

ВШЭ и НСПК

MIR Links

Версия 1.0.0 от 22.03.2021

Выполнили студенты ВШЭ:

Абдельсалам Шади (ФКН ПИ 2 курс)

Дондик Ярослав (ФКН ПИ 2 курс)

Кирюхин Андрей (ФКН ПИ 2 курс)

Терехова София (ФКН ПИ 2 курс)

Точилина Полина (ФКН ПИ 2 курс)

Москва, 2021

СОДЕРЖАНИЕ

1. MIR Links. Введение	5
1.1. Назначение и содержание.....	5
1.2. Описание проблемы	5
2. Концептуальная архитектура сервиса	6
2.1. Архитектурная схема сервиса.....	6
2.2. Описание компонентов.....	7
3. Концептуальная схема работы сервиса	10
3.1. Схема использования сервиса.....	10
3.2. Пользовательские сценарии	11
3.2.1. User script: Авторизация в банковском приложении	11
3.2.2. User script: Создание короткой ссылки	11
3.2.3. User script: Перевод(оплата) по короткой ссылке	11
3.2.4. User script: Удаление короткой ссылки	12
3.2.5. User script: Организация сбора денег	12
3.3. Схема взаимодействия модулей сервиса	13
3.3.1. Взаимодействие компонентов при создании ссылки	13
3.3.2. Взаимодействие компонентов при совершении перевода	14
3.3.3. Взаимодействие компонентов при удалении ссылки	15
3.3.4. Взаимодействие компонентов при денежном переводе (в контексте банков и ядра МИР)	16
4. Описание методов API	17
4.1. Конечные точки(Endpoints).....	17
4.1.1. POST /links	18
4.1.2. GET /links/{linkId}	19
4.1.3. GET /links.....	20
4.1.4. DELETE /links/{linkId}.....	21
4.1.5. POST /links/auth.....	22
4.1.6. GET /links/send	23
4.1.7. GET /links/transactions/{linkId}	24
4.1.8. GET /links/last_transactions/{linkId}.....	25
4.2. Модель данных	26
4.2.1. LinkInfo.....	26
4.2.2. LinkCreationResponse	28
4.2.3. LinkIdArray	29
4.2.4. ErrorResponseObject	30
4.2.5. CreditorAgentData	31

4.2.6.	AuthData	32
4.2.7.	AccountData	33
4.2.8.	PaymentResponse	34
4.2.9.	TransactionsArray	35
4.2.10.	LastTransactionsArray	36
5.	Описание Android приложения	37
5.1.	Компоненты приложения	37
5.1.1.	Основные компоненты	37
5.1.2.	Основные activities	37
5.1.3.	Основные adapters:	38
5.2.	Требования к экранам	39
5.2.1.	Login Activity	39
5.2.2.	Main Activity	39
5.2.3.	CreateNewLinkView Activity	39
5.2.4.	ExistingLinks Activity	40
5.2.5.	LinkInfo Activity	40
5.2.6.	SendMoney Activity	40
5.3.	Экраны	41
5.3.1.	Login Activity	41
5.3.2.	Main Activity	41
5.3.3.	CreateNewLinkView Activity	42
5.3.4.	ExistingLinks Activity	42
5.3.5.	LinkInfo Activity	43
5.3.6.	SendMoney Activity	43
6.	Тестирование	44
6.1.	Тестирование web-версии	44
6.1.1.	Веб-интерфейс страницы сбора денег	44
6.1.2.	Тестирование кнопки перевода	44
6.1.3.	Тестирование блока с информацией о прогрессе сбора средств	45
6.1.4.	Тестирование автообновления списка переводов	45
6.1.5.	Веб-интерфейс страницы выбора банка	46
6.1.6.	Тестирование поиска банка	47
6.2.	Тестирование Android приложения	47
6.2.1.	Тестирование экрана авторизации	47
6.2.2.	Тестирование главного экрана	48
6.2.3.	Тестирование экрана создания ссылки	49
6.2.4.	Тестирование экрана перевода денег	50

6.2.5.	Тестирование экрана информации о короткой ссылке.....	52
--------	---	----

1. MIR Links. Введение

1.1. Назначение и содержание

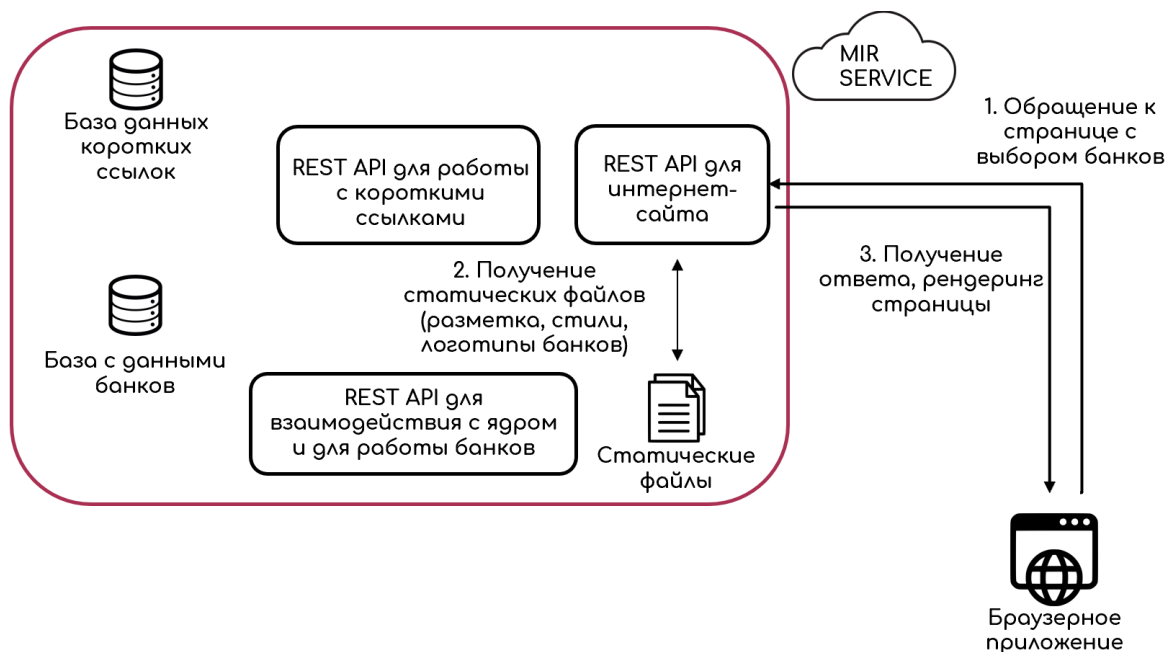
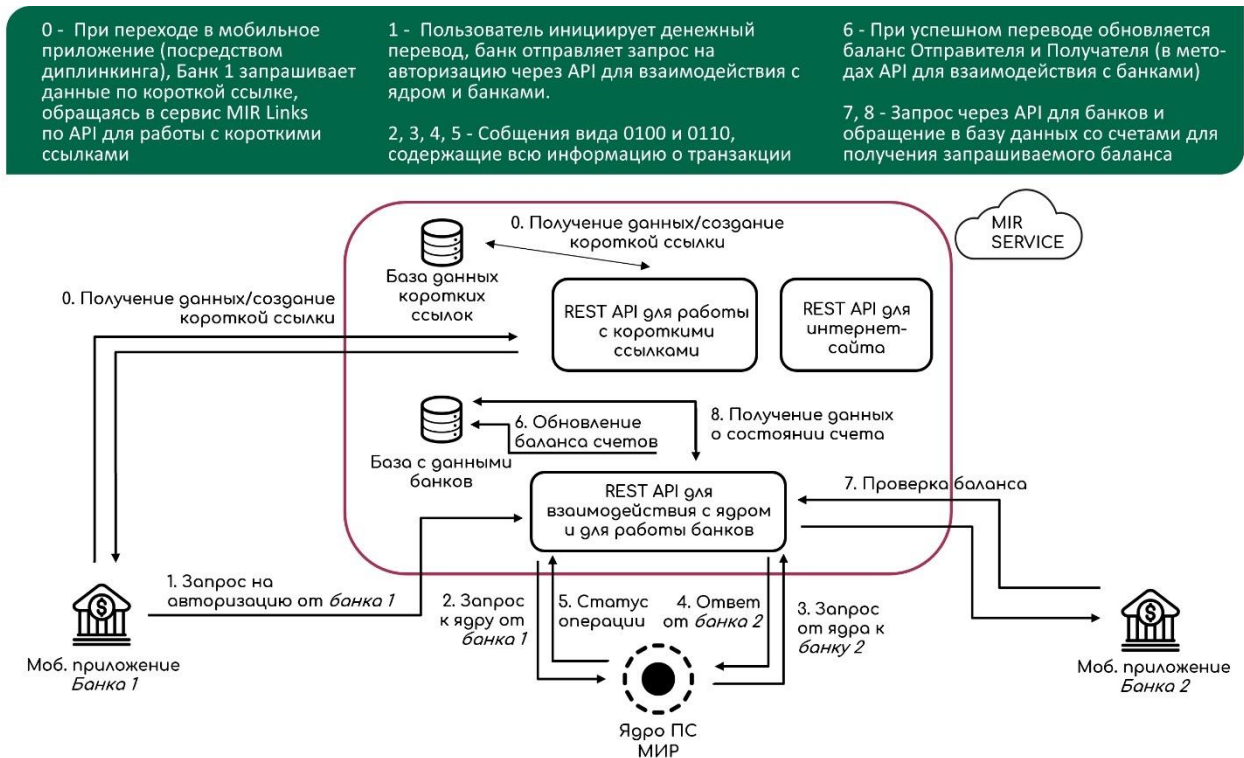
Представленный документ исчерпывающе описывает “общение” с сервисом, интерфейс, внешнее представление, какие элементы бизнес данных задействованы в информационных потоках, а также логику внутреннего устройства и правила взаимодействия компонентов.

1.2. Описание проблемы

Данный сервис предоставляет возможность клиенту банка создать уникальную короткую ссылку для перевода денег по ней. При этом не раскрываются данные получателя (номер карты /номер телефона).

