

ЗАО «НАЦИОНАЛЬНАЯ СЕРВИСНАЯ КОМПАНИЯ»

Протокол RURUPay

версия 1.9.17

01.10.2019

История изменений

Версия	Дата	Внесённые изменения	Автор
0.3	01.03.2011	Исходная версия	Петров А.К.
0.5	11.04.2011	Техническое описание протокола	Петров А.К.
1.0	12.07.2011	Версия протокола 1.0	Стефанов Р.Н.
1.1	02.11.2011	Версия протокола 1.1 (добавлен Amount)	Стефанов Р.Н.
1.2	16.11.2011	Версия протокола 1.2. Клиентская часть	Мялицин В.В.
1.3	24.11.2011	Версия протокола 1.3. WillCallback	Мялицин В.В.
1.4	08.12.2011	Версия протокола 1.4 Добавлен опциональный параметр externalId в метод Init	Шиков И.А.
1.5	18.01.2012	Исправления в клиентской части	Стефанов Р.Н.
1.6	09.02.2012	Спецификация для серверной части	Стефанов Р.Н.
1.7	02.03.2012	Поддержка ЭЦП Изменён интерфейс серверной части (добавлен spld во все запросы)	Стефанов Р.Н.
1.8	12.03.2012	Изменена схема url-encoding (кодируются только params и signature)	Стефанов Р.Н.
1.8.1	14.03.2012	Обязательное указание spld при использовании ЭЦП	Стефанов Р.Н.
1.8.2	02.05.2012	Внесены уточнения для action=cancelinit	Стефанов Р.Н.
1.8.3	16.06.2012	Добавлен параметр amount в клиентскую часть	Стефанов Р.Н.
1.8.4	05.07.2012	Зафиксирован порядок следования параметров запроса в серверной части	Стефанов Р.Н.
1.8.5	09.07.2012	Расширены коды ошибок в клиентской части	Стефанов Р.Н.
1.9	09.07.2012	Добавлены описания сценариев взаимодействия с поставщиком услуг	Стефанов Р.Н.
1.9.1	13.12.2012	Расширена спецификация кодов ошибок	Стефанов Р.Н.
1.9.2	10.06.2013	Добавлено описание метода получения статуса покупки Добавлено описание метода предварительной инициации покупки Расширена спецификация кодов ошибок	Еманов А.Н.
1.9.3	19.06.2013	Добавлено описание кодов Reason	Еманов А.Н.
1.9.4	24.07.2013	Расширен набор кодов Reason	Стефанов Р.Н.
1.9.5	15.01.2014	Исправлены ссылки	Еманов А.Н.
1.9.6	17.02.2014	Исправлено описание обязательности параметров	Рохин А.А.
1.9.7	05.03.2014	Уточнения обязательности параметров, соответствие параметров клиентского и серверного протоколов	Рохин А.А.
1.9.8	01.09.2014	Новые методы cancelpayment, Cancel, Notice	Мялицин В.В.
1.9.9	07.10.2014	Особенности подписи метода Notice	Лагутин М.А.
1.9.10	24.04.2015	Косметические изменения Добавлен пример кода на PHP для инициации транзакции Исправлен порядок параметров в ответе Response метода Init Из списка поддерживаемых протоколов исключён SSL	Еманов А.Н.
1.9.11	24.04.2015	Формирование ЭЦП для метода Notice	Еманов А.Н.
1.9.12	14.01.2016	Добавлено описание дополнительных параметров (parameters) во входных параметрах метода Init серверной части протокола	Еманов Е.Н.
1.9.13	12.02.2016	Уточнены параметры, участвующие в формировании	Еманов Е.Н.

		подписи методов PreInit и Init серверной части протокола.	
1.9.14	09.03.2016	Документ разделён на две части	Еманов А.Н.
1.9.15	12.04.2018	Добавлено описание дополнительных параметров (parameters) во входных параметрах метода Init серверной части протокола	Еманов Е.Н.
1.9.16	24.06.2019	Обновлено WSDL-описание серверной части	Еманов Е.Н.
1.9.17	01.10.2019	Новый метод CancelLink	Комаров М.А

Оглавление

История изменений.....	2
1. Термины и сокращения	6
2. Общие положения.....	7
3. Общие сведения о протоколе	7
3.1. Транспортный уровень	7
4. Спецификация клиентской части протокола	7
4.1. Типы запросов	7
4.1.1. Типы входных параметров	7
4.1.2. Формат ответного сообщения	8
4.2. Формирование ЭЦП.....	9
4.3. Коды ошибок.....	10
4.4. Запросы от ПС к ТСП	11
4.4.1. Проверка параметров/Резервирование услуги (Init)	11
4.4.2. Покупка товара (Payment).....	12
4.4.3. Отмена резервирования (CancelInit).....	12
5. Спецификация серверной части протокола.....	13
5.1. Транспортный уровень	13
5.2. Формирование ЭЦП.....	13
5.3. Коды ошибок.....	13
5.3.1. ErrorCode	13
5.3.2. Reason	14
5.4. Запросы	14
5.4.1. Инициация покупки (Init)	15
5.4.2. Подтверждение резервирования (ReserveCallback).....	17
5.4.3. Подтверждение покупки (PurchaseCallback)	18
5.4.4. Получение статуса покупки (GetTransactionStatus)	18
5.4.5. Отмена покупки (Cancel)	19
5.4.6. Удаление привязки (CancelLink)	20
5.5. Структуры	21
5.5.1. Parameter.....	21
6. Стандартные сценарии интеграции.....	22
Варианты подключения и необходимые протоколы.....	22
Реализация методов клиентской части.....	22
Реализация методов серверной части.	23

6.1.	Инициация с витрины RuRu	24
6.1.1.	Мобильная коммерция. Синхронное взаимодействие.	24
6.1.2.	Мобильная коммерция. Асинхронное взаимодействие.	29
6.1.3.	Банковская карта. Синхронное взаимодействие. Успешный сценарий.	33
6.2.	Инициация с витрины ТСП	34
6.2.1.	Запрос статуса транзакции со стороны партнёра	34
6.2.2.	Мобильная коммерция. Синхронное взаимодействие без шага резервирования. Успешный сценарий.	35
6.2.3.	Мобильная коммерция. Синхронное взаимодействие с резервированием. Успешный сценарий.	36
6.3.	Инициация через Widget	37
	Приложение А. WSDL-описание серверной части.....	38
	Приложение Б. Связь между клиентским и серверным протоколами	39
	Приложение В. Примеры кода.....	39
	Подпись запроса на создание транзакции (PHP)	39

1. Термины и сокращения

Аос	Advice of charge/Запрос на согласие абонента совершить покупку
Привязка	<p>Связь [Идентификатор услуги]-[Идентификатор пользователя у партнёра]-[Сумма (опционально)]-[Телефонный номер пользователя], хранящаяся на стороне НСК.</p> <p>Последний атрибут - это и номер счёта, с которого будут выполняться списания средств, и номер телефона, через который пользователь и система привязок могут взаимодействовать посредством СМС.</p> <p>[Идентификатор пользователя у партнёра] и [Телефонный номер пользователя] могут совпадать.</p> <p>Если существует привязка номера телефона пользователя к услуге, то списание средств с телефона будет выполняться без Аос'а.</p>
ПС	Платёжная система РУРУ
СП	Сервис провайдер. Используется как синоним ТСП
ТСП	Торгово-сервисное предприятие
Услуга	Услуга или товар, предоставляемый ТСП
ЭЦП	Электронно-цифровая подпись

2. Общие положения

Протокол предназначен для организации информационного обмена между платёжной системой RuRu (ПС) и Торгово-сервисным Предприятием/Поставщиком услуг (ТСП). Протокол состоит из двух частей: серверной и клиентской. Серверная часть протокола используется партнёром для инициации транзакции в платёжной системе, уведомлении об окончании процедуры резервирования и покупки товара в случае асинхронного взаимодействия. Клиентская часть предназначена для уведомления ТСП со стороны ПС о необходимости резервирования/покупки и отмены резервирования товара.

3. Общие сведения о протоколе

3.1. Транспортный уровень

Информационное взаимодействие осуществляется с использованием протокола безопасной передачи данных HTTPS (протокол прикладного уровня HTTP с использованием TLS). Аутентификация и авторизация происходит путём предъявления цифрового сертификата формата X.509 v3. Сертификаты предъявляются и клиентской (ПС) и серверной стороной (ТСП), либо только серверной при условии использования ЭЦП.

4. Спецификация клиентской части протокола

Параметры запроса передаются из ПС в ТСП посредством метода GET согласно RFC 1945, RFC 2616, RFC 3986. ТСП может рассчитывать на то, что параметры будут передаваться в запросе в том же порядке, что присутствуют в описании. Длина URL запроса не превышает 8192 символа.

Параметры передаются в кодировке «windows-1251». ТСП в свою очередь формирует ответ в XML-формате в кодировке «utf-8».

ТСП предоставляет базовый адрес сервиса, реализующего протокол. ПС формирует URL запрос в формате ***https://<базовый адрес сервиса>/?<имя параметра 1>=<значение>&<имя параметра 2>=<значение>*** и т.д.

4.1. Типы запросов

ПС может отправлять в ТСП следующие запросы:

- 1) Запрос на проверку параметров (резервирование Услуги).
- 2) Запрос на отмену резервирования Услуги.
- 3) Запрос на покупку Услуги.

Тип запрашиваемой услуги ПС передаёт в параметре ***action***.

4.1.1. Типы входных параметров

Для каждого типа запроса ПС передаёт фиксированный набор параметров в строке запроса. ТСП может потребоваться получать дополнительные параметры для предоставления Услуги. Такие параметры передаются в параметре ***params*** в формате ***<имя параметра 1> <значение>;<имя параметра N> <значение>***; Имя и значение параметров разделяются символом пробела. Значение ***params*** передаётся в формате url-encoded.

Имя параметра	Назначение	Формат	Обязательный
action	Определяет тип запроса: <ul style="list-style-type: none"> ✓ <i>init</i> – запрос на инициализацию платежа (резервирование Услуги) ✓ <i>cancelinit</i> – запрос на отмену резервирования Услуги ✓ <i>payment</i> – запрос на оплату Услуги 	Одно из фиксированных значений	Да
id	Идентификатор запроса в ПС	Целое число длиной 8 байт	Да
externalid	Идентификатор запроса на стороне ТСП	Строка 16 символов	Да, кроме запроса init
reason	Код причины отказа от резервирования или оплаты	Целое число. Допустимые значения: 0 – клиент отказался от резервирования или оплаты 1 – технический сбой 2 – ошибка источника средств	Только для запросов на отмену Список кодов будет расширяться.
code	Код услуги	Строка 20 символов Допустимые символы: a-Z_0-9	Да, кроме запросов cancelinit
account	Идентификатор клиента в услуге (Например, номер телефона абонента при оплате услуг связи)	Строка 64 символа	Нет
amount	Сумма в копейках	Целое число	Нет
date	Дата регистрации запроса на стороне ПС	yyyy-MM-dd HH:mm:ss	Да, кроме запросов cancelinit
params	Дополнительные параметры, определяемые ТСП	<i>имя1 значение; имяN значение;</i>	Нет
signature	ЭЦП	Строка Base64, передаётся в формате url-encoded	Только при взаимодействии по ЭЦП

Необязательность параметра подразумевает то, что в строке запроса он может отсутствовать (для конкретных видов запросов), а не то, что значение параметра можно задать пустым.

