

code = 211 - счет не оплачен

code = 306 - идентификатор (номер) электронного кошелька EasyPay содержит недопустимые символы

code = 501 - несуществующий идентификатор (номер) электронного кошелька EasyPay

code = 330 - сумма меньше 100 руб. Необходимо увеличить сумму для проведения операции

code = 30 - неверный формат переданных данных

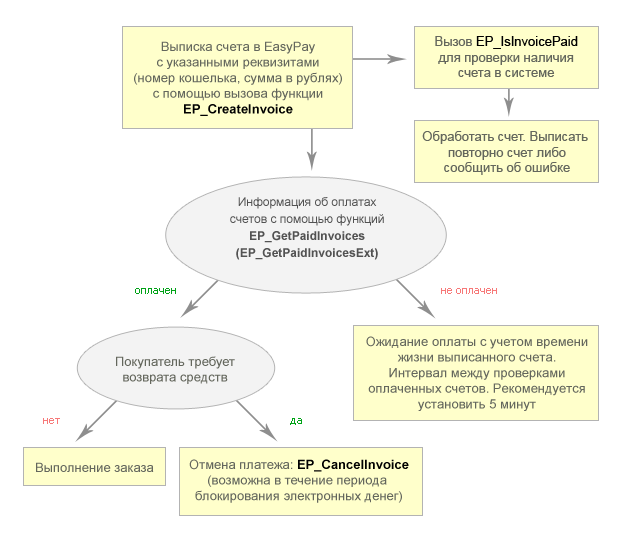
code = 31 - ошибка соединения с сервером EasyPay

code = 32 - неверный формат ответа сервера EasyPay

code = 500 - неизвестная внутренняя ошибка. При возврате ошибки с кодом "500" обратитесь к Администратору

*\*При выставлении счетов Покупателям отображайте полученное сообщение об ошибке вместе с кодом (номером).*

**Схема взаимодействия Поставщика с Процессинговым центром**



**Пример xml-пакетов**

Xml-пакеты формируются неявно при взаимодействии с SOAP-сервером EasyPay при вызове функций.

Рассмотрим xml-запрос и xml-ответ на примере функции:

(code, message) = EP\_IsInvoicePaid(mer\_no, pass, order)

Xml-запрос:

<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"

xmlns:ns1="http://easypay.by/">

<SOAP-ENV:Body>

<ns1:EP\_IsInvoicePaid>

<ns1:mer\_no>ok\*\*\*\*</ns1:mer\_no>

<ns1:pass>\*\*\*\*\*\*</ns1:pass>

<ns1:order>1170338164</ns1:order>

</ns1:EP\_IsInvoicePaid>

</SOAP-ENV:Body>

</SOAP-ENV:Envelope>

Xml-ответ:

<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"

xmlns:ns1="http://easypay.by/">

<SOAP-ENV:Body>

<ns1:EP\_IsInvoicePaidResponse>

<ns1:status>

<ns1:code>200</ns1:code>

<ns1:message>

</ns1:message>

</ns1:status>

</ns1:EP\_IsInvoicePaidResponse>

</SOAP-ENV:Body>

</SOAP-ENV:Envelope>

**Пример клиента на PHP5 с использованием библиотеки SOAP**

<?php

/\* client.php требования: PHP5 с расширением SOAP

SOAP-клиент для демонстрации обращения к SOAP-серверу EasyPay.

Для каждого конкретного Поставщика должна быть своя реализация SOAP-клиента

в соответствии с внутренней логикой работы и схемой взаимодействия с EasyPay

с соответствующей обработкой ошибок и исключительных ситуаций.

