

IboxPro API

Руководство по интеграции .NET

V 1.3.0

---

## История изменений

Версия	Дата	Описание
1.0.0	12.04.2016	Исходная версия
1.0.1	20.04.2016	Исправлено Transaction.Card, вызов события PaymentCancelled. Добавлено событие ReverseEvent.CantReverse
1.0.2	30.05.2016	Добавлен enum Currency, исправлены ошибки округления
1.0.3	04.08.2016	Добавлены частичные отмены/возвраты. Актуализирована процедура EMV платежей, добавлена однофакторная авторизация. Добавлены свойства класса Transaction. Общие улучшения
1.0.4	11.08.2016	Добавлена оплата наличными
1.1.0	09.12.2016	Изменен фреймворк на .NET4.5. Добавлены считыватели карт Wisepad_2 и QPos_mini.
1.2.0	21.03.2017	Добавлена генерация слип-чека Общие улучшения
1.3.0	10.05.2017	Поддержка NFC для ридера P17 Изменены названия ридеров

## Содержание

История изменений.....	2
Обзор .....	4
Пространство имен Ibox.Pro.SDK.External .....	5
Перечисления(enums) .....	5
Класс PaymentController .....	7
Пространство имен Ibox.Pro.SDK.External.Context .....	14
Перечисления(enums) .....	14
Класс PaymentContext.....	16
Класс RegularPaymentContext.....	17
Класс PaymentResultContext.....	18
Пространство имен Ibox.Pro.SDK.External.Entry .....	19
Перечисления(enums) .....	19
Класс Transaction.....	20
Класс Card .....	22
Класс Schedule.....	23
Пространство имен Ibox.Pro.SDK.External.Result.....	24
Класс APIResult.....	24
Класс APIGetHistoryResult .....	25
Приложение 1: Печать слипа.....	26

## Обзор

## Пространство имен Ibox.Pro.SDK.External

### Перечисления(enums)

#### ReaderType

Набор поддерживаемых типов считывателей карт

Тип	Описание
P15	Считыватель карт «Chip&Pin», P15
P16	Считыватель карт «Chip&Pin», P16
P17	Считыватель карт «Chip&Pin NFC», P17

#### ReaderEvent

Набор возможных событий, которые могут быть переданы считывателем карт

Тип	Описание
Connected	Считыватель карт был подключен
Disconnected	Считыватель карт был отключен
Startinit	Начало инициализации
InitSuccessfully	Инициализация завершена успешно
InitFailed	Произошла ошибка инициализации
EjectCardTimeout	Не используется
SwipeCard	Обнаружено проведение магнитной полосой
TransactionStarted	Начата чиповая транзакция
WaitingForCard	Ожидание проведения магнитной полосой или вставки чиповой карты
PaymentCanceled	Платеж отменен пользователем
EjectCard	Пользователь может извлечь карту (возникает при ошибке проведения транзакции)
BadSwipe	Не удалось считать данные магнитной ленты
LowBattery	Уровень заряда батареи считывателя карт менее 10%

#### PaymentError

Набор возможных ошибок, которые могут возникнуть в процессе выполнения платежа

Тип	Описание
ConnectionError	Ошибка соединения с сервером
ServerError	Ошибка выполнения транзакции
PaymentRuntimeError	Ошибка выполнения платежа
TransactionNullOrEmpty	Ошибка создания транзакции
EmvError	
EmvTerminated	
EmvDeclined	
Тип	Описание

EmvCancel	
EmvCardError	
EmvCardBlocked	
EmvDeviceError	
EmvCardNotSupported	
EmvZeroTranEmv	
EmvNotAllowed	Чиповая транзакция не разрешена

#### ReverseEvent

События изменения состояния транзакции отмены(возврата)

Тип	Описание
TransactionNotFound	Транзакция не была найдена, либо не уникальна
SwitchedToCNP	Отмена транзакции будет выполнена в режиме CNP
CantReverse	Для транзакции не может быть выполнена операция отмены(возврата)

#### Currency

Валюты, которыми можно провести оплату

Тип	Описание
RUB	Российский рубль
VND	Вьетнамский донг

## Класс PaymentController

Данный класс является центральным в библиотеке. Содержит методы для создания и отмены(возврата) транзакций, передачи в них дополнительных параметров а также инкапсулирует работу со считывателями карт. Также с помощью этого класса осуществляется передача дополнительных данных платежа и получение истории платежей.

