

Software Requirements Specification

для

Мобильного приложения «Ugra-Mobile»

Версия 1.0

Разработано

Дата: 31.07.2022

ВВЕДЕНИЕ	3
Цели	3
Краткая сводка возможностей	3
Определения, акронимы и сокращения	3
Краткое содержание	5
ОБЗОР СИСТЕМЫ	6
Обзор вариантов использования	6
Описание вариантов использования	6
Сменить тариф	6
Авторизоваться по номеру телефона	7
Диаграмма классов	9
Архитектура REST API	9
ДЕТАЛЬНЫЕ ТРЕБОВАНИЯ	10
Пользовательские требования функционального характера	10
Функциональные требования функционального характера	10
Нефункциональные требования	10
Приложение 1	12

1. ВВЕДЕНИЕ

1.1 Цели

Основной целью создания мобильного приложения «Ugra-Mobile» является – разгрузка персонала мобильной связи и ее оптимизация в расширенности действий абонента, без долгих ожиданий обратной связи оператора.

1.2 Краткая сводка возможностей

Приложение «Ugra-Mobile» предназначена для информационного обеспечения граждан, в части следующих процессов:

- авторизация по номеру телефона;
- просмотр текущего баланса;
- получение информации о тарифе;
- получение информации о всех доступных тарифах;
- возможность изменить тариф;
- подключение/отключение дополнительных опций к тарифу.

1.3 Определения, акронимы и сокращения

Таблица №1. Глоссарий

1	ТЗ	Техническое задание
2	Дизайн приложения	Визуальный облик программы, который подстроен под пользовательское поведение. Хороший дизайн одновременно эстетичный и удобный для пользователя.
3	Мобильное приложение	Мобильное приложение – это специально разработанное приложение под конкретную мобильную платформу (iOS, Android, Windows Phone).
4	Нативное мобильное приложение	Приложение можно назвать «родным» для операционных систем – Android, IOS, WinPhone . Такие мобильные приложения пишутся на языках программирования, утвержденных

		разработчиками программного обеспечения под каждую конкретную платформу, а потому органично встраиваются в сами операционные системы. Приложения загружаются через магазины приложений (App Store, Google Play и т.д.) и соответствуют требованиям этих магазинов.
5	Онлайн-администратор	Лицо, осуществляющее от имени Заказчика информационную поддержку приложения
6	.pdf	межплатформенный открытый формат электронных документов, изначально разработанный фирмой Adobe Systems с использованием ряда возможностей языка PostScript. В первую очередь предназначен для представления полиграфической продукции в электронном виде.
7	Спринт	это повторяемый фиксированный промежуток времени, в течение которого создается продукт «Готово» с максимально возможной ценностью.
8	ПО	Программное обеспечение
9	Лицензирование ПО	Это правовой инструмент, определяющий использование и распространение программного обеспечения, защищённого авторским правом
10	База данных	совокупность данных, хранимых в соответствии со схемой данных, манипулирование которыми выполняют в соответствии с правилами средств моделирования данных.
11	Предположение	это утверждение, которое предполагается верным в отсутствие знаний или доказательств иного.
12	Вариант использования	в разработке программного обеспечения и системном проектировании это описание поведения системы, когда она взаимодействует с кем-то из внешней среды.

1.4 Краткое содержание

Данное приложение предназначено для улучшения качества обслуживания абонентов мобильной связи “Ugra-Mobile”, оно помогает абоненту быстро получить нужную ему информацию. С помощью него абонент сможет узнать текущий баланс, посмотреть свой тариф, изменить тариф и многое другое, при этом не нагружая онлайн-администраторов и не тратя время на его ответ.

Стоимость данного продукта составит не менее 20 000 000 рублей (двадцать миллионов рублей). Сроки разработки на первый спринт составят не менее 2-4 недель согласно Договору.

Данное приложение будет свободного использования типа лицензий (бесплатная или открытая).