При использовании https-соединения в конструкторе клиента при необходимости

дополнительно указывать путь к сертификату и пароль на ключ сертификата

\*/

/\* Cписок доступных функции EasyPay

'EP\_CreateInvoice',

'EP\_IsInvoicePaid',

'EP\_GetPaidInvoices',

'EP\_SendMessage',

'EP\_GetPaidInvoicesExt',

'EP\_CheckCard',

'EP\_CreatePayment',

'EP\_CheckPayment',

'EP\_CancelInvoice',

'EP\_ConfirmInvoice',

'EP\_MerchantBalance',

\*/

//--------------------------------ПАРАМЕТРЫ----------------------------------

$wsdl = 'https://ssl.easypay.by/\*\*\*\*'; //адрес SOAP-сервера

$encoding = 'windows-1251'; //кодировка счета (кодировка на сайте Поставщика)

//массив параметров для вызова функции выписки счета (EP\_CreateInvoice)

$params\_EP\_CreateInvoice = array(

'mer\_no' =>'ok\*\*\*\*', //тестовый Поставщик

'pass' =>'\*\*\*\*\*', //пароль на тестовый Поставщик

'order' =>'4000', //номер счета

'sum' =>'100', //сумма в белорусских рублях

'exp' =>'4', //время жизни счета

'card' =>'00539959', //идентификатор (номер) электронного кошелька EasyPay

'comment' =>substr('Комментарий счета', 0, 50), //комментарий счета (до 50 симв.)

'info' =>substr('Информация о счете', 0 , 2000), //подробный ком. (до 2000 с.)

'xml' =>'<xml>Это xml-данные</xml>', //xml-данные

);

//параметры для вызова функции получения оплаченных счетов (EP\_GetPaidInvoices)

$params\_EP\_GetPaidInvoices = array(

'mer\_no' =>'ok\*\*\*\*', //тестовый Поставщик

'pass' =>'\*\*\*\*\*', //пароль на тестовый Поставщик

'order' =>'0', //номер счета. 0 - для получения первых 100 оплаченных счетов.

);

//---------------------------------------------------------------------------

main();

function main () {

global $wsdl, $encoding, $params\_EP\_CreateInvoice, $params\_EP\_GetPaidInvoices;

//создание клиента

$client = new SoapClient($wsdl, array('trace' => 1, 'encoding' => $encoding));

//может возникнуть исключение SoapFault при неявном формировании

//входных и выходных xml-пакетов

try {

//вызов функции EP\_CreateInvoice

$answer = $client->EP\_CreateInvoice($params\_EP\_CreateInvoice);

} catch (SoapFault $fault) {

print "ExeptionIsCatchedByUser!<br>faultcode: $fault->faultcode;

faultstring: $fault->faultstring<br>";

}

print\_r ($answer); //отображение структуры ответа сервера

/\* структура ответа сервера

stdClass Object (

[status] => stdClass Object

(

[code] => 200

[message] =>

)

)

\*/

/\*

обращаться к полям структуры можно так:

print $answer->status->code; //код ответа сервера EasyPay

print $answer->status->message; //сообщение, поясняющее код

возвращаемый объект можно преобразовать в массив с помощью (array)

\*/

//может возникнуть исключение SoapFault при неявном формировании

//входных и выходных xml-пакетов

try {

//вызов функции EP\_CreateInvoice

$answer = $client->EP\_GetPaidInvoices($params\_EP\_GetPaidInvoices);

} catch (SoapFault $fault) {

print "ExeptionIsCatchedByUser!<br>faultcode: $fault->faultcode;

faultstring: $fault->faultstring<br>";

}

//массив оплаченных счетов. Количество счетов = количеству элементов

print\_r ((array)$answer->invoices);

/\* вид массива оплаченных счетов. В этом примере есть 3 оплаченных счета

Array (

[0] => stdClass Object

(

[order\_mer\_code] => ad\_1200407111

[purch\_date] => 2008-01-15 16:25:50

)

[1] => stdClass Object

(

[order\_mer\_code] => ad\_1200567652

[purch\_date] => 2008-01-17 13:01:18

)

[2] => stdClass Object

(

[order\_mer\_code] => 4000

[purch\_date] => 2008-01-18 17:18:53

)

)

\*/

//массив полей первого счета в системе (если он есть) массива счетов

print\_r ((array)$answer->invoices[0]);

/\* вид массива полей оплаченного счета

Array (

[order\_mer\_code] => ad\_1200407111

[purch\_date] => 2008-01-15 16:25:50

)