2. Концептуальная архитектура сервиса

2.1. Архитектурная схема сервиса



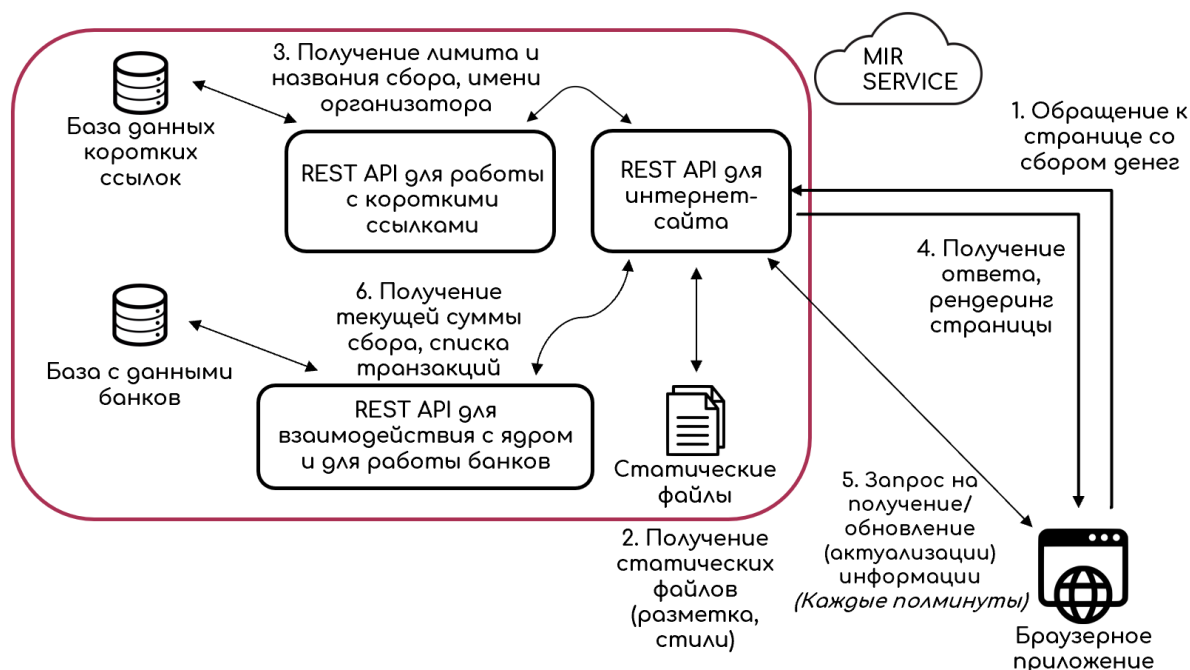


Рисунок 3. Общая схема работы при обращении к странице с денежным сбором

2.2. Описание компонентов

Компонент	Описание	Контекст использования
REST API для работы с короткими ссылками	REST API, реализующее методы управления: получения/отправки, создания/удаления коротких ссылок. Регулировка одноразовых ссылок, ссылок с датой истечения, ссылок для сборов денег (целей)	Создание ссылки в приложении банка, отображение списка созданных ссылок, удаление ссылки из приложения банка, перевод денег с использованием информации, полученной по короткой ссылке
REST API для работы взаимодействия с ядром и для работы с банками	REST API, реализующее методы получения баланса, обработки/сохранения/получения транзакций, взаимодействие с ядром (сообщения 0100 и 0110), получение 10 последних транзакций и актуальной суммы сбора денег	Денежный перевод по короткой ссылке, обработка запросов от браузера на получение актуальной информации о сборе (транзакции, текущая сумма сбора), получение баланса в мобильном приложении

REST API для интернет сайта	REST API, реализующее методы маршрутизации при обращении по адресу сайта, получение статических файлов, рендеринг страничек (подстановка серверных данных в статические файлы)	Обращение к страничке с выбором банков, обращение к страничке со сбором денег, динамическое обновление актуальных данных (каждые полминуты): последние транзакции, текущая сумма сбора
База данных коротких ссылок	<p>Хранение таких данных, как:</p> <ul style="list-style-type: none"> • Пользовательское название ссылки • Имя Получателя • Номер счета Получателя • Тип ссылки (одноразовая, многократная) • Публичное описание • Фиксированная сумма перевода • Лимит сбора (если это сбор денег) • Дата истечения ссылки (после которой ссылка перестает работать) • Статус ссылки (активная, истекшая, удаленная) • <u>Ключи идемпотентности</u> и ответы на соответствующие запросы 	Получение информации о ссылке, обновление статуса ссылки (при удалении, истечении, одноразовости перевода), создание ссылки с заданными параметрами
База данных с данными банков	<p>Хранение таких данных, как:</p> <ul style="list-style-type: none"> • Токен аутентификации • Идентификатор банка • Название банка • Номер счета Получателя • Имя Получателя • Баланс Получателя • Прошедшие транзакции: <ul style="list-style-type: none"> ○ ID транзакции ○ ID короткой ссылки ○ Сумма ○ Дата совершения транзакции ○ Номер счета получателя/отправителя 	Денежный перевод по короткой ссылке, обработка запросов от браузера на получение актуальной информации о сборе (транзакции, текущая сумма сбора), получение баланса в мобильном приложении

Мобильное приложение банка	<p>Приложение банка представляет собой функциональный интерфейс, который позволяет:</p> <ul style="list-style-type: none"> • Производить авторизацию, выходить из аккаунта • Создавать короткие ссылки для перевода с заданными параметрами • Просматривать созданные ссылки • Получать информацию о каждой ссылке (сколько всего перечислено, какой прогресс у денежного сбора, все транзакции-поступления по короткой ссылке) • Возможность скопировать, поделиться (Android sharing), удалить ссылку • Переходить из браузера в приложение (по диплинкингу) • Переводить денежные средства посредством короткой ссылки) 	Является инициатором практически всех различных действий в системе (будь то создание ссылки или перевод денег). Мобильное приложение запускает связанные процессы, описанные в соседнем столбце и запускает большинство методов API для работы приложения
Браузерное приложение	<p>Приложение на стороне клиента, которое запрашивает интернет-странички сервиса, такие как:</p> <ul style="list-style-type: none"> • Страница с выбором банков • Страница со сбором денег <p>На страничке со сбором денег динамически (т.е. без обновления страницы), запрашивает актуальную информацию по сбору (последние транзакции и текущую сумму сбора денег).</p>	

3. Концептуальная схема работы сервиса

3.1. Схема использования сервиса



Рисунок 4. Общая схема взаимодействия пользователей с сервисом



Рисунок 5. Общая схема взаимодействия при денежном переводе

3.2. Пользовательские сценарии

3.2.1. User script: Авторизация в банковском приложении

Предусловия: Наличие пароля от аккаунта банка

Базовый сценарий:

- Пользователь переходит в приложение своего банка. На главном экране отображается форма авторизации (по логину и паролю)
- Пользователь вводит корректные данные, и видит следующий экран (с балансом и меню взаимодействия)
- Если Пользователь введет некорректные данные, приложение об этом сообщит

3.2.2. User script: Создание короткой ссылки

Триггер: Необходимость получить денежный перевод, при этом не раскрывая своих данных

Базовый сценарий:

- Пользователь переходит в приложение своего банка, открывает раздел для создания коротких ссылок.
- Выбирает необходимые параметры (ключевое слово для валидации ссылки; одноразовая/многократная ссылка; сумма, которая будет введена при открытии ссылки по умолчанию)
- Нажимает кнопку сгенерировать ссылку и в ответ получает готовую короткую ссылку, которую можно скопировать в буфер обмена

3.2.3. User script: Перевод(оплата) по короткой ссылке

Предусловия: наличие ссылки, по которой будет совершен перевод (ссылка может быть полученным любым способом, пример: в личных сообщениях, на QR коде)

Базовый сценарий:

- Отправитель переходит по ссылке и попадает на сайт, где ему предлагают выбрать свой банк.
- Пользователь выбирает (кликает) на иконку/название своего банка
 - Если установлено банковское приложение, то происходит переадресация по диплинку и открывается приложение банка с уже введенными данными и суммой, покупателю остается только проверить секретную фразу и подтвердить перевод.
 - Если банковское приложение не установлено, пользователь может совершить перевод прямо со страницы, введя данные своей карты (аналог обычной покупки в интернет-магазине)

3.2.4. User script: Удаление короткой ссылки

Предусловия: наличие активной ссылки (по которой можно сделать перевод)

Базовый сценарий:

- Пользователь заходит в мобильное приложение своего банка
- Пользователь переходит в раздел со списком активных ссылок
- Пользователь выбирает (кликает) строчку со ссылкой, действие которой он хочет прекратить
 - Приложение отправляет запрос на удаление ссылки и получает результат этой операции
 - Статус ссылки меняется на *deleted*, и её дальнейшее использование становится невозможным

3.2.5. User script: Организация сбора денег

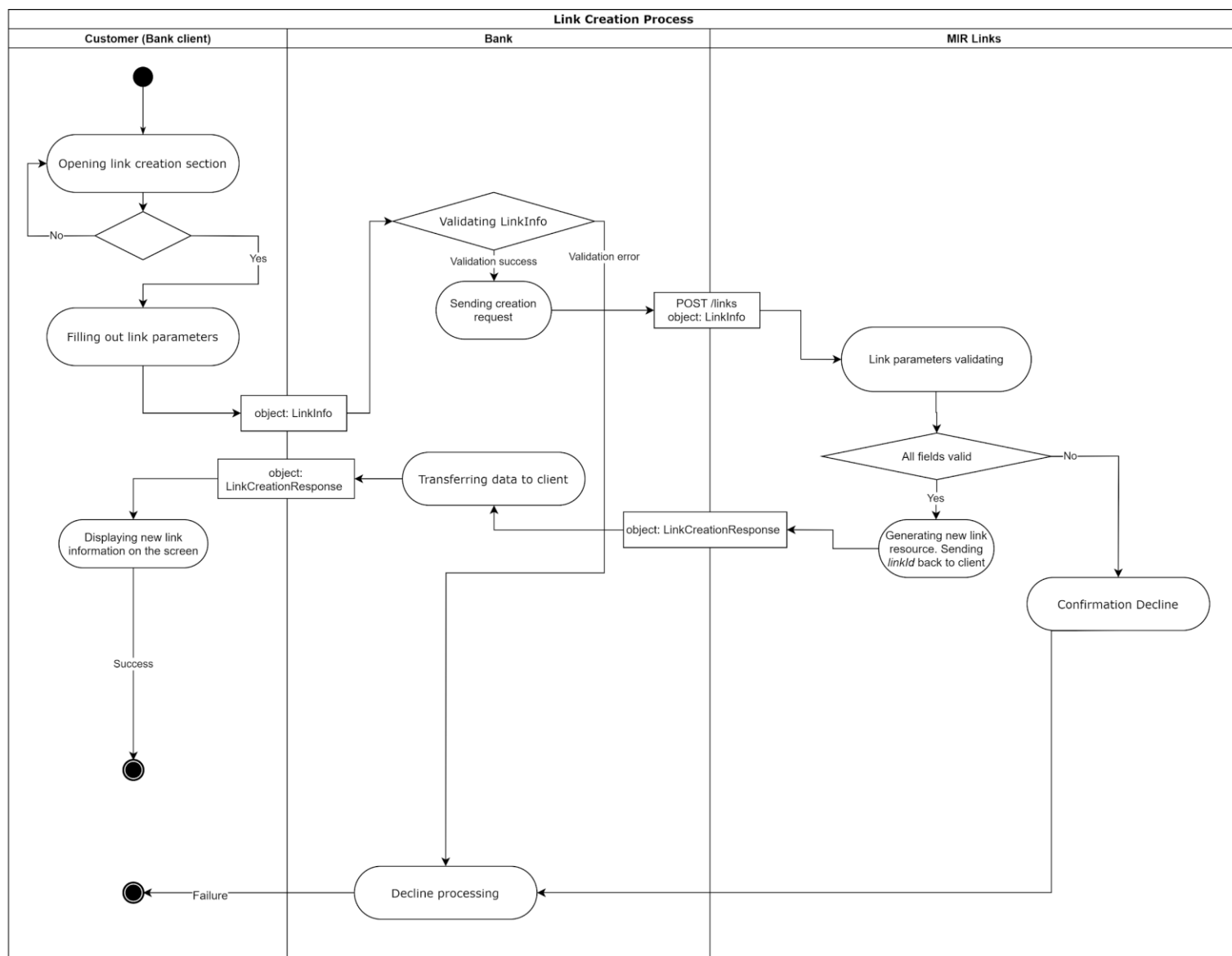
Предусловия: наличие ссылки, по которой будет организован сбор денег (ссылка может быть полученным любым способом, пример: в личных сообщениях, на QR коде)

Базовый сценарий:

- Отправитель переходит по ссылке и попадает на сайт, где представлена информация о текущей сумме сбора денег, о лимите сбора, об организаторе и последних транзакциях
- Отправитель ожидает несколько минут, при этом каждые полминуты автоматически обновляется информация о текущей сумме сбора и последних 10 транзакциях
- Пользователь имеет возможность навигации по последним транзакциям (нажимая на стрелочки), может просматривать несколько страниц транзакций (не более 10 транзакций на страницу)
- Пользователь кликает на кнопку «Перевести» и попадает на страницу с выбором банка. Дальнейшее взаимодействие происходит по скрипту, описанным в п.3.2.2

3.3. Схема взаимодействия модулей сервиса

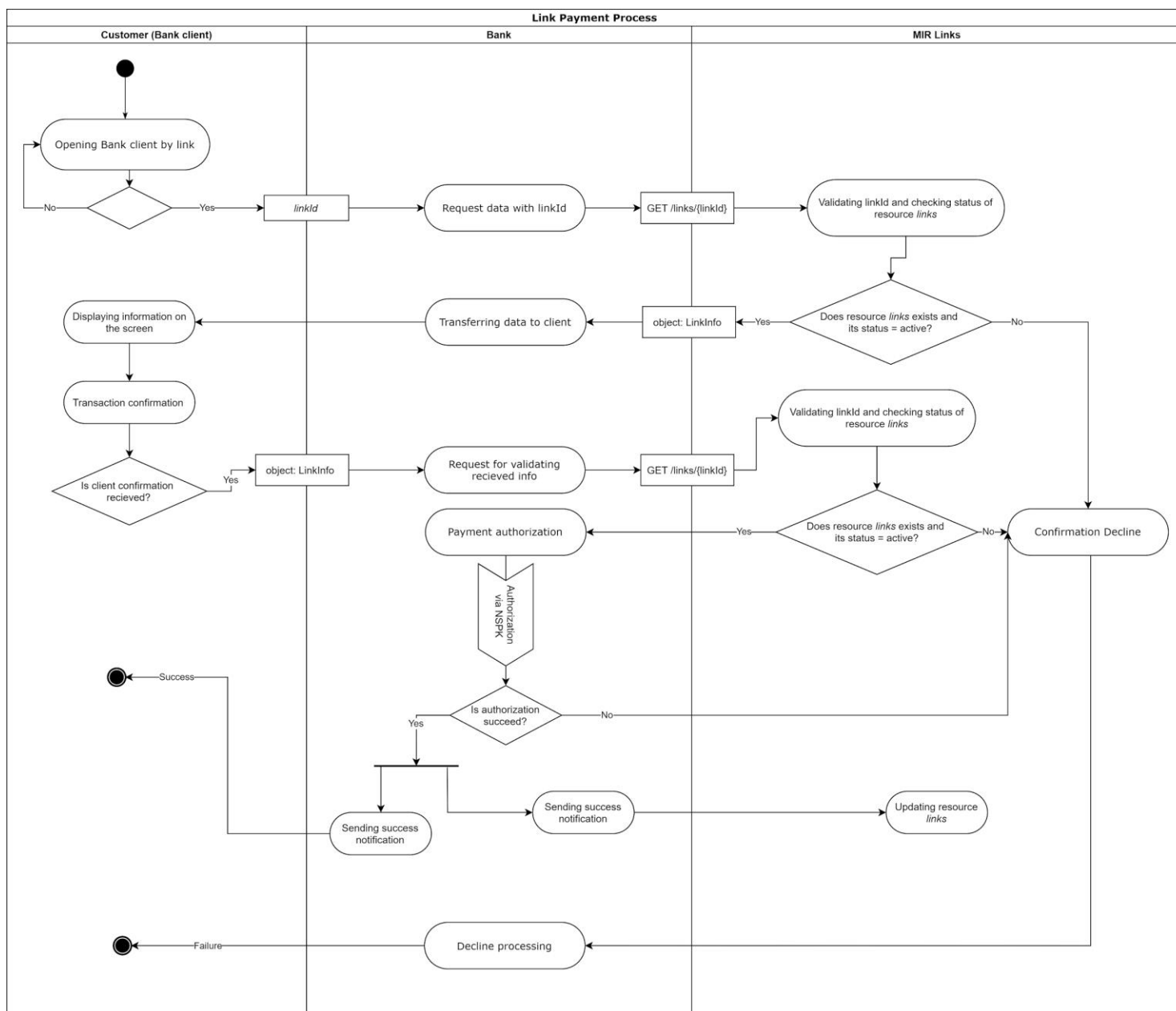
3.3.1. Взаимодействие компонентов при создании ссылки



3.3.1.1. Примечания к процессу создания ссылки

- Часть данных валидирует Банк (например номер счет, идентификатор Банка), однако остальные данные все равно дополнительно валидируются MIR Links, т.к. допускается возможность отправки Банком некорректной информации

3.3.2. Взаимодействие компонентов при совершении перевода

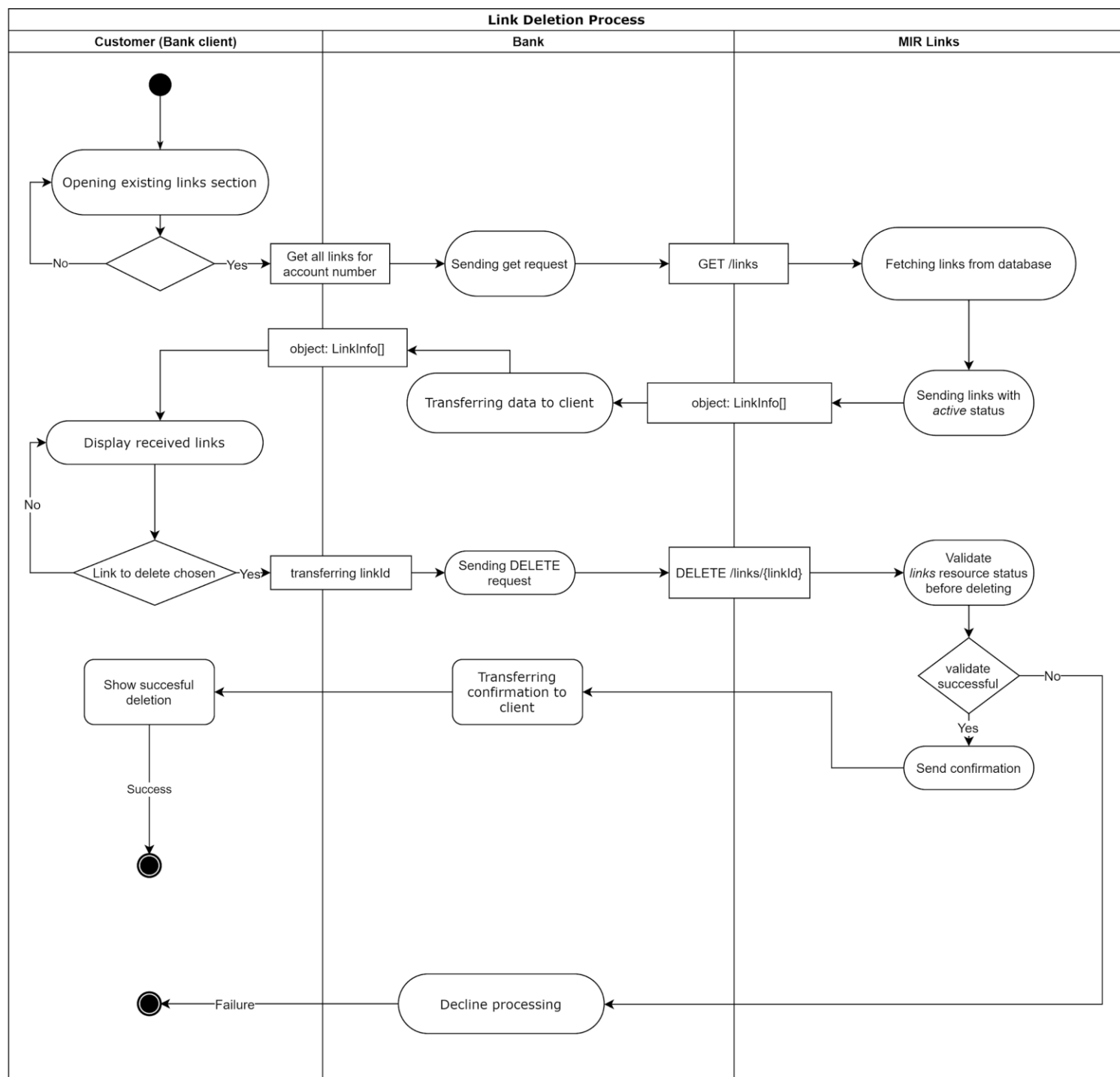


3.3.2.1. Примечания к процессу совершения перевода

- Так как между открытием банковского приложения и подтверждением перевода клиентом может пройти продолжительное время, за которое ссылка может:
 - Истечь (*expiryDate* < текущего времени)
 - Ссылка могла быть удалена, что ведет к невозможности перевода по ней
 - Если она была одноразовой, то кто-то мог оплатить раньше и ссылка "закроется"

Для этого после подтверждения перевода отправляется заново запрос на проверку данных в компонент MIR Links.

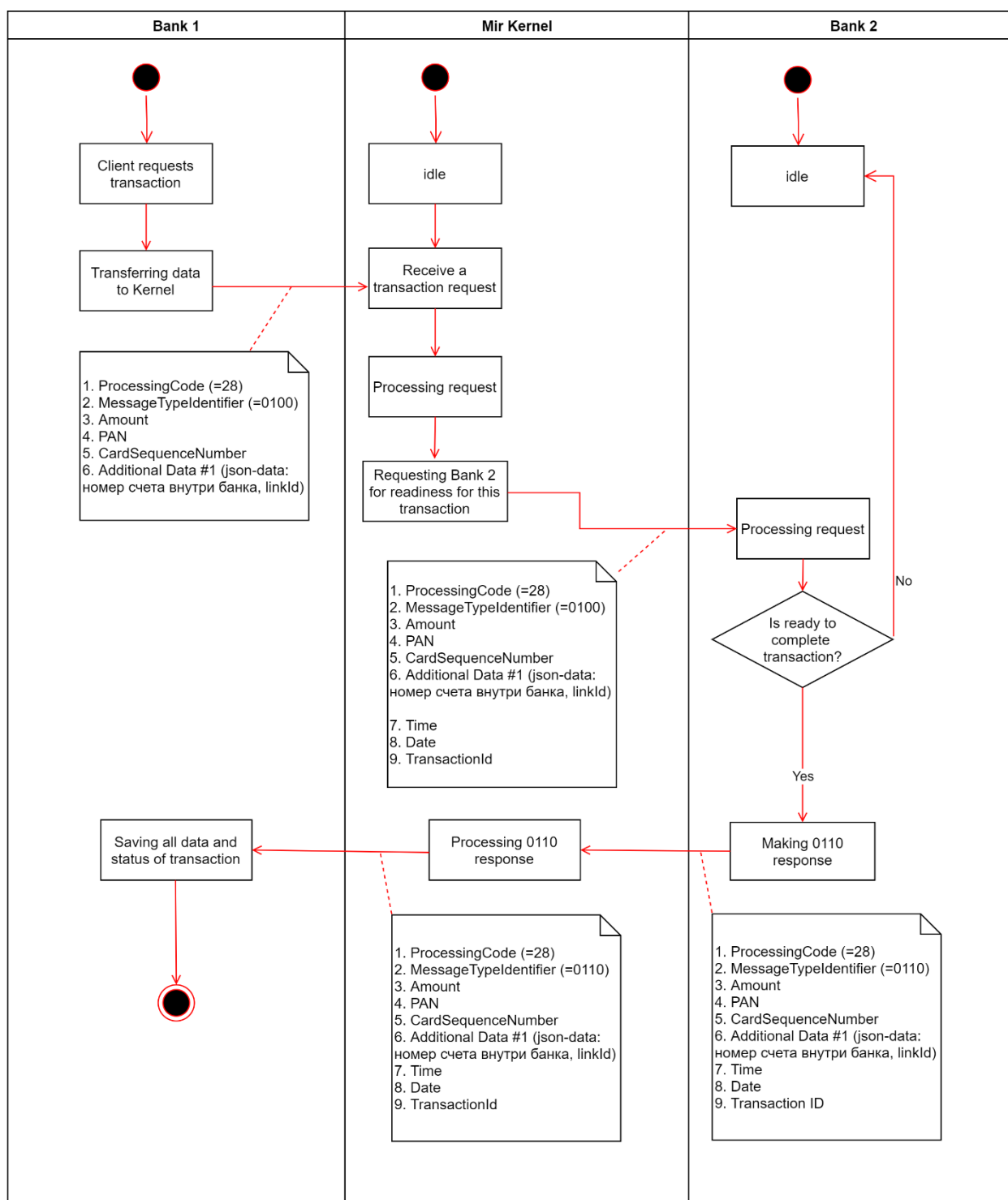
3.3.3. Взаимодействие компонентов при удалении ссылки



3.3.3.1. Примечания к процессу удаления ссылки

- В данном процессе Банк является только посредником в общении Приложения и сервера MIR Links. Запросы, ответы, состояния он может и не хранить.
- В данном процессе инициатором действия является тот же субъект, что и в процессе, описанном в пункте 3.3.1. То есть удалить ссылку может только тот клиент, который её создал.

