

Описание взаимодействия

В данном разделе описывается взаимодействие микросервисов посредством сценариев использования, описания логики процессов и диаграмм последовательности. В сценариях использования и диаграммах активности опущено взаимодействие через брокер сообщений в целях упрощения изложения, однако остальные элементы включают в себя описание асинхронного взаимодействия.

- Генерация и обработка данных о звонках абонентов
 - Сценарии использования
 - Описание логики
 - Диаграммы
- Авторизация
 - Сценарии использования
 - Описание логики
 - Диаграммы
- Пополнение баланса
 - Сценарии использования
 - Описание логики
 - Диаграммы
- Смена тарифа
 - Сценарии использования
 - Описание логики
 - Диаграммы
- Создание абонента
 - Сценарии использования
 - Описание логики
 - Диаграммы

Генерация и обработка данных о звонках абонентов

В UC-1 и UC-1.1 описан основной сценарий использования системы: обработка данных о вызовах, сгенерированных микросервисом CDRg. Процесс затрагивает 3 микросервиса: CDRg, BRT и HRS, причём инициирует взаимодействие CDRg.

Сценарии использования

Идентификатор и название	UC-1 Обработка записей о звонках абонентов
Взаимодействующие системы	CDRg, BRT, HRS
Предусловия	1. CDRg сгенерировал 10 записей о звонках абонентов 2. CDRg сформировал файл CDR из этих записей
Триггер	CDRg отправляет CDR в BRT
Результат	Баланс абонента изменён в соответствии с обработанной информацией о вызове
Основной поток	Точка расширения: UC-1.1 Списание абонентской платы по ежемесячному тарифу

	<ol style="list-style-type: none"> 1. BRT осуществляет выбор записей из CDR о звонках, принадлежащих абонентам оператора “Ромашка”. 2. BRT подтверждает корректность каждой записи. 3. BRT преобразует каждую запись в объект, дополняя необходимой информацией. 4. BRT отправляет итоговый объект в HRS. 5. HRS подтверждает корректность полученных данных. 6. HRS вычисляет длительность и стоимость каждого вызова в соответствии с тарифами абонентов. 7. HRS отправляет полученные данные в BRT. 8. BRT подтверждает корректность полученных данных. 9. BRT списывает средства со счёта абонента в соответствии с переданными данными.
Альтернативный поток	<p><i>1a. В CDR отсутствуют записи о звонках абонентов оператора “Ромашка”:</i></p> <p>1a.1 BRT завершает выполнение сценария.</p> <p><i>2a. Некорректная запись о вызове:</i></p> <p>2a.1 BRT выводит ошибку в лог.</p> <p>2a.2 BRT переходит к рассмотрению следующих записей.</p> <p>2a.3 Переход к шагу 3 основного потока.</p> <p><i>5a. В HRS переданы некорректные данные:</i></p> <p>5a.1 HRS выводит ошибку в лог.</p> <p>5a.2 HRS завершает выполнение сценария.</p> <p><i>8a. В BRT переданы некорректные данные:</i></p> <p>8a.1 BRT выводит ошибку в лог.</p> <p>8a.2 BRT завершает выполнение сценария.</p>

Идентификатор и название	UC-1.1 Списание абонентской платы по помесечному тарифу
Взаимодействующие системы	BRT, HRS
Триггер	BRT получил CDR с записью за следующий месяц
Предусловия	Нет
Постусловия	<ol style="list-style-type: none"> 1. Со счёта абонентов с помесечным тарифом списана абонентская плата. 2. Остаток минут абонентов с помесечным тарифом обновлён.
Основной поток	<ol style="list-style-type: none"> 1. BRT запрашивает у HRS данные о помесечных тарифах. 2. HRS отправляет необходимую информацию в BRT. 3. BRT подтверждает корректность данных. 4. BRT изменяет данные абонентов с помесечным тарифом, списывая абонентскую плату и обновляя количество минут в соответствии с ответом HRS.

Альтернативный поток	<p>За. В BRT переданы некорректные данные:</p> <p>За.1 BRT выводит ошибку в лог.</p> <p>За.2 BRT завершает выполнение сценария.</p> <p>Зб. BRT получил пустой массив (среди переданных идентификаторов нет месячного тарифа):</p> <p>Зб.1 BRT завершает выполнение сценария.</p>
----------------------	--

Описание логики

Суть процесса заключается в выполнении следующих шагов:

1. CDRg генерирует записи о звонках абонентов.

▼ Про записи звонков

- таблица *subscribers* в БД должна содержать не менее 10 абонентов;
- записи генерируются параллельно для абонентов, присутствующих в таблице *subscribers*;
- информация о звонках добавляется в таблицу *calls*;
- записи генерируются в рамках одного года;
- каждая запись содержит:
 - тип вызова (01 - исходящий, 02 - входящий);
 - телефонный номер обслуживаемого абонента;
 - телефонный номер абонента, с которым осуществлялась связь;
 - дата и время начала звонка (Unix time);
 - дата и время окончания звонка (Unix time).
- количество и длительность звонков определяются случайным образом;
- если генерируется исходящий звонок от абонента, то должен присутствовать и соответствующий входящий вызов для абонента, с которым осуществлялась связь.

2. Из каждых 10 записей формируется CDR.

▼ Про CDR

- формат файла - .txt;
- разделитель данных - запятая, разделитель записей - перенос строки;
- записи в файле должны идти в хронологическом порядке.

Пример записей:

```
01,79123456789,79876543221,1709798657,1709799601
02,79876543221,79123456789,1709798657,1709799601
01,79996667755,79876543221,1709899870,1709905806
02,79876543221,79996667755,1709899870,1709905806
```

3. Сформированный CDR отправляется в топик *cdr_events* брокера сообщений.