\*/

//Остальные функции вызываются аналогичным образом

//получить структуру неявно формируемого xml-пакета можно так

//отображение xml-пакета для анализа и отладки (запрос)

$request = ($client->\_\_getLastRequest());

//для более наглядного отображения и анализа xml-пакета

$request = preg\_replace("/></", ">\n<", $request);

//отображение xml-пакета для анализа и отладки (ответ)

$response = ($client->\_\_getLastResponse());

//для более наглядного отображения и анализа xml-пакета

$response = preg\_replace("/></", ">\n<", $response);

print "<pre>\n";

print "Запрос:\n".$request."\n";

print "Ответ:\n".$response."\n";

print "</pre>";

}

?>

# ПРИЛОЖЕНИЕ Б

**Описание XML-интерфейсов системы WebMoney**

[XML-интерфейсы](https://wiki.webmoney.ru/projects/webmoney/wiki/XML-%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81%D1%8B) предназначены для использования в программах сторонних разработчиков.

В данном разделе представлены интерфейсы системы, основанные на обращениях (запросах) к специальному Web-серверу системы по протоколу https (SSL 128 бит) в формате XML с использованием специального модуля аутентификации для ключей WM Keeper Classic или стандартных сертификатов x.509 (сертификаты WM Keeper Light).

Все интерфейсы, использующие аутентификацию с ключами [WM Keeper Classic](https://wiki.webmoney.ru/projects/webmoney/wiki/WM_Keeper_Classic) имеют адрес вида [https://w3s.webmoney.ru/asp/XML\_название\_.asp](https://w3s.webmoney.ru/asp/XML_%D0%BD%D0%B0%D0%B7%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_.asp). Интерфейсы использующие аутентификацию со стандартными сертификатами (сертификаты [WM Keeper Light](https://wiki.webmoney.ru/projects/webmoney/wiki/WM_Keeper_Light)) - [https://w3s.wmtransfer.com/asp/XML\_название\_Cert.asp](https://w3s.wmtransfer.com/asp/XML_%D0%BD%D0%B0%D0%B7%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_Cert.asp).

В случае аутентификации с использованим ключей [WM Keeper Classic](https://wiki.webmoney.ru/projects/webmoney/wiki/WM_Keeper_Classic) в запросе передается подпись, сформированная модулем [WMSigner](https://wiki.webmoney.ru/projects/webmoney/wiki/WMSigner). Сформированная подпись передается в параметре <sign>...</sign>.  
В случае аутентификации с использованием сертификатов [WM Keeper Light](https://wiki.webmoney.ru/projects/webmoney/wiki/WM_Keeper_Light), сертификат должен использоваться при установлении соединения с сервером <https://w3s.wmtransfer.com>. Параметр <sign>...</sign> в этом случае в запросе не используется.

В случае сложной схемы взаимодействия кошельков разных [WMID](https://wiki.webmoney.ru/projects/webmoney/wiki/WMID), запросы также могут быть подписаны [доверенным WMID](https://wiki.webmoney.ru/projects/webmoney/wiki/%D0%94%D0%BE%D0%B2%D0%B5%D1%80%D0%B5%D0%BD%D0%BD%D1%8B%D0%B9_WMID).

Вне зависимости от типа используемой аутентификации, формат запросов одинаков и отличается только способом формирования и передачи подписи запроса. Ниже приведены примеры структуры запросов с различными типами аутентификации:

Структура запроса с аутентификацией по ключам [WM Keeper Classic](https://wiki.webmoney.ru/projects/webmoney/wiki/WM_Keeper_Classic) (подпись запроса формируется с помощью модуля [WMSigner](https://wiki.webmoney.ru/projects/webmoney/wiki/WMSigner))

<w3s.request>

<reqn></reqn> - номер запроса

<wmid></wmid> - WM идентификатор подписавшего запрос

<sign></sign> - подпись запроса сформированная с помощью модуля WMSigner

<тип\_запроса>

... - параметры запроса

</тип\_запроса>

</w3s.request>

Структура запроса с аутентификацией по стандартным сертификатам WM Keeper Light (параметры wmid и sign не используются; сертификат WMID, который вызывает интерфейс должен использоваться при установлении соединения с сервером). Более подробную информацию об использовании сертификатов смотрите на <http://msdn.microsoft.com>.