По мере развития ПС фиксированный набор параметров может расширяться, что не должно влиять на корректность обработки запросов ТСП.

4.1.2. Формат ответного сообщения

На все запросы ПС ТСП отвечает XML сообщением **ServiceResponse**

```
<ServiceResponse
  xmlns="http://ruru.service.provider"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
```



```

>
<ErrorCode></ErrorCode>
<ErrorDescription></ErrorDescription>
<WillCallback></WillCallback>
<Signature></Signature>
<ResponseBody></ResponseBody>
</ServiceResponse>

```

Элементы секции **ServiceResponse**

Имя элемента	Описание	Формат	Обязательный
ErrorCode	Код ошибки	Целое число (см.)	Да
ErrorDescription	Текстовое описание ошибки, которое может быть отображено клиенту.	Строка	Нет
WillCallback	Признак того, что ТСП вернёт результат операции асинхронно по клиентскому протоколу (только для операций резервирования и оплаты услуги)	Boolean <i>true/false</i>	Нет, значение по умолчанию <i>false</i>
Signature	ЭЦП ТСП	Строка Base64	Только при взаимодействии по ЭЦП
ResponseBody	Содержимое зависит от конкретного типа запроса	XML	Да, в случае если ErrorCode = 0

4.2. Формирование ЭЦП

Подписываемое сообщение является строкой, в которой перечислены значения всех передаваемых параметров¹. Если в подписи используются не все параметры, то в этом случае необходимые параметры перечисляются в описании метода. В настоящий момент поддерживается тип подписи HMACSHA1. Для запросов со стороны RuRu в ТСП тело для подписи формируется из следующих значений и в следующем порядке:

```
<action><id><externalId><code><account><amount><date><params><reason>
```

В случае если какой-либо из параметров отсутствует – он не участвует в подписи. Пример:

```
https://<базовый_адрес>/?action=init&id=232456456&externalId=123456&code=55&account=9030000000&amount=50000&date=2012-03-02 22:12:13&params=pname
pvalue;&signature=DjnQ8jSocJmiSd0Nyfiq9QerjFE=
```

Строка для подписи будет выглядеть таким образом:

```
init232456456123456559030000000500002012-03-02 22:12:13pname pvalue;
```

Все параметры подписываются до URL-кодирования.

В ответном сообщении ТСП также формирует подпись путём перечисления возвращаемых значений. Порядок значений следующий:

¹ Здесь и далее подразумеваются все параметры, кроме параметра *signature*, предназначенного для передачи значения подписи.

<ErrorCode><ErrorDescription><ResponseBody.Amount><ResponseBody.Date><ResponseBody.ExternalId><ResponseBody.Info><ResponseBody.Id>

Параметр WillCallback не участвует в подписи. В случае если какой-либо из параметров отсутствует – он также не участвует в подписи. Пример:

```
<ServiceResponse
  xmlns="http://ruru.service.provider"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
>
  <ErrorCode>0</ErrorCode>
  <ErrorDescription>Успех</ErrorDescription>
  <Signature>DjnQ8jSocJmiSd0Nyfiq9QerjFE=</Signature>
  <WillCallback>>false</WillCallback>
  <ResponseBody>
    <Amount>50000</Amount>
    <Date>2012-03-02 22:12:12</Date>
    <ExternalId>12345</ExternalId>
    <Info>за телефон</Info>
    <Id>6758</Id>
  </ResponseBody>
</ServiceResponse>
```

Строка для подписи будет выглядеть таким образом:

0Успех500002012-03-02 22:12:1212345за телефон6758

4.3. Коды ошибок

Код ошибки	Описание
0	Успех
30	Истёк таймаут на резервирование/покупку - (услуга не оказана)
100	Клиент существует, но не является допустимым для платежа
101	Информация в поле запроса <i>account</i> не соответствует идентификатору клиента или отсутствует
102	Отсутствует или неправильный идентификатор запроса ПС (поле запроса <i>id</i>)
103	Отсутствует или неправильный идентификатор запроса, полученный от ТСП (поле запроса <i>externalId</i>)
104	Поле даты отсутствует или имеет неправильный формат
200	Сработали фрод-ограничения на стороне ТСП
201	Сумма платежа ниже минимально допустимой
202	Сумма платежа выше максимально допустимой
203	Неверная сумма платежа
204	Товар отсутствует в наличии
300	Ошибка в одном из дополнительных параметров. В поле <i>ErrorDescription</i> содержится описание, в каком именно.
400	Нарушение ожидаемой логики вызовов. Например, вызов <i>payment</i> без <i>init</i> , если иное не предусмотрено в данной услуге
500	Запрос в параметре <i>action</i> не распознан
600	Ошибка ЭЦП
999	Внутренняя ошибка (ошибка ТСП не попадающая ни под одну из перечисленных выше)
>2000	Ошибки, которые может специфицировать сам ТСП

По мере развития интерфейса коды ошибок могут расширяться.

4.4. Запросы от ПС к ТСП

4.4.1. Проверка параметров/Резервирование услуги (Init)

Если для Услуги указана необходимость выполнения первого шага (инициализация / резервирование), перед запросом на платёж ПС отправляет ТСП запрос следующего вида:

```
https://<базовый_адрес>/?action=init&id={id}&externalId={externalId}&code={code}&account={account}&amount={amount}&date={date}&params={parameters}&signature={signature}
```

Параметр `account` может быть пустым или отсутствовать в случае, если данная услуга является безадресной.

Параметр `params` может быть пустым или отсутствовать в случае, если данная услуга не требует дополнительных параметров.

Параметр `externalId` может отсутствовать, но не может быть пустым!

В ответ ТСП возвращает структуру **ServiceResponse**, содержащую секцию **ResponseBody** в формате:

```
<ServiceResponse
  xmlns="http://ruru.service.provider"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
>
  <ErrorCode></ErrorCode>
  <ErrorDescription></ErrorDescription>
  <WillCallback></WillCallback>
  <Signature></Signature>
  <ResponseBody>
    <Amount></Amount>
    <Date></Date>
    <ExternalId></ExternalId>
    <Info></Info>
    <Id></Id>
  </ResponseBody>
</ServiceResponse>
```

Если операция завершилась успешно, значение секции **ErrorCode** равно 0. Если результат операции будет выслан по клиентскому протоколу, значение секции **WillCallback** равно true, а тело может отсутствовать.

Элементы секции **ResponseBody**

Имя элемента	Описание	Формат	Обязательный
Date	Дата регистрации запроса в ТСП	yyyy-MM-dd HH:mm:ss	Нет
ExternalId	Идентификатор запроса в ТСП	Строка 16 символов	Нет
Info	Текст от ТСП, который в случае если запрос успешный будет (по желанию ТСП) передан клиенту	Строка до 128 символов	Нет

Имя элемента	Описание	Формат	Обязательный
Id	Идентификатор запроса в ПС	Строка	Да
Amount	Сумма в копейках от ТСП	Целое число	Нет

4.4.2. Покупка товара (Payment)

Если резервирование услуги прошло успешно и от клиента получено подтверждение на оплату, либо, когда резервирование не требовалось, ПС отправляет ТСП запрос на предоставление Услуги в формате

```
https://<базовый_адрес>/?action=payment&id={id}&externalId={externalId}&code={code}&account={account}&amount={amount}&date={date}&params={parameters}&signature={signature}
```

Параметр `account` может быть пустым или отсутствовать в случае, если данная услуга является безадресной.

Параметр `params` может быть пустым или отсутствовать в случае, если данная услуга не требует дополнительных параметров.

ТСП отвечает сообщением

```
<ServiceResponse
  xmlns="http://ruru.service.provider"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
>
  <ErrorCode></ErrorCode>
  <ErrorDescription></ErrorDescription>
  <Signature></Signature>
  <WillCallback></WillCallback>
  <ResponseBody>
    <Amount></Amount>
    <Date></Date>
    <ExternalId></ExternalId>
    <Info></Info>
    <Id></Id>
  </ResponseBody>
</ServiceResponse>
```

Формат ответного сообщения от ТСП описан в **Элементы** секции **ResponseBody**. Если результат операции будет выслан по клиентскому протоколу, значение секции **WillCallback** равно true, а тело может отсутствовать.

4.4.3. Отмена резервирования (CancelInit)

В случае если резервирование средств /списание с клиента по каким-либо причинам не состоялось, ПС отправляет ТСП запрос следующего формата

```
https://<базовый_адрес>/?action=cancelinit&id={id}&externalId={externalId}&reason={reason}&signature={signature}
```

ТСП отвечает сообщением

```
<ServiceResponse
```

```

xmlns="http://ruru.service.provider"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
>
  <ErrorCode></ErrorCode>
  <ErrorDescription></ErrorDescription>
  <Signature></Signature>
  <ResponseBody>
    <Date></Date>
    <ExternalId></ExternalId>
    <Id></Id>
  </ResponseBody>
</ServiceResponse>

```

Формат ответного сообщения от ТСП описан в **Элементы секции ResponseBody**

5. Спецификация серверной части протокола

5.1. Транспортный уровень

Информационное взаимодействие осуществляется с использованием протокола безопасной передачи данных HTTPS (протокол прикладного уровня HTTP с использованием TLS). Аутентификация и авторизация происходит путём предъявления цифрового сертификата формата X.509 v3. Сертификаты предъявляются и клиентской (ТСП) и серверной стороной (ПС).

WSDL-описание интерфейса ТСП находится в приложении А. ПС предоставляет сервис, реализующий протокол, по адресу:

https://<базовый адрес сервиса>/RuRu.FrontEnd.ServiceProvider/TransactionService.svc

5.2. Формирование ЭЦП

Подписываемое сообщение является строкой в формате UTF-8, в которой по умолчанию перечислены значения всех передаваемых параметров². В некоторых методах для подписи используются не все параметры, в этом случае необходимые параметры перечисляются в описании метода. В настоящий момент поддерживается тип подписи HMACSHA1. **Порядок параметров в теле подписи такой же, как и в таблицах с описаниями параметров в соответствующих методах, если в описании не оговорено иное.** В HTTP GET запросах значение подписи передаётся в формате url-encoded.

5.3. Коды ошибок

5.3.1. ErrorCode

Код ошибки	Описание
0	Успех
1	Внутренняя ошибка
2	Невозможно определить оператора по номеру телефона плательщика
3	Некорректный формат даты
17	Неверный канал инициирования покупки

² Здесь и далее подразумеваются все параметры, кроме параметра signature, предназначенного для передачи значения подписи

Код ошибки	Описание
18	Услуга не найдена или не активна
20	Ошибка определения маршрута оплаты услуги или комиссии с неё по источнику средств
21	Доступный маршрут оплаты услуги не найден
22	Не указан номер телефона плательщика
24	Ошибочный формат цены
27	Ошибка доступа ТСП (проверьте клиентский сертификат)
28	Нет прав на инициацию покупки
29	ТСП не найден или не активен
30	Таймаут при обработке транзакции (истекло время на покупку или резервирование)
31	Транзакция принадлежит другому пользователю
32	Транзакция не найдена
35	Текущий статус транзакции не позволяет выполнить операцию
36	У плательщика отсутствует номер счёта в источнике средств
37	Пользователь (плательщик) не найден
38	Сумма покупки меньше минимально допустимой
39	Сумма покупки больше максимально допустимой
60	Транзакция обрабатывается
61	Транзакция завершена с ошибкой
600	Ошибка ЭЦП

5.3.2. Reason

Список ошибок приведён ниже и может расширяться.