2. ОБЗОР СИСТЕМЫ

2.1. Обзор вариантов использования

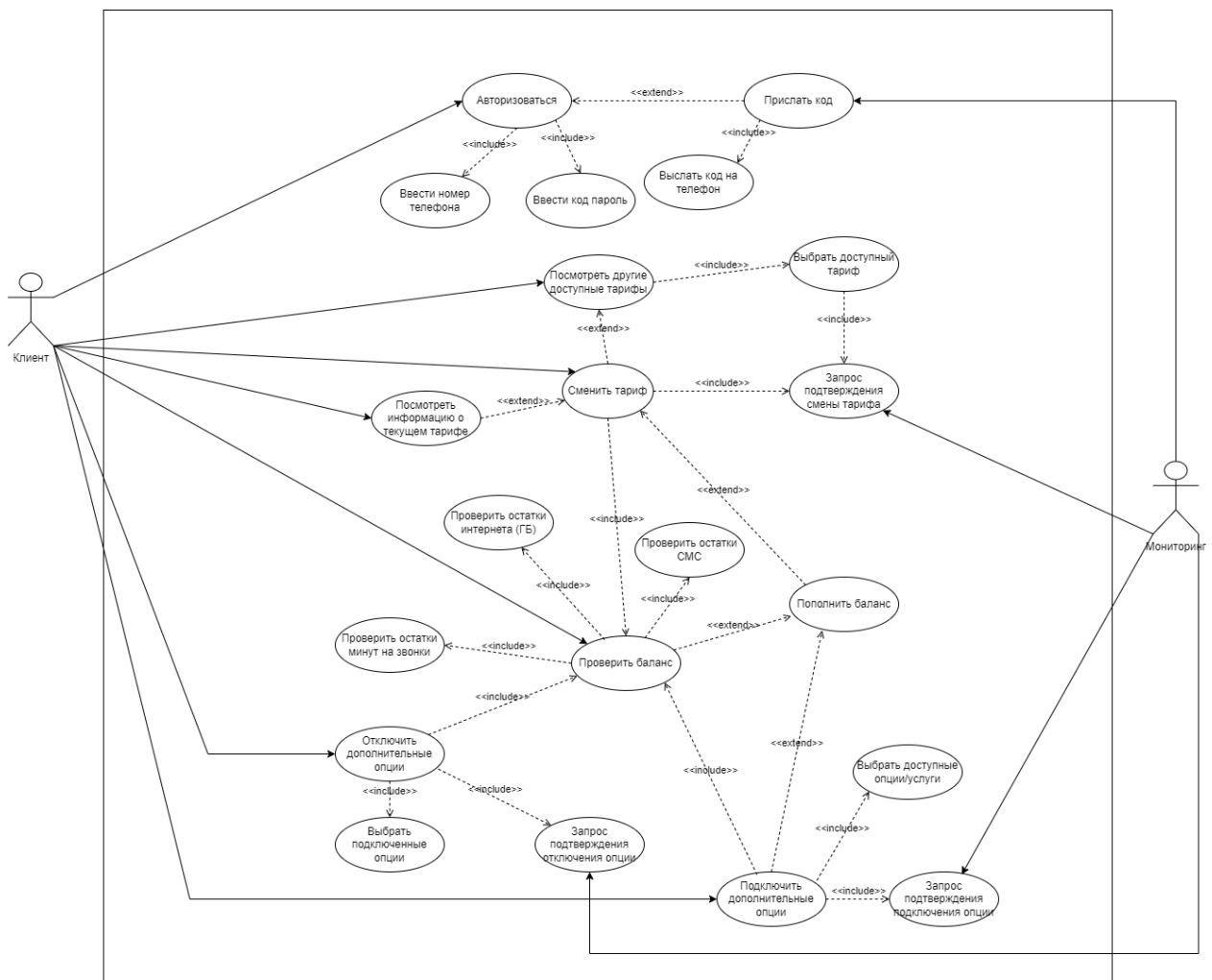


Рисунок 1 – Диаграмма вариантов использования

2.2. Описание вариантов использования

2.2.1. Сменить тариф

УС-УМ-1: Сменить тариф.

Краткое описание: Абонент меняет тариф.

Действующие лица: Клиент.

Триггер: Клиент нажимает кнопку “сменить тариф”.

