

Министерство образования и науки Российской Федерации  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)  
Институт прикладной математики и компьютерных наук  
Кафедра теоретических основ информатики

ДОПУСТИТЬ К ЗАЩИТЕ В ГЭК

Руководитель ООП

Доцент, к.т.н.

А.Л.Фукс

« 7 » 06 2018 г.

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА**  
**РАЗРАБОТКА КЛИЕНТСКОЙ КОМПОНЕНТЫ ВЕБ-ПРИЛОЖЕНИЯ**  
**«ДЕНЕЖНЫЕ ПЕРЕВОДЫ»**

по основной образовательной программе подготовки бакалавров  
020302 – Фундаментальная информатика и информационные технологии

Сорокин Артем Витальевич

Руководитель ВКР,  
ст. преподаватель кафедры  
программирования ФПМК  
А.А.Мирютов

« 7 » 06 2018 г.

Автор работы  
студент группы № 1441

А.В. Сорокин

## Оглавление

Введение .....	3
1. Обзор предметной области .....	4
2. Обзор и анализ существующих клиентских фреймворков .....	6
2.1. Обзор фреймворка Angular .....	7
2.2. Обзор фреймворка Vue.....	8
2.3. Обзор фреймворка React .....	8
3. Реализация приложение «Денежные переводы» .....	10
Заключение .....	17
Список использованных источников и литературы .....	18
Приложение А. Доступное API. ....	19
<u>TransferTariffInfo</u> .....	19
<u>TransferTariff</u> .....	20
<u>Person</u> .....	21
<u>Transfer</u> .....	22
<u>TransferRefundRequest</u> .....	25
<u>TransferReceiver</u> .....	25
<u>TransferReceivingOption</u> .....	26
<u>TransferReceivingInfo</u> .....	27
<u>TransferReceivingFixation</u> .....	28
<u>TransferMetaData</u> .....	29

## Введение

В 2015 году для единства технологического стека клиентской и серверной компонент веб-приложения «Золотая Корона – Денежные переводы», его клиентская компонента была реализована на фреймворке GWT (Google Web Toolkit), который в данный момент является устаревшим и невостребованным. Ввиду этого возникла необходимость заменить GWT на другой фреймворк.

**Цель данной работы** заключается в выборе подходящего фреймворка для клиентской части веб-приложения и реализации клиентской компоненты на этом фреймворке.

В рамках указанной цели были поставлены и успешно решены следующие задачи:

- изучить и проанализировать существующие клиентские фреймворки;
- на основе этого анализа выбрать наиболее подходящий фреймворк для реализации клиентской компоненты веб-приложения «Золотая Корона – Денежные переводы»;
- реализовать клиентскую компоненту веб-приложения «Золотая Корона – Денежные переводы» на основе выбранного фреймворка.

## 1. Обзор предметной области

Широкая либерализация внешнеэкономической деятельности, с одной стороны, и неравномерность развития государств, с другой, — главные факторы, стимулирующие глобальные миграционные процессы. Перемещение трудовых ресурсов из относительно отсталых регионов в экономически развитые есть, в конечном счете, следствие указанной неравномерности, а результирующий вектор движения мигрантов направлен, соответственно, в сторону передовых стран, обладающих необходимой производственной инфраструктурой. Связанные же с производственной деятельностью мигрантов трансграничные переводы денежных средств осуществляются в противоположном направлении.

Денежный перевод — это перевод (движение) денежных средств от отправителя к получателю с помощью операторов платежных систем через национальные или международные платежные системы с целью зачисления денежных средств на счет получателя или выдачи ему их в наличной форме. В структуре денежного перевода всегда присутствует отправитель, получатель и посредник — оператор платежной системы, взимающий за свои услуги определенную плату.

В современном мире денежные переводы стали неотъемлемой частью жизни. Необходимость перевести близким, находящимся за тысячи километров, деньги в срочном порядке или же самому получить необходимую сумму — обыденные задачи для рядового пользователя.

За 2017 год физическими лицами из России за рубеж было осуществлено денежных переводов на 9,186 млрд долларов в эквиваленте (против 7,335 млрд в 2016-м): в страны СНГ было отправлено переводов на 7,004 млрд долларов, в страны дальнего зарубежья — на 2,183 млрд (в 2016 году эти показатели составляли 5,596 млрд и 1,739 млрд соответственно)<sup>1</sup>.

В январе 2018 года объем денежных переводов, отправленных физическими лицами из России за границу и поступивших физлицам в РФ из-за рубежа, уменьшился в долларовом эквиваленте по сравнению с предыдущим месяцем, но вырос в годовом выражении (по сравнению с январем-2017). Об этом свидетельствуют данные сводки ЦБ «Трансграничные переводы, осуществленные через платежные системы в 2018 году» и статистика предыдущих периодов<sup>2</sup>.

---

<sup>1</sup> <http://www.banki.ru/news/lenta/?id=10305831> (дата обращения 03.03.2018)

<sup>2</sup> [https://www.cbr.ru/statistics/CrossBorder/35-rem\\_18.xlsx](https://www.cbr.ru/statistics/CrossBorder/35-rem_18.xlsx) (дата обращения 03.03.2018)

В настоящее время владельцы банковских карт международных платежных систем VISA, MasterCard, Maestro или МИР любого банка Российской Федерации могут отправить денежный перевод через сайт с помощью веб-приложения «Золотая Корона».

«Золотая Корона – Денежные переводы» - это сервис мгновенных безадресных денежных переводов без открытия счета, который предоставляет возможность отправить перевод любым удобным для вас способом: наличными в офисах банков, салонах-магазинах крупнейших ритейлеров в РФ и в пунктах партнеров «Золотой Короны» в ближнем и дальнем зарубежье, онлайн с банковской карты с получением наличных в широкой сети точек обслуживания, а также с карты на карту.