4. BRT регулярно проверяет топик на наличие новых сообщений. Как только в топике *cdr_events* появляется сообщение с CDR, он его забирает для обработки.

5. Если пришли данные за следующий месяц (прим.: например, программа была запущена 27.04.2024, а в CDR есть дата 01.05.2024):

5.1. BRT отправляет в HRS GET-запрос на получение данных о месячных тарифах.

▼ Про запрос

- тип запроса - *GET*;
- путь запроса - */tariffs?type=monthly*

5.3. HRS отправляет необходимую информацию в BRT:

```
1 [
2   {
3     "tariffId": 12,
4     "minutesAmount": 50,
5     "cost": 100
6   }
7 ]
```

5.5 BRT списывает со счёта абонентов с помесечным тарифом абонентскую плату и обновляет количество минут.

▼ Про абонентскую плату

- абонентская плата списывается:
 - при получении CDR, в котором есть запись, датированная следующим месяцем;
 - только по происшествии месяца. Т.е. в мае списывается плата за апрель, в июне - за май и т.п.;
 - даже если абонент не разговаривал в предыдущем месяце;
 - баланс может уходить в минус.

6. BRT выбирает из CDR только записи абонентов оператора “Ромашка”.

▼ Про БД BRT

- у BRT должна быть собственная БД с таблицей *subscribers*, которая также должна содержать не менее 10 записей.

7. BRT преобразует запись о каждом звонке, дополняя её следующей информацией:

- а. о тарифе абонента;
- б. о текущем остатке минут на счёте абонента;
- с. о том, является ли *вызываемый* абонент также абонентом оператора “Ромашка”.

▼ Про отправляемые данные

- остаток минут на счёте отправляется для разрешения ситуаций, когда абонент с помесечным тарифом потратил оставшийся месячный лимит и необходимо рассчитать сумму списания по классическому тарифу;
- флаг *isToStranger* нужен для определения стоимости одной минуты по тарифу “Классика” (1.5 для “локальных” абонентов, 2.5 - для “чужих”);
- пример тела сообщения:

```
1 {
2   "msisdn": "75558005353",
3   "tariffId": 11,
4   "amountMinutes": 0,
5   "type": 02,
6   "isToStranger": false,
7   "startTime": "2024-04-10 18:15:06",
8   "endTime": "2024-04-10 18:15:46"
9 }
```

8. BRT отправляет полученный объект в топик *call_events*.

9. HRS забирает сообщение из топика *call_events*.

10. HRS вычисляет длительность и стоимость вызова.

▼ Про расчёт

- в базе данных HRS хранится информация о тарифах. Тарифы на “старте”:

id	Название	Исходящие	Входящие
11	Классика	Для абонентов “Ромашки” - 1.5 у.е./мин Для абонентов других операторов - 2.5 у.е./мин	Бесплатно
12	Помесячный	50 льготных минут (на входящие и исходящие вызовы) - 100 у.е./мес С 51-ой минуты расчет продолжается по тарифу “Классика”	

- длительность вызова округляется в большую сторону (например, 20 секунд считаются за одну минуту);
- стоимость вызова вычисляется исходя из информации о тарифах, хранящейся в таблице *tariffs*. Для каждого тарифа должна быть прописана своя логика, соответствующая данным в атрибуте *details*. Например, если тариф месячный, длительность звонка составила 6 минут, остаток минут на счёте абонента - 4 минуты, при этом звонок совершался не абоненту оператора “Ромашка”, то сообщение, передаваемое HRS, будет выглядеть следующим образом:

```

1 {
2   "msisdn": "78005553535",
3   "cost": 5.0,
4   "minutesCost": 4
5 }
```

11. HRS отправляет полученный объект в топик *withdraw_events*.
12. BRT забирает сообщение из топика *withdraw_events*.
13. BRT списывает средства со счёта абонента в соответствии с полученными данными.

▼ Про списание средств

- если у абонента на счёте не хватает средств, баланс уходит в минус.