Предусловия: Клиент абонент мобильной связи Ugra-Mobile. Абонент авторизован в системе.

1. Система предлагает список доступных ему тарифов.
2. Клиент читает условия и описание тарифов.
3. Клиент выбирает подходящий ему тариф.
4. **Если** оплата по текущему тарифу прошла более 7 дней назад, то система переходит на следующий шаг.
5. Система запрашивает подтверждение на смену тарифа.
6. Система списывает деньги со счета для оплаты тарифа.

7. **Если** у клиента хватает денежных средств на счету для оплаты тарифы, то система переходит на следующий шаг.
8. Система отправляет СМС-уведомление о смене тарифного плана.
9. Система благодарит клиента за пользования услугами компании.
10. Вариант использования завершает свою работу.

Альтернативный поток:

4а. Списание средств произошло менее 7 дней назад.

1. Система оповещает клиента, о смене тарифного плана в следующем месяце.
2. Система переходит на шаг 8 основного потока.

7а. Не хватает средств на балансе.

1. Система оповещает клиента о недостаточности средств для оплаты тарифа.
2. Система предлагает пополнить баланс.
3. Если клиент пополняет баланс, то шаг переходит на 8 основного потока.

Поток исключения:

3b. Клиент отказывается пополнять баланс

1. Вариант использования завершает свою работу.

Постусловие: Клиент меняет тариф.

Результат: В случае успешного выполнения основного потока, клиент меняет тарифный план.

2.2.2 Авторизоваться по номеру телефона

УС-УМ-2: Авторизация по номеру телефона.

Краткое описание: Абонент проходит авторизацию в мобильном приложении.

Действующие лица: Клиент.

Триггер: Клиент нажимает кнопку “Авторизоваться”.

Предусловия: Клиент пользователь мобильной связи Ugra-Mobile. Клиент имеет мобильное приложение.

1. Система предлагает авторизоваться клиенту.
2. Клиент набирает номер телефона для авторизации.
3. **Если** система номер телефона введен верно, то система переходит на следующий шаг.
4. Клиент нажимает кнопку “Отправить код-пароль”.

5. Система отправляет код-пароль клиенту.
6. **Если** клиенту приходит код-пароль, то система переходит на следующий шаг.
7. Клиент вводит код-пароль в системе.
8. **Если** клиент вводит код-пароль правильно, то система переходит на следующий шаг.
9. Система приветствует пользователя.
10. Вариант использования завершает свою работу.

Альтернативный поток:

3а. Номер введен неверно.

1. Система оповещает клиента, о том что такого пользователя не существует.
5. Система переходит на шаг 1 основного потока.

6а. Не пришел код-пароль

1. Система проверяет кол-во попыток ввода код-пароля.
2. Если кол-во попыток меньше трех, то система предлагает отправить код повторно.
2. Система предлагает отправить код повторно через 30 секунд.
3. Клиент нажимает кнопку "Отправить код повторно".
5. Система увеличивает кол-во попыток ввода код-пароля на 1.
4. Система переходит на шаг 7 основного потока

Поток исключения:

2а. Кол-во попыток больше трех.

1. Вариант использования завершает свою работу.

8а. Клиент вводит код-пароль не верно

1. Система отображает о неправильно введенном код-пароле.
2. Система предлагает заново ввести код-пароль.
3. Клиент вновь вводит код-пароль.
4. Вводит код-пароль 3 раза подряд неправильно.
5. Система блокирует попытки входа на 4 часа.
6. Вариант использования завершает свою работу.

Постусловие: Клиент прошел авторизацию

Результат: В случае успешного выполнения основного потока, клиент проходит авторизацию в мобильном приложении.