## 2. Обзор и анализ существующих клиентских фреймворков

Веб-приложение является технически сложным и объемным программным продуктом, реализующим масштабную бизнес-логику и сложные взаимодействия с конечным пользователем. В связи с чем встает вопрос о максимальном упрощении его реализации и последующей поддержки. Для упрощения и унификации кода разработчики веб-приложений, как правило, используют такие вспомогательные средства как фреймворки.

Фреймворки – это программные продукты, которые упрощают создание и поддержку технически сложных или нагруженных проектов. Фреймворк, как правило, содержит только базовые программные модули, а все специфичные для проекта компоненты реализуются разработчиком на их основе. Тем самым достигается не только высокая скорость разработки, но и большая производительность и надёжность решений.

На сегодняшний день существует огромное количество клиентских фреймворков. Для того, чтобы их сравнить и выбрать наиболее подходящий, необходимо сформулировать критерии, по которым будет происходить оценка фреймворков. Критерии должны учитывать, как специфику проекта, так и диктуемые бизнесом требования, а именно:

- необходимость длительной поддержки проекта;
- добавления нового функционала;
- поиск кадров для работы с проектом;
- возможность использования его для других проектов компании
- открытость фреймворка, чтобы гарантировать его безопасность для конечного пользователя.

Проанализировав наши потребности и потребности бизнеса, мы вывели итоговые критерии выбора фреймворка:

- зрелость фреймворка (время на рынке и его востребованность);
- перспективность фреймворка (активная поддержка и инвестиции автором);
- объем сообщества вокруг фреймворка (количество разработчиков, поддерживающих данный фреймворк);
- количество разработчиков, владеющих фреймворком;
- возможность реализации крупного веб-приложения;
- совместимость с диктуемой стратегией разработки;
- доступность исходного кода и наличие документации.

Для сравнения нами было выбрано 3 наиболее крупных современных фреймворка, которые могут заменить GWT, а именно: Angular, Vue и React. Далее рассмотрим более подробно каждый из них.

## 2.1. Обзор фреймворка Angular

Angular представляет собой фреймворк от компании Google для создания клиентских приложений. Прежде всего он нацелен на разработку SPA-решений (Single Page Application), то есть одностраничных приложений.

Первый выпуск Angular – сентябрь 2016 года, и с тех пор фреймворк получил уже 6 мажорных версий.

Одной из ключевых особенностей Angular является то, что он использует в качестве языка программирования TypeScript. Поэтому перед началом работы рекомендуется ознакомиться с основами данного языка.

Angular 5 предоставляет такую функциональность, как двустороннее связывание, позволяющее динамически изменять данные в одном месте интерфейса при изменении данных модели в другом, шаблоны, маршрутизация и так далее.

Angular является наследником AngularJS. Однако это не новая версия AngularJS, а принципиально новый фреймворк, который не обладает обратной совместимостью с AngularJS, что привело к разделению общества Angular-разработчиков на тех, кто поддерживает проекты на старой версии, и тех, кто реализует проекты на новой версии. Вследствие этого, не было произведено естественной миграции со старого инструментария на новый, что, в свою очередь, замедлило развитие нового фреймворка – Angular.

Кроме того, чтобы максимально результативно использовать данный фреймворк, необходимо в качестве основного языка разработки клиентской компоненты веб-приложения использовать TypeScript, что для нас является усложнением процесса разработки, единственной целью которого было бы использование фреймворка Angular.

В качестве оценки популярности мы взяли недельное количество скачиваний фреймворка на сайте npmjs.com, что составило 200 032 скачиваний в неделю<sup>3</sup>. В качестве оценки количества разработчиков были взяты данные с сайта поиска работы HeadHunter, который сообщает о наличии 2 367 резюме по России, в которых указано владение Angular<sup>4</sup>

---

<sup>3</sup> <https://www.npmjs.com/package/angular> (дата обращения 02.04.2018)

<sup>4</sup> [https://tomsk.hh.ru/search/resume?exp\\_period=all\\_time&order\\_by=relevance&text=angular&pos=full\\_text&logic=normal&clusters=true](https://tomsk.hh.ru/search/resume?exp_period=all_time&order_by=relevance&text=angular&pos=full_text&logic=normal&clusters=true) (дата обращения 02.04.2018)

(ввиду схожести названий текущего фреймворка и его предшественника, нет уверенности в репрезентативности).

## 2.2. Обзор фреймворка Vue

Создателем Vue является Evan You, бывший сотрудник Google и Meteor Dev Group. Разработка фреймворка началась в 2013-м году, а уже в феврале 2014-го состоялся первый публичный релиз. Vue широко используется среди таких китайских компаний, как: Alibaba, Baidu, Xiaomi, Sina Weibo и др. Он входит в ядро продуктов Laravel и PageKit. Свободная система управления репозиториями GitLab недавно тоже перешла на Vue.

В конце сентября 2016-го вышел в релиз Vue 2.0, в котором основной упор сделан на производительность — теперь в фреймворке используется виртуальный DOM, поддерживается серверный рендеринг, возможность использовать JSX и т.д. Хотя сейчас он поддерживается только сообществом, Vue 2.0 держится достойно даже на уровне продуктов таких гигантов, как Google и Facebook (Angular2 и React 15), и постепенно догоняет их по популярности.

В качестве оценки популярности было так же взято недельное количество скачиваний фреймворка на сайте npmjs.com, что для Vue составляет 229 957 скачиваний в неделю<sup>5</sup>. В качестве оценки количества разработчиков были так же взяты данные с сайта поиска работы HeadHunter, в котором по России выдается 1 798 резюме с указанием владения Vue<sup>6</sup>.