Перед проведением транзакций необходимо установить пользовательские Email и пароль, необходимые для аутентификации, с помощью метода **SetCredentials** и установить тип считывателя карт с помощью метода **SetReaderType**. иначе будет сгенерировано исключение **InvalidOperationException**. Также необходимо вызвать метод **Enable**. Вызов метода **Disable** приведет к отключению считывателя карт, но НЕ прервет обработку текущей транзакции. Если перед началом платежа не будут определены свойства **SelectApplicationDelegate**, **ConfirmScheduleDelegate** и **ScheduleCreationFailedDelegate** или не будет вызван метод **Enable** - будет сгенерировано исключение **InvalidOperationException**. При указании суммы платежа, разрядность десятичной части которой превышает разрядность десятичной части валюты, количество знаков после запятой такой суммы будет урезано **без округления**.

### Свойства:

Название	Описание
Instance	Экземпляр класса
IsPaymentInProgress	<b>true</b> , если платеж все еще выполняется
SinglestepEMV	Признак однофакторной авторизации
SelectApplicationDelegate	Обработчик выбора приложения карты
ConfirmScheduleDelegate	Обработчик подтверждения регулярного платежа
ScheduleCreationFailedDelegate	Обработчик запроса повторной попытки создания расписания

### События:

Название	Описание
TransactionStartedEvent	Начало выполнения транзакции
TransactionFinishedEvent	Успешное окончание выполнения транзакции
ReaderEvent	Изменилось состояние считывателя карт
ErrorEvent	При выполнении платежа произошла ошибка
ReverseEvent	Состояние транзакции отмены(возврата) изменилось

## Делегаты:

### SelectApplicationCallback

Сигнатура	Int SelectApplicationCallback(List<string> apps)
Входные параметры	apps – список названий приложений
Возвращаемое значение	Порядковый номер выбранного приложения (начиная с 0)
Описание	Вызывается при выполнении чиповой транзакции, если чиповая карта содержит более 1 приложения

### ConfirmScheduleCallback

Сигнатура	bool ConfirmScheduleCallback(List<KeyValuePair<DateTime, decimal>> steps, decimal totalAmount)
Входные параметры	steps – список шагов выполнения расписания, состоящий из пар типа <Дата списания, Сумма к списанию> totalAmount – итоговая сумма по всем дням
Возвращаемое значение	Признак того, что плательщик подтверждает правильность расписания
Описание	Вызывается при создании регулярного платежа

### ScheduleCreationFailedCallback

Сигнатура	bool ScheduleCreationFailedCallback(PaymentError error, string description = null)
Входные параметры	error – тип ошибки description – сообщение об ошибке. Используется только когда error == SERVER_ERROR
Возвращаемое значение	<b>true</b> , если необходимо повторить попытку создания расписания
Описание	Вызывается в случае возникновения ошибки при создании расписания регулярного платежа

### TransactionStartedEventHandler

Сигнатура	void TransactionStartedEventHandler(string transactionID)
Входные параметры	transactionID – ID созданной транзакции
Возвращаемое значение	нет
Описание	Обработчик события <b>TransactionStartedEvent</b>

### TransactionFinishedEventHandler

Сигнатура	void TransactionFinishedEventHandler(PaymentResultContext result)
Входные параметры	result – данные проведенной транзакции в представлении <b>PaymentResultContext</b>
Возвращаемое значение	нет
Описание	Обработчик события <b>TransactionFinishedEvent</b>