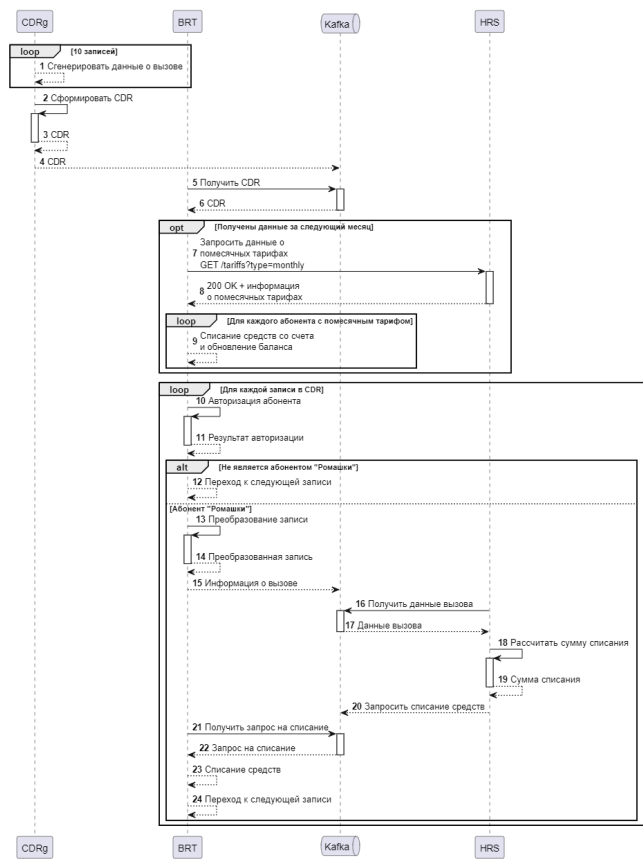


Диаграмма последовательности

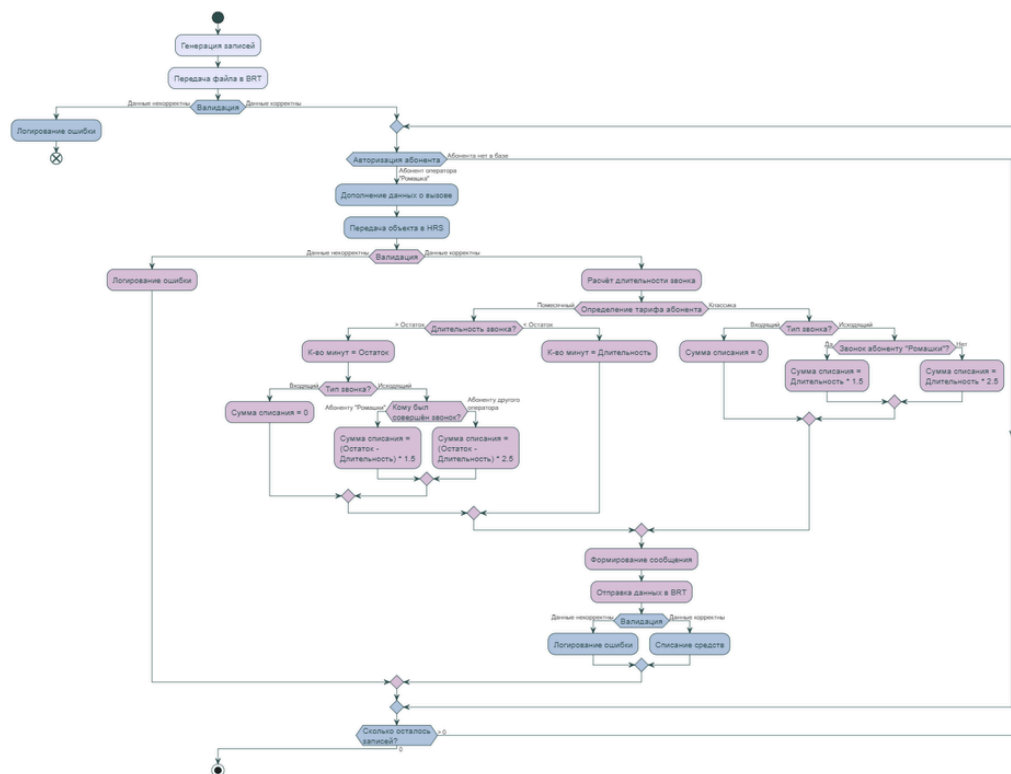


Диаграмма активности

Зоны ответственности микросервисов на диаграмме активности разделены по цветам:

- CDRg - голубой
- BRT - синий
- HRS - розовый

Авторизация

Сценарии использования

Идентификатор и название	UC-2.1 Авторизация пользователя с ролью “Абонент”
Действующие лица	Абонент, CRM, BRT
Предусловия	Отсутствуют
Триггер	Абонент отправляет запрос в CRM на авторизацию по номеру телефона
Результат	Абоненту предоставлен токен для доступа к системе
Основной поток	<ol style="list-style-type: none"> 1. CRM подтверждает корректность данных. 2. CRM отправляет в BRT запрос на идентификацию абонента. 3. BRT отправляет подтверждение, что такой номер телефона присутствует в базе данных. 4. CRM формирует и отправляет абоненту токен для доступа к системе.
Альтернативный поток	<p><i>1а. В CRM переданы некорректные данные:</i></p> <p>1а.1 CRM отображает сообщение об ошибке при составлении запроса.</p> <p>1а.2 CRM завершает выполнение сценария.</p> <p><i>3а. Введённый номер телефона не найден в базе данных:</i></p> <p>3а.1 BRT отправляет сообщение об ошибке идентификации в CRM.</p> <p>3а.2 CRM отображает сообщение об ошибке.</p> <p>3а.3 CRM завершает выполнение сценария.</p> <p><i>3б. В BRT переданы некорректные данные:</i></p> <p>3б.1 BRT выводит ошибку в лог.</p> <p>3б.2 BRT отправляет сообщение об ошибке составления запроса в CRM.</p> <p>3б.3 CRM отображает сообщение об ошибке.</p> <p>3б.4 CRM завершает выполнение сценария.</p>

Идентификатор и название	UC-2.2 Авторизация пользователя с ролью “Менеджер”
Действующие лица	Менеджер, CRM
Предусловия	Отсутствуют

Триггер	Менеджер отправляет в CRM запрос на авторизацию по логину и паролю
Результат	Менеджеру предоставлен токен для доступа к системе
Основной поток	<ol style="list-style-type: none"> 1. CRM подтверждает корректность данных. 2. CRM осуществляет идентификацию и аутентификацию пользователя. 3. CRM формирует и отправляет менеджеру токен для доступа к системе.
Альтернативный поток	<p><i>1a. Переданы некорректные данные:</i></p> <p>1a.1 CRM отображает сообщение об ошибке при составлении запроса.</p> <p>1a.2 CRM завершает выполнение сценария.</p> <p><i>2a. Неверный логин или пароль:</i></p> <p>2a.1 CRM отправляет сообщение об ошибке аутентификации.</p> <p>2a.2 CRM завершает выполнение сценария.</p>

Описание логики

1. Пользователь отправляет в CRM POST-запрос на авторизацию.

Роль пользователя	Абонент	Менеджер
Путь	/subscribers/login	managers/login
Пример	<pre> 1 { 2 "msisdn": "78005553535" 3 }</pre>	<pre> 1 { 2 "username": "admin", 3 "password": "admin" 4 }</pre>

2. Система осуществляет идентификацию и аутентификацию пользователя.