3.3.4. Взаимодействие компонентов при денежном переводе (в контексте банков и ядра МИР)



3.3.4.1. Примечания к процессу взаимодействия банков и ядра

- В данном процессе Банк предоставляет следующие данные: Код операции, код сообщения, сумму операции, номер карты Отправителя, доп. номер (для идентификации карты), номер счета внутри банка (опционально), id короткой ссылки
- Ядро МИР в обязательном порядке добавляет следующие поля: время совершения операции, дата совершения операции, id транзакции в системе Ядра МИР

4. Описание методов API

4.1. Конечные точки(Endpoints)

HTTP Operation	Endpoint	Grant Type	Idempotency Key	Request Object	Response Object
POST	POST /links	Authorization Token	Да	LinkInfo	LinkCreationResponse или ErrorResponseObject
GET	GET /links/{linkId}	Authorization Token	Нет	N/A	LinkInfo
GET	GET /links	Authorization Token	Нет	N/A	LinkIdArray
DELETE	DELETE /links/{linkId}	Authorization Token	Нет	N/A	N/A
POST	POST /links/auth	Username and password	Нет	AuthData	AccountData
GET	GET /send	Authorization Token	Да	N/A	PaymentResponse или ErrorResponseObject
GET	GET /links/transactions/{linkId}	Authorization Token	Нет	N/A	TransactionsArray
GET	GET /links/last_transactions/{linkId}	Нет	Нет	N/A	LastTransactionsArray

4.1.1. POST /links

Метод POST, применяемый к эндпоинту /links, позволяет Банку запрашивать у сервиса MIR Links создание нового ресурса link. В запросе Банк должен передать информацию LinkInfo

4.1.1.1. Требования к заголовкам запроса

Header	Обязательный	Описание
AuthToken	Да	Токен авторизации Банка
x-Idempotency-Key	Да	Ключ идемпотентности

- Действие POST указывает сервису MIR Links, что ресурс link (короткая ссылка) должен быть создан.
- В результате действия POST компонент MIR Links создает ресурс link и отвечает LinkCreationResponse, содержащим уникальным идентификатором (linkId) созданного ресурса, по которому можно получить всю информацию о нем.

4.1.1.2. Статус (status)

Статус по умолчанию - Active (можно получить информацию о ресурсе) сразу после создания ресурса link

Статус
Active

4.1.1.3. Возможные ответы (Responses)

При успешном добавлении нового ресурса link:

- Response (HTTP Code: 200 OK, Body: LinkCreationResponse)

Возможные ошибки во время запроса:

- Response (HTTP Code: 400 Bad Request, Body: ErrorResponseObject) - при неверных переданных параметрах.
- Response (HTTP Code: 401 Unauthorized, Body: ErrorResponseObject) - при неверном переданном токене авторизации.

4.1.2. GET /links/{linkId}

Метод GET, применяемый к эндпоинту /links/{linkId}, позволяет банку извлечь данные из существующего и валидного ресурса link (возвращает LinkInfo).

4.1.2.1. Требования к заголовкам запроса

Header	Обязательный	Описание
AuthToken	Да	Токен авторизации Банка

4.1.2.2. Статус (status)

- Если во время запроса к данному ресурсу его параметр expiryDate имеет значение, меньшее чем дата запроса - статус ресурса изменится на Expired
Доступные коды состояния (Status Codes) для ресурса links приведены в таблице.

Статус
Active
Expired
Deleted

4.1.2.3. Возможные ответы (Responses)

Если ресурс link существует и не истёк:

- Response (HTTP Code: 200 OK, Body: LinkInfo)

Возможные ошибки при запросе:

- Response (HTTP Code: 401 Unauthorized, Body: ErrorResponseObject) - при неверном переданном токене авторизации.
- Response (HTTP Code: 404 Not found, Body: ErrorResponseObject) - когда ссылка не существует.
- Response (HTTP Code: 410 Gone, Body: ErrorResponseObject) - когда ссылка устарела (параметр expiryDate имеет значение, меньшее чем дата запроса)

4.1.3. GET /links

Метод GET, применяемый к эндпоинту /links, позволяет банку извлечь данные из ресурса *links*, состоящий из массива *linkId* (уникальные id валидных ссылок).

4.1.3.1. Требования к заголовкам запроса

Header	Обязательный	Описание
AuthToken	Да	Токен авторизации Банка

4.1.3.2. Query-параметры запроса

Параметр	Обязательный	Описание
AccountNumber	Да	Номер счета Получателя
BankIdentificationNumber	Да	Идентификатор Банка

Оба параметра в совокупности позволяют идентифицировать Получателя единственным образом.

4.1.3.3. Возможные ответы (Responses)

При успешном получении данных из ресурса:

- Response (HTTP Code: 200 OK, Body: LinkIdArray)

Возможные ошибки во время запроса:

- Response (HTTP Code: 400 Bad Request, Body: ErrorResponseObject) - при неверных переданных параметрах.
- Response (HTTP Code: 401 Unauthorized, Body: ErrorResponseObject) - при неверном переданном токене авторизации.
- Response (HTTP Code: 404 Not Found, Body: ErrorResponseObject) - при отсутствии действующих ссылок у номера счета, переданного в запросе

4.1.4. DELETE /links/{linkId}

Метод DELETE, применяемый к эндпоинту /links/{linkId}, позволяет банку удалить данный ресурс link.

4.1.4.1. Требования к заголовкам запроса

Header	Обязательный	Описание
AuthToken	Да	Токен авторизации Банка

4.1.4.2. Статус (status)

- Если во время запроса на удаление, ресурс link имеет параметр expiryDate меньше чем дата запроса - статус ресурса будет изменён на Expired
- Если ресурс link был успешно удален в сервисе MIR Links, то статус данного ресурса будет изменен на Deleted (этот ресурс будет ещё некоторое время хранится на сервисе, но будет недоступен для чтения с помощью API)

Статус
Deleted

4.1.4.3. Возможные ответы (Responses)

При успешном удалении нового ресурса link:

- Response (HTTP Code: 200 OK, Body: N/A)

При ошибке во время удаления:

- Response (HTTP Code: 401 Unauthorized, Body: ErrorResponseObject) - при неверном переданном токене авторизации.
- Response (HTTP Code: 404 Not found, Body: ErrorResponseObject) - когда ссылка не существует.
- Response (HTTP Code: 410 Gone, Body: ErrorResponseObject) - когда ссылка устарела (параметр expiryDate имеет значение, меньшее чем дата запроса, т.е ссылка имеет статус Expired или Deleted)

4.1.5. POST /links/auth

Метод POST, применяемый к эндпоинту /links/auth, позволяет банку авторизовать своего клиента по логину и паролю

4.1.5.1. Требования к телу запроса

Request Object	Обязательный	Описание
AuthData	Да	Логин и пароль пользователя

4.1.5.2. Возможные ответы (Responses)

При успешной авторизации клиента:

- Response (HTTP Code: 200 OK, Body: AccountData)

При ошибке во время авторизации:

- Response (HTTP Code: 401 Unauthorized, Body: ErrorResponseObject) – при неверном логине и пароле
- Response (HTTP Code: 404 Not found, Body: ErrorResponseObject) - когда логин пользователя не существует.

4.1.5.3. Примечание

- Переданный пароль хешируется с помощью алгоритма SHA3-512 (Кеccak), затем сравнивается с хешем в базе данных, на основе чего и принимается решение, об авторизации

-

4.1.6. GET /links/send

Метод GET, применяемый к эндпоинту /links/send, позволяет банку инициировать процесс перевода денег на другой счет в другом банке

4.1.6.1. Требования к заголовкам запроса

Header	Обязательный	Описание
AuthToken	Да	Токен авторизации Банка
x-Idempotency-Key	Да	Ключ идемпотентности
AccountNumber	Да	Счет Получателя денег (куда перечислить средства)
Sum	Да	Сумма перевода
LinkId	Да	Id короткой ссылке, по которой проходит перевод

4.1.6.2. Возможные ответы (Responses)

При успешной переводе денег:

- Response (HTTP Code: 200 OK, Body: PaymentResponse)

При ошибке во время перевода денег:

- Response (HTTP Code: 400 Bad request, Body: ErrorResponseObject) – если параметры (заголовки) перевода некорректны
- Response (HTTP Code: 400 Bad request, Body: ErrorResponseObject) – если не хватает денег на счете Отправителя
- Response (HTTP Code: 404 Not found, Body: ErrorResponseObject) – если номер счета Получателя или Отправителя денег не существует

4.1.6.3. Примечание

- При успешном переводе генерируется 64-символьный хэш транзакции и записывается в базу данных наравне с другими данными по транзакции (номер счета Получателя/Отправителя, суммы транзакции, даты и id короткой ссылки)

4.1.7. GET /links/transactions/{linkId}

Метод GET, применяемый к эндпоинту /links/transactions/{linkId}, позволяет банку получить список всех транзакций по конкретной короткой ссылке

4.1.7.1. Требования к заголовкам запроса

Header	Обязательный	Описание
AuthToken	Да	Токен авторизации Банка

4.1.7.2. Возможные ответы (Responses)

При успешном запросе списка ссылок:

- Response (HTTP Code: 200 OK, Body: TransactionsArray)

При ошибке во время запроса списка ссылок:

- Response (HTTP Code: 401 Not authorized, Body: ErrorResponseObject) – неверный токен авторизации

4.1.7.3. Примечание

- При запрашивании списка транзакций у несуществующей ссылки – метод вернет пустой список

-

4.1.8. GET /links/last_transactions/{linkId}

Метод GET, применяемый к эндпоинту /links/last_transactions/{linkId}, позволяет любому пользователю (т.к. публичный метод) получить список из 10 последних транзакций по ссылке (с ограниченной информацией), текущую сумму сбора и общее количество транзакций

4.1.8.1. Требования к заголовкам запроса

Header	Обязательный	Описание
offset	Да	Сдвиг относительно количества всех транзакций

4.1.8.2. Возможные ответы (Responses)