Reason	Описание
0	Отказ клиента от услуги
1	Неустановленная ошибка
2	Нет средств
3	Неверный номер счёта/суммы платежа
4	Превышение лимитов платежа
5	Таймаут
6	Отказ источника в предоставлении услуги
7	Срок действия банковской карты истёк
8	Банковскую карту нельзя использовать в сети Интернет
9	Сервис отключён для данного номера телефона
10	Предыдущий платёж ещё не был обработан

5.4. Запросы

ТСП может отправлять в ПС следующие запросы и подтверждения:

- 1) Запрос на инициацию покупки.
- 2) Подтверждение резервирования.
- 3) Подтверждение покупки.
- 4) Получение статуса покупки

5.4.1. Инициация покупки (Init)

ТСП может инициировать покупку от имени клиента вызовом метода Init.

5.4.1.1. Входные параметры

Имя элемента	Описание	Формат	Обязательный
spTrnId	Идентификатор платежа в ТСП	Строка	Нет
spTrnDate	Дата платежа в ТСП	Строка Формат: YYYY-MM-dd HH:mm:ss	Нет
phone	Номер телефона в источнике средств	Строка 10 цифр 9xxxxxxxxx	Да
info	Информация, отправляемая клиенту в момент подтверждения списания	Строка	Нет
spAccount	Номер счёта клиента в ТСП	Строка	Нет
productId	Идентификатор услуги или товара в ПС	Целое число	Да
amount	Сумма покупки в копейках	Целое число	Нет
spld	Идентификатор ТСП	Целое число	Да
parameters	Дополнительные параметры	Parameter[]	Нет
trnId	Идентификатор транзакции, полученный во время предварительной инициализации покупки	long	Нет
signature	ЭЦП ТСП	Строка Base64	Только при взаимодействии по ЭЦП

Примечание: Для подписи должны использоваться все параметры, кроме **parameters**.

Описание дополнительных параметров (parameters) во входных параметрах

Имя элемента	Описание	Формат	Обязательный
Links_CreateLink	Признак "Создать привязку номера телефона пользователя к услуге": 1 - создать; 0 - не создавать.	Целое	Нет
Links_UserId	Идентификатор пользователя в системе партнёра (логин, телефон, адрес электронной почты, номер банковской карты, и пр.)	Строка	Нет
Cheque_Email	Адрес электронной почты, на который клиенту должна быть отправлена информация об электронных чеках, в случае их создания. Если адрес не указан, информация об электронных чеках будет отправлена в СМС-сообщении на номер телефона из запроса.	Строка	Нет

Попытка создания привязки будет создана только если указаны оба дополнительных параметра (Links_CreateLink и Links_UserId) и в Links_CreateLink передано значение 1.

Электронный чек должен быть создан и отправлен клиенту в одном из следующих случаев:

- с клиента берётся комиссия в пользу ПС;
- у ПС с поставщиком услуг заключён договор, по которому электронными чеками вместо поставщика услуг занимается ПС.

Пример запроса:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ser="http://ruru.ru/serviceprovider/">
  <soapenv:Header/>
  <soapenv:Body>
    <ser:Init>
      <ser:spTrnId nil="true"/>
      <ser:spTrnDate>2013-06-11 13:17:04</ser:spTrnDate>
      <ser:phone>9069060906</ser:phone>
      <ser:info></ser:info>
      <ser:spAccount>9069060609</ser:spAccount>
      <ser:productId>919</ser:productId>
      <ser:amount>1000</ser:amount>
      <ser:spId>1036</ser:spId>
      <ser:signature>LqHzbt4FP99Cx6cb1720idRG2KI=</ser:signature>
    </ser:Init>
  </soapenv:Body>
</soapenv:Envelope>
```

Примечание. Порядок следования параметров в запросе имеет значение. Параметры должны следовать в порядке, описанном в таблице (Входные параметры метода Init).

5.4.1.2. Выходные параметры

Имя элемента	Описание	Формат	Обязательный
Id	Идентификатор транзакции в ПС	Целое число	Да
ErrorCode	Код ошибки, либо 0 в случае успеха	Целое число	Да
Info	Дополнительная информация	Строка	Да
TimeOut1	Таймаут резервирования	Целое число, минуты	Да
TimeOut2	Таймаут покупки	Целое число, минуты	Да
Signature	ЭЦП ПС	Строка Base64	Только при взаимодействии по ЭЦП

В случае ошибки поле идентификатора платежа может принимать значение -1.

Пример ответа:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <InitResponse xmlns="http://ruru.ru/serviceprovider/">
      <InitResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <ErrorCode>1</ErrorCode>
        <Id>3115858</Id>
        <Info>Внутренняя ошибка</Info>
        <Signature>Crb8c0gZLo8XSQi7LJWmQ9MfLMs=</Signature>
        <TimeOut1>30</TimeOut1>
        <TimeOut2>30</TimeOut2>
      </InitResult>
    </InitResponse>
  </s:Body>
</s:Envelope>
```



```
</s:Body>
</s:Envelope>
```

5.4.2. Подтверждение резервирования (ReserveCallback)

При асинхронном взаимодействии ТСП должен подтвердить резервирование покупки вызовом метода ReserveCallback. Выходные параметры аналогичны параметрам метода Init.

5.4.2.1. Входные параметры

Имя элемента	Описание	Формат	Обязательный
trnId	Идентификатор платежа в ПС	Целое число	Да
spTrnId	Идентификатор платежа в ТСП	Строка	Нет
spTrnDate	Дата платежа в ТСП	Строка Формат: YYYY-MM-dd HH:mm:ss	Нет
info	Информация, отправляемая клиенту (в момент подтверждения списания)	Строка	Нет
amount	Сумма покупки в копейках	Целое число	Нет
spId	Идентификатор ТСП	Целое число	Обязательный если используется ЭЦП
result	Код ошибки в ТСП	Целое число, либо 0 в случае успеха	Да
errorDescription	Описание ошибки	Строка	Да
signature	ЭЦП ТСП	Строка Base64	Только при взаимодействии по ЭЦП

Пример запроса:

```
<soapenv:Envelope xmlns:ser="http://ruru.ru/serviceprovider/" xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <ser:ReserveCallback>
      <!--Optional:-->
      <ser:trnId>5060</ser:trnId>
      <ser:spTrnId>12345</ser:spTrnId>
      <ser:spId>1090</ser:spId>
      <ser:result>0</ser:result>
    </ser:ReserveCallback>
  </soapenv:Body>
</soapenv:Envelope>
```

Примечание. Порядок следования параметров в запросе имеет значение. Параметры должны следовать в порядке, описанном в таблице (Входные параметры метода ReserveCallback).

Пример ответа:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <ReserveCallbackResponse xmlns="http://ruru.ru/serviceprovider/">
      <ReserveCallbackResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <ErrorCode>32</ErrorCode>
      </ReserveCallbackResult>
    </ReserveCallbackResponse>
  </s:Body>
</s:Envelope>
```

```

        <Id>5060</Id>
        <Info>Транзакция с Id=5060 отсутствует в оперативных данных</Info>
        <Signature i:nil="true"/>
        <TimeOut1>0</TimeOut1>
        <TimeOut2>0</TimeOut2>
    </ReserveCallbackResult>
</ReserveCallbackResponse>
</s:Body>
</s:Envelope>

```

5.4.3. Подтверждение покупки (PurchaseCallback)

При асинхронном взаимодействии ТСП должен подтвердить покупку вызовом метода PurchaseCallback. Входные и выходные параметры аналогичны параметрам метода ReserveCallback.

Пример запроса:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ser="http://ruru.ru/serviceprovider/">
  <soapenv:Header/>
  <soapenv:Body>
    <ser:PurchaseCallback>
      <ser:trnId>305656898</ser:trnId>
      <ser:spTrnId>12345</ser:spTrnId>
      <ser:spId>1090</ser:spId>
      <ser:result>0</ser:result>
    </ser:PurchaseCallback>
  </soapenv:Body>
</soapenv:Envelope>

```

Пример ответа:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <PurchaseCallbackResponse xmlns="http://ruru.ru/serviceprovider/">
      <PurchaseCallbackResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <ErrorCode>0</ErrorCode>
        <Id>305656898</Id>
        <Info></Info>
        <Signature i:nil="true"/>
        <TimeOut1>0</TimeOut1>
        <TimeOut2>0</TimeOut2>
      </PurchaseCallbackResult>
    </PurchaseCallbackResponse>
  </s:Body>
</s:Envelope>

```

5.4.4. Получение статуса покупки (GetTransactionStatus)

В случае, если статус покупки при асинхронном взаимодействии ТСП не получен, ТСП может вызвать метод GetTransactionStatus для проверки текущего статуса покупки.

5.4.4.1. Входные параметры

Имя элемента	Описание	Формат	Обязательный
trnId	Идентификатор транзакции в RURU	long	Да
spId	Идентификатор ТСП	Целое число	Да
signature	ЭЦП ТСП	Строка Base64	Только при взаимодействии по ЭЦП

5.4.4.2. Выходные параметры

Имя элемента	Описание	Формат	Обязательный
TransactionId	Идентификатор транзакции в RURU	long	Да
ErrorCode	Код ошибки, либо 0 в случае успеха	Целое число	Да
Reason	Причина ошибки в случае неудачи	Целое число	Да
Description	Описание ошибки	Строка	Да
Signature	ЭЦП ПС	Строка Base64	Только при взаимодействии по ЭЦП

В случае ошибки поле идентификатора платежа может принимать значение -1.

Пример запроса:

```
<soapenv:Envelope xmlns:ser="http://ruru.ru/serviceprovider/" xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <ser:GetTransactionStatus>
      <ser:trnId>305656898</ser:trnId>
      <ser:spId>10</ser:spId>
      <ser:signature></ser:signature>
    </ser:GetTransactionStatus>
  </soapenv:Body>
</soapenv:Envelope>
```

Пример ответа:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetTransactionStatusResponse xmlns="http://ruru.ru/serviceprovider/">
      <GetTransactionStatusResult
        xmlns:a="http://schemas.datacontract.org/2004/07/RuRu.FrontEnd.ServiceProvider.Contract"
        xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Description>SignatureError</a:Description>
        <a:ErrorCode>600</a:ErrorCode>
        <a:Reason>-1</a:Reason>
        <a:Signature i:nil="true"/>
        <a:TransactionId>-1</a:TransactionId>
      </GetTransactionStatusResult>
    </GetTransactionStatusResponse>
  </s:Body>
</s:Envelope>
```

5.4.5. Отмена покупки (Cancel)

При невозможности выдать Услугу ТСП должен инициировать отмену покупки вызовом метода Cancel. Входные и выходные параметры аналогичны параметрам метода PurchaseCallback.