▼ Про идентификацию абонента

- для идентификации абонента необходимо узнать, есть ли абонент с таким номером в базе данных оператора “Ромашка”:
 - CRM отправляет запрос, аналогичный указанному в таблице, в BRT;
 - BRT проверяет, существует ли абонент с указанным *msisdn* в таблице *subscribers*;
 - если абонент не найден, BRT отправляет ответ с кодом 401 Unauthorized;
 - если абонент существует, то BRT отправляет ответ 200 OK с телом сообщения:

```

1 {
2   "id": 1,
3   "msisdn": 78005553535
4 }
```

▼ Про аутентификацию менеджера

- аутентификация менеджера производится в соответствии с данными, указанными в таблице *managers*;
- пароли в базе данных не должны храниться в “сыром” виде - необходимо использовать SHA256.

3. При ошибке проверки доступа CRM отображает пользователю ошибку (см. Swagger).

4. При успешной проверке доступа CRM формирует JWT.

5. CRM отправляет JWT пользователю.

▼ Про JWT

- JWT (JSON Web Tokens) - способ передачи информации между клиентом и сервером. Состоит из трёх частей, разделённых точкой:
 - заголовки (headers), содержащие метаданные: алгоритм шифрования (SHA256) и тип токена (JWT);
 - полезная нагрузка (payload), содержащая необходимую информацию о пользователе и токене: время истечения срока действия токена, роль пользователя, логин (в случае с менеджером) или номер телефона (в случае с абонентом) и т.п.;
 - подпись (signature) - доказательство корректности токена.
- секретный ключ можно выбрать на своё усмотрение;
- в дальнейшем JWT передаётся в заголовке Authorization при запросах, требующих авторизации (пополнение счёта, смена тарифа абонента, создание абонента).

Диаграммы

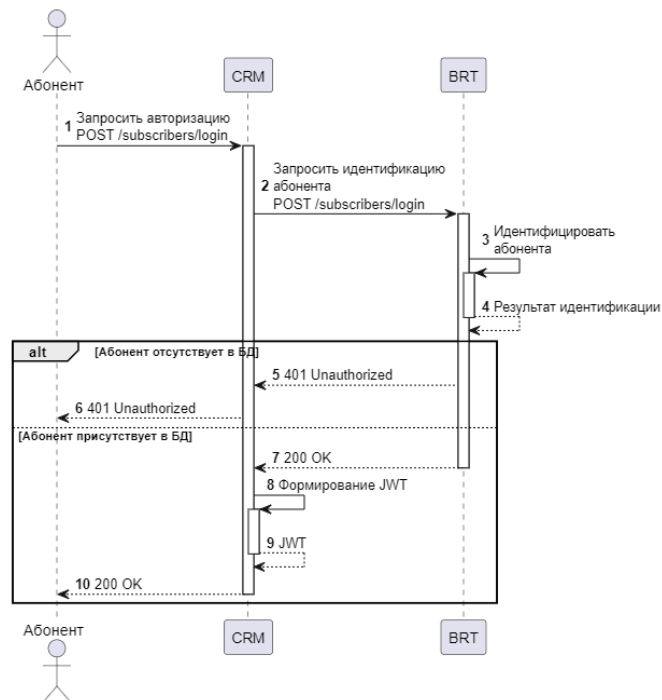


Диаграмма последовательности: авторизация абонента

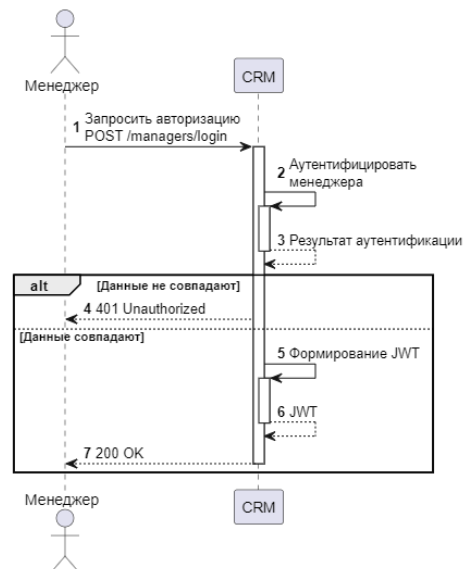


Диаграмма последовательности:
авторизация менеджера

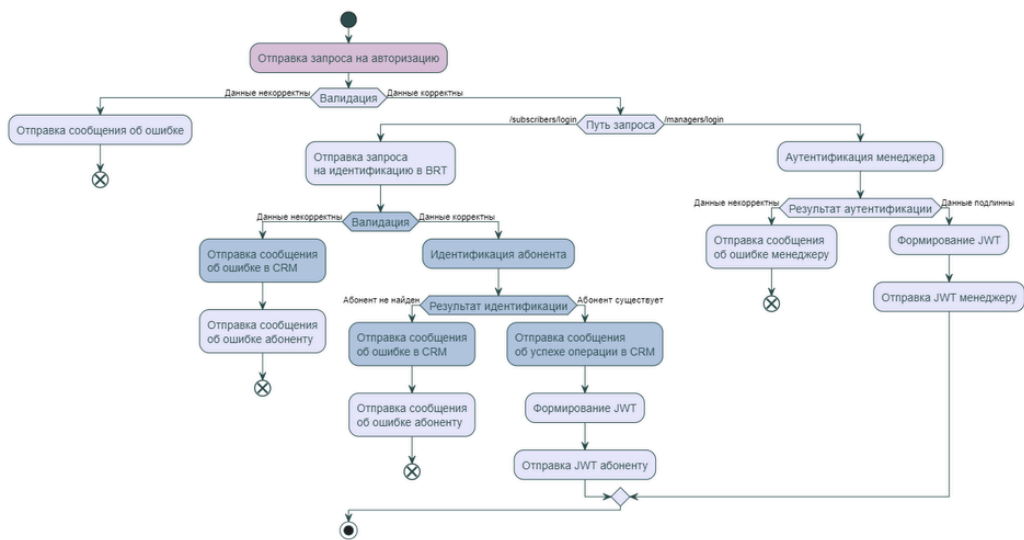


Диаграмма активности

Зоны ответственности микросервисов на диаграмме активности разделены по цветам:

- пользователь - розовый;
- CRM - голубой;
- BRT - синий.