<w3s.request>

<reqn></reqn> - номер запроса

<тип\_запроса>

... - параметры запроса

</тип\_запроса>

</w3s.request>

Формат ответа системы не зависит от способа аутентификации запроса. Ниже приведен пример структуры ответа сервера:

<w3s.response>

<reqn></reqn> - номер запроса, на который высылается ответ

<retval></retval> - код ошибки выполнения запроса: 0 - выполнен успешно

<retdesc></retdesc> - описание ошибки если retval != 0

<тип\_ответа>

... - параметры ответа

</тип\_ответа>

</w3s.response>

**Интерфейс X1.** [**Выписывание счета**](https://wiki.webmoney.ru/projects/webmoney/wiki/%D0%92%D1%8B%D0%BF%D0%B8%D1%81%D0%BA%D0%B0_%D1%81%D1%87%D0%B5%D1%82%D0%B0) **от одного участника (магазина, ресурса) другому участнику (покупателю).**

Кипер, подписывающий запросы, должен иметь [персональный аттестат](https://wiki.webmoney.ru/projects/webmoney/wiki/%D0%BF%D0%B5%D1%80%D1%81%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D0%B0%D1%82%D1%82%D0%B5%D1%81%D1%82%D0%B0%D1%82) или выше.

URL для передачи запроса с авторизацией по ключам [WM Keeper Classic](https://wiki.webmoney.ru/projects/webmoney/wiki/WM_Keeper_Classic) - <https://w3s.webmoney.ru/asp/XMLInvoice.asp>.

URL для передачи запроса с авторизацией по ключам [WM Keeper Light](https://wiki.webmoney.ru/projects/webmoney/wiki/WM_Keeper_Light) - <https://w3s.wmtransfer.com/asp/XMLInvoiceCert.asp>.

метод - [POST](http://ru.wikipedia.org/wiki/HTTP)

формат запроса:

<w3s.request>

<reqn></reqn>

<wmid></wmid>

<sign></sign>

<invoice>

<orderid></orderid>

<customerwmid></customerwmid>

<storepurse></storepurse>

<amount></amount>

<desc></desc>

<address></address>

<period></period>

<expiration></expiration>

<onlyauth></onlyauth>

</invoice>

</w3s.request>

параметры запроса:

|  |  |  |
| --- | --- | --- |
| **название** | **назначение** | **примечание** |
| reqn | номер запроса | целое без знака, макс. количество цифр - 15; всегда должен быть больше номера предыдущего запроса на выписывание счета!!! |
| wmid | [WMID](https://wiki.webmoney.ru/projects/webmoney/wiki/WMID) подписавшего запрос | используется только при авторизации с ключами [WM Keeper Classic](https://wiki.webmoney.ru/projects/webmoney/wiki/WM_Keeper_Classic) |
| sign | подпись запроса | формируется из параметров: orderid+customerwmid+storepurse+amount+desc+address+period+expiration+reqn используется только при авторизации с ключами [WM Keeper Classic](https://wiki.webmoney.ru/projects/webmoney/wiki/WM_Keeper_Classic) |
| invoice | параметры выписываемого [счета](https://wiki.webmoney.ru/projects/webmoney/wiki/%D0%A1%D1%87%D0%B5%D1%82) |  |
| invoice\  orderid | номер счета | номер [счета](https://wiki.webmoney.ru/projects/webmoney/wiki/%D0%A1%D1%87%D0%B5%D1%82) в системе учета магазина; любое целое число без знака |
| invoice\  customerwmid | [WMID](https://wiki.webmoney.ru/projects/webmoney/wiki/WMID) покупателя |  |
| invoice\  storepurse | [кошелек](https://wiki.webmoney.ru/projects/webmoney/wiki/WM-%D0%BA%D0%BE%D1%88%D0%B5%D0%BB%D0%B5%D0%BA) для оплаты | номер кошелька, на который необходимо [оплатить счет](https://wiki.webmoney.ru/projects/webmoney/wiki/%D0%9E%D0%BF%D0%BB%D0%B0%D1%82%D0%B0_%D1%81%D1%87%D0%B5%D1%82%D0%B0) |
| invoice\  amount | сумма [счета](https://wiki.webmoney.ru/projects/webmoney/wiki/%D0%A1%D1%87%D0%B5%D1%82) | число с плавающей точкой (разделитель - .(точка) ) и для десяти с половиной может выглядеть так: 10.5; незначащие нули в конце и точка, если число целое, должны отсутствовать, например, 10.50 - не верно, 10.5 - верно, 9. - не верно, 9 - верно) |
| invoice\  desc | описание товара или услуги | произвольная строка от 0 до 255 символов; пробелы в начале или конце не допускаются, кодировка win-1251 |
| invoice\  address | адрес доставки товара | произвольная строка от 0 до 255 символов; пробелы в начале или конце не допускаются |