### ReaderEventHandler

Сигнатура	void ReaderEventHandler(ReaderEvent readerEvent)
Входные параметры	readerEvent – событие считывателя карт
Возвращаемое значение	нет
Описание	Обработчик события <b>ReaderEvent</b>

### ErrorEventHandler

Сигнатура	void ErrorHandler(PaymentError error, string description = null)
Входные параметры	error – возникнувшая ошибка description – описание
Возвращаемое значение	нет
Описание	Обработчик события <b>ErrorEvent</b>

### ReverseEventHandler

Сигнатура	void ReverseEventHandler(ReverseEvent reverseEvent, string description = null)
Входные параметры	event – событие description – описание
Возвращаемое значение	нет
Описание	Обработчик события <b>ReverseEvent</b>

### Методы класса:

#### Enable

Сигнатура	void Enable()
Входные параметры	Нет
Возвращаемое значение	Нет
Описание	Начинает работу со считывателем карт

#### Disable

Сигнатура	void Disable()
Входные параметры	Нет
Возвращаемое значение	Нет
Описание	Завершает работу со считывателем карт

#### SetCredentials

Сигнатура	void SetCredentials(string email, string password)
Входные параметры	email – email пользователя password – пароль пользователя
Возвращаемое значение	Нет
Описание	Задаёт данные пользователя, необходимые для работы

## Auth

Сигнатура	APIAuthResult Auth()
Входные параметры	Нет
Возвращаемое значение	APIAuthResult
Описание	Проверяет правильность введенных учетных данных и возвращает информацию об учетной записи

## SetReaderType

Сигнатура	void SetReaderType(ReaderType readerType, string readerBTPort = null)
Входные параметры	readerType – тип считывателя карт readerBTPort – название COM порта, через который подключен Bluetooth-ридер.
Возвращаемое значение	Нет
Описание	Изменяет тип текущего считывателя карт. При попытке изменить тип считывателя во время проведения платежа будет сгенерировано исключение <b>InvalidOperationException</b>

## GetReaderType

Сигнатура	ReaderType? GetReaderType()
Входные параметры	Нет
Возвращаемое значение	Текущий тип считывателя карт
Описание	Возвращает текущий тип считывателя карт

## StartPayment

Сигнатура	void StartPayment(PaymentContext paymentContext)
Входные параметры	paymentContext – данные платежа
Возвращаемое значение	Нет
Описание	Начинает выполнение платежа. Перед вызовом необходимо установить данные пользователя, свойства <b>SelectApplicationDelegate</b> , <b>ConfirmScheduleDelegate</b> , <b>ScheduleCreationFailedDelegate</b> и убедиться, что тип считывателя карт задан, а сам считыватель подключен, иначе будет сгенерировано исключение <b>InvalidOperationException</b> . При попытке начать новый платеж/отмену платежа до окончания предыдущего будет сгенерировано исключение <b>InvalidOperationException</b> .

## StartReverse

Сигнатура	void StartReverse(string transactionID, ReverseMode mode, decimal? amountToReverse)
-----------	---

Входные параметры	transactionID – ID транзакции отменяемого платежа mode – тип отмены amountToReverse – сумма, на которую будет выполнена отмена/возврат. Для полной отмены/возврата передавать <b>null</b>
Возвращаемое значение	Нет
Описание	Начинает выполнение отмены платежа. Перед вызовом необходимо установить данные пользователя и убедиться, что тип считывателя карт задан, а сам считыватель подключен, иначе будет сгенерировано исключение <b>InvalidOperationException</b> . При попытке начать новый платеж/отмену платежа до окончания будет сгенерировано исключение <b>InvalidOperationException</b> .