Пополнение баланса [↗](#)

Сценарии использования [↗](#)

Идентификатор и название	UC-3 Пополнение баланса
Действующие лица	Абонент, CRM, BRT
Предусловия	Отсутствуют

Триггер	Абонент отправляет запрос в CRM на пополнение баланса
Результат	Баланс абонента пополнен
Основной поток	<ol style="list-style-type: none"> 1. CRM подтверждает корректность данных. 2. CRM отправляет запрос на пополнение счёта в BRT. 3. BRT подтверждает корректность полученных данных. 4. BRT обновляет значение баланса абонента. 5. BRT отправляет в CRM обновлённый баланс абонента. 6. CRM подтверждает корректность полученных данных. 7. CRM отображает абоненту обновлённый баланс.
Альтернативный поток	<p><i>1а. У пользователя нет доступа к системе:</i></p> <p>1а.1 CRM отображает сообщение об ошибке аутентификации.</p> <p>1а.2 CRM завершает выполнение сценария.</p> <p><i>1б. Неверный формат данных:</i></p> <p>1б.1 CRM отображает сообщение о некорректных значениях атрибутов.</p> <p>1б.2 CRM завершает выполнение сценария.</p> <p><i>1в. У пользователя нет прав совершать эту операцию:</i></p> <p>1в.1 CRM отображает сообщение об отсутствии прав на совершение операции.</p> <p>1в.2 CRM завершает выполнение сценария.</p> <p><i>3а. Переданы некорректные данные:</i></p> <p>3а.1 BRT выводит ошибку в лог.</p> <p>3а.2 BRT отправляет сообщение об ошибке в CRM.</p> <p>3а.3 CRM отображает сообщение об ошибке выполнения операции абоненту.</p> <p><i>6а. Переданы некорректные данные:</i></p> <p>6а.1 CRM выводит ошибку в лог.</p> <p>6а.2 CRM отображает сообщение об ошибке выполнения операции абоненту.</p> <p>6а.3 CRM завершает выполнение сценария.</p>

Описание логики

1. Абонент отправляет в CRM PATCH-запрос на пополнение счёта.

▼ Про запрос

- путь запроса: *subscribers/pay*;
- метод - *PATCH*;
- в заголовке *Authorization* должен быть указан токен абонента для доступа к системе (*Bearer + JWT*);
- пример тела запроса:

```
1 {
```

```

2  "money": 100.1
3  }

```

2. CRM передаёт запрос на пополнение счёта в BRT с указанием номера телефона абонента и суммы, на которую он хочет пополнить счёт.

▼ Про запрос

- CRM берёт номер телефона (*msisdn*) из JWT;
- пример тела запроса:

```

1  {
2  "msisdn": "78005553535",
3  "money": 100.1
4  }

```

3. BRT обновляет баланс абонента с указанным номером телефона, прибавляя к текущему балансу сумму из тела запроса.

4. BRT отправляет обновлённые данные в CRM.

5. CRM отображает обновлённые данные абоненту.

Диаграммы [↗](#)

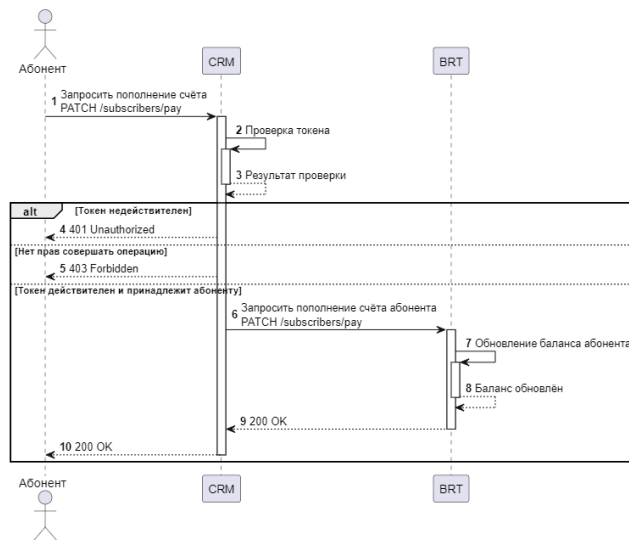


Диаграмма последовательности

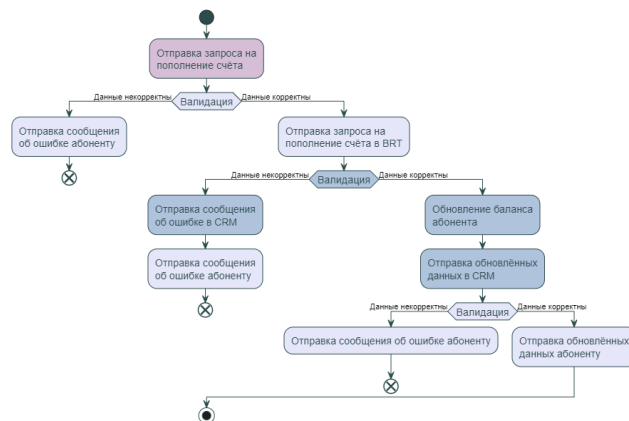


Диаграмма активности

Зоны ответственности микросервисов на диаграмме активности разделены по цветам:

- абонент - розовый;
- CRM - голубой;
- BRT - синий.