|  |  |  |
| --- | --- | --- |
| **название** | **назначение** | **примечание** |
| invoice\  period | срок протекции сделки | максимально допустимый срок протекции в днях; при оплате счета целое число от 0 до 255; если 0 - протекция сделки при оплате счета не разрешена. При выписке счета для получения средств на [WMD-кошелек](https://wiki.webmoney.ru/projects/webmoney/wiki/WMD-%D0%BA%D0%BE%D1%88%D0%B5%D0%BB%D0%B5%D0%BA) – максимальный срок возврата кредита в днях. |
| invoice\  expiration | срок оплаты [счета](https://wiki.webmoney.ru/projects/webmoney/wiki/%D0%A1%D1%87%D0%B5%D1%82) | максимально допустимый срок оплаты в днях; целое число от 0 до 255; если 0 - срок оплаты не определен |
| invoice\  onlyauth | учитывать разрешение получателя | если 0 - выписка счета будет выполняться без учета разрешает ли получатель выписывать ему счета; 1 - выписка счета будет выполняться только если получатель разрешает выписывать ему счета (в противном случае код возврата - 35). Получатель может запретить выписывать себе счета в двух случаях. В первом, если отправитель является корреспондентом получателя счета и получатель явно в свойствах корреспондента (раздел ограничения) запретил ему выписывать счета и во-втором, если отправитель не является корреспондентом и получатель запретил всем неавторизованным (не являющимся его корреспондентами) выписывать ему счета. |

формат ответа:

<w3s.response>

<reqn></reqn>

<retval></retval>

<retdesc></retdesc>

<invoice id="n1" ts="n2">

<orderid></orderid>

<customerwmid></customerwmid>

<storepurse></storepurse>

<amount></amount>

<desc></desc>

<address></address>

<period></period>

<expiration></expiration>

<state>0</state>

<datecrt></datecrt>

<dateupd></dateupd>

</invoice>

</w3s.response>

параметры ответа:

|  |  |  |
| --- | --- | --- |
| **название** | **назначение** | **примечание** |
| reqn | номер запроса |  |
| retval | код выполнения | 0 - запрос выполнен успешно (остальные значения в таблице код выполнения запроса) |
| retdesc | расшифровка кода выполнения |  |
| invoice | параметры выписанного счета | атрибут id - уникальный номер счета в системе учета WebMoney; атрибут ts - служебный номер счета в системе учета WebMoney |
| invoice\  orderid | номер счета | номер счета в системе учета магазина; любое целое число без знака |
| invoice\  customerwmid | [WMID](https://wiki.webmoney.ru/projects/webmoney/wiki/WMID) покупателя |  |
| invoice\  storepurse | номер [кошелька](https://wiki.webmoney.ru/projects/webmoney/wiki/WM-%D0%BA%D0%BE%D1%88%D0%B5%D0%BB%D0%B5%D0%BA) для оплаты |  |
| invoice\  amount | сумма счета для оплаты | число с плавающей точкой (разделитель - .(точка) ) и для десяти с половиной может выглядеть так: 10.5; незначащие нули в конце и точка, если число целое, должны отсутствовать, например, 10.50 - не верно, 10.5 - верно, 9. - не верно, 9 - верно) |
| invoice\  desc | описание товара или услуги | произвольная строка от 0 до 255 символов; пробелы в начале или конце не допускаются |
| invoice\  address | адрес доставки товара | произвольная строка от 0 до 255 символов; пробелы в начале или конце не допускаются |