При успешном запросе списка ссылок:

- Response (HTTP Code: 200 OK, Body: LastTransactionsArray)

4.1.8.3. Примечание

- Параметр *offset* показывает, начиная с какой по счету транзакции следует выводить следующие 10 транзакций (в порядке убывания даты транзакции)
- При несуществующей ссылке вернет пустой список

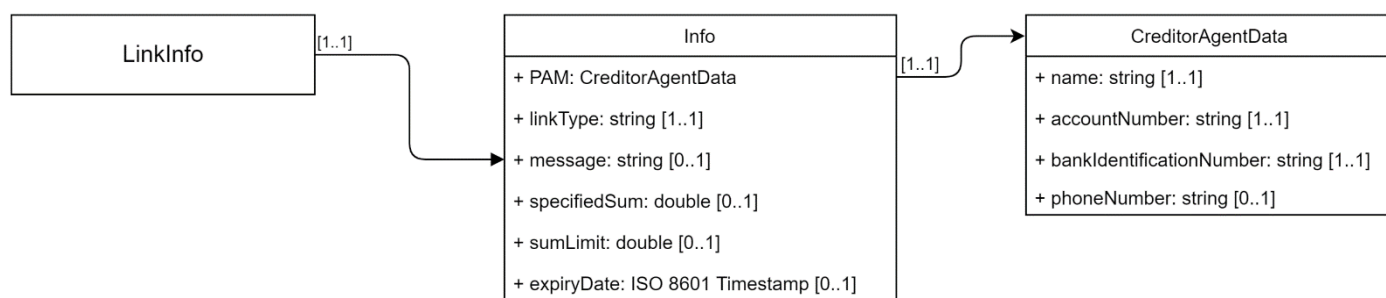
4.2. Модель данных

4.2.1. LinkInfo

Объект класса LinkInfo, состоит из секции:

- info, который представлен следующими входными элементами:
 - PAM, который представлен объектом CreditorAgentData
 - *linkType*
 - *message*
 - *specifiedSum*
 - *sumLimit*
 - *expiryDate*

4.2.1.1. UML - диаграмма



4.2.1.2. Примечание

- Поле *linkType* содержит один из следующих типов:

<i>linkType</i>	Описание
one-time	После первого перевода по этой ссылке, её status автоматически меняется на deleted
perpetual	Количество переводов по этой ссылке неограниченно

- Поле *message* представлено строкой - комментарий получателя к данному переводу по ссылке
Если поле *message* отсутствует:
Получатель не указал комментарий к переводу
- Поле *specifiedSum* может иметь следующие значения:

<i>specifiedSum</i>	Описание
{double_number}	Вещественное число до двух знаков после запятой, обозначающее сумму перевода, указанную Получателем. При этом Отправитель не может изменить сумму перевода

Если поле *specifiedSum* отсутствует:

сумма перевода не задана Получателем, т.е. при переводе по ссылке, Отправитель может указать свою сумму перевода

- Поле *sumLimit* может иметь следующие значения:

<i>sumLimit</i>	Описание
{double_number} > 0	Вещественное число до двух знаков после запятой, обозначающее лимит сбора денег
0	По ссылке не предусмотрен сбор денег

- Поле *expiryDate* может иметь следующие значения:

<i>expiryDate</i>	Описание
{ISO 8601 Timestamp}	Дата, после которой получение информации методом GET /{linkId} становится недоступным. Формат: YYYY-MM-DD hh:mm:ss

Если поле *expiryDate* отсутствует:

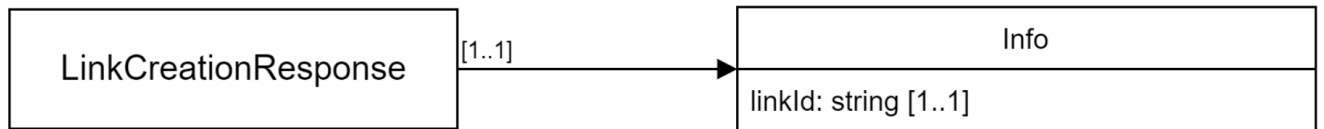
Дата окончания ссылки не задана, т.е. ссылка будет доступна, пока для нее будет вызван метод DELETE /links/{linkId}

4.2.2. LinkCreationResponse

Объект класса LinkCreationResponse, состоит из секции:

- info, который представлен единственным элементом:
 - *linkId*

4.2.2.1. UML - диаграмма



4.2.2.2. Примечание

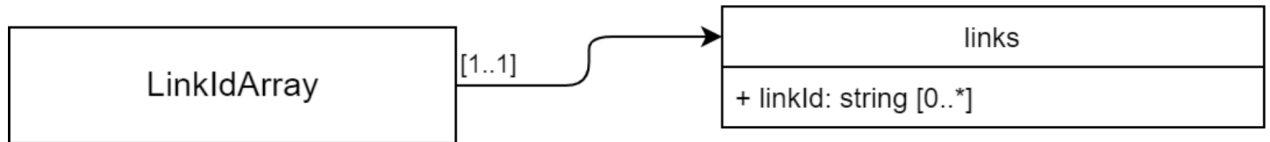
- Поле *linkId* представлено уникальной (в пределах сервиса MIR Links) строкой, который является указателем (id) на ресурс link

4.2.3. LinkIdArray

Объект класса LinkCreationResponse, состоит из секции:

- *links*, который представлен массивом следующих элементов:
 - *linkId*

4.2.3.1. UML - диаграмма



4.2.3.2. Примечание

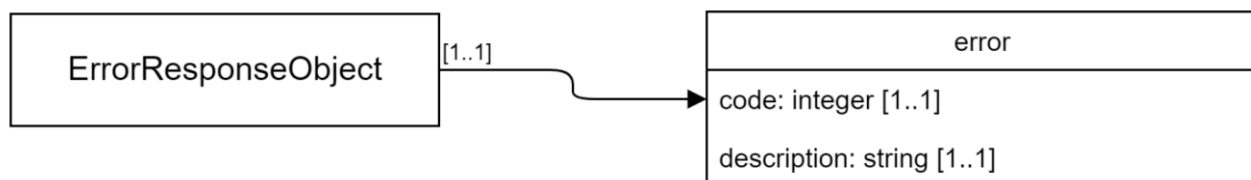
- Массив строк, состоящий из *linkId*, содержит уникальные id ссылок, которые являются доступными для использования на момент запроса

4.2.4. ErrorResponseObject

Объект класса ErrorResponseObject, состоит из секций:

- *error*, который представлен следующими элементами:
 - *code*
 - *description*

4.2.4.1. UML - диаграмма



4.2.4.2. Примечание

- Поле *description* содержит детальное описание ошибки
- Поле *code* может содержать одно из следующих значений:

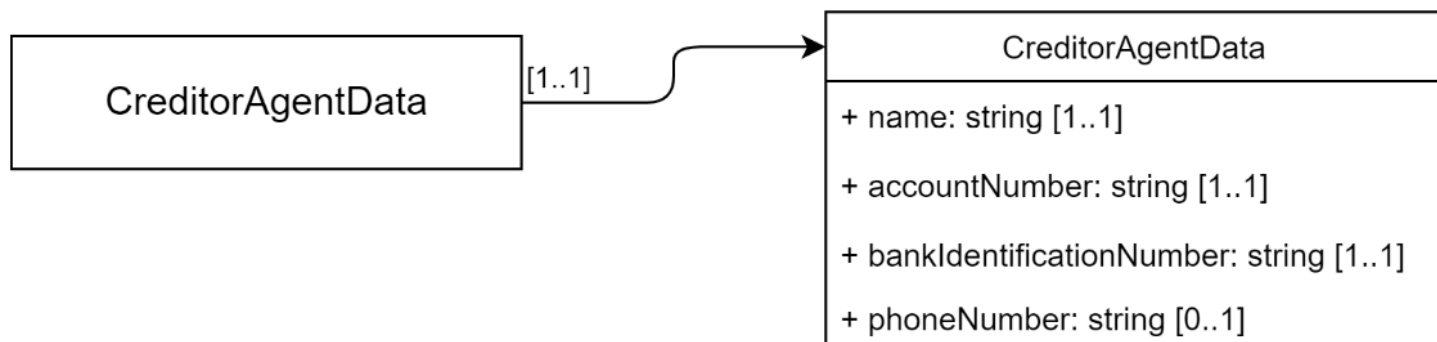
<i>code</i>	Описание
1	Неверный токен авторизации
2	Передан неправильный запрос. (Подробнее об ошибке в поле <i>description</i>)
3	Запрашиваемый ресурс не найден

4.2.5. CreditorAgentData

Объект класса CreditorAgentData, состоит из элементов:

- *name*
- *accountNumber*
- *bankIdentificationNumber*
- *phoneNumber*

4.2.5.1. UML - диаграмма



4.2.5.2. Примечание

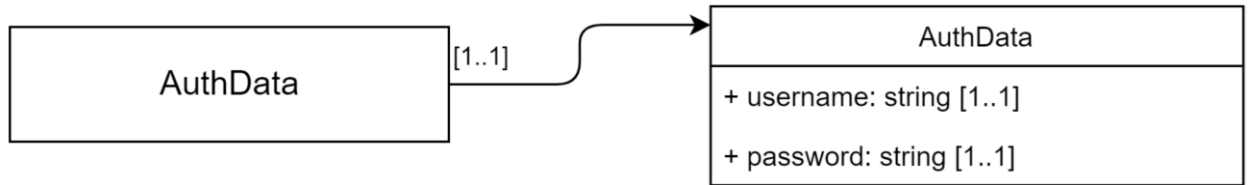
- Поле *name* содержит имя и фамилию получателя
- Поле *accountNumber* является уникальным номером счета в пределах Банка (20 цифр)
- Поле *bankIdentificationNumber* является уникальным идентификатором Банка
- Поле *phoneNumber* содержит номер телефона получателя

4.2.6. AuthData

Объект класса AuthData, состоит из элементов:

- *username*
- *password*

4.2.6.1. UML – диаграмма



4.2.6.2. Примечание

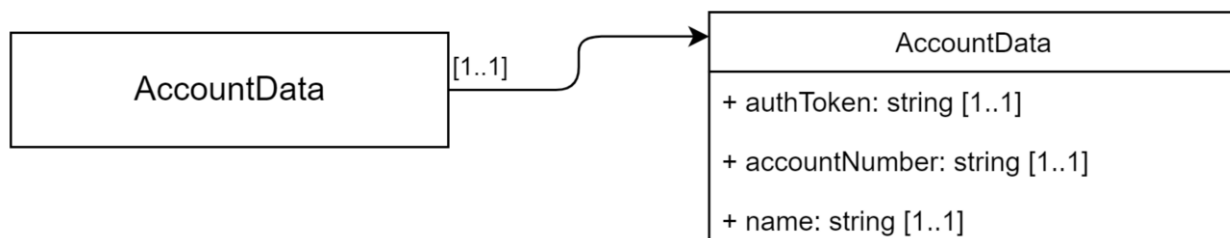
- Поле *username* содержит логин клиента банка
- Поле *password* содержит пароль от аккаунта клиента

4.2.7. AccountData

Объект класса AccountData, состоит из элементов:

- *authToken*
- *accountNumber*
- *name*

4.2.7.1. UML – диаграмма



4.2.7.2. Примечание

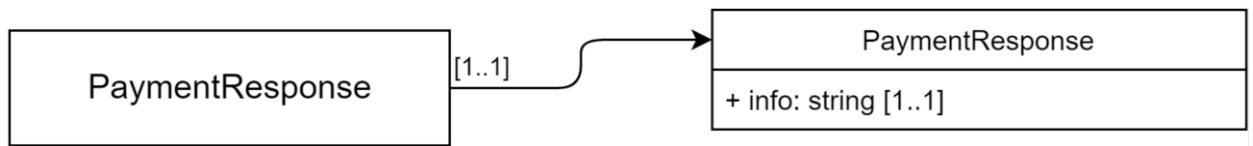
- Поле *authToken* содержит 20 латинских букв или цифр, служит для авторизации банка и доступа к закрытым методам
- Поле *accountNumber* является уникальным номером счета в пределах Банка (20 цифр)
- Поле *name* содержит имя и фамилию получателя

4.2.8. PaymentResponse

Объект класса PaymentResponse, состоит из элементов:

- *info*

4.2.8.1. UML – диаграмма



4.2.8.2. Примечание

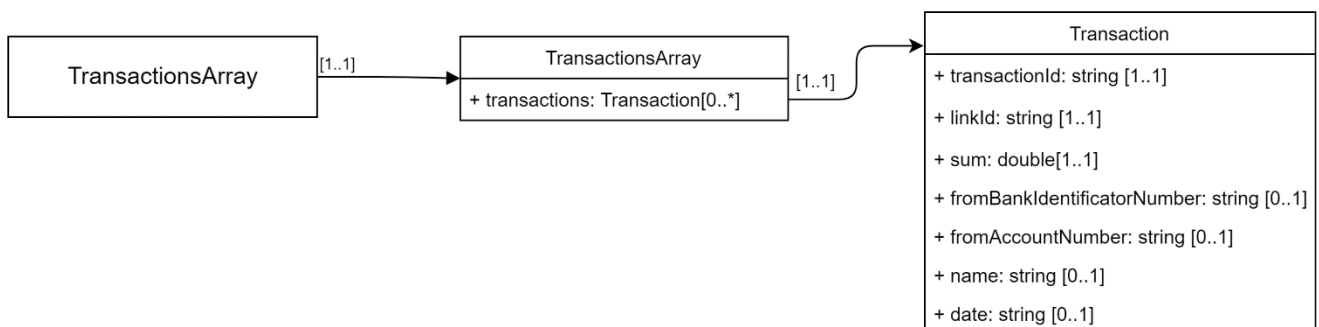
- При корректном взаимодействии, поле *info* принимает единственное значение: “*success*”

4.2.9. TransactionsArray

Объект класса TransactionsArray, состоит из элементов:

- *transactions*, который в свою очередь состоит из:
 - *transactionId*
 - *linkId*
 - *sum*
 - *fromBankIdentificationNumber*
 - *fromAccountNumber*
 - *name*
 - *date*

4.2.9.1. UML – диаграмма



4.2.9.2. Примечание

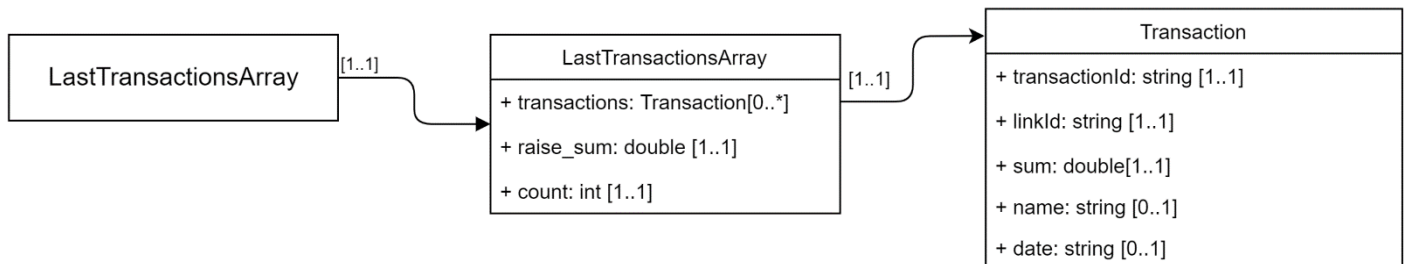
- Поле *transactionId* содержит 64 латинских букв и/или цифр, является уникальным среди всех транзакций через сервис MIR Links
- Поле *linkId* представлено уникальной (в пределах сервиса MIR Links) строкой, который является указателем (id) на ресурс link
- Поле *sum* содержит сумму транзакции
- Поле *fromBankIdentificationNumber* содержит БИК банка Отправителя
- Поле *fromAccountNumber* содержит 20 номер счета Отправителя внутри банка
- Поле *name* содержит имя и фамилию Отправителя
- Поле *date* содержит дату совершения транзакции

4.2.10. LastTransactionsArray

Объект класса LastTransactionsArray, состоит из элементов:

- *transactions*, который в свою очередь состоит из:
 - *transactionId*
 - *linkId*
 - *sum*
 - *name*
 - *date*
- *count*, общее количество транзакций
- *raise_sum*, текущая сумма сбора

4.2.10.1. UML – диаграмма



4.2.10.2. Примечание

- Поле *transactionId* содержит 8 первых латинских букв и/или цифр настоящего поля *transactionId*
- Поле *linkId* представлено уникальной (в пределах сервиса MIR Links) строкой, который является указателем (id) на ресурс link
- Поле *sum* содержит сумму транзакции Поле *name* содержит имя и фамилию Отправителя
- Поле *date* содержит дату совершения транзакции
- Так как метод является публичным – часть информации отсутствует, а id транзакции состоит из 8 первых символов настоящего поля *transactionId*

5. Описание Android приложения

5.1. Компоненты приложения

5.1.1. Основные компоненты

- Activities
- Adapters
- Api
- DataClasses
- MainActivity

5.1.2. Основные activities

5.1.2.1. CreateNewLinkActivity

Экран для создания новой ссылки. На этапе заполнения данных происходит их валидация, на сервер отправляются корректные данные при помощи классов из DataClasses с использованием ApiClient (о нем подробнее написано дальше). При успешном создании ссылки её адрес копируется в буфер обмена устройства пользователя.

5.1.2.2. ExistingLinksActivity

На экране показывается список всех личных ссылок. Список формируется при помощи запроса к серверной части MirLinks и предоставления токенов, необходимых для этого действия.

5.1.2.3. LinkInfoActivity

На экране просмотра конкретной короткой ссылки реализован следующий функционал:

- Кнопка “Показать QR-код” – открывает экран с QR-кодом (содержит короткую ссылку) и его расшифровкой
- Кнопка “Скопировать” – копирует ссылку для перевода в буфер обмена.
- Кнопка “Поделиться” – открывает окно для размещения ссылки для перевода в других источниках.
- Кнопка “Удалить” – удаляет ссылку и возвращает пользователя в ExistingLinks Activity.

Также отображается общая сумма сбора по ссылке, прогресс бар – текущий прогресс сбора денег, список всех транзакций по этой короткой ссылке, с указанием имени Отправителя, суммы и даты транзакции.

5.1.2.4. LoginActivity

Экран для аутентификации пользователя. Если введены данные, то происходит попытка авторизоваться в приложении, при помощи запроса к серверной части MirLinks с использованием ApiAuth. При неправильном логине или пароле, отображается ошибка, полученная с сервера. После успешной аутентификации данные пользователя сохраняются для дальнейшего использования.

5.1.2.5. SendMoneyActivity

Экран для перевода денег по короткой ссылке, открывается по deep link.

5.1.3. Основные adapters:

5.1.3.1. LinkInfoRecyclerAdapter

Адаптер для RecyclerView для отображения списка ссылок в приложении.

Каждый элемент списка представлен внутренним названием ссылки, id ссылки, датой истечения и фиксированной суммой перевода, если таковая была указана при создании ссылки

5.1.3.2. TransactionRecyclerAdapter

Адаптер для RecyclerView для отображения списка транзакций.

Api:

- ApiClient

Получение объекта Retrofit (который реализует интерфейс ApiAuthService) для осуществления запросов, объявленных в интерфейсе ApiAuthService. Запросы происходят с использованием паттернов реактивного программирования (с использованием RxJava). Это позволяет не останавливать UI и производить запросы в других потоках, а получать и обрабатывать значения при помощи паттерна Observable.

- ApiClientService

Интерфейс, описывающий запросы и необходимые к ним параметры. Здесь представлена основная логика по взаимодействию с линками на серверной части.

- ApiAuth

Получение объекта для осуществления аутентификации.

- ApiAuthService

Интерфейс для аутентификации в приложении.

DataClasses – используются для взаимодействия с серверной частью:

- LinkObject

Объект, содержащий все необходимые классы, описывающие все сущности короткой ссылки (присутствуют вложенные data классы)

- LoginResult

Data класс для получения информации из запроса при успешном входе.

MainActivity – основное окно приложения, откуда происходит переход в другие Activity, при помощи кнопок. На главном экране отображается текущий баланс и номер счета, обновить баланс можно свайпом сверху-вниз.

5.2. Требования к экранам

5.2.1. Login Activity

Позволяет пользователям авторизоваться в приложении. Если пользователь был авторизован ранее, происходит переадресация на Main Activity.

Элементы ввода данных:

1. Логин – уникальный идентификатор пользователя
2. Пароль – пароль учетной записи пользователя

Элементы управления:

1. Кнопка “Войти” – осуществляет авторизацию. В случае успеха направляет пользователя в Main Activity.

Возможные ошибки:

1. Несуществующий уникальный идентификатор (логин) пользователя.
2. Несовпадение данных для авторизации пользователя.

5.2.2. Main Activity

Отображает номер счета и баланс. Позволяет пользователю перейти к основным функциям приложения (Создание новой ссылки, просмотр существующих ссылок, осуществление перевода), а также выйти из аккаунта.

Элементы управления:

1. Кнопка “Создать новую ссылку” – Направляет пользователя в CreateNewLinkView Activity.
2. Кнопка “Посмотреть существующие” – Направляет пользователя в ExistingLinks Activity.
3. Кнопка “Осуществить перевод” – открывает сканнер QR-кодов и в случае успешного считывания переадресовывает пользователя на SendMoney Activity.
4. Кнопка “Выход” – Осуществляет деавторизацию пользователя. Направляет пользователя в Login Activity.

5.2.3. CreateNewLinkView Activity

Отображает номер счета пользователя. Позволяет пользователям создать новую ссылку для переводов.

Обязательные элементы ввода данных:

1. Название ссылки – Приватное название ссылки, которое будет отображаться в списке ссылок.
2. Тип – тип ссылки (единоразовый/многократный перевод). По умолчанию – многократный.
3. Время действия – дата, до которой будут возможны переводы по данной ссылке. По умолчанию – бессрочный.

Опциональные элементы ввода данных:

1. Сумма – фиксированная сумма перевода по ссылке.
2. Публичное описание – описание, которое видят люди при попытке перевода денег по данной ссылке.
3. Публичная цель – целевая сумма сбора.

Элементы управления:

1. Кнопка “Создать новую ссылку” – создает ссылку с описанными параметрами. В случае успеха направляет пользователя в Main Activity и копирует новую ссылку в буфер обмена.

Возможные ошибки:

1. Незаполненное или состоящие только из пробелов поле названия ссылки.

5.2.4. ExistingLinks Activity

Отображает все созданные пользователем ссылки. Позволяет открыть полную информацию о каждой ссылке.

Каждый элемент списка представлен внутренним названием ссылки, id короткой ссылки, датой истечения и фиксированной суммой перевода (если было указано при создании).

Элементы управления:

1. Каждый элемент кликабельный. В случае нажатия направляет пользователя в LinkInfo Activity для выбранной ссылки.