Пример запроса:

```
<soapenv:Envelope xmlns:ser="http://ruru.ru/serviceprovider/" xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <ser:Cancel>
      <ser:trnId>305656898</ser:trnId>
      <ser:spTrnId>12345</ser:spTrnId>
      <ser:result>0</ser:result>
      <ser:signature></ser:signature>
    </ser:Cancel>
  </soapenv:Body>
</soapenv:Envelope>
```

Пример ответа:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <CancelResponse xmlns="http://ruru.ru/serviceprovider/">
      <CancelResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <ErrorCode>0</ErrorCode>
        <Id>305656898</Id>
        <Signature i:nil="true"/>
      </CancelResult>
    </CancelResponse>
  </s:Body>
</s:Envelope>

```

5.4.6. Удаление привязки (CancelLink)

5.4.6.1. Входные параметры

Имя элемента	Описание	Формат	Обязательный
spId	Идентификатор ТСП	Целое число	Да
linkId	Идентификатор привязки	long	Да
signature	ЭЦП ТСП	Строка Base64	Только при взаимодействии по ЭЦП

5.4.6.2. Выходные параметры

Имя элемента	Описание	Формат	Обязательный
ErrorCode	1 в случае ошибки, либо 0 в случае успеха	Целое число	Да
Description	Описание ошибки	Строка	Да
Signature	ЭЦП ПС	Строка Base64	Только при взаимодействии по ЭЦП

Пример запроса:

```

<soapenv:Envelope xmlns:ser="http://ruru.ru/serviceprovider/" xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <ser:CancelLink>
      <ser:spId>5060</ser:spId>
      <ser:linkId>12345</ser:linkId>
      <ser:signature/>
    </ser:CancelLink>
  </soapenv:Body>
</soapenv:Envelope>

```

Пример ответа:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <CancelLink xmlns="http://ruru.ru/serviceprovider/">
      <CancelLinkResponse xmlns:a="http://schemas.datacontract.org/2004/07/RuRu.FrontEnd.ServiceProvider.Contract" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:ErrorCode>0</a:ErrorCode>
        <a:Description/>
        <a:Signature i:nil="true"/>
      </CancelLinkResponse>
    </CancelLink>
  </s:Body>
</s:Envelope>

```

```
</CancelLinkResponse>  
</CancelLink>  
</s:Body>  
</s:Envelope>
```

5.5. Структуры

5.5.1. Parameter

Имя элемента	Описание	Формат	Обязательный
Name	Имя параметра	Строка	Да
Value	Значение параметра	Строка	Да

6. Стандартные сценарии интеграции

В данном разделе описываются типовые сценарии интеграции поставщика услуг и RuRu. Для каждого поставщика услуг определяется наиболее подходящий сценарий. В зависимости от сценария интеграции могут быть задействованы либо обе части протокола RuRuPay, либо лишь одна из них. Также определяется набор реализуемых/используемых методов из каждой части. Сценарий определяется исходя из нескольких факторов, например, с чей витрины происходит инициация, должен ли в Аос присутствовать текст от ТСП, будут ли использоваться источники средств иные, чем мобильный счёт клиента и т.д.

Варианты подключения и необходимые протоколы

	Витрина RuRu	Витрина ТСП (только мобильная коммерция)	Витрина ТСП (МК и остальные источники средств)/Кнопка RuRu
Widget*	-	-	Да
Клиентская часть RuRuPay	Да	Да	Опционально
Серверная часть RuRuPay	Опционально (только для асинхронного взаимодействия)	Да	Опционально (только для асинхронного взаимодействия)

*Описание протокола widget выходит за рамки данного документа

Реализация методов клиентской части.

	Витрина RuRu	Витрина ТСП (только мобильная коммерция)	Витрина ТСП (МК и остальные источники средств)/Кнопка RuRu
init	Опционально (в случае, когда необходимо проверить параметры покупки или зарезервировать услугу)	Опционально (рекомендуется для проверки доступности шлюза ТСП до того, как списаны средства)	Опционально (в случае, когда необходимо проверить параметры покупки / зарезервировать услугу, либо передать текст для Аос*)
payment	Да	Да	Опционально (в случае, когда уведомление о статусе покупки запрашивается не через widget и не планируется включить в финальную СМС текст от ТСП)
cancelinit	Да	Да	Опционально (в случае, когда уведомление о статусе покупки запрашивается не через widget)

*текст для Аос от поставщика услуг в настоящее время можно передать только для оператора Билайн, для остальных операторов будет использоваться стандартный текст Аос для услуги настроенный оператором.

Реализация методов серверной части.

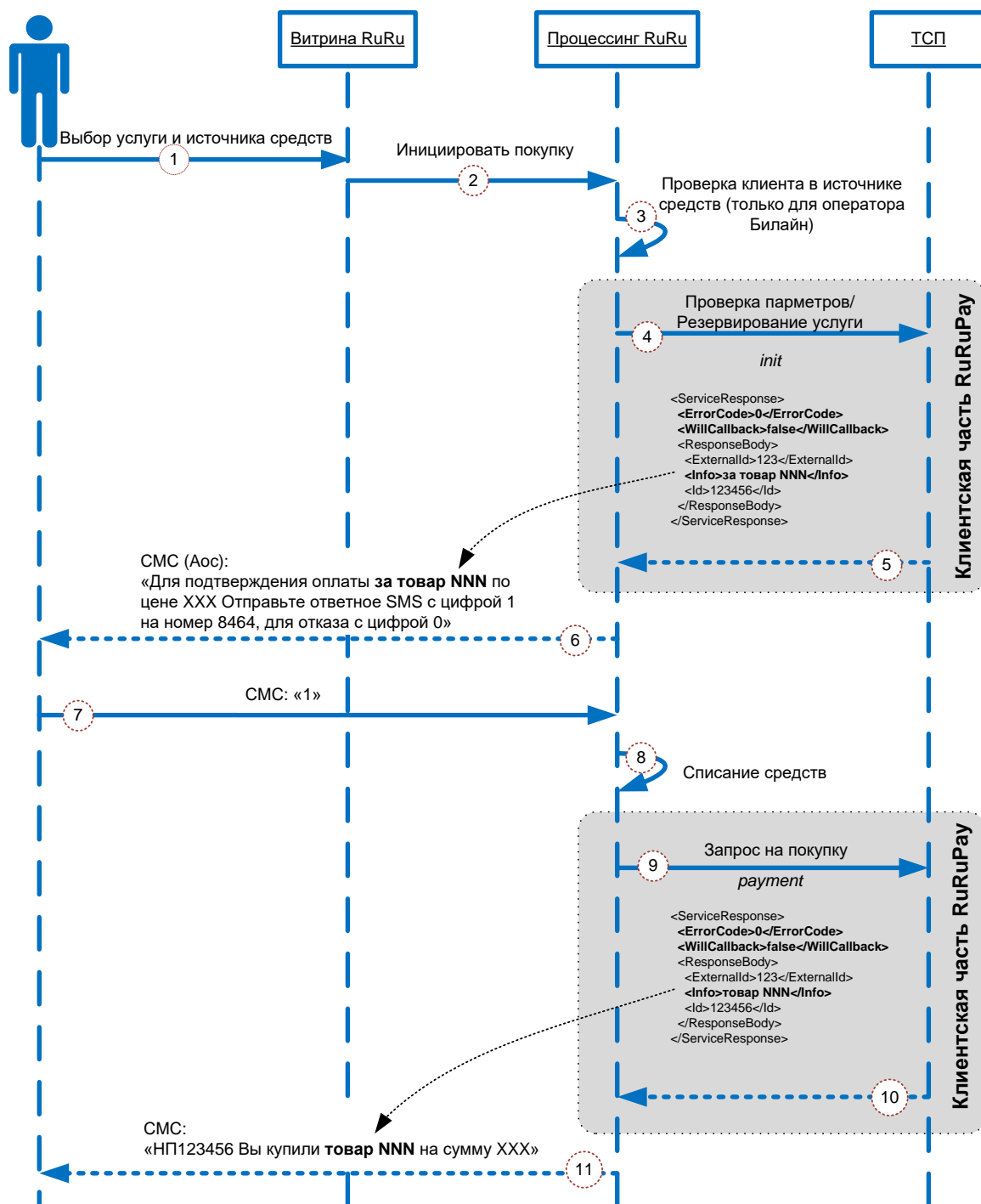
	Витрина RuRu	Витрина ТСП (только мобильная коммерция)	Витрина ТСП (МК и остальные источники средств)/Кнопка RuRu
Init	-	Да	-
ReserveCallback*	Опционально (только для асинхронного взаимодействия)	Опционально (только для асинхронного взаимодействия)	Опционально (только для асинхронного взаимодействия)
PurchaseCallback*	Опционально (только для асинхронного взаимодействия)	Опционально (только для асинхронного взаимодействия)	Опционально (только для асинхронного взаимодействия)
PurchaseCallback*	Опционально (только для асинхронного взаимодействия)	Опционально (только для асинхронного взаимодействия)	Опционально (только для асинхронного взаимодействия)
GetTransactionStatus	-	Опционально	-
Cancel	Опционально (для возврата средств на счёт клиента, зависит от Услуги и источника средств)	Опционально (для возврата средств на счёт клиента, зависит от Услуги и источника средств)	Опционально (для возврата средств на счёт клиента, зависит от Услуги и источника средств)

* методы ReserveCallback и PurchaseCallback вызываются только в случае, если при ответе на методы init/payment клиентской части протокола значение параметра WillCallback=true