## 2.3. Обзор фреймворка React

React — JavaScript-фреймворк с открытым исходным кодом для разработки пользовательских интерфейсов. Он разрабатывается и поддерживается такими компаниями как Facebook, Instagram и сообществом отдельных разработчиков и корпораций. Впервые React был выпущен в марте 2013 года и с тех пор является одним из самых востребованных фреймворков.

React может использоваться для разработки одностраничных и мобильных приложений. Его цель — предоставить высокую скорость, простоту и масштабируемость.

---

<sup>5</sup> <https://www.npmjs.com/package/vue> (дата обращения 02.04.2018)

<sup>6</sup> [https://tomsk.hh.ru/search/resume?text=vue&logic=normal&pos=full\\_text&exp\\_period=all\\_time&order\\_by=relevance&clusters=true](https://tomsk.hh.ru/search/resume?text=vue&logic=normal&pos=full_text&exp_period=all_time&order_by=relevance&clusters=true) (дата обращения 02.04.2018)

В частности, для предоставления высокой скорости React предлагает Виртуальный DOM (Document Object Model).

Недельное количество скачиваний фреймворка на сайте npmjs.com для React составляет 1 587 904 скачиваний в неделю<sup>7</sup>. На сайте поиска работы HeadHunter выдается 3 423 резюме по России, в которых указано владение React<sup>8</sup>.

## **2.4. Выбор наиболее подходящего фреймворка**

Исходя из сформулированных критериев становится очевидно, что больше всего им соответствует React, так как, являясь самым старым из них, он не потерял актуальности и до сих пор лидирует по численным показателям скачиваний и количеству разработчиков.

---

<sup>7</sup> <https://www.npmjs.com/package/react> (дата обращения 02.04.2018)

<sup>8</sup> [https://tomsk.hh.ru/search/resume?exp\\_period=all\\_time&order\\_by=relevance&text=react&pos=full\\_text&logic=normal&clusters=true](https://tomsk.hh.ru/search/resume?exp_period=all_time&order_by=relevance&text=react&pos=full_text&logic=normal&clusters=true) (дата обращения 02.04.2018)

### 3. Реализация приложение «Денежные переводы»

При реализации программного продукта, в первую очередь, необходимо определить его функционал:

Таблица 1 – Функции приложения «Денежные Переводы»

Функция	Содержание
Рассчитать стоимость перевода (как один из этапов оформления перевода)	рассчитать комиссию и/или курс валют в зависимости от суммы, валюты и страны получателя перевода
Отправить перевод	оформить и оплатить безадресный перевод (указываются минимальные данные получателя и отправителя в соответствии с требованиями страны получателя перевода, сумма и валюта перевода)
Изменить перевод	до момента выдачи перевода отправитель может изменить данные получателя
Вернуть перевод	до момента выдачи перевода отправитель может вернуть переведенные средства обратно на карту, с потерей комиссии
Получить выписку по переводам	скачать данные по оформленным переводам в одном из трех форматов (Web: pdf, html, xls, МП: pdf)
Получить чек	получить на email электронный чек по операции перевода
Настроить регистрацию карт	отметить сохранять ли данные карт, с которых оформляются переводы
Удалить карту	удалить сохраненные данные по карте
Повторить перевод	оформить новый перевод на основании одного из оформленных ранее
Поделиться промо-кодом	опубликовать промо-код через соц.сети. Данная возможность предоставляется пользователю совершившему перевод на сумму более 5000 руб (либо эквивалент в иностранной валюте)
Использовать купон	при расчете стоимости перевода указать промо-код для снижения комиссии или для применения

	акционного курса валют (если у пользователя есть купон, он используется автоматически). Купон выдается пользователю, который "привел друга" (по промо-коду выданному пользователю совершил перевод новый отправитель)
Использовать промо-код	при расчете стоимости перевода указать промо-код для снижения комиссии или для применения акционного курса валют
Фильтровать историю переводов	фильтрация списка оформленных переводов на основании ряда заданных критериев
Проверить статус перевода	узнать статус перевода по его номеру
Повторно запросить СМС с кодом подтверждения	запрос повторной отправки СМС с кодом для различных операций подтверждения действий
Просмотреть детали перевода	просмотр полной информации по переводу (отправитель, получатель, перевод)
Просмотреть список пунктов выдачи	поиск и просмотр сведений пунктов выдачи
Обратиться в техподдержку	отправка сообщения в службу техподдержки по средствам формы обратной связи на сайте, WhatsApp, Telegram, email или телефона
Ознакомиться с правовыми документами	просмотреть размещенные на сайте материалы (для просмотра материалов из МП происходит переадресация на сайт)
Войти в личный кабинет	вход в личный кабинет по номеру телефона с одноразовым паролем (СМС с кодом)
Зарегистрироваться	регистрация происходит автоматически, пользователь регистрируется по номеру телефона при первом его подтверждении (первый вход или первая отправка перевода)

Данный функционал был определен бизнес-аналитиками с учетом потребностей конечных пользователей, и частично был реализован в версии веб-приложения на GWT.

Процесс взаимодействия пользователя с приложением выглядит следующим образом:

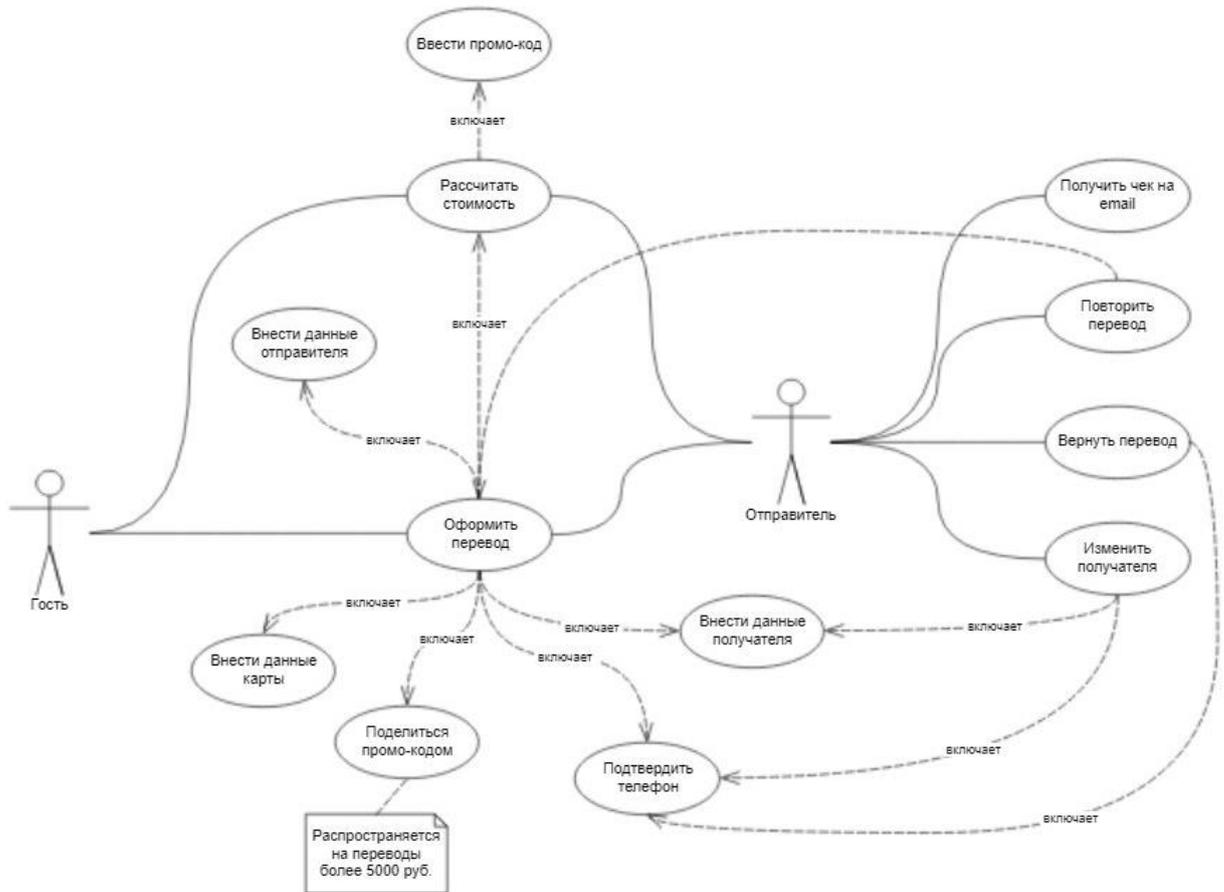


Рисунок 1 – Диаграмма вариантов использования. Оформление перевода

Разработка клиентской компоненты веб-приложения Денежные Переводы – подразумевает под собой реализацию Front-end части веб-приложения

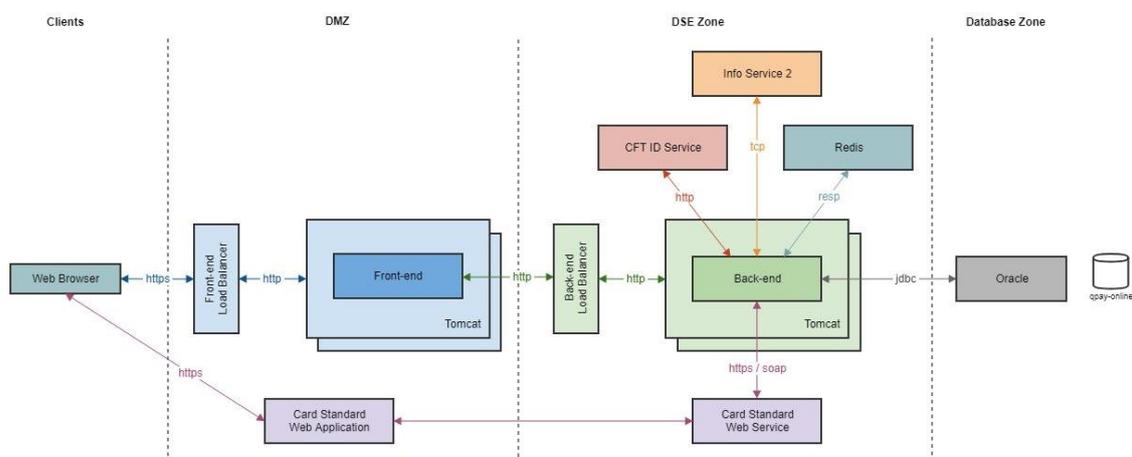


Рисунок 2 – Структура веб-приложения

Разработка клиентской компоненты веб-приложения, в данном случае, представляет собой не только создание интерактивной проксирующей прослойки между конечным пользователем и API, предоставляемым серверной частью приложения, но и реализация большого числа неочевидного функционала, а именно:

- роутинг;
- навигация;
- серверный рендеринг;
- логика взаимодействия пользователя с API.

Для реализации роутинга и навигации используется React Router v4. При старте проекта нужно определить какой тип роутера использовать. Для браузерных проектов есть BrowserRouter и HashRouter компоненты. BrowserRouter — следует использовать, когда на сервере обрабатываются динамические запросы, а HashRouter когда у вас статический веб сайт.

Обычно предпочтительнее использовать BrowserRouter, но если ваш сайт расположен на статическом сервере, то использовать HashRouter это хорошее решение проблемы.

В случае реализации клиентской компоненты веб-приложения «Денежные Переводы» необходимо использовать компонент BrowserRouter. Каждый Router создает объект history который хранит путь к текущему location и перерисовывает интерфейс сайта когда происходят какие то изменения пути.