5.2.5. LinkInfo Activity

Отображает полную информацию о ссылке: публичное название; текущую сумму сбора; публичную цель; прогресс сбора; ссылку для пополнения; список транзакций по данной ссылке с указанием отправителя, даты и суммы транзакции).

Элементы управления:

1. Кнопка “Показать QR-код” – открывает экран с QR-кодом (содержит короткую ссылку) и его расшифровкой
2. Кнопка “Скопировать” – копирует ссылку для перевода в буфер обмена.
3. Кнопка “Поделиться” – открывает окно для размещения ссылки для перевода в других источниках.
4. Кнопка “Удалить” – удаляет ссылку и возвращает пользователя в ExistingLinks Activity.

5.2.6. SendMoney Activity

Отображает счет и владельца, которому будет осуществлен перевод, а также публичное сообщение для более полной информации о переводе.

Элементы ввода данных:

1. Сумма – сумма перевода. Если сумма по ссылке фиксирована, то она будет отображаться без возможности изменения.

Элементы управления:

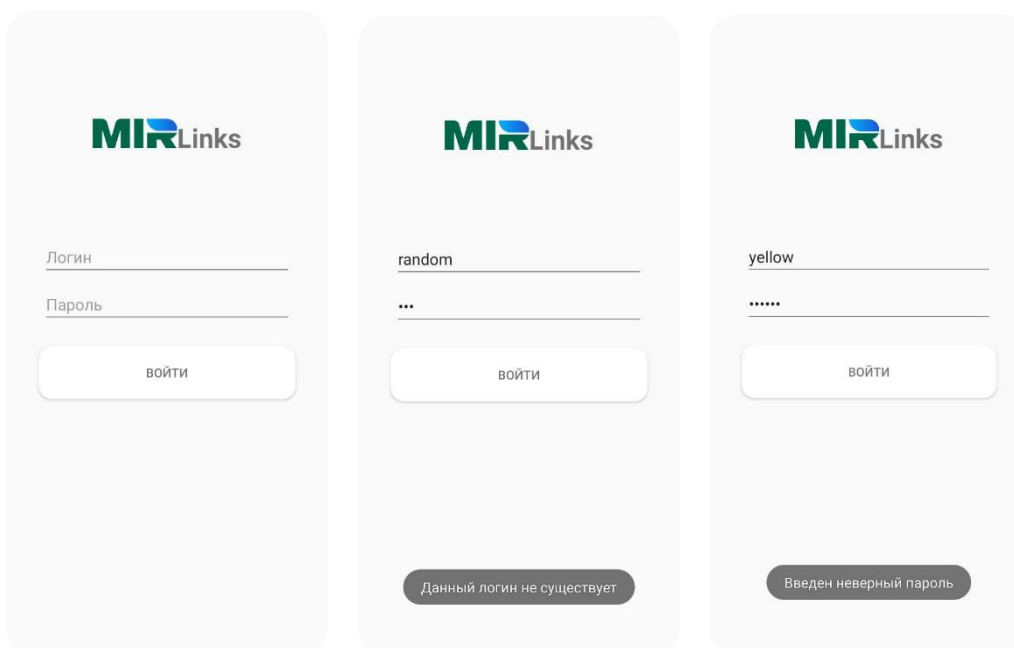
1. Кнопка “Осуществить перевод” – осуществляет транзакцию с указанными параметрами. В случае успеха направляет пользователя в Main Activity.

Возможные ошибки:

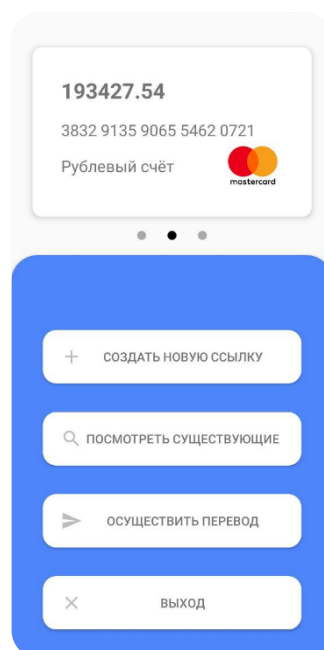
1. Не введена сумма перевода.
2. Осуществление перевода по собственной ссылке (самому себе).

5.3. Экраны

5.3.1. Login Activity



5.3.2. Main Activity



5.3.3. CreateNewLinkView Activity

MIRLinks

Счет3832 9135 9065 5462 0721

Название ссылки

Тип

Единоразовый перевод

Многоразовый перевод

Время действия

Бессрочный

Выбрать дату

Опциональные настройки

Сумма:сумма по умолчанию

Публичное описание:

описание

Публичная цель

сумма для сбора

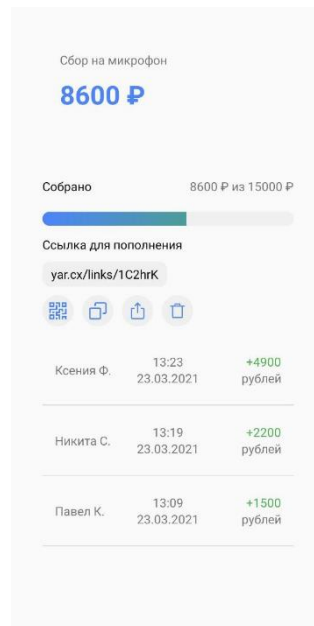
+

создать новую ссылку

5.3.4. ExistingLinks Activity

mylink		
ZLXtST	∞	
Моя первая ссылка		
YX3Nqj	∞	
Test name 63		
xgUAIF	∞	500 ₺
Test name 62		
AY1zxb	∞	1337 ₺
Test name 58		
iBaH34	∞	
Test name 46		
nSQ2SW	∞	
Test name 41		
zm55cx	2020-03-27	9690.5 ₺
Test name 40		
ISsPIg	∞	
Test name 36		

5.3.5. LinkInfo Activity



5.3.6. SendMoney Activity

MIR Links

Счет 1432 1985 2329 8966 4322

Имя Екатерина З.

Сообщение Оплата за проект

Сумма Сумма для перевода

➤ **ОСУЩЕСТВИТЬ ПЕРЕВОД**

6. Тестирование

6.1. Тестирование web-версии

6.1.1. Веб-интерфейс страницы сбора денег

Интерфейс страницы накоплений (рис. 6) состоит из следующих частей:

- Блок с информацией о названии сбора и кнопкой для перевода;
- Блок с информацией о целевой и собранной суммах накопления, шкалой прогресса;
- Список из десяти последних переводов, состоящий из суммы, имени и даты перевода;
- Стрелки навигации, позволяющие перейти к предыдущим или следующим переводам.

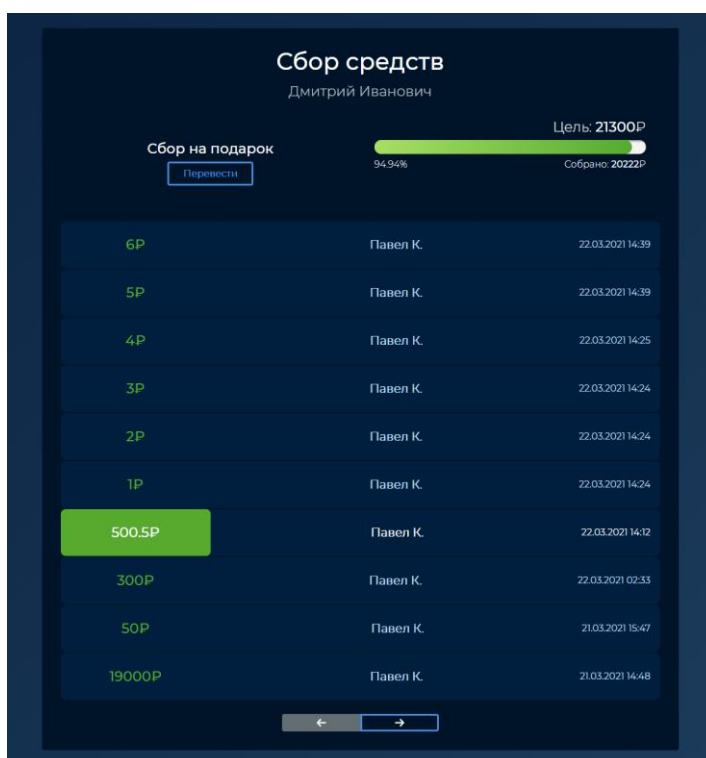


Рис. 6 - страница сбора средств

6.1.2. Тестирование кнопки перевода

При нажатии на кнопку “Перевести” (рис. 7) произойдёт переход на страницу переводов, при наведении изменится внешний вид кнопки.

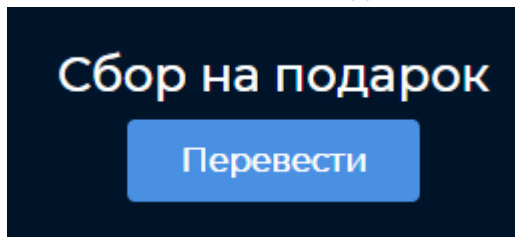


Рис. 7 - кнопка “Перевести” для перехода на страницу переводов

6.1.3. Тестирование блока с информацией о прогрессе сбора средств

Текущая собранная сумма и соответствующая ей шкала прогресса изменяются без перезагрузки страницы при обновлении списка транзакций:



Рис. 8 - блок с информацией о прогрессе сбора средств

Теперь осуществим тестовый перевод суммы в 1000 рублей и посмотрим на изменения в интерфейсе (рис. 9):



Рис. 9 - текст с собранной суммой и шкала изменились после перевода

6.1.4. Тестирование автообновления списка переводов

Список переводов обновляется с заданной периодичностью (20 секунд), при этом во время выполнения серверного запроса список блокируется и появляется соответствующий индикатор загрузки (рис. 10)

1000R	Павел К.	23.03.2021 12:41
6R	Павел К.	22.03.2021 14:39
5R	Павел К.	22.03.2021 14:39
4R	Павел К.	22.03.2021 14:25
3R	Павел К.	22.03.2021 14:24
2R	Павел К.	22.03.2021 14:24
1R	Павел К.	22.03.2021 14:24
500.5R	Павел К.	22.03.2021 14:12
300R	Павел К.	22.03.2021 02:33
50R	Павел К.	21.03.2021 15:47

Рис. 10 - индикатор обновления списка

Также, после тестирования предыдущего пункта, видно, как появился новый элемент в списке транзакций с суммой в 1000 рублей и список автоматически обновился.

Тестирование кнопок навигации

Кнопки навигации (рис. 11) автоматически становятся заблокированными (обозначаются серым цветом) при невозможности загрузить предыдущие или следующие транзакции.



Рис. 11 - кнопки для навигации по страницам транзакций

Попробуем перейти на следующую страницу (рис. 12)

19000₽	Павел К.	21.03.2021 14:48
150.35₽	Павел К.	09.03.2021 08:00
200.15₽	Павел К.	09.03.2021 07:29





Рис. 12 - вторая страница транзакций

Список транзакций после перехода на следующую страницу изменился, как и состояние кнопок навигации.

6.1.5. Веб-интерфейс страницы выбора банка

Интерфейс страницы выбора банка (рис. 13) состоит из следующих частей:

- Строка поиска по названию банка;
- Список карточек банков с логотипами банков.