2.3. Диаграмма классов

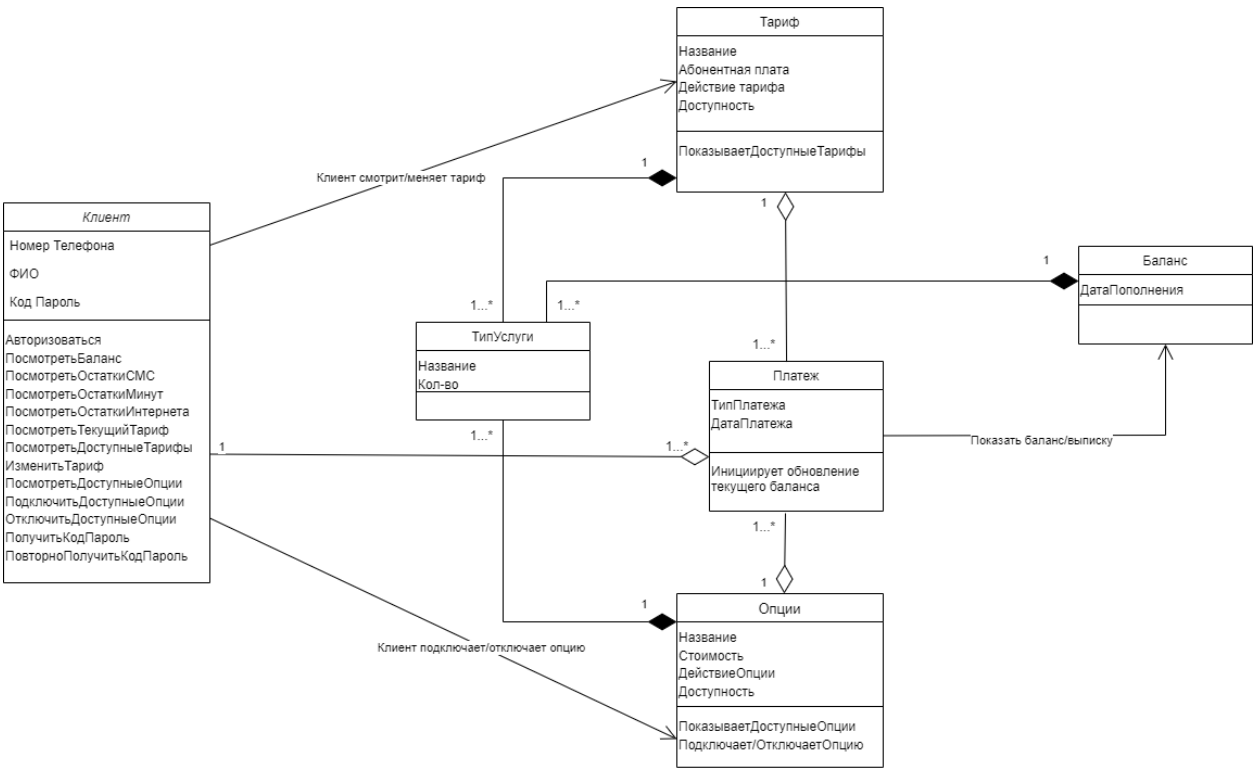


Рисунок 1 – Диаграмма классов

2.4. Архитектура REST API

Swagger-документ представлен в Приложении 1.

3. ДЕТАЛЬНЫЕ ТРЕБОВАНИЯ

3.1. Пользовательские требования функционального характера

ПТ001. Пользователь должен иметь возможность посмотреть текущий баланс в системе;

ПТ002. Пользователь должен иметь возможность посмотреть текущий тариф;

ПТ003. Пользователь должен иметь возможность посмотреть доступные ему тарифы;

ПТ004. Пользователь должен иметь возможность изменить тариф;

ПТ005. Пользователь должен иметь возможность подключить дополнительные услуги/опции;

ПТ006. Пользователь должен иметь возможность посмотреть остатки по минутам звонков, гб интернета и кол-во смс.

3.2. Функциональные требования функционального характера

ФТ001. Система должна иметь возможность отправить код-пароль для авторизации пользователя;

ФТ002. Система должна предоставить возможность пользователю ввести код-пароль при авторизации повторно в случае, если пользователь ввел неверный код-пароль;

ФТ003. Система должна автоматически определять регион пользователя, для отображения корректного списка доступных ему тарифов;

ФТ004. Система должна иметь возможность хранить информацию о всех действиях клиента.