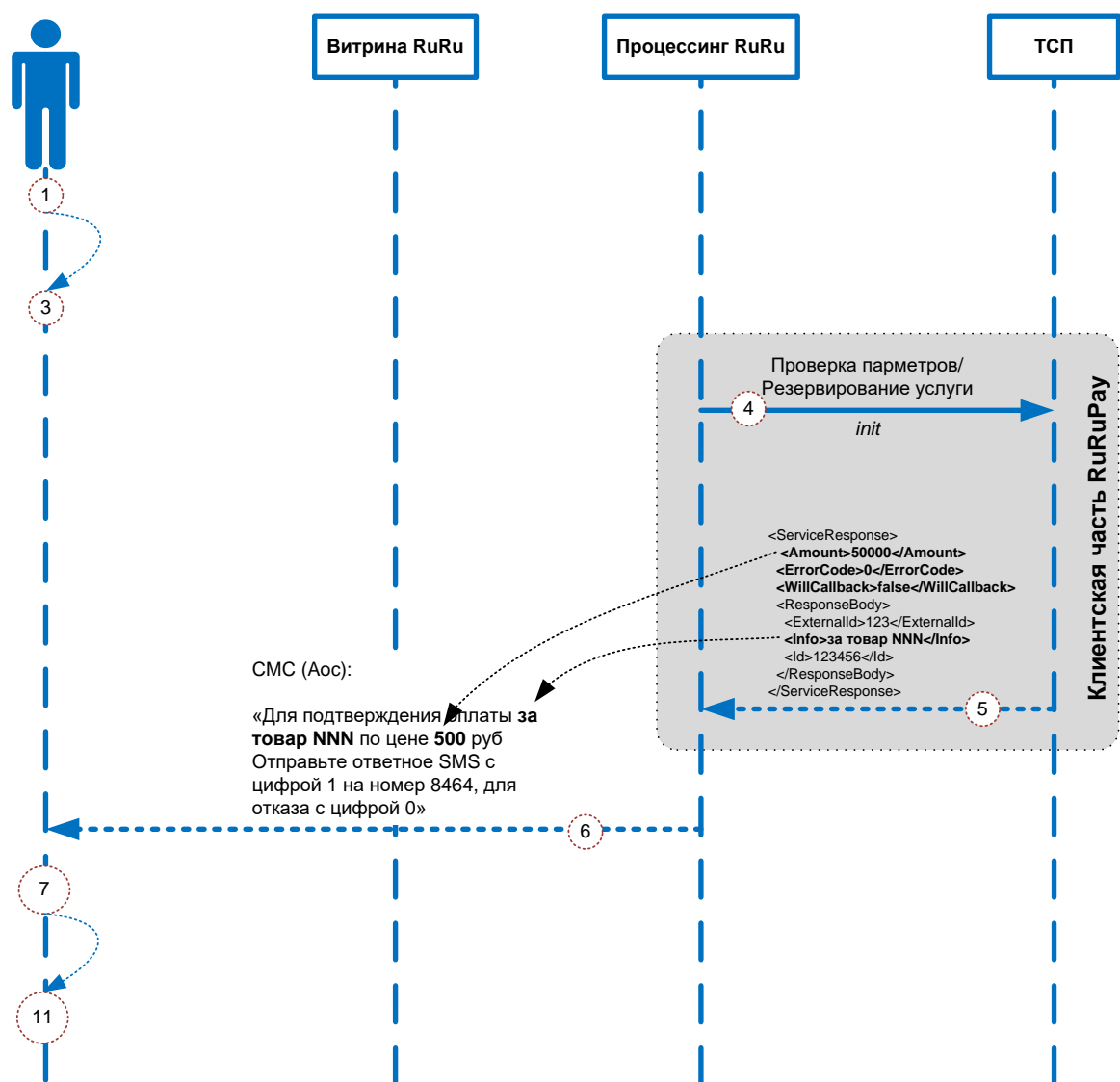
6.1. Инициация с витрины RuRu

6.1.1. Мобильная коммерция. Синхронное взаимодействие.

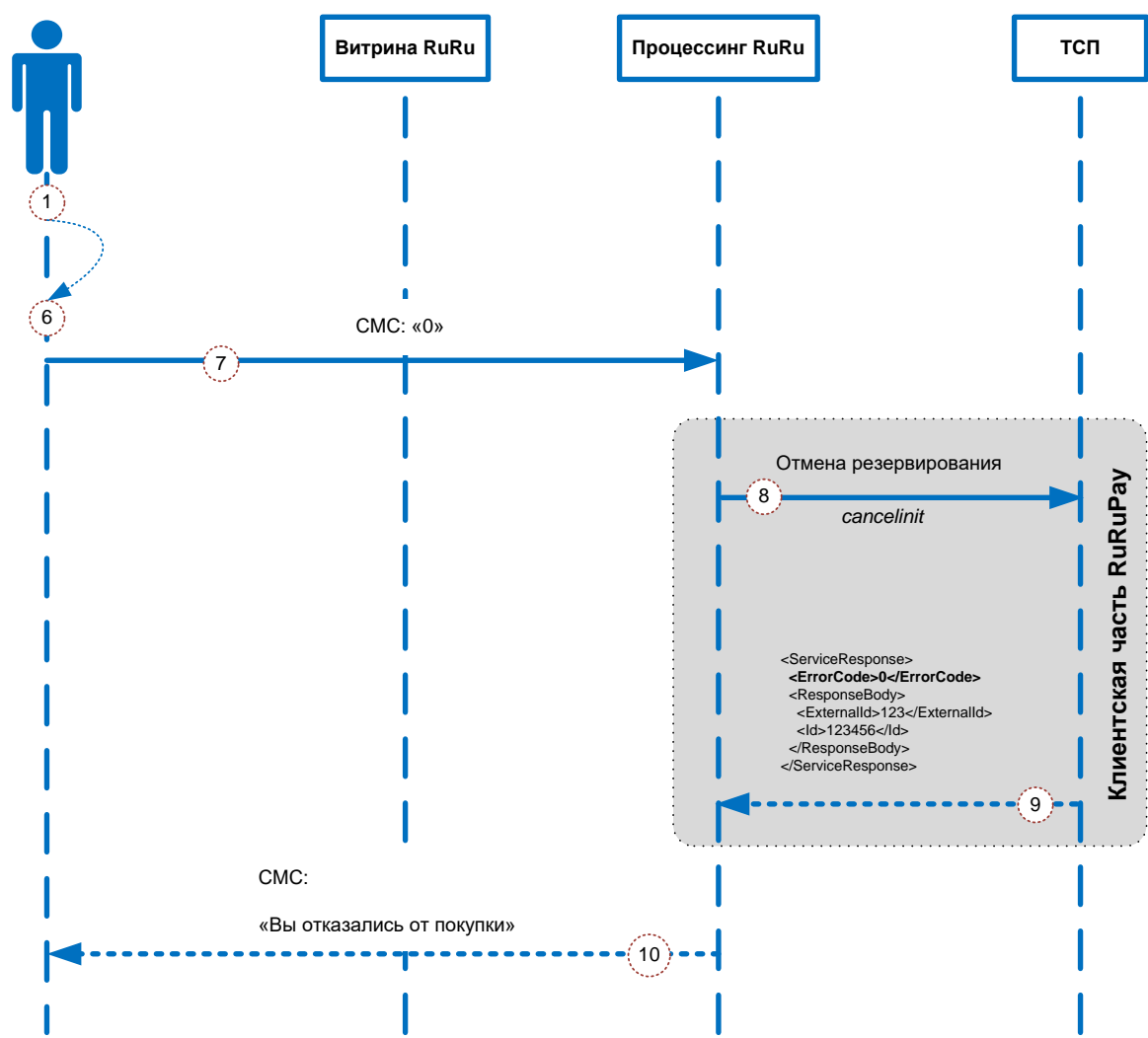
6.1.1.1. Успешный сценарий.



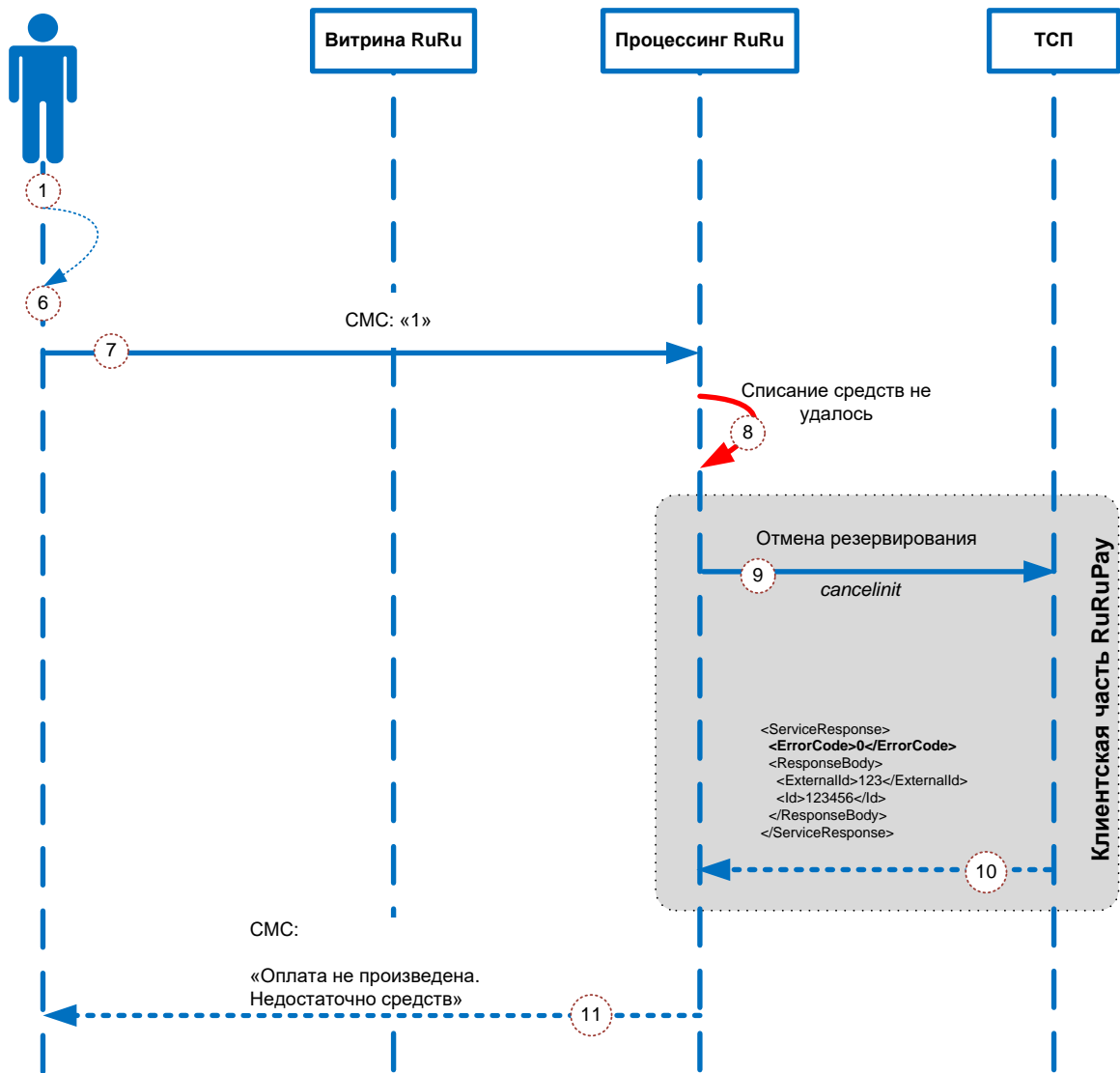
6.1.1.2. Успешный сценарий. Сумма определяется при резервировании.