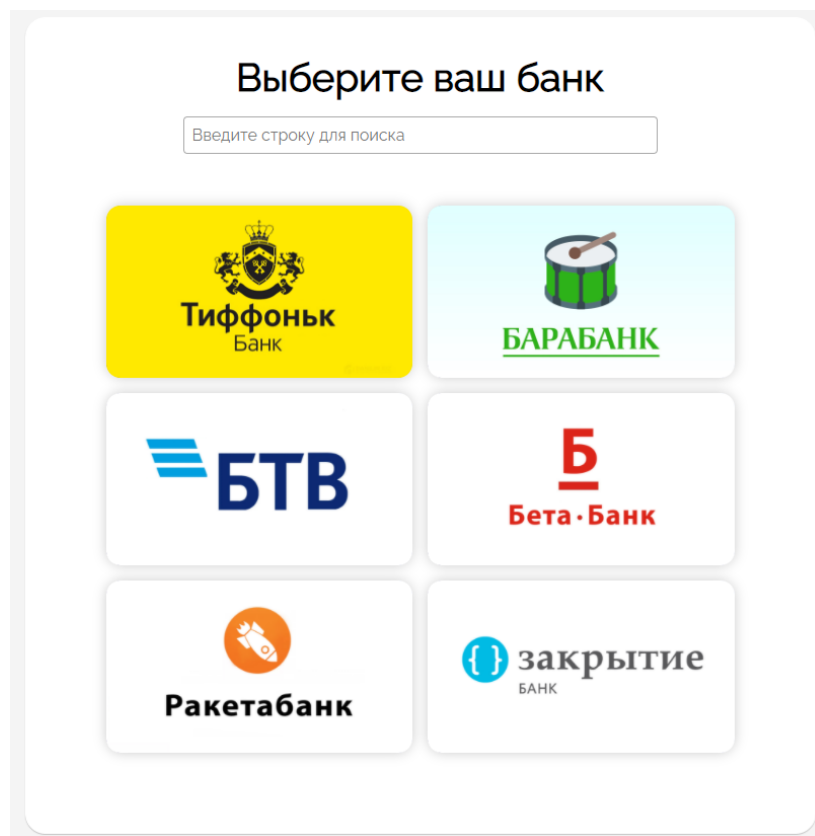


Рис. 13 - страница поиска банков

6.1.6. Тестирование поиска банка

Попробуем ввести “Банк” в строку поискового запроса. Список банков автоматически изменится без необходимости обновления страницы (рис. 14)

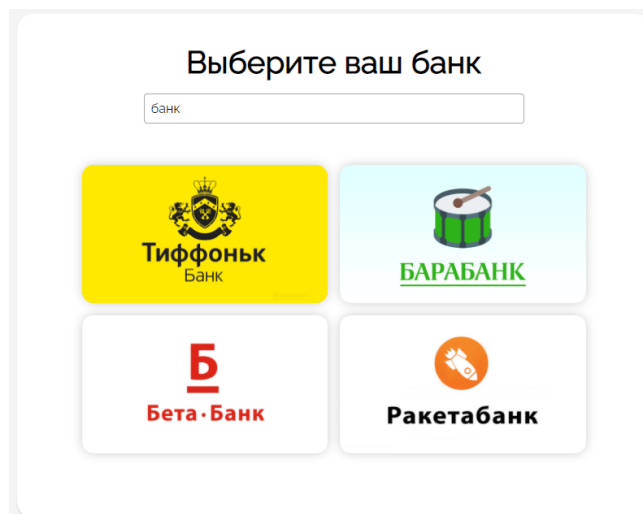


Рис. 14 - Список банков с примененной фильтрацией

6.2. Тестирование Android приложения

6.2.1. Тестирование экрана авторизации

При первом открытии приложения нас должен встретить экран авторизации (рис. 15):

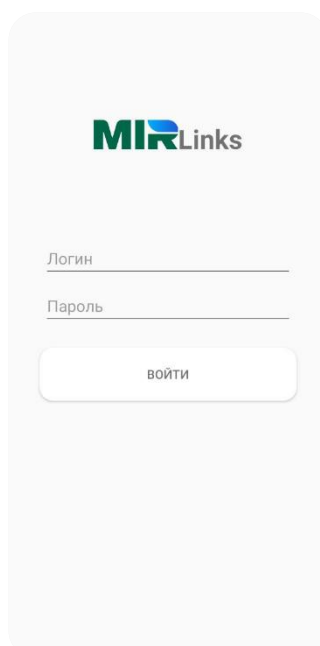


Рис. 15 – Экран авторизации

Попробуем ввести некорректный (несуществующий логин) “random” (рис. 16):



Рис. 16 – Несуществующий логин
Приложение об этом нам немедленно сообщает.

Попробуем ввести верный логин, но неправильный пароль (рис. 17):

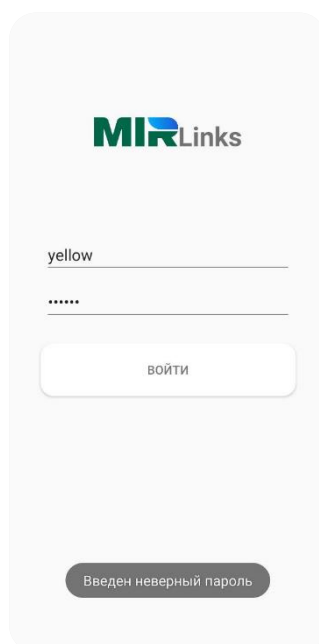


Рис. 17 – Неверный пароль
Приложение также об этом нам немедленно сообщает.

6.2.2. Тестирование главного экрана

Главный экран имеет интерфейс в точности соответствующий требованиям (рис. 18):

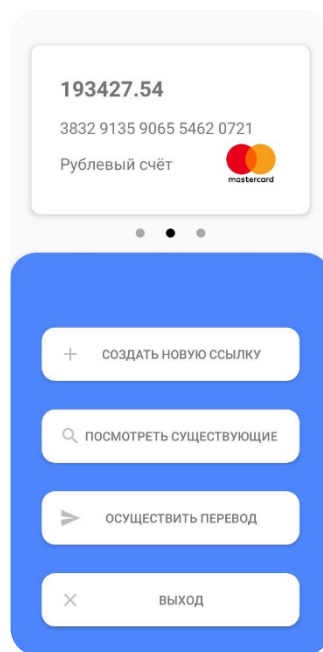


Рис. 18 – Главный экран

- При нажатии на “Создать новую ссылку”, открывается соответствующий интерфейс создания ссылки
- При нажатии на “Посмотреть существующие”, открывается соответствующий интерфейс со списком всех созданных ссылок
- При нажатии на “Осуществить перевод”, открывается соответствующий интерфейс сканера QR-кода
- При нажатии на “Выход” нас перекидывает на экран авторизации

Все кнопки работает в соответствии с их назначением

6.2.3. Тестирование экрана создания ссылки

Экран создания ссылки имеет интерфейс в точности соответствующий требованиям (рис. 19):

The screenshot shows the 'MIR Links' app interface. At the top is the 'MIR Links' logo. Below it, the account number 'Счет' is displayed as '3832 9135 9065 5462 0721'. There is a text input field for 'Название ссылки' (Link name). The 'Тип' (Type) section has two radio buttons: 'Единоразовый перевод' (One-time transfer) and 'Многоразовый перевод' (Recurring transfer), with the latter being selected. The 'Время действия' (Validity period) section has two radio buttons: 'Бессрочный' (Indefinite) and 'Выбрать дату' (Select date), with the former being selected. The 'Опциональные настройки' (Optional settings) section includes a 'Сумма' (Amount) field set to 'сумма по умолчанию' (default amount), a 'Публичное описание:' (Public description:) field, a checkbox for 'Публичная цель' (Public purpose), and a 'сумма для сбора' (collection amount) field. At the bottom is a button with a plus sign and the text 'создать новую ссылку' (create new link).

Рис. 18 – Экран создания ссылки

- Если ввести пустое название ссылки, приложение об этом сообщит и не будет отправлять запрос на создание ссылки
- Если ввести корректные данные, и нажать на кнопку “Создать новую ссылку” -> только что созданная ссылка скопируется в буфер обмена

6.2.4. Тестирование экрана перевода денег

Экран перевода денег имеет интерфейс в точности соответствующий требованиям (рис. 19):

MIR Links

Счет 1432 1985 2329 8966 4322

Имя Екатерина З.

Сообщение
Оплата за проект

Сумма Сумма для перевода

➤ ОСУЩЕСТВИТЬ ПЕРЕВОД

Рис. 19 – Экран перевода денег по ссылке

- Если ввести корректную сумму перевода, и нажать на кнопку “Осуществить перевод” -> у создателя ссылки обновится список транзакций по этой ссылке, а баланс уменьшится и увеличится у Отправителя и Получателя, соответственно
- Если создателю ссылки попробовать отправить по ней деньги, то приложение выдаст ошибку (рис. 20)

MIR Links

Счет 1432 1985 2329 8966 4322

Имя Екатерина З.

Сообщение
Оплата за проект

Сумма Сумма для перевода

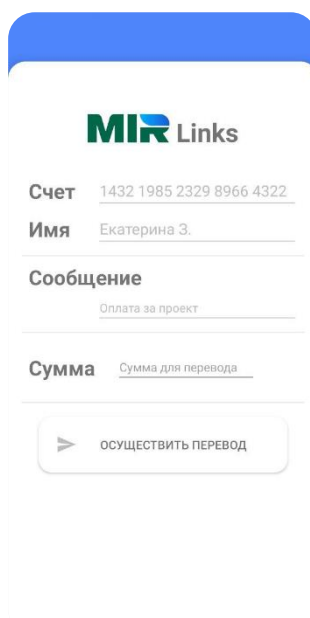
➤ ОСУЩЕСТВИТЬ ПЕРЕВОД

Вы не можете совершить перевод самому себе

Рис. 20 – Экран перевода денег по ссылке (ошибка)

6.2.5. Тестирование экрана информации о короткой ссылке

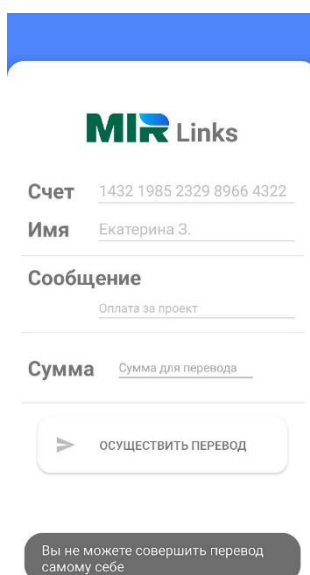
Экран перевода денег имеет интерфейс в точности соответствующий требованиям (рис. 19):



The screenshot shows a mobile application interface for MIR Links. At the top is a blue header. Below it, the MIR Links logo is displayed. The form contains the following fields: 'Счет' (Account) with the value '1432 1985 2329 8966 4322', 'Имя' (Name) with the value 'Екатерина З.', 'Сообщение' (Message) with the value 'Оплата за проект', and 'Сумма' (Sum) with the value 'Сумма для перевода'. At the bottom is a button with a right arrow and the text 'ОСУЩЕСТВИТЬ ПЕРЕВОД'.

Рис. 19 – Экран перевода денег по ссылке

- Если ввести корректную сумму перевода, и нажать на кнопку “Осуществить перевод” -> у создателя ссылки обновится список транзакций по этой ссылки, а баланс уменьшится и увеличится у Отправителя и Получателя, соответственно
- Если создателю ссылки попробовать отправить по ней деньги, то приложение выдаст ошибку (рис. 20)



This screenshot is identical to the one in Figure 19, showing the MIR Links transfer form. However, at the bottom of the screen, there is an additional dark grey error message box that reads: 'Вы не можете совершить перевод самому себе' (You cannot transfer money to yourself).

Рис. 20 – Экран перевода денег по ссылке (ошибка)