3.3. Нефункциональные требования

О001. Система должна работать на операционных системах IOS и Android;

СТ001. Система должна обрабатывать 4000 одновременных запросов с временем отклика, не превышающим 3 сек. Максимальное время ожидания абонентом обработки своего запроса (без учета задержки до серверов) составляет 5 секунд;

СТ002. Система должна быть функционирующей 99,9% времени, то есть простаивать не более 9 часов в год;

СТ003. Графические интерфейсы должны выводить сообщение об ошибке в случае возникновения некорректных или ошибочных действий системы;

СТ004. Графические интерфейсы должны быть защищены от несанкционированных и нетипичных действий пользователя;

СТ005. Система мониторинга системы должна обнаруживать аномальную сетевую активность, аномальную активность приложений, сообщать об обнаруженных угрозах администратору и принимать меры по предотвращению такой активности;

СТ006. При открытии вкладки “дополнительные опции/услуги”, система должна предоставить список доступных опций абоненту;

СТ007. При открытии вкладки “сменить тариф”, система должна предоставить список доступных тарифов абоненту.

АК001. Пользователь системы имеет доступ к информации, доступной для его класса;

АК002. Все данные системы, в том числе базы данных, а также программный и исполняемый код компонентов системы должны иметь две резервные копии;

АК003. Резервное копирование данных должно осуществляться во время наименьшей нагрузки на систему (с 03:00 до 06:00 по местному времени нахождения сервера) каждый день;

АК004. При открытии приложения, пользователь должен видеть на главной странице свой текущий баланс;

АК005. При открытии приложения, пользователь должен видеть на главной странице свой текущий тариф;