Смена тарифа ↗

Сценарии использования ↗

Идентификатор и название	UC-4 Смена тарифа
Действующие лица	Менеджер, CRM, BRT, HRS
Предусловия	Отсутствуют
Триггер	Менеджер отправляет в CRM запрос на смену тарифа абонента
Результат	Тариф абонента изменён
Основной поток	<div>1. CRM подтверждает корректность данных.</div> <div>2. CRM отправляет запрос на смену тарифа в BRT.</div> <div>3. BRT подтверждает корректность данных.</div> <div>4. BRT запрашивает у HRS данные о тарифах.</div> <div>5. HRS отправляет данные о тарифах в BRT.</div> <div>6. BRT подтверждает корректность данных.</div> <div>7. BRT обновляет баланс абонента в соответствии с данными о тарифах.</div> <div>8. BRT изменяет идентификатор тарифа абонента.</div> <div>9. BRT отправляет в CRM обновлённую информацию об абоненте.</div> <div>10. CRM подтверждает корректность данных.</div> <div>11. CRM отображает менеджеру обновлённую информацию об абоненте.</div>
Альтернативный поток	<div>1а. У пользователя нет доступа к системе:</div> <div>1а.1 CRM отображает сообщение об ошибке аутентификации.</div> <div>1а.2 CRM завершает выполнение сценария.</div> <div>1б. Неверный формат данных:</div> <div>1б.1 CRM отображает ошибку о некорректных значениях атрибутов.</div> <div>1б.2 CRM завершает выполнение сценария.</div> <div>1в. У пользователя нет прав совершать эту операцию:</div> <div>1в.1 CRM отображает сообщение об отсутствии прав на совершение операции.</div> <div>1в.2 CRM завершает выполнение сценария.</div> <div>3а. Переданы некорректные данные:</div> <div>3а.1 BRT выводит ошибку в лог.</div>

	3а.2 BRT отправляет сообщение об ошибке в CRM.
	3а.3 CRM отображает сообщение об ошибке выполнения операции менеджеру.
	3а.4 CRM завершает выполнение сценария.
	5а. В HRS отсутствует информация о тарифе:
	5а.1 HRS выводит ошибку в лог.
	5а.2 HRS отправляет сообщение об ошибке в BRT.
	5а.3 BRT отправляет сообщение об ошибке в CRM.
	5а.4 CRM отображает сообщение об ошибке выполнения операции менеджеру.
	5а.5 CRM завершает выполнение сценария.
	6а. Переданы некорректные данные:
	6а.1 BRT выводит ошибку в лог.
	6а.2 BRT отправляет сообщение об ошибке в CRM.
	6а.3 CRM отображает сообщение об ошибке выполнения операции менеджеру.
	6а.4 CRM завершает выполнение сценария.
	10а. Переданы некорректные данные:
	10а.1 CRM выводит ошибку в лог.
	10а.2 CRM отображает сообщение об ошибке выполнения операции менеджеру.
	10а.3 CRM завершает выполнение сценария.

Описание логики

1. Менеджер отправляет в CRM PATCH-запрос на смену тарифа абонента.

▼ Про запрос

- путь запроса: `subscribers/{msisdn}/changeTariff`;
- метод - `PATCH`;
- в заголовке `Authorization` должен быть указан токен менеджера для доступа к системе (`Bearer + JWT`);
- пример тела запроса:

```

1 {
2   "tariffId": 12
3 }
```

2. CRM проверяет действительность токена менеджера, после чего передаёт аналогичный PATCH-запрос на изменение тарифа в BRT.

3. BRT проверяет существование абонента с указанным MSISDN в базе данных, после чего передаёт GET-запрос в HRS, чтобы получить информацию об абонентской плате по текущему тарифу пользователя (если она есть) и о предполагаемом будущем тарифе.

▼ Про запрос

- путь запроса: `tariffs?type=monthly&id=11&id=12`;

- метод - *GET*;
- *type* в параметрах запроса служит фильтром для переданных идентификаторов тарифов (если указан тип - помесечный, то HRS вернёт информацию только о тарифах, у которых в атрибутах есть *monthlyFee* и/или *minutesAmount*);
- логика здесь следующая: поскольку абонентская плата за определённый месяц списывается только по его прошествии, при смене тарифа оператору необходимо взимать плату за этот месяц (чтобы не потерять деньги). Также необходимо начислить пользователю количество минут, которое ему полагается по будущему тарифу.

4. HRS проверяет существование тарифов и передаёт информацию о тех из них, что включают в себя количество доступных минут и/или абонентскую плату (если таких тарифов нет, возвращается пустой массив с кодом 200 ОК).

▼ Про ответ

```

1  [
2  {
3    "id": 12,
4    "name": "Помесечный",
5    "monthlyFee": 100,
6    "minutesAmount": 50
7  }
8  ]

```

5. BRT сопоставляет переданные и полученные тарифы. Если предыдущий тариф абонента является помесечным, то BRT списывает абонентскую плату и обнуляет баланс минут. Если будущий тариф абонента является помесечным, то BRT начисляет абоненту количество минут по тарифу. После этих проверок BRT меняет тариф абонента.