|  |  |  |
| --- | --- | --- |
| **название** | **назначение** | **примечание** |
| invoice\  period | срок протекции сделки | максимально допустимый срок протекции при оплате счета в днях; целое число от 0 до 255; если 0 - протекция сделки при оплате счета не разрешена. При выписке счета для получения средств на [WMD-кошелек](https://wiki.webmoney.ru/projects/webmoney/wiki/WMD-%D0%BA%D0%BE%D1%88%D0%B5%D0%BB%D0%B5%D0%BA) – максимальный срок возврата кредита в днях. |
| invoice\  expiration | срок оплаты счета | максимально допустимый срок оплаты счета в днях; целое число от 0 до 255; если 0 - срок оплаты не определен |
| invoice\  state | состояние счета | 0 - не оплачен 1 - оплачен по протекции 2 - оплачен окончательно или без протекции 3 – отказан |
| invoice\  datecrt | дата и время создания счета | ГГГГММДД ЧЧ:ММ:СС |
| invoice\dateupd | дата и время изменения состояния счета | ГГГГММДД ЧЧ:ММ:СС |

код выполнения запроса:

|  |  |
| --- | --- |
| **код возврата** | **описание** |
| -100 | общая ошибка при разборе команды. неверный формат команды. |
| -9 | неверное значение поля w3s.request/reqn |
| -8 | неверное значение поля w3s.request/sign |
| -1 | неверное значение поля w3s.request/invoice/orderid |
| -2 | неверное значение поля w3s.request/invoice/customerwmid |
| -3 | неверное значение поля w3s.request/invoice/storepurse |
| -5 | неверное значение поля w3s.request/invoice/amount |
| -6 | слишком длинное поле w3s.request/invoice/desc |
| -7 | слишком длинное поле w3s.request/invoice/address |
| -11 | идентификатор, переданный в поле w3s.request/wmid не зарегистрирован |
| -12 | проверка подписи не прошла |
| 102 | не выполнено условие постоянного увеличения значения параметра w3s.request/reqn |
| **код возврата** | **описание** |
| 111 | попытка выставление счета для [кошелька](https://wiki.webmoney.ru/projects/webmoney/wiki/WM-%D0%BA%D0%BE%D1%88%D0%B5%D0%BB%D0%B5%D0%BA) не принадлежащего [WMID](https://wiki.webmoney.ru/projects/webmoney/wiki/WMID), которым подписывается запрос; при этом доверие не установлено. |
| 6 | получатель счета не найден |
| 5,7 | отправитель счета не найден |
| 35 | плательщик не авторизован корреспондентом для выполнения данной операции. Это означает, что магазин пытается выписать счет плательщику, который, либо не добавил ВМИД магазина к себе в список корреспондентов и при этом запретил неавторизованным (не являющимся его корреспондентами) выписывать себе счета (для Кипер Классик - в главном меню вверху - Инструменты - Парметры программы -Ограничения ), либо плательщик добавил ВМИД магазина к себе в корреспонденты, но именно для ВМИДа этого магазина запретил выписку себе счетов. Без действий со стороны плательщика избежать этой ошибки магазин не может, необходимо показать плательщику ВМИД магазина с инструкцией о том, что ВМИД магазина должен быть добавлен плательщиком в список корреспондентов и для ВМИДа должна быть разрешена выписка счета |
| 61 | Превышен лимит долговых обязательств заемщика |