Приложение 1

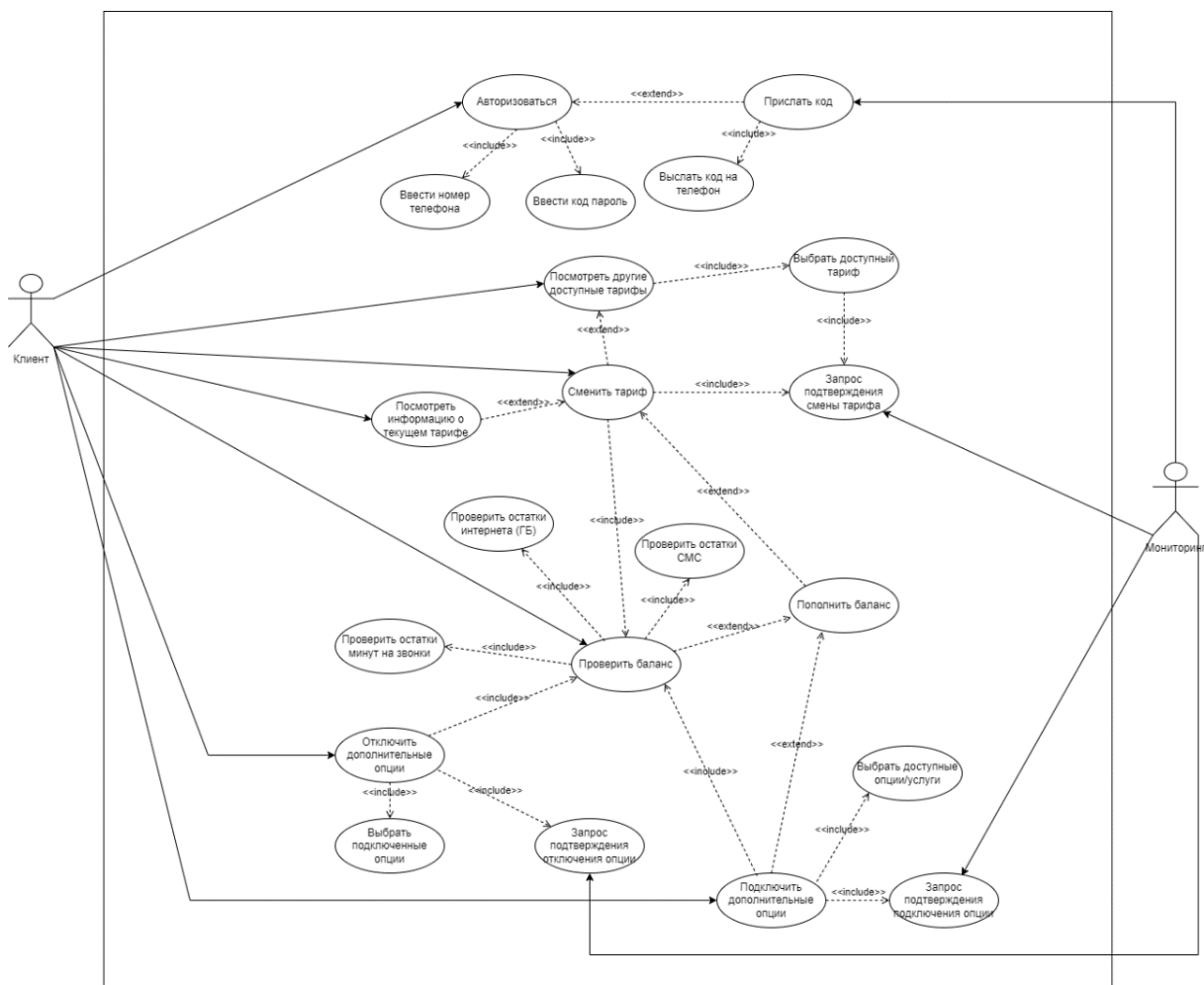


Рисунок 3 – Диаграмма вариантов использования

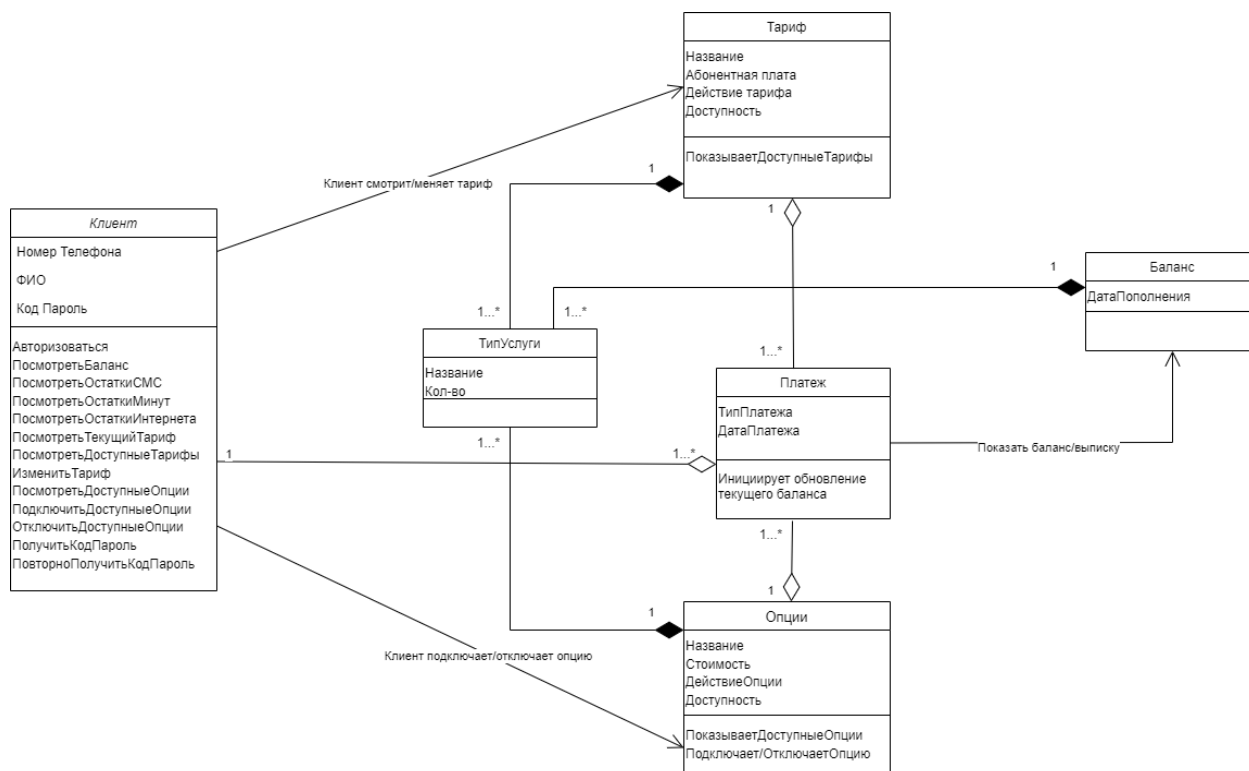


Рисунок 4 – Диаграмма классов

Swagger-документация:

<https://app.swaggerhub.com/apis/zorki2020/CH3/1.0>

Мобильная связь “Ugra-Mobile” компания, предоставляющая услуги сотовой связи физическим лицам. Мобильная связь имеет сайт, через который осуществляется следующие функции:

- изменить тариф;
- пополнить баланс;
- посмотреть баланс;
- посмотреть доступные тарифы;
- подключить доступные опции.

Также есть операторы, которые обрабатывают запросы клиентов и помогают им в решении их вопросов.

Все данные о об обращениях хранятся и обрабатываются на одном сервере.

Описание структуры хранения данных

Rate

Поле (атрибут)	Тип	Описание
rate_id	int	Идентификатор тарифа
name_rate	string	Название тарифа
action	string	Действие
availability_rate	boolean	Доступность тарифа
price_rate	string	Список тарифов

Balance

Поле (атрибут)	Тип	Описание
check_id	int	Идентификатор баланса
replenishment_date	string	Дата пополнения
amount_of_money	number	Оставшееся количество денег
subscriber_id	int	Идентификатор абонента
Service_type	string	Типы услуг
amount_service_type	int	Кол-во типов услуг

Options

Поле (атрибут)	Тип	Описание
option_id	int	Идентификатор опции
name_option	string	Имя опции
price_option	int	Список опций
action	string	Действие
availability_option	boolean	Доступность опции

Service_type

Поле (атрибут)	Тип	Описание
name_service	string	Название услуги
id_service	int	Идентификатор услуги
amount	string	Кол-во услуг

Client

