

Я как клиент банка хочу изменить информацию о себе.

История версий

Задача	Тип изменения	Цвет изменения	Дата	Автор
Ссылка на задачу	Создано		26.03.2024	Ситникова Е.А.
Ссылка на задачу	Изменено (указать раздел и пункт)	синий	27.03.2024	Ситникова Е.А.

Бизнес- анализ

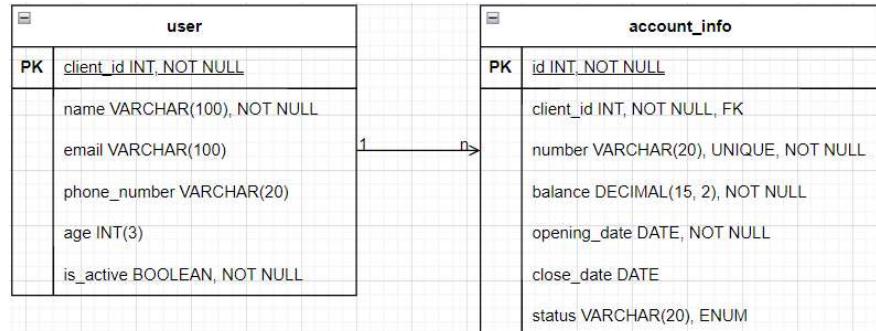
- ссылка

Назначение и цель

Изменение параметров пользователя.

Предварительные действия

- Реализована модель данных в СУБД:



- Пользователь прошел аутентификацию и авторизацию

Ограничения по ролям

- Доступ к интерфейсу возможен для авторизованного клиента с ролью CLIENT_ALL

Макеты

- ссылка на Figma/Pixco/draw.io итд

Основной сценарий (успешный сценарий)

1. Пользователь переходит в ЛК https..... и переходит в раздел «Мои банковские счета»
2. Front отправляет запрос на Back GET/accounts (ссылка на API)
 - a. Если Back вернул ошибки 4xx, 5xx см. альтернативный сценарий 1
3. Front отображает список счетов
4. Пользователь нажимает на любой из счетов в списке всех счетов
5. Front отправляет запрос на Back GET/accounts/{accountId} (ссылка на API)
 - a. Если Back вернул ошибки 4xx, 5xx см. альтернативный сценарий 1
6. Front отображает информацию по счету в модальное окне в соответствии с п.Макеты
7. Пользователь вносит изменения в о себе и нажимает кнопку «Сохранить».
8. Front отправляет запрос на Back PUT /users/{userId} (ссылка на API)
 - a. Если Back вернул ошибки 4xx, 5xx см. альтернативный сценарий 1
9. Front отображает измененные параметры счета.

Альтернативный сценарий

1. Front отобразить ошибку от Back

Требования к интерфейсу

Название атрибута	Тип	Длина	Редактируемость	Маска ввода /отображения	Обязательность	Сортировка	Значение по умолчанию	Источник данных	Примечания
ФИО	label	250	+	Допускаются только буквы	-	-	-	Back	
email	label	100	+	Должен быть знак “@”	-	-	-	Back	
Возраст	label	3	+	3-х значное число	-	-	-	Back	

Результат

- Клиент в ЛК вносит изменения в параметры счета.