6. BRT отправляет в CRM обновлённую информацию об абоненте.

7. CRM отправляет менеджеру обновлённую информацию об абоненте.

Диаграммы

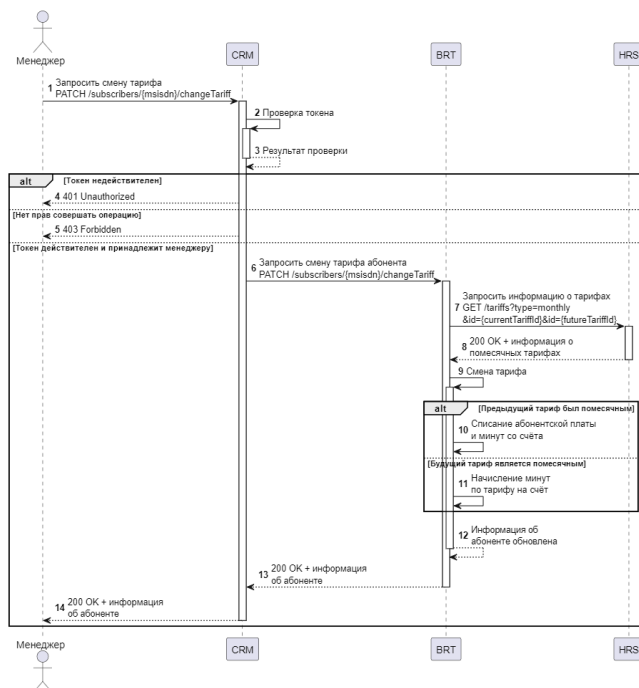


Диаграмма последовательности

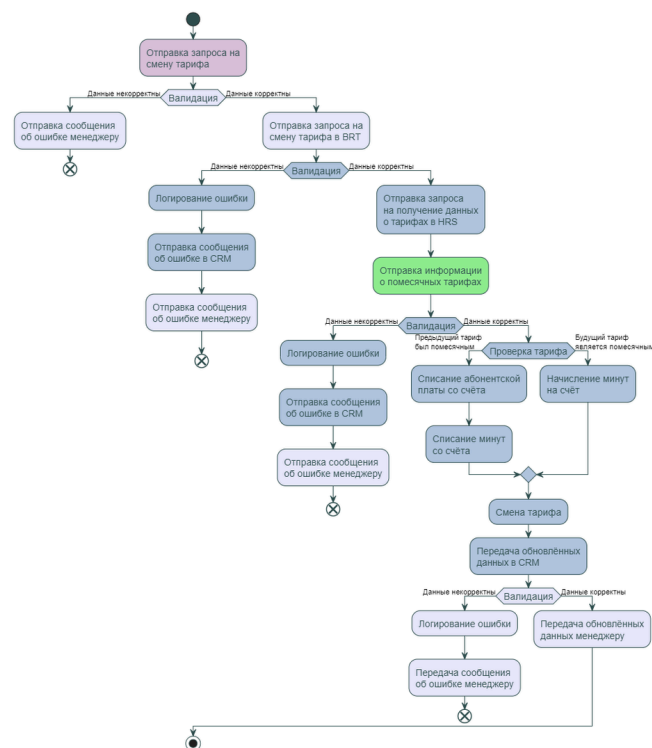


Диаграмма активности

Зоны ответственности микросервисов на диаграмме активности разделены по цветам:

- менеджер - розовый;
- CRM - голубой;
- BRT - синий;
- HRS - зелёный.

Создание абонента [↗](#)

Сценарии использования [↗](#)

Идентификатор и название	UC-5 Создание нового абонента
Действующие лица	Менеджер, CRM, BRT
Предусловия	Отсутствуют
Триггер	Менеджер отправляет в CRM запрос на создание абонента
Результат	Создан новый абонент оператора "Ромашка"
Основной поток	<ol style="list-style-type: none"> 1. CRM подтверждает корректность запроса. 2. CRM отправляет в BRT запрос на создание абонента. 3. BRT подтверждает корректность данных. 4. BRT запрашивает у HRS данные о тарифе. 5. HRS отправляет в BRT данные о тарифе. 6. BRT подтверждает корректность данных. 7. BRT создаёт нового абонента с заданными параметрами.

	<p>8. BRT отправляет в CRM информацию о созданном абоненте.</p> <p>9. CRM подтверждает корректность данных.</p> <p>10. CRM отображает менеджеру информацию о созданном абоненте.</p>
Альтернативный поток	<p><i>1а. У пользователя нет доступа к системе:</i></p> <p>1а.1 CRM отображает сообщение об ошибке аутентификации.</p> <p>1а.2 CRM завершает выполнение сценария.</p> <p><i>1б. Неверный формат данных:</i></p> <p>1б.1 CRM отображает ошибку о некорректных значениях атрибутов.</p> <p>1б.2 CRM завершает выполнение сценария.</p> <p><i>1в. У пользователя нет прав совершать эту операцию:</i></p> <p>1в.1 CRM отображает сообщение об отсутствии прав на совершение операции.</p> <p>1в.2 CRM завершает выполнение сценария.</p> <p><i>3а. Переданы некорректные данные:</i></p> <p>3а.1 BRT выводит ошибку в лог.</p> <p>3а.2 BRT отправляет сообщение об ошибке в CRM.</p> <p>3а.3 CRM отображает сообщение об ошибке выполнения операции менеджеру.</p> <p>3а.4 CRM завершает выполнение сценария.</p> <p><i>5а. В HRS отсутствует информация о тарифе:</i></p> <p>5а.1 HRS выводит ошибку в лог.</p> <p>5а.2 HRS отправляет сообщение об ошибке в BRT.</p> <p>5а.3 BRT отправляет сообщение об ошибке в CRM.</p> <p>5а.4 CRM отображает сообщение об ошибке выполнения операции менеджеру.</p> <p>5а.5 CRM завершает выполнение сценария.</p> <p><i>6а. Переданы некорректные данные:</i></p> <p>6а.1 BRT выводит ошибку в лог.</p> <p>6а.2 BRT отправляет сообщение об ошибке в CRM.</p> <p>6а.3 CRM отображает сообщение об ошибке выполнения операции менеджеру.</p> <p>6а.4 CRM завершает выполнение сценария.</p> <p><i>9а. Переданы некорректные данные:</i></p> <p>9а.1 CRM выводит ошибку в лог.</p> <p>9а.2 CRM отображает сообщение об ошибке выполнения операции менеджеру.</p> <p>9а.3 CRM завершает выполнение сценария.</p>

Описание логики

1. Менеджер отправляет в CRM POST-запрос на создание абонента.