Остальные функции, предоставляемые в React Router, полагаются на доступность объекта history через context, поэтому они должны рендериться внутри компонента Router.

Использование React Router позволяет переложить проблему роутинга и навигации на правильное использование и настройку компонента.

Серверный рендеринг используется для упрощения и увеличения эффективности индексации веб-приложения поисковыми роботами. Суть серверного рендеринга довольно проста: на сервере нам нужно определить на основе правил роутера, какой компонент будет показан на странице, выяснить, какие данные ему нужны для работы, запросить эти данные, отрендерить HTML, и выслать этот HTML вместе с данными на клиент. Переход к серверному рендерингу обеспечивается грамотной настройкой фреймворка React.

Используя Redux при инициализации приложения необходимо передать начальное состояние приложения в функцию createStore, которая из переданных данных формирует новый объект Store, с которым в дальнейшем будут работать компоненты приложения.

Одним из основных принципов Redux является то, что в каждый момент времени при одном и том же состоянии Store приложение должно также иметь одно и то же состояние.

Компоненты в React делятся на, так называемые, «Умные» компоненты и «Глупые» компоненты. Их отличие состоит в том, что «Глупые» компоненты не имеют зависимостей от остального приложения и определяют только визуальную составляющую элементов приложения, а «Умные» компоненты предоставляют данные для «Глупых».

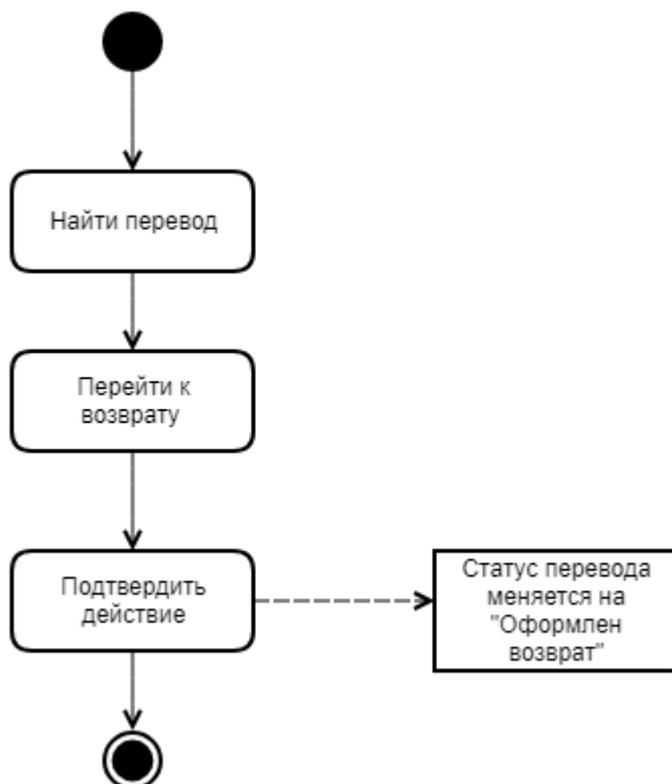


Рисунок 3 – Диаграмма действий «Возврат перевода».