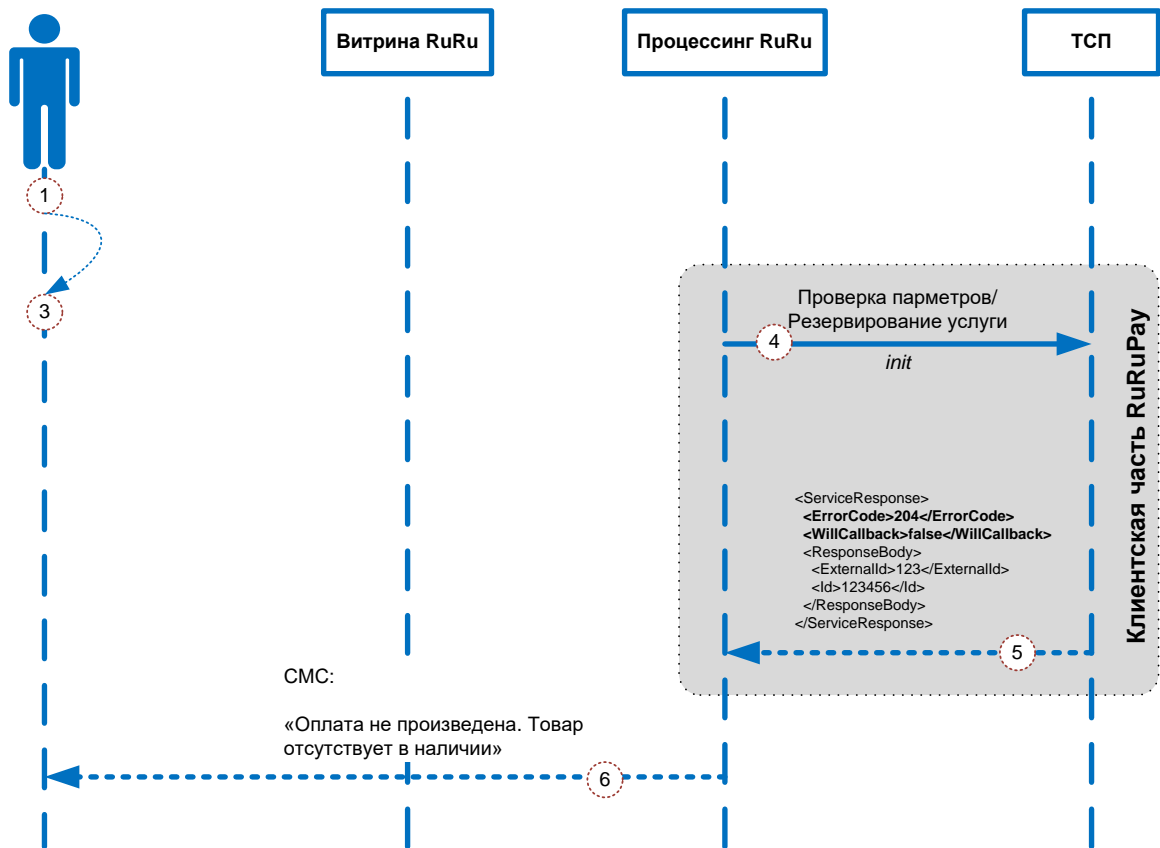
6.1.1.3. Отказ клиента от покупки.



6.1.1.4. Недостаточно средств.

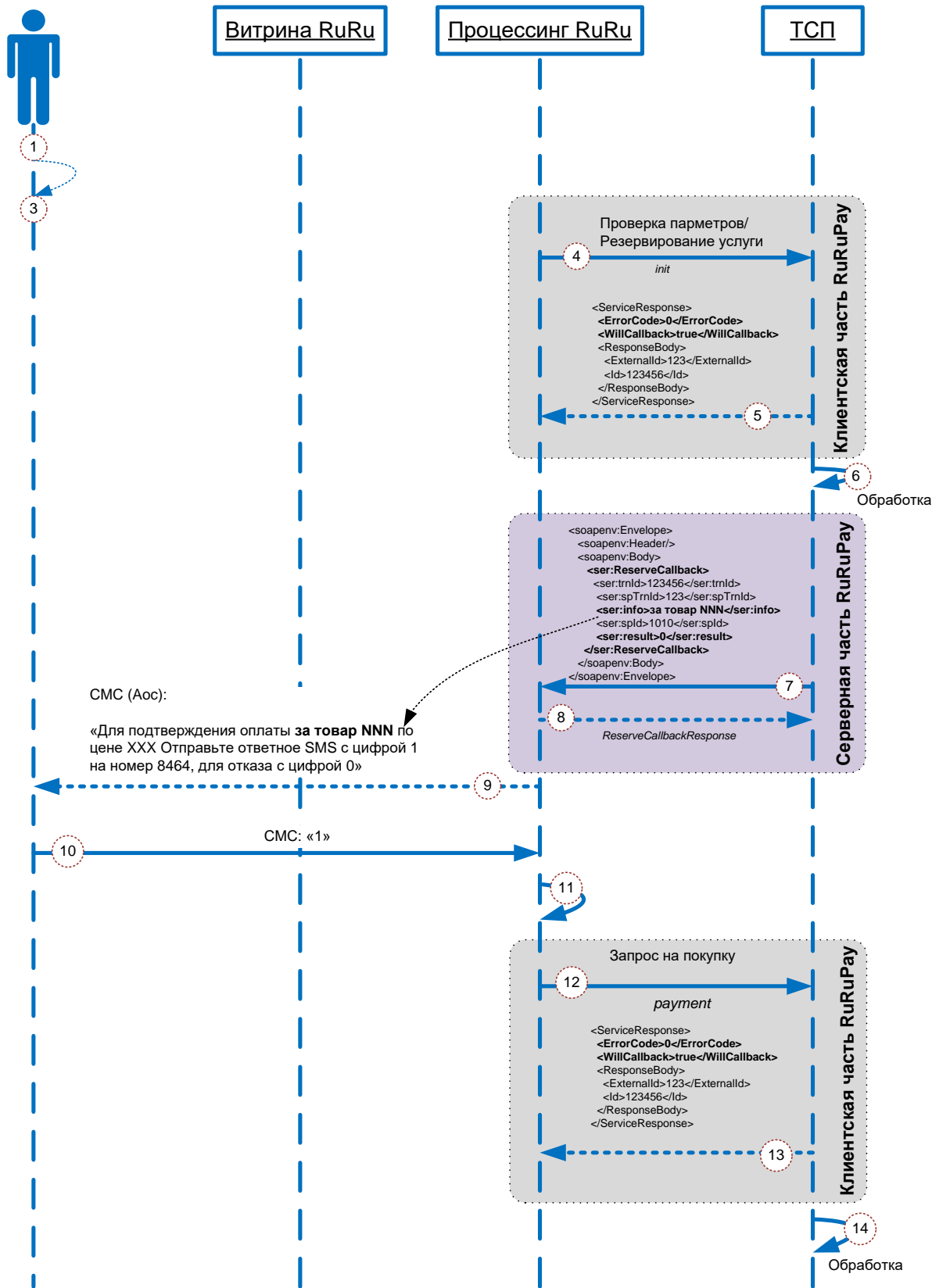


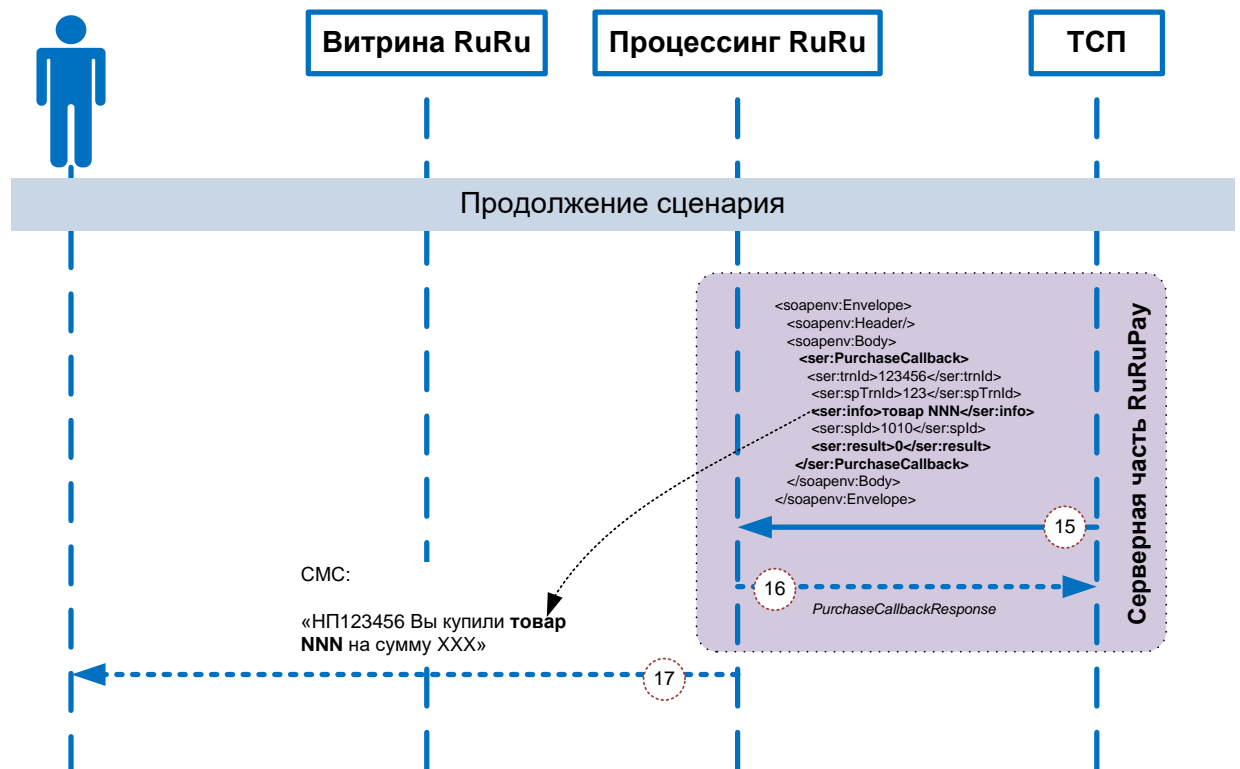
6.1.1.5. Ошибка резервирования товара.