▼ Про запрос

- путь запроса: *subscribers/save*;
- метод - *POST*;
- в заголовке *Authorization* должен быть указан токен менеджера для доступа к системе (*Bearer + JWT*);
- пример тела запроса:

```
1 {  
2   "msisdn": "78005553535",  
3   "tariffId": 12,  
4   "money": 100.1  
5 }
```

- параметр *money* не обязателен, при его отсутствии выставляется значение по умолчанию (100).

2. CRM проверяет действительность токена менеджера, после чего передаёт аналогичный POST-запрос на создание абонента в BRT.

3. BRT передаёт GET-запрос в HRS, чтобы получить информацию о переданном тарифе (при условии, что он является помесечным).

▼ Нажмите здесь, чтобы развернуть...

- путь запроса: *tariffs?type=monthly&id=12*;
- метод - *GET*;
- данный запрос необходим, чтобы проверить, существует ли данный тариф в базе HRS и является ли он помесечным (чтобы начислить количество минут по тарифу).

4. HRS проверяет существование тарифов и передаёт информацию о тех из них, что включают в себя количество доступных минут и/или абонентскую плату (если таких тарифов нет, возвращается пустой массив).

▼ Про ответ

```
1 [  
2   {  
3     "id": 12,  
4     "name": "Помесечный",  
5     "monthlyFee": 100,  
6     "minutesAmount": 50  
7   }  
8 ]
```

5. BRT создаёт нового абонента с указанными данными (если тариф помесечный, то баланс минут приравнивается к количеству минут по тарифу) и отправляет ответ в CRM.

6. CRM отображает менеджеру информацию о новом абоненте.

Диаграммы

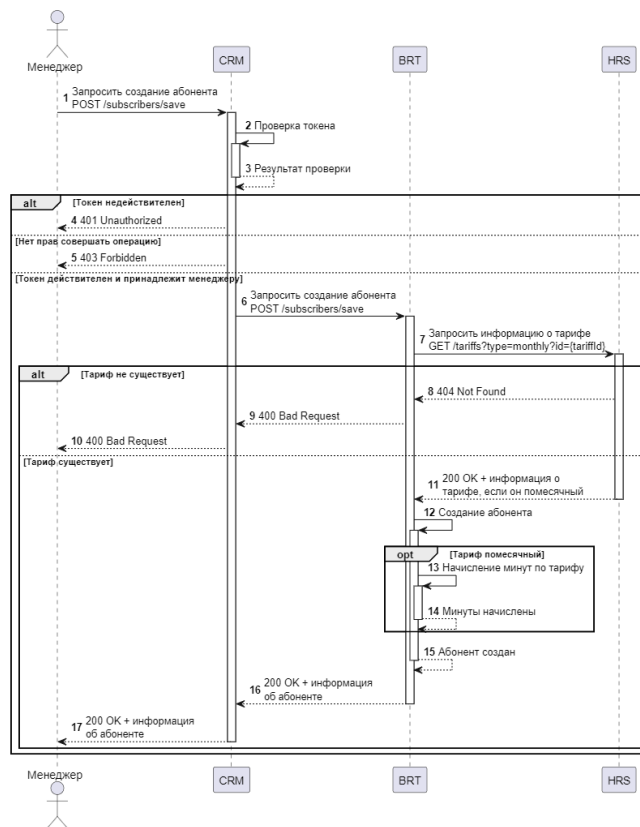


Диаграмма последовательности

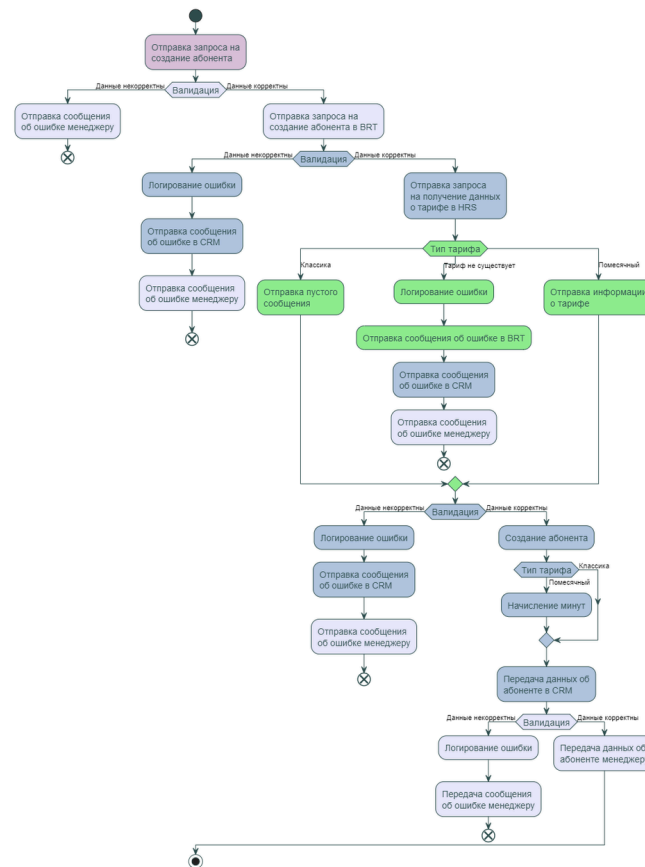


Диаграмма активности

Зоны ответственности микросервисов на диаграмме активности разделены по цветам:

- менеджер - розовый;
- CRM - голубой;
- BRT - синий;
- HRS - зелёный.