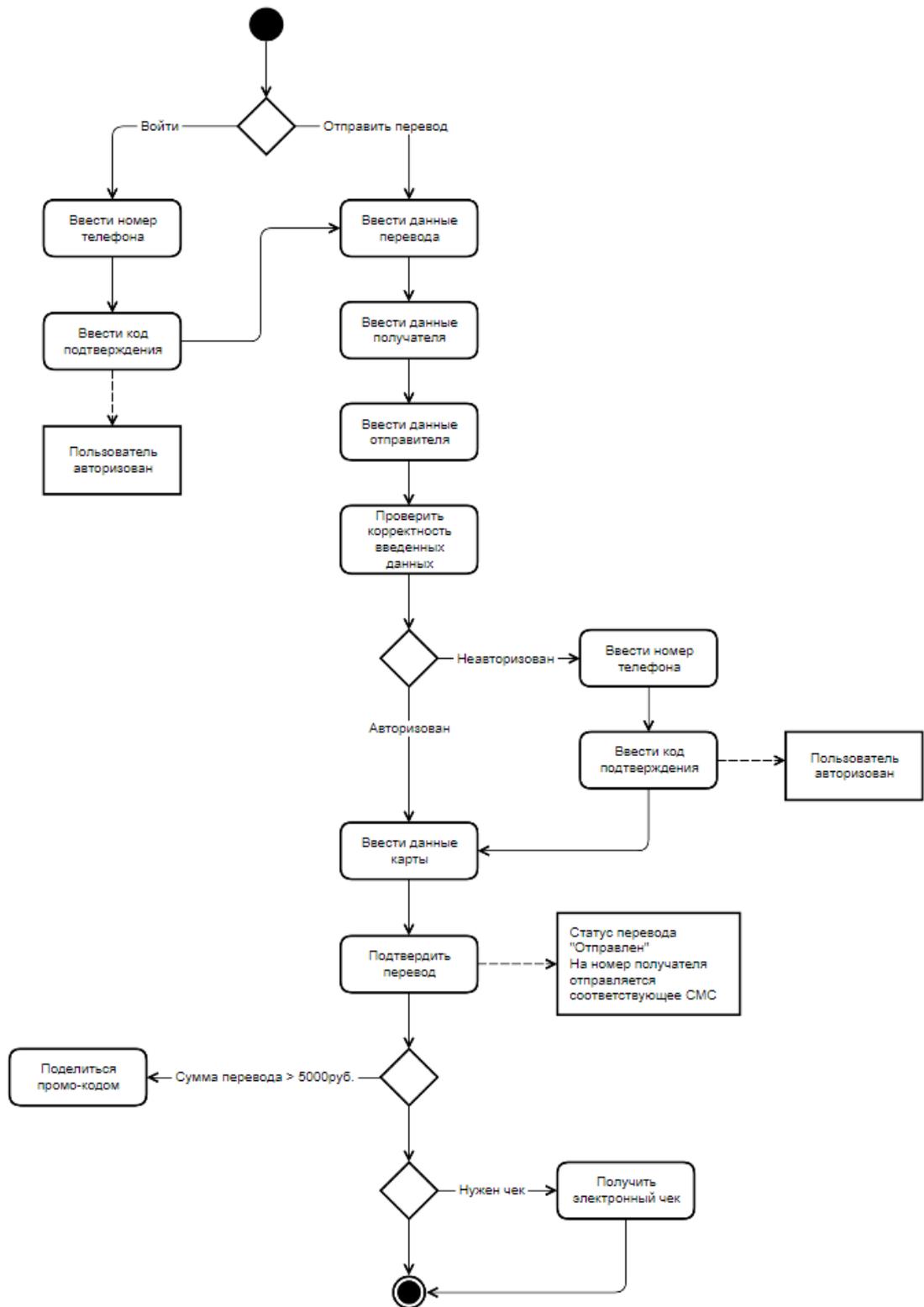


Рисунок 4 – Диаграмма действий «Отправка перевода».



Рисунок 5 –Диаграмма действий «Изменение перевода».

Получение данных для работы веб-приложения «Денежные Переводы» и передача их на сервер для дальнейшего сохранения, валидации или исполнения того функционала, который по тем или иным причинам не должен выполняться на стороне клиента осуществляется с помощью уже существующего API, детальное описание которого находится в приложении А.

## Заключение

1. Для достижения поставленной цели были изучены и проанализированы фреймворки Vue, Angular и React.
2. Из изученных фреймворков был выбран подходящий, который в данный момент стал стандартом разработки веб-приложений в нашей компании.
3. Было развернуто и настроено окружение разработки под React и Redux.
4. Был разработан программный продукт клиентская компонента веб-приложения «Денежные Переводы».
5. Разработанный продукт прошел большинство этапов тестирования внутри компании.
6. В данный момент продукт готовится к внедрению в эксплуатацию. Предполагаемая дата релиза 04.06.2018гг.

## Список использованных источников и литературы

1. HeadHunter [Электронный ресурс] // официальный сайт – URL: <https://tomsk.hh.ru> (дата обращения: 02.04 2018).
2. Node.js Package Manager [Электронный ресурс] // официальный сайт – URL: <https://www.npmjs.com> (дата обращения: 02.04 2018).
3. Бэнкс А. React и Redux: функциональная веб-разработка / А. Бэнкс, Е. Порселло. – Спб: Питер, 2018. - 336с.
4. Официальная документация Angular [Электронный ресурс] // официальный сайт – URL: <https://angular.io> (дата обращения: 05.02 2018).
5. Официальная документация React.js [Электронный ресурс] // официальный сайт – URL: <https://reactjs.org> (дата обращения: 15.02 2018).
6. Официальная документация Vue.js [Электронный ресурс] // официальный сайт – URL: <https://ru.vuejs.org> (дата обращения: 15.02 2018).
7. Центральный банк Российской Федерации [Электронный ресурс] // официальный сайт – URL: <https://www.cbr.ru> (дата обращения 03.03.2018).
8. Центробанк: объем денежных переводов из РФ и в Россию в 2017 году увеличился [Электронный ресурс] // независимый финансовый портал – URL: <http://www.banki.ru> (дата обращения 03.03.2018).

## Приложение А. Доступное API.

### TransferTariffInfo

Сущность, содержащая краткую информацию о тарифе для денежного перевода.

Поле	Тип данных	Обязательное	Комментарий
sendingCurrency	<u>Currency</u>	Да	Валюта отправки для денежного перевода
receivingCurrency	<u>Currency</u>	Да	Валюта получения для денежного перевода
minAmount	long	Да	Минимальная сумма денежных средств, которая может быть отправлена
maxAmount	long	Да	Максимальная сумма денежных средств, которая может быть отправлена
comment	String (200)	Нет	Комментарий, содержащий описание тарифа. Например, комиссия 1%, минимум 50 RUR
warnings	[ string ]	Нет	Коллекция, содержащая уведомления о лимитах на выдачу в стране назначения.

## TransferTariff

Сущность, содержащая детальную информацию о тарифе для денежного перевода.

Поле	Тип данных	Обязательное	Комментарий
sendingCurrency	<u>Currency</u>	Да	Валюта, используемая для отправки перевода
sendingAmount	long	Да	Сумма денежных средств (в валюте отправки), которая будет заплачена отправителем перевода
sendingAmount Discount	long	Да	Сумма скидки (в валюте отправки), которая будет получена отправителем
sendingCommission	long	Да	Сумма комиссии (в валюте отправки), которая будет заплачена отправителем
sendingCommission Discount	long	Да	Сумма скидки для комиссии (в валюте отправки)
receivingCurrency	<u>Currency</u>	Да	Валюта, используемая для получения перевода
receivingAmount	long	Да	Сумма денежных средств (в валюте получения), которые будут выданы получателю перевода
exchangeRate	double	Нет	Курс обмена валюты, используемый для перевода денежной суммы из валюты отправки в валюту получения. Используется, если валюта отправки не совпадает с валютой получения перевода
exchangeRate Discount	double	Нет	Размер скидки на курс обмена валюты. Используется, если валюта отправки не совпадает с валютой получения перевода

## Person

Сущность, содержащая информацию о физическом лице, участнике денежных переводов.