Поле (атрибут)	Тип	Описание
subscriber_id	int	Идентификатор абонента
first name	string	Имя пользователя
last name	string	Фамилия пользователя
code	int	Код-пароль
current_rate	string	Текущий тариф
option_id	int	Идентификатор опции

Payment

Поле (атрибут)	Тип	Описание
payment_id	int	Идентификатор платежа
payment_type	string	Тип платежа
replenishment_date	string	Дата пополнения
balance_id	int	Идентификатор баланса
rate_id	int	Идентификатор тарифа
subscriber_id	int	Идентификатор абонента
option_id	int	Идентификатор опции

Взаимодействие с API

Взаимодействовать с API будут:

- Опции
- Тариф
- Баланс

Действия API

API будет выполнять следующие действия:

- Получить информацию о текущем балансе;
- Изменить текущий тариф на доступный;
- Удалить опцию.

Методы использования API

Действие, которое будет выполнять API	HTTP метод
Получить информацию о текущем балансе	GET
Изменить текущий тариф на доступный	PUT
Удалить опцию	DELETE
Подключить дополнительную опцию	POST

Параметры запроса

1. PUT /Client

Изменить текущий **тариф** абонента на доступный

1. Запрос
 - a. rate_id
 - b. subscriber_id
2. Ответ
 - a. 200 OK

2. GET /balance

Получить информацию о текущем **балансе**

1. Запрос
 - a. check_id
 - b. subscriber_id
2. Ответ
 - a. amount_of_money
 - b. subscriber_id
 - c. replenishment_date

3. GET /Options

Получить список всех доступных опций

1. Запрос
 - a. availability_option
2. Ответ
 - a. name_option
 - b. option_id
 - c. price_option

4. GET /Rate

Получить список всех доступных тарифов

1. Запрос
 - a. availability_rate
2. Ответ
 - a. rate_id
 - b. name_rate
 - c. price_rate