МИНОБРНАУКИ РОССИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ**

**«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**Сервис доставки**

Редакция от 16.01.2017

Выполнили: студенты 3 курса, группы 3.1

Симонов Е. И., Бердников О. В.

Руководитель:

Хлебостроев В.Г.

Воронеж 2017

# Содержание

[**Содержание**](#_z4r0e9h4elm4)[**2**](#_z4r0e9h4elm4)

[**Введение**](#_wkjsz7qyepns)[**4**](#_wkjsz7qyepns)

[**Анализ предметной области**](#_azf0sc91wkn1)[**5**](#_azf0sc91wkn1)

[1.1. Глоссарий](#_90tae3v52ou3) [5](#_90tae3v52ou3)

[1.2. Анализ существующих решений](#_reu0rblbdzmp) [5](#_reu0rblbdzmp)

[1.3. Целевая аудитория](#_hihnoft1ucu4) [6](#_hihnoft1ucu4)

[**2. Постановка задачи**](#_n4plbkwjc7kc)[**7**](#_n4plbkwjc7kc)

[**3. Общие сведения о проекте**](#_i