**Интерфейс X4. Получение истории выписанных счетов по кошельку. Проверка оплаты счета.**

URL для передачи запроса с авторизацией по ключам [WM Keeper Classic](https://wiki.webmoney.ru/projects/webmoney/wiki/WM_Keeper_Classic) - <https://w3s.webmoney.ru/asp/XMLOutInvoices.asp>

URL для передачи запроса с авторизацией по ключам [WM Keeper Light](https://wiki.webmoney.ru/projects/webmoney/wiki/WM_Keeper_Light) - <https://w3s.webmoney.ru/asp/XMLOutInvoicesCert.asp>

метод - [POST](http://ru.wikipedia.org/wiki/HTTP)

формат запроса:

<w3s.request>

<reqn></reqn>

<wmid></wmid>

<sign></sign>

<getoutinvoices>

<purse></purse>

<wminvid></wminvid>

<orderid></orderid>

<datestart></datestart>

<datefinish></datefinish>

</getoutinvoices>

</w3s.request>

параметры запроса:

|  |  |  |
| --- | --- | --- |
| **название** | **назначение** | **Примечание** |
| reqn | номер запроса | целое без знака, макс. количество цифр - 15; **всегда должен быть больше номера предыдущего запроса!!!** |
| wmid | [WMID](https://wiki.webmoney.ru/projects/webmoney/wiki/WMID) подписавшего запрос | используется только при авторизации с ключами [WM Keeper Classic](https://wiki.webmoney.ru/projects/webmoney/wiki/WM_Keeper_Classic) |
| sign | подпись запроса - формируется из параметров: **purse+reqn** | используется только при авторизации с ключами [WM Keeper Classic](https://wiki.webmoney.ru/projects/webmoney/wiki/WM_Keeper_Classic) |
| getoutinvoices | параметры запрашиваемого счета (счетов), по которым ведется поиск | **обязательные параметры purse, datestart, datefinish**, остальные можно не задавать или указывать 0 |
| getoutinvoices\  purse | номер [кошелька](https://wiki.webmoney.ru/projects/webmoney/wiki/WM-%D0%BA%D0%BE%D1%88%D0%B5%D0%BB%D0%B5%D0%BA) для оплаты на который выписывался счет |  |
| getoutinvoices\wminvid | номер счета (в системе [WebMoney](https://wiki.webmoney.ru/projects/webmoney/wiki/WebMoney)) | целое число > 0 |
| getoutinvoices\orderid | номер счета | номер счета в системе учета магазина |
| getoutinvoices\datestart | минимальное время и дата создания счета | ГГГГММДД ЧЧ:ММ:СС |
| getoutinvoices\datefinish | максимальное время и дата создания счета | ГГГГММДД ЧЧ:ММ:СС |

формат ответа:

<w3s.response>

<reqn></reqn>

<retval></retval>

<retdesc></retdesc>

<outinvoices cnt="n" >

<outinvoice id="n1" ts="n2">

<orderid></orderid>

<customerwmid></customerwmid>

<storepurse></storepurse>

<amount></amount>

<desc></desc>

<address></address>

<period></period>

<expiration></expiration>

<state></state>

<datecrt></datecrt>

<dateupd></dateupd>

<wmtranid></wmtranid>

</outinvoice>

<outinvoice>...</outinvoice>

</outinvoices>

</w3s.response>

параметры ответа:

|  |  |  |
| --- | --- | --- |
| **название** | **назначение** | **примечание** |
| reqn | номер запроса |  |
| retval | код выполнения запроса | 0 - запрос выполнен |
| retdesc | расшифровка кода выполнения запроса |  |
| outinvoices | список выписанных счетов, полученных по запросу | атрибут **n** - кол-во счетов, удовлетворяющих запросу |
| outinvoices\  outinvoice | параметры счета | атрибут **id** - уникальный номер счета в системе учета [WebMoney](https://wiki.webmoney.ru/projects/webmoney/wiki/WebMoney) атрибут **ts** - служебный номер счета в системе учета [WebMoney](https://wiki.webmoney.ru/projects/webmoney/wiki/WebMoney) |
| outinvoice\  orderid | номер счета | номер счета перевода в системе учета отправителя |
| **название** | **назначение** | **примечание** |
| outinvoice\  customerwmid | [WMID](https://wiki.webmoney.ru/projects/webmoney/wiki/WMID) покупателя, которому был выписан счет |  |
| outinvoice\  storepurse | номер [кошелька](https://wiki.webmoney.ru/projects/webmoney/wiki/WM-%D0%BA%D0%BE%D1%88%D0%B5%D0%BB%D0%B5%D0%BA), на который необходимо оплатить счет |  |
| outinvoice\  amount | сумма счета, выставленная для оплаты покупателю |  |
| outinvoice\  desc | описание товара или услуги, на который выписывается счет |  |
| outinvoice\  address | адрес доставки товара |  |
| outinvoice\  period | максимально возможный при оплате счета срок протекции сделки в днях | При выписке счета для получения средств на [WMD-кошелек](https://wiki.webmoney.ru/projects/webmoney/wiki/WMD-%D0%BA%D0%BE%D1%88%D0%B5%D0%BB%D0%B5%D0%BA) - максимальный срок возврата кредита в днях |
| outinvoice\  expiration | максимально допустимый срок оплаты счета в днях |  |
| outinvoice\  state | состояние оплаты счета | 0 - не оплачен 1 - оплачен по протекции 2 - оплачен окончательно 3 – отказан |
| outinvoice\  datecrt | дата и время создания счета |  |
| outinvoice\  dateupd | дата и время последнего изменения состояния счета |  |
| **название** | **назначение** | **примечание** |
| outinvoice\  wmtranid | номер операции в системе [WebMoney](https://wiki.webmoney.ru/projects/webmoney/wiki/WebMoney), если счет оплачен |  |
| outinvoice\  customerpurse | [кошелек](https://wiki.webmoney.ru/projects/webmoney/wiki/WM-%D0%BA%D0%BE%D1%88%D0%B5%D0%BB%D0%B5%D0%BA) плательщика, если счет оплачен |  |

# ПРИЛОЖЕНИЕ В

**Описание структуры XML сообщений для доступа в режиме on-line из центрального узла АИС ЕРИП к биллинговым системам производителей услуг**

Настоящий документ содержит описание структуры XML сообщений, которые используются для доступа в режиме on-line из центрального узла АИС ЕРИП к биллинговым системам производителей услуг в процессе выполнения операций оплаты услуг.

Документ включает в себя способ доставки сообщений, перечень используемых сооб­щений, общие требования к структуре сообщений и представлению элементов данных в со­общениях, а также описание структуры конкретных сообщений.

В настоящей редакции документа рассматриваются протоколы взаимодействия только с теми производителями услуг, которые выставляют задолженность плательщика в виде суммы (случаи оплаты по счетчикам не рассматриваются).

Помимо описанных в настоящем документе on-line сообщений, для обмена данными с производителем услуг используются сообщения, описанные в документе УАРС. 1403.02 «Протоколы обмена данными между производителем услуг и региональным узлом», которые передаются в режиме off-line. Обязательно производителю услуг передается сообщение 210 (реестр оплаченных платежей).

**СПИСОК ЛИТЕРАТУРЫ**

Взаимодействие Поставщика и EasyPay по протоколу SOAP : [Электронный ресурс] / ООО "Открытый контакт" – 2004. – Режим доступа: <https://ssl.easypay.by/soap>. – Дата доступа: 01.03.2013.

Описание XML-интерфейсов системы [Электронный ресурс] / WebMoney – 1998. – Режим доступа: http://www.webmoney.ru/rus/developers/interfaces/xml/. – Дата доступа: 01.03.2013.

Описание структуры XML сообщений для доступа в режиме on-line из центрального узла АИС ЕРИП к биллинговым системам производителей услуг: [Текст] / ОАО «[Центр банковских технологий](http://www.cbt.by)». – Минск, 2009.