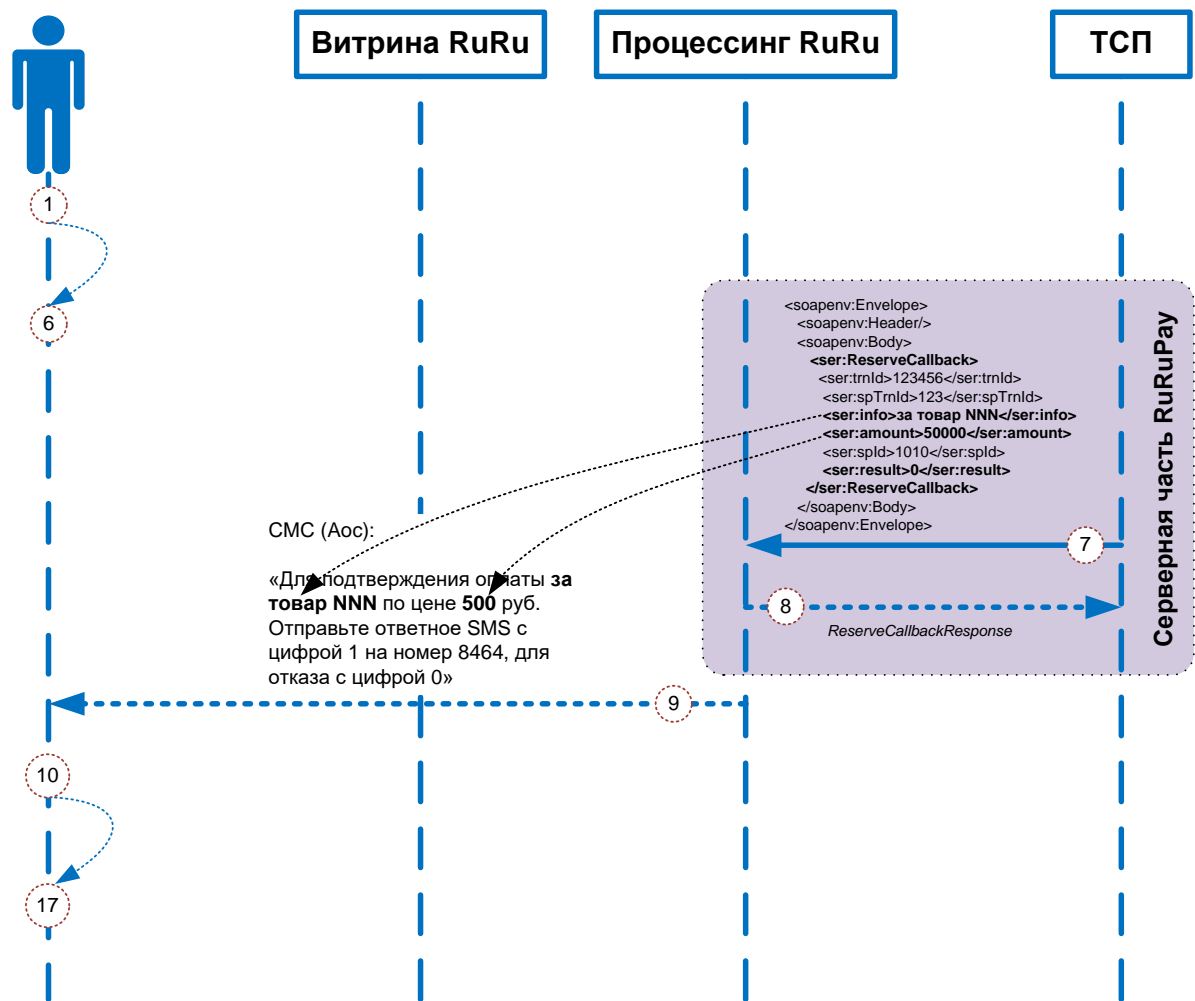
6.1.2. Мобильная коммерция. Асинхронное взаимодействие.

6.1.2.1. Успешный сценарий.

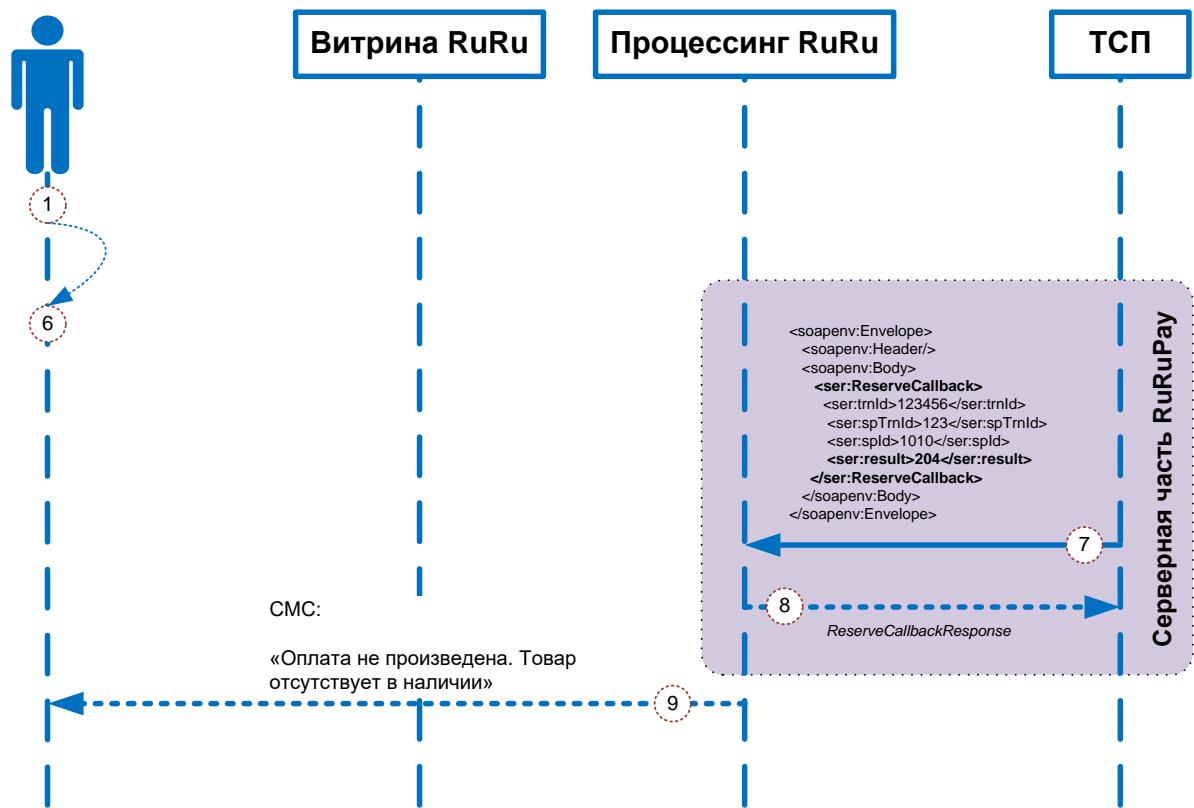




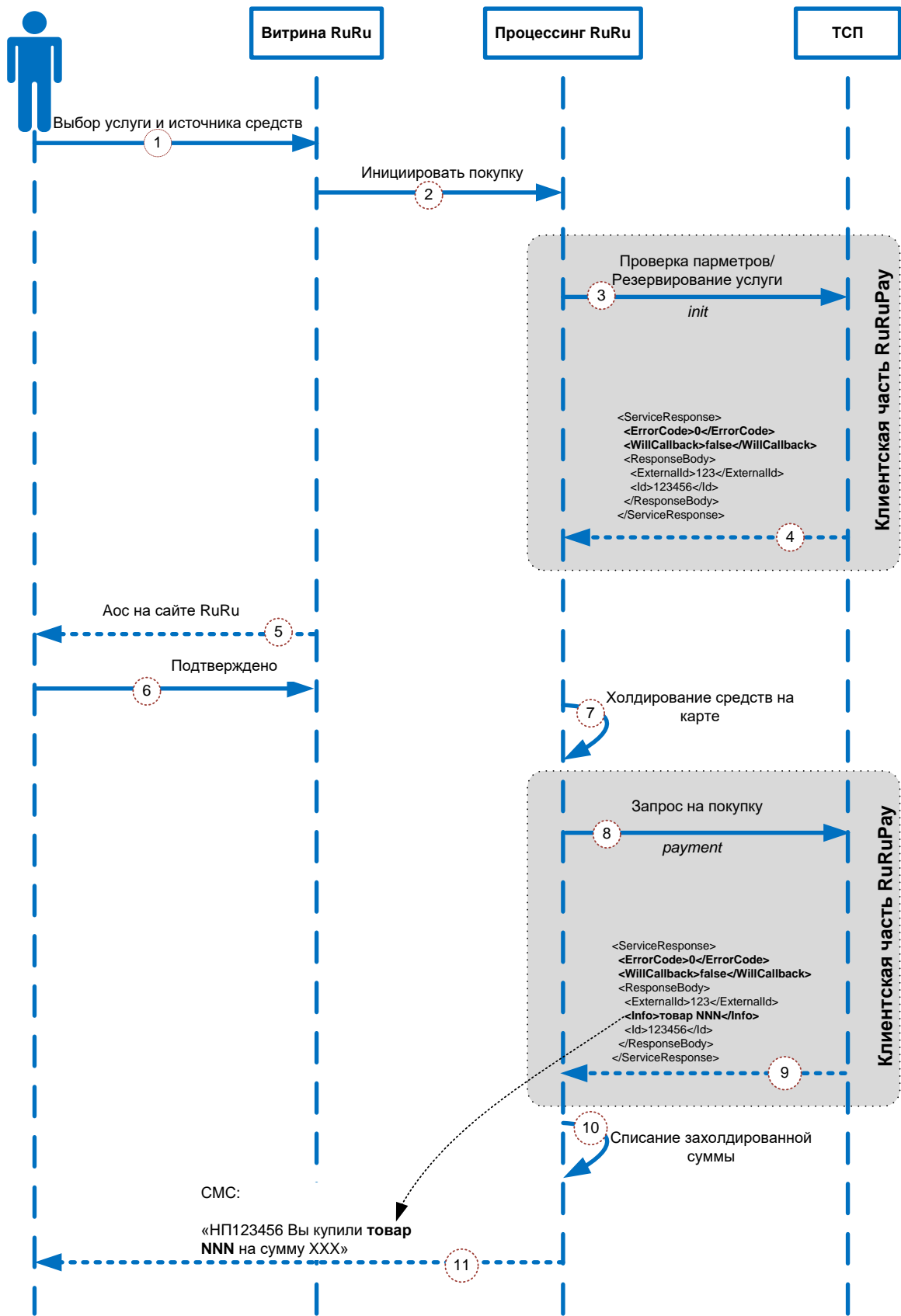
6.1.2.2. Успешный сценарий. Сумма определяется при резервировании.



6.1.2.3. Ошибка резервирования.



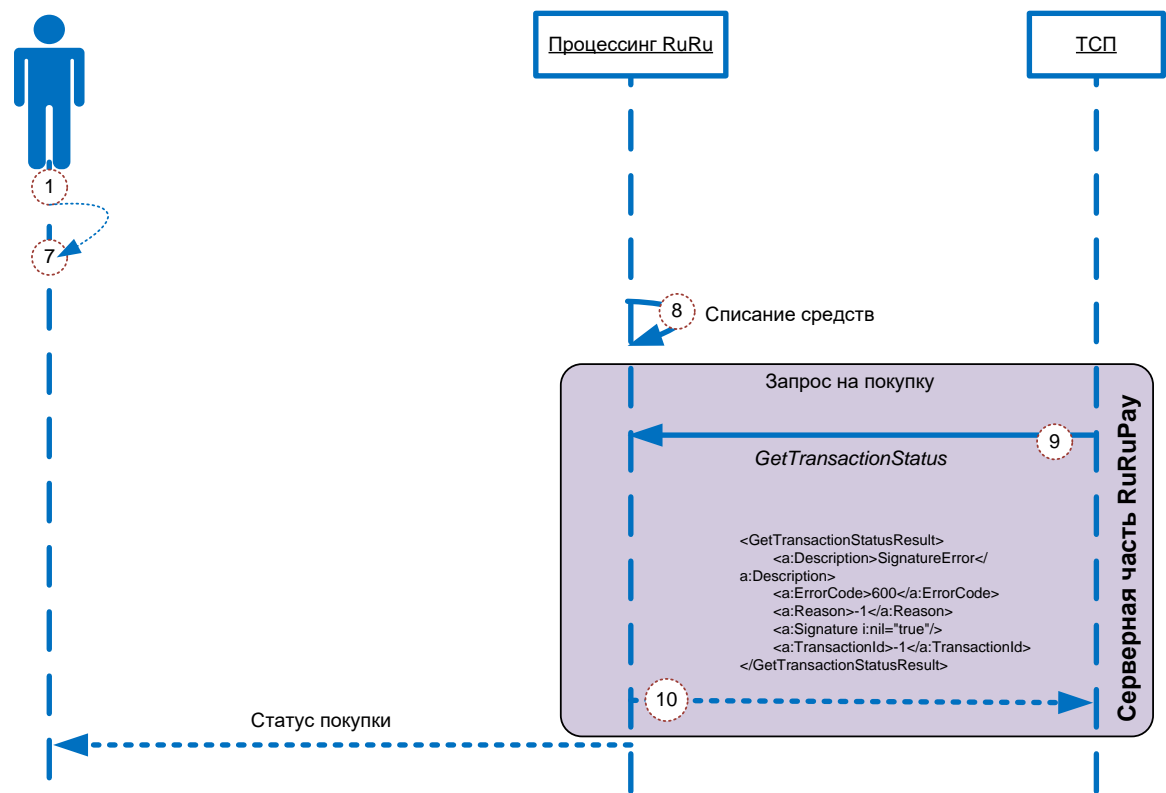
6.1.3. Банковская карта. Синхронное взаимодействие. Успешный сценарий.