#### Adjust

Сигнатура	APIResult Adjust(string transactionID, string email, string phone)
Входные параметры	transactionID – ID транзакции, для которой требуется отправить дополнительные данные email – email для отправки чека phone – номер телефона для отправки чека
Возвращаемое значение	Результат отправки данных
Описание	Используется для отправки подписи и чека для транзакции единичного платежа

#### AdjustRegular

Сигнатура	APIResult AdjustRegular(string transactionID, string email, string phone)
Входные параметры	transactionID – ID транзакции, для которой требуется отправить дополнительные данные email – email для отправки чека phone – номер телефона для отправки чека
Возвращаемое значение	Результат отправки данных
Описание	Используется для отправки подписи и чека для регулярного платежа

#### AdjustReverse

Сигнатура	APIResult AdjustReverse(string transactionID, string email, string phone)
Входные параметры	transactionID – ID транзакции, для которой требуется отправить дополнительные данные email – email для отправки чека phone – номер телефона для отправки чека

Возвращаемое значение	Результат отправки данных
Описание	Используется для отправки подписи и чека для транзакции отмены(возврата) платежа

## GetHistory

Сигнатура	APIGetHistoryResult GetHistory(int page)
Входные параметры	page – номер страницы
Возвращаемое значение	Объект APIGetHistoryResult, содержащий набор транзакций
Описание	Позволяет получить историю транзакций в постраничном виде

## GetTranscationByID

Сигнатура	APIGetHistoryResult GetTransactionByID(string transactionID)
Входные параметры	transactionID – ID целевой транзакции
Возвращаемое значение	Объект APIGetHistoryResult, содержащий набор транзакций
Описание	Позволяет получить данные транзакции по ее ID

## Пространство имен Ibox.Pro.SDK.External.Context

### Перечисления(enums)

#### RepeatType

Набор возможных типов регулярного платежа

Тип	Описание
DelayedOnce	Платеж будет выполнен один раз
Weekly	Еженедельный платеж
Monthly	Ежемесячный платеж
Quarterly	Ежеквартальный платеж
Annual	Ежегодный платеж
ArbitraryDays	Платеж будет выполняться в заданные дни

#### EndType

Набор возможных способов окончания выполнения регулярного платежа

Тип	Описание
ByQuantity	Окончание по количеству повторов
AtDay	Окончание в заданный день

#### ReverseMode

Набор возможных способов отмены платежа

Тип	Описание
Cancel	Отмена платежа
Return	Возврат платежа
CancelCNP	Отмена в режиме CNP

#### InputType

Набор возможных способов оплаты

Тип	Описание
Swipe	Оплата с помощью проката карты магнитной лентой
Chip	Оплата с помощью чипа на карте
Cash	Оплата наличными



## Класс PaymentContext

Контейнер данных, необходимых для выполнения разового платежа.

### Свойства класса:

Название	Описание
Amount	Сумма платежа
Currency	Валюта платежа
Description	Описание платежа
Image	Изображение, прикрепленное к платежу
PaymentProductCode	Код продукта в платеже
PaymentProductTextDictionary	Набор значений текстовых полей продукта в формате <Код поля, значение>
PaymentProductImageDictionary	Набор изображений для графических полей продукта в формате <Код поля, изображение>
Cash	Признак оплаты наличными

### Методы класса:

#### Clear

Сигнатура	Clear()
Входные параметры	Нет
Возвращаемое значение	Нет
Описание	Выполняет очистку полей объекта



## Класс RegularPaymentContext

Расширение класса **PaymentContext**, дополнительно содержащее свойства, необходимые для создания регулярного платежа. Для выполнения платежа в последний день месяца свойство **DayOfWeek** должно иметь значение, равное константе **LAST\_DAY\_OF\_MONTH**.