Поле	Тип данных	Обязательное	Комментарий
lastName	string(40)	Нет	Фамилия
firstName	string(40)	Нет	Имя
middleName	string(40)	Нет	Отчество
phone	long	Нет	Номер телефона в международном формате
birthDate	string(10)	Нет	Дата рождения в формате dd.ММ.yyyy
birthPlace	<u>Place</u>	Нет	Место рождения
registrationAddress	<u>Address</u>	Нет	Адрес регистрации
identityCard	<u>IdentityCard</u>	Нет	Удостоверение личности. Поля IdentityCard.issuer, IdentityCard.issueDate являются обязательными
identityNumber	string(20)	Нет	Индивидуальный идентификационный номер
citizenship	string(60)	Нет	Гражданство. Наименование страны, гражданином которой является физическое лицо
resident	boolean	Нет	Признак, определяющий, является ли физическое лицо резидентом страны, из которой выполняется операция отправки денежного перевода

## Transfer

Сущность, содержащая информацию о денежном переводе.

Поле	Тип данных	Обязательное	Комментарий
id	string(20)	Да	Идентификатор
number	string(20)	Нет	Номер перевода
type	string	Да	Тип перевода. Возможные значения:  in - входящий перевод; out - исходящий перевод; inOut - перевод, являющийся входящим и исходящим одновременно (перевод, отправленный самому себе)
sendingMode	string	Да	Режим отправки перевода. Возможные значения:  online - перевод отправлен с использованием online-сервиса; offline - перевод отправлен из банка или торговой сети
creationTime	long	Да	Дата создания перевода. Количество миллисекунд с 01.01.1970 UTC
sender	<u>Person</u>	Да	Отправитель перевода
sendingCountry	<u>Country</u>	Да	Страна, из которой будет (был) отправлен перевод (в которой расположен эмитент банковской карты)
sendingCurrency	<u>Currency</u>	Да	Валюта, используемая для отправки перевода

Поле	Тип данных	Обязательное	Комментарий
sendingAmount	long	Да	Сумма денежных средств (в валюте отправки), которые будут (были) заплачены отправителем перевода
sendingAmount Discount	long	Да	Сумма скидки (в валюте отправки), которая будет получена отправителем
sendingCommission	long	Да	Сумма комиссии (в валюте отправки), которая будет (была) заплачена отправителем
sendingCommission Discount	long	Да	Сумма скидки для комиссии (в валюте отправки)
receivingCountry	<u>Country</u>	Да	Страна, в которой будет (был) получен перевод
receiver	<u>Person</u>	Да	Получатель перевода
receivingCurrency	<u>Currency</u>	Да	Валюта, используемая для получения перевода
receivingAmount	long	Да	Сумма денежных средств, которая будет (была) выдана получателю перевода
exchangeRate	double	Нет	Курс обмена валюты, используемый для перевода денежной суммы из валюты отправки в валюту получения.  Используется, если валюта отправки не совпадает с валютой получения перевода
exchangeRate Discount	double	Нет	Размер скидки на курс обмена валюты.  Используется, если валюта отправки не совпадает с валютой получения перевода
card	<u>Card</u>	Нет	Банковская карта, которая использовалась для оплаты перевода

Поле	Тип данных	Обязательное	Комментарий
status	string	Да	<p>Статус перевода.</p> <p>Возможные значения:</p> <p>registered - зарегистрирован;  payed - оплачен отправителем;  received - получен;  refunded - оформлен возврат;  cancelled - отменен  ready - готов к выдаче  receivedOnCard - зачислен на карту</p>
actions	[ string ]	Нет	<p>Список идентификаторов действий, которые могут быть выполнены с переводом.</p> <p>Возможные значения для идентификатора действия:</p> <p>repeat - повтор перевода с тем же перечнем реквизитов;  refund - возврат платежа на банковскую карту;  change - изменение данных получателя перевода;  creditingToCard - зачисление перевода на банковскую карту;  fixReceivingCurrency - фиксация курса конвертации и валюты получения перевода</p>

## TransferRefundRequest

Сущность, содержащая информацию о запросе на возврат денежного перевода.

Поле	Тип данных	Обязательное	Комментарий
currency	<u>Currency</u>	Да	Валюта, используемая для возврата перевода
amount	long	Да	Сумма денежных средств, которая будет перечислена на банковскую карту отправителя перевода
exchangeRate	double	Нет	Курс обмена валюты, используемый для расчета суммы денежных средств, подлежащих возврату.  Используется, если валюта отправки не совпадает с валютой получения перевода
expiryTime	long	Нет	Дата истечения срока действия запроса на возврат. Количество миллисекунд с 01.01.1970 UTC

## TransferReceiver

Сущность, содержащая информацию о получателе денежного перевода.

Поле	Тип данных	Обязательное	Комментарий
lastName	string(40)	Да	Фамилия
firstName	string(40)	Да	Имя
middleName	string(40)	Нет	Отчество
phone	long	Да	Номер телефона в международном формате
lastTransferId	string(20)	Да	Идентификатор последнего денежного перевода, который был отправлен данному получателю

## TransferReceivingOption

Сущность, содержащая информацию о варианте получения перевода.

Поле	Тип данных	Обязательное	Комментарий
currency	<u>Currency</u>	Да	Валюта получения перевода
amount	long	Да	Сумма денежных средств, которая будет выдана получателю перевода
conversionRate	<u>Currency</u> <u>ConversionRate</u>	Нет	Курс конвертации валюты, указанной отправителем перевода, в валюту, заданную в данном варианте получения
receivingMethods	[ string ]	Да	Список способов получения перевода. Возможные значения для способа получения:  card - зачисление на банковскую карту; cash - получение наличными
fixationExpiryTime	long	Нет	Дата истечения срока действия фиксации. Количество миллисекунд с 01.01.1970 UTC.

## TransferReceivingInfo

Сущность, содержащая информацию о способах получения перевода.