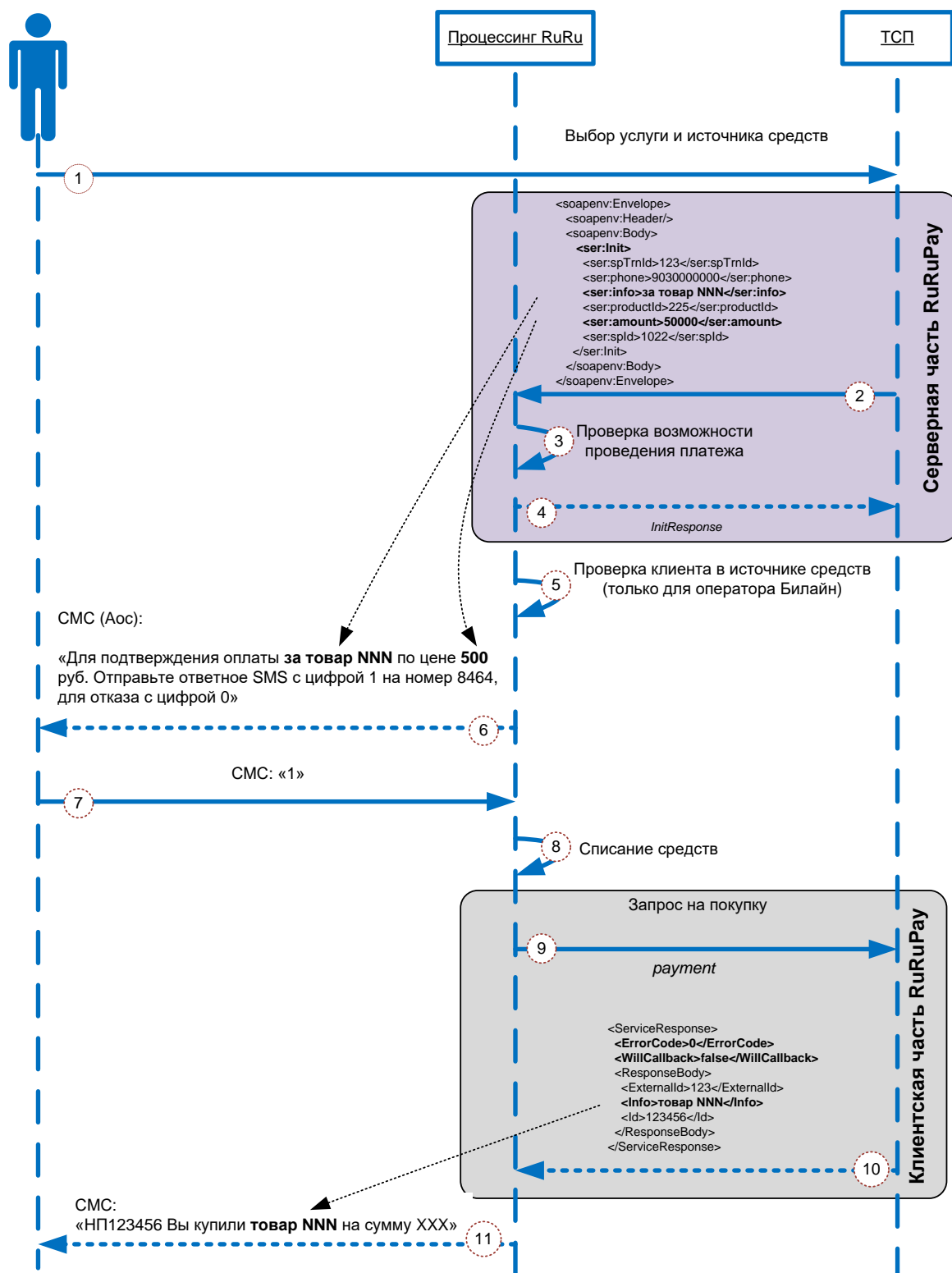
Сценарии для банковской карты отличаются от сценариев для мобильной коммерции наличием шага предварительного холдирования. Списание средств выполняется по завершении покупки.

6.2. Инициация с витрины ТСП

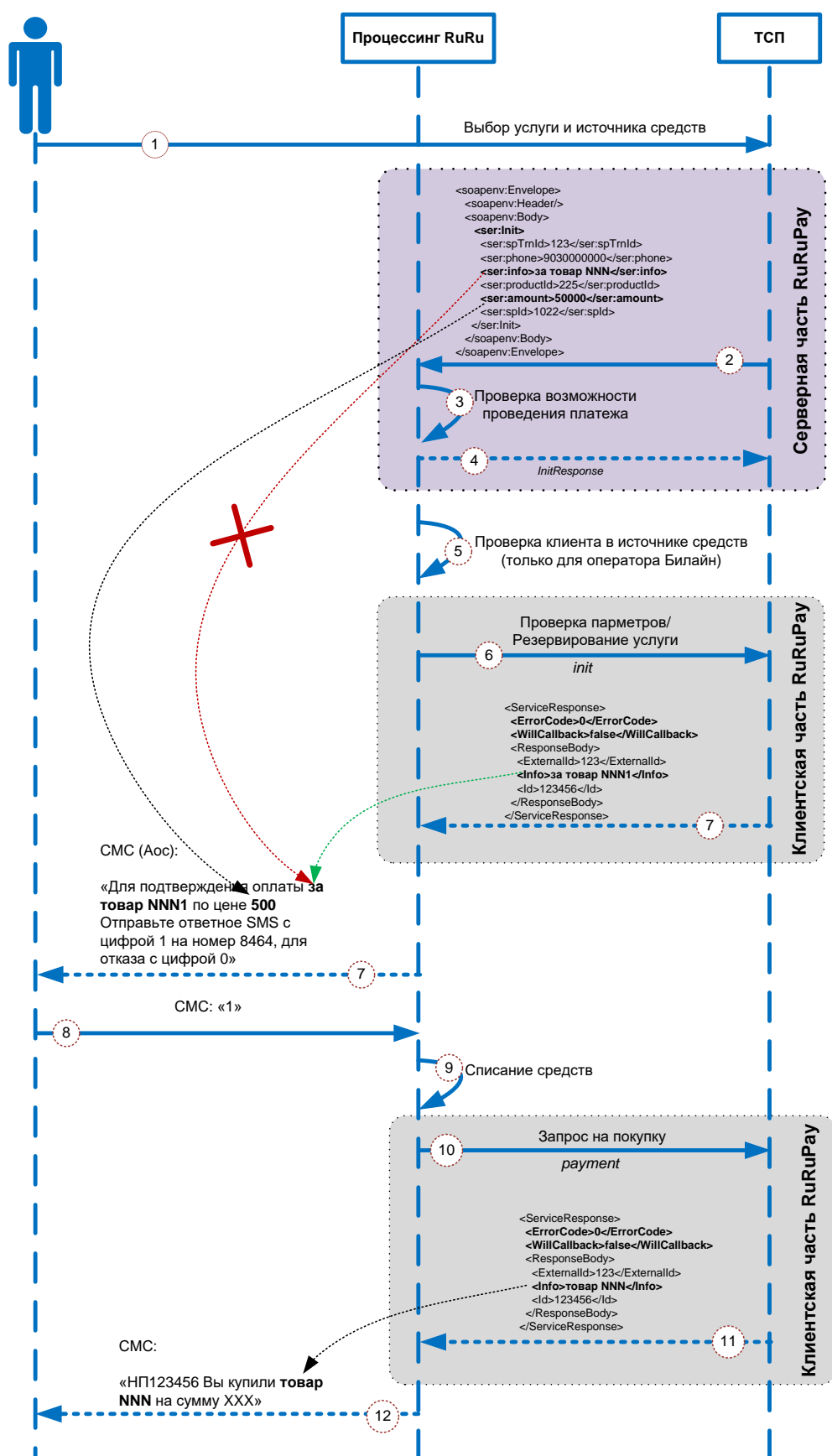
6.2.1. Запрос статуса транзакции со стороны партнёра



6.2.2. Мобильная коммерция. Синхронное взаимодействие без шага резервирования. Успешный сценарий.



6.2.3. Мобильная коммерция. Синхронное взаимодействие с резервированием. Успешный сценарий.



Сценарии инициации с витрины ТСП отличаются от сценариев инициации с витрины RuRu вызовом метода Init серверной части протокола

6.3. Инициация через Widget

Widget инициация транзакций в случае использования протокола RuRuPay, является полным аналогом инициации с витрины RuRu.

Приложение А. WSDL-описание серверной части



TransactionService.wsdl.xml

Приложение Б. Связь между клиентским и серверным протоколами

Клиентский и серверный протоколы имеют существенное различие в наименовании параметров. Ниже приведено соответствие наименования параметров между протоколами.

RuRuPay клиентский	RuRuPay серверный
externalId	spTrnId
Code	productId
Account	spAccount

Приложение В. Примеры кода

Подпись запроса на создание транзакции (PHP)

```
<?php

// WSDL для запросов SOAP
$wsdl = 'https://178.20.234.188/RuRu.FrontEnd.ServiceProvider2/TransactionService.svc?wsdl';

// Параметры для создания транзакции.
$params = array(
    "spTrnId" => "1800",
    "spTrnDate" => "2015-03-18 13:25:17",
    "phone" => "9031111111",
    "info" => "Тестовая оплата",
    "productId" => "3952",
    "amount" => "1100",
    "spId" => "1492",
    "signature" => ""
);

// Секретный ключ партнёра
$encoded_key = 'RFNGSHNyZTU3bHM0NW1qZSUkRSZTNGgzeWg=';

// Формирование ЭЦП
$params['signature'] = base64_encode(hash_hmac('sha1', implode(' ', $params),
base64_decode($encoded_key), true));

// Создание клиента
$client = new SoapClient($wsdl);

// Вызов метода
$result = $client->Init($params);
?>
```