### Свойства класса:

Название	Описание
PaymentRepeatType	Тип регулярного платежа
PaymentEndType	Способ завершения выполнения регулярного платежа
StartDate	Дата начала выполнения регулярного платежа
EndDate	Дата окончания выполнения регулярного платежа (если окончание по дате)
RepeatCount	Количество выполнений регулярного платежа (если окончание по количеству повторов)
ArbitraryDays	Дни, заданные для выполнения платежа (если тип платежа – по заданным датам)
Month	Месяц для выполнения платежа, [1,12]
MonthOfQuarter	Месяц для выполнения ежеквартального платежа, [1,4]
Day	День для выполнения платежа, [1,31]
DayOfWeek	День недели для выполнения платежа, [0,7], 0 – Воскресенье)
Hour	Час выполнения платежа
Minute	Минута выполнения платежа
ReceiptEmail	Email для отправки чека
ReceiptPhone	Телефон для отправки чека

Набор необходимых заполненных свойств зависит от типа платежа:

Тип платежа	Набор свойств
Never	StartDate
Weekly	StartDate, (EndDate или RepeatCount)
Monthly	StartDate, (EndDate или RepeatCount), Day
Quarterly	StartDate, (EndDate или RepeatCount), MonthOfQuarter, Day
Annual	StartDate, (EndDate или RepeatCount), Month, Day
ArbitraryDays	ArbitraryDays

Параметры RepeatType, EndType, ReceiptEmail, ReceiptPhone являются обязательными для всех типов регулярных платежей.

Параметры Hour, Minute являются необязательными для всех типов регулярных платежей.

## Класс PaymentResultContext

Контейнер данных, полученных при успешном проведении платежа или отмене(возврате) платежа.

**Свойства класса:**

Название	Описание
TransactionItem	Данные о транзакции платежа/отмены платежа в представлении <b>TransactionItem</b>
ScheduleItem	Данные о транзакции регулярного платежа в представлении <b>ScheduleItem</b>
RequiresSignature	признак необходимости отправки подписи плательщика после оплаты
TerminalName	Терминал
EmvData	Набор данных EMV(чиповой) транзакции в представлении <b>HashMap&lt;String, String&gt;</b>

## Пространство имен Ibox.Pro.SDK.External.Entry

### Перечисления(enums)

#### InputType

Набор возможных способов оплаты

Тип	Описание
SWIPE	Оплата с помощью проката карты магнитной лентой
CHIP	Оплата с помощью чипа на карте
CASH	Оплата наличными

#### DisplayMode

Тип отображения транзакции

Тип	Описание
Declined	Отклоненная транзакция
Success	Успешная транзакция
Reverse	Транзакция отмены/возврата
Reversed	Платеж отменен/выполнен возврат
NonFinancial	

## Класс Account

Содержит информацию об учетной записи

Свойства класса:

Название	Описание
SingleStepAuth	Признак доступности одношаговой авторизации
Name	Имя агента
BranchName	Название филиала
BranchAddress	Адрес филиала
BranchPhone	Телефон филиала
ClientName	Название компании
ClientLegalName	Юридическое название компании
ClientLegalAddress	Юридический адрес компании
ClientRealAddress	Фактический адрес компании
ClientPhone	Телефон компании
ClientWeb	Сайт компании
BankName	Название банка
TerminalName	Номер терминала

## Класс Transaction

Объектное представление транзакции. Содержит набор свойств, определяющих ее

Свойства класса:

Название	Описание
ID	ID транзакции
Date	Время и дата выполнения транзакции, согласно GMT устройства
Description	Описание транзакции
Invoice	Номер чека
Terminal	Номер терминала
AcquirerApprovalCode	Код подтверждения транзакции
ScheduleID	ID регулярного платежа
ScheduleStepID	ID списания для рекуррентного платежа
Amount	Сумма транзакции
InputType	Способ оплаты в представлении <b>InputType</b>
Operation	Название операции
Latitude	Географическая широта места выполнения транзакции
Longitude	Географическая долгота места выполнения транзакции