Поле	Тип данных	Обязательное	Комментарий
number	string(20)	Да	Номер перевода
countryId	string(3)	Да	Идентификатор страны получения перевода
cityName	string(200)	Да	Полное наименование населенного пункта, в котором должен быть получен перевод
currency	<u>Currency</u>	Да	Валюта получения, указанная отправителем перевода
amount	long	Да	Сумма денежных средств в валюте получения
options	[ <u>Transfer Receiving Option</u> ]	Да	Список вариантов получения перевода в разных валютах
fixationExpiryTime	long	Нет	Дата истечения срока действия фиксации. Количество миллисекунд с 01.01.1970 UTC.  В случае, если fixationExpiryTime != null массив options содержит только один элемент, соответствующий ранее зафиксированному варианту получения

## TransferReceivingFixation

Сущность, содержащая информацию о фиксации варианта получения перевода.

Поле	Тип данных	Обязательное	Комментарий
number	string(20)	Да	Номер перевода
currencyId	string(3)	Да	Идентификатор валюты получения перевода
amount	long	Да	Сумма денежных средств, которая будет выдана получателю перевода
Receiving Method	string	Да	Способ получения перевода. Возможные значения:  card - зачисление на банковскую карту; cash - получение наличными
expiryTime	long	Да	Дата истечения срока действия фиксации. Количество миллисекунд с 01.01.1970 UTC

## TransferMetaData

Сущность, содержащая метаинформацию о полях, которые нужно заполнить для перевода.

Поле	Тип данных	Обязательное	Комментарий
senderFields	[ string ]	Да	<p>Список идентификаторов обязательных полей для отправителя перевода.</p> <p>Идентификаторы полей:</p> <p>fullName phone birthDate birthPlace registrationAddress identityCard identityNumber citizenship resident</p>
receiverFields	[ string ]	Да	<p>Список идентификаторов обязательных полей для получателя перевода.</p> <p>Идентификаторы полей:</p> <p>fullName phone birthDate birthPlace registrationAddress identityCard identityNumber citizenship resident</p>
senderName Alphabets	[ string ]	Да	<p>Список идентификаторов алфавитов, которые могут быть использованы для написания имени отправителя перевода.</p> <p>Идентификаторы алфавитов:</p> <p>cyrillic - кириллический алфавит; latin - латинский алфавит</p>

Поле	Тип данных	Обязательное	Комментарий
receiverName Alphabets	[ string ]	Да	<p>Список идентификаторов алфавитов, которые могут быть использованы для написания имени получателя перевода.</p> <p>Идентификаторы алфавитов:</p> <p>cyrillic - кириллический алфавит; latin - латинский алфавит</p>

# Отчет о проверке на заимствования №1

Автор: 40inart@gmail.com / ID: 5850614

Проверяющий: (40inart@gmail.com / ID: 5850614)

Отчет предоставлен сервисом «Антиплагиат»- <http://www.antiplagiat.ru>

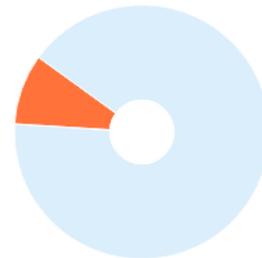
## ИНФОРМАЦИЯ О ДОКУМЕНТЕ

№ документа: 1  
Начало загрузки: 12.06.2018 08:37:04  
Длительность загрузки: 00:00:01  
Имя исходного файла: DiplomSorokin  
Размер текста: 880 кБ  
Символов в тексте: 29227  
Слов в тексте: 3540  
Число предложений: 155

## ИНФОРМАЦИЯ ОБ ОТЧЕТЕ

Последний готовый отчет (ред.)  
Начало проверки: 12.06.2018 08:37:06  
Длительность проверки: 00:00:02  
Комментарии: не указано  
Модули поиска:

ЗАИМСТВОВАНИЯ	ЦИТИРОВАНИЯ	ОРИГИНАЛЬНОСТЬ
9,02% 	0% 	90,98% 



Заимствования — доля всех найденных текстовых пересечений, за исключением тех, которые система отнесла к цитированиям, по отношению к общему объему документа.  
Цитирования — доля текстовых пересечений, которые не являются авторскими, но система посчитала их использование корректным, по отношению к общему объему документа. Сюда относятся оформленные по ГОСТу цитаты; общепотребительные выражения; фрагменты текста, найденные в источниках из коллекций нормативно-правовой документации.

Текстовое пересечение — фрагмент текста проверяемого документа, совпадающий или почти совпадающий с фрагментом текста источника.

Источник — документ, проиндексированный в системе и содержащийся в модуле поиска, по которому проводится проверка.

Оригинальность — доля фрагментов текста проверяемого документа, не обнаруженных ни в одном источнике, по которым шла проверка, по отношению к общему объему документа.

Заимствования, цитирования и оригинальность являются отдельными показателями и в сумме дают 100%, что соответствует всему тексту проверяемого документа.

Обращаем Ваше внимание, что система находит текстовые пересечения проверяемого документа с проиндексированными в системе текстовыми источниками. При этом система является вспомогательным инструментом, определение корректности и правомерности заимствований или цитирований, а также авторства текстовых фрагментов проверяемого документа остается в компетенции проверяющего.

№	Доля в отчете	Доля в тексте	Источник	Ссылка	Актуален на	Модуль поиска	Блоков в отчете	Блоков в тексте
[01]	2,54%	2,54%	Июнь   2017   SavePearlHar...	<a href="http://savepearlharbor.com">http://savepearlharbor.com</a>	21 Окт 2017	Модуль поиска Интернет	5	5
[02]	2,17%	2,17%	не указано	<a href="http://uralpress.ru">http://uralpress.ru</a>	раньше 2011	Модуль поиска Интернет	1	1
[03]	1,49%	1,49%	Денежный перевод	<a href="http://ru.wikipedia.org">http://ru.wikipedia.org</a>	17 Дек 2016	Модуль поиска Интернет	2	2

Еще источников: 3  
Еще заимствований: 2,81%