PUT /users/{userId} Обновить информацию о пользователе

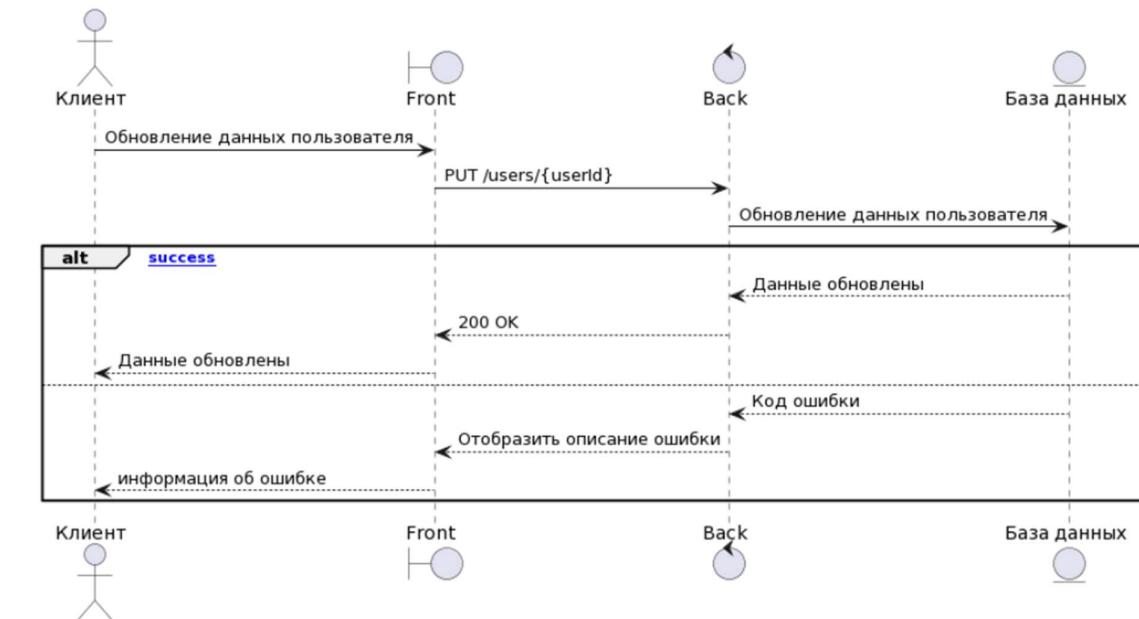
История версий

Задача	Тип изменения	Цвет изменения	Дата	Автор
Ссылка на задачу	Создано		26.03.2024	Ситникова Е.

Сервис

Счета. Микросервис account-number

Алгоритм вызова



@startuml
actor "Клиент" as customer

```

boundary "Front" as Front
control "Back" as Back
entity "База данных" as db
customer -> Front: Обновление данных пользователя
Front-> Back: PUT /users/{userId}
Back -> db : Обновление данных пользователя
alt [success]
Back <-- db : Данные обновлены
Front <-- Back: 200 OK
Front--> customer : Данные обновлены
else
db --> Back: Код ошибки
Back --> Front : Отобразить описание ошибки
Front --> customer : информация об ошибке
end

@enduml

```

Свойства вызова

Свойство	Описание
Тип взаимодействия	REST/Kafka/Артемис MQ/gRPC/GraphQL
Формат запроса/ответа	JSON/XML
Тип вызова	Синхронный/Асинхронный
Протокол взаимодействия	HTTPS/MTLS
Способ аутентификации	Bearer Token/ API key/ JWT

Формат запроса

Параметр	Тип	Длина	Расположение	Обяз	Описание
userId	string	20	path	+	Уникальный идентификатор клиента

Пример

PUT /users/40781000287777000002

Тело запроса

Параметр	Тип	Длина	Расположение	Обяз	Описание	Место для сохранения
name	string	250	-	-	ФИО	User.name
email	string	100	-	-	Электронная почта	User.email
age	string	3	-	-	Возраст	User.age

Пример

```
{  
  "name": "Alice Smith",  
  "email": "alice@example.com",  
  "age": 30  
}
```

Формат ответа

Параметр	Тип	Длина	Обязательность	Описание	Источник данных
userId	string	-	+	Уникальный идентификатор клиента	User.client_id
name	string	250	+	ФИО	User.name
email	string	100	+	Электронная почта	User.email
age	string	3	+	Возраст	User.age

Пример

```
{  
  "userId": «70203333»,  
  "name": "Иванов Иван Иванович",  
  "email": name@ya.ru,  
  "age": "30"  
}
```

Коды ответов

Код	Название	Описание	Поведение системы
200	OK	Успешный запрос	Продолжить процесс
401	Bad Request	Не авторизован	Отобразить ошибку
403	Forbidden	Доступ запрещен	Отобразить ошибку
404	Not found	Данные не найдены	Отобразить ошибку
500	Internal Server Error	Сервер не доступен	Отобразить ошибку

Алгоритм обработки запроса

1. Back при получении запроса, обращается в таблицу user, находит параметры пользователя и заменяет найденные параметры новыми значениями.
2. (можно приложить пример SQL запроса)
3. Back по запрошенному пользователю, обращается в таблицу user и отдает ответ на front в соответствии с п.Формат ответа

Альтернативные процессы

Ошибки 4xx, 5xx

1. Back вернуть ошибку с текстом на Front