HasPhoto	Признак наличия приложенного изображения
PhotoUrl	URL приложенного изображения
RequiresSignature	Признак требования подписи. False, если оплата была подтверждена вводом PIN кода
HasSignature	Признак наличия приложенной подписи
SignatureUrl	URL приложенной подписи
StateDisplay	Описание состояния транзакции
CardholderName	Имя держателя карты
Card	Данные карты, которая была использована для оплаты, в представлении <b>Card</b>
EMVData	Список EMV тэгов карты, необходимых для печати в слипе
CanCancel	Признак возможности проведения отмены платежа
CanReturn	Признак возможности проведения возврата платежа
CanCancelPartial	Признак возможности проведения частичной отмены платежа
CanReturnPartial	Признак возможности проведения частичного возврата платежа
Canceled	Признак того, что для платежа выполнена отмена или возврат
DisplayMode	Тип отображения транзакции в представлении <b>DisplayMode</b>
SubstateDisplay	Описание подсостояния транзакции

## Класс Card

Содержит данные о платежной карте.

### Свойства класса:

Название	Описание
IIN	Тип карты или "cash"(в случае оплаты наличными)
BIN	Внутренний идентификатор банка
EXP	Срок действия карты
PANMasked	Первые и последние 4 цифры номера карты, разделенные символом "*"
PANEnding	Последние 4 цифры номера карты

### Методы класса:

#### IsCash

Сигнатура	IsCash()
Входные параметры	Нет
Возвращаемое значение	<b>true</b> , если оплата была проведена наличными
Описание	Возвращает признак того, что оплата была проведена наличными

## Класс Schedule

Является объектным представлением данных регулярного платежа

Свойства класса:

Название	Описание
ID	ID регулярного платежа
Card	Данные карты, которая была использована для оплаты, в представлении <b>TransactionItem.Card</b>

## Пространство имен Ibox.Pro.SDK.External.Result

### Класс APIResult

Является примитивной сущностью, содержащую ответ от сервера

**Свойства класса:**

Название	Описание
ErrorCode	Код ошибки. 0 – если ответ не содержит сообщений об ошибке.
ErrorMessage	Сообщение об ошибке



## Класс **APIAuthResult**

Дочерний класс **APIResult**. Содержит набор транзакций, полученных в ответ на запрос истории.

### Свойства класса:

Название	Описание
Account	Информация об учетной записи

## Класс **APIGetHistoryResult**

Дочерний класс **APIResult**. Содержит набор транзакций, полученных в ответ на запрос истории.

### Свойства класса:

Название	Описание
Transactions	Набор транзакций, содержащихся в ответе

## Приложение 1: Печать слипа

Данные для слипа приходят в событии onPaymentFinished  
Реквизиты клиента можно получить при помощи метода PaymentController.Auth().

### Поля слипа:

Название	Описание
Банк	Account.BankName
Название компании	Account. ClientName
Название юридического лица	Account. ClientLegalName
Телефон компании	Account. ClientPhone
WEB-сайт компании	Account. ClientWeb
Дата и время операции	result.TransactionItem.Date
Номер терминала	result.TransactionItem.TerminalName
Номер чека	result.TransactionItem.Invoice
Код подтверждения	result.TransactionItem.AcquirerApprovalCode
Номер и тип карты	result.TransactionItem.Card.IIN, result.TransactionItem.Card.PANMasked
EMV тэги транзакции	result.TransactionItem.EMVData, печатаются в виде ключ-значение
Тип операции	result.TransactionItem.Operation
Сумма операции	result.TransactionItem.Amount
Комиссия	0.00 р.
Статус	Успешно
Подпись клиента	Место для подписи в случае, если result.TransactionItem.RequiresSignature==true, в ином случае «Подтверждено вводом PIN»

### Пример слипа:

ВТБ 24  
Тестовый клиент  
ООО "Тестовый клиент"  
+7 916 111 2233  
www.testclient.com  
Дата и время операции: 21.03.2017 15:47:34  
Терминал: II040001  
Чек: RM7ZEDMAAE7L  
Код подтверждения: SIMULATION  
Карта: mastercard \*\*\*\* 5631  
AID: A0000000041010  
TSI: 6800  
TVR: 8020008000  
Операция: Purchase  
Итого: 33 р  
Комиссия: 0.00 р  
Статус: Успешно  
Подтверждено вводом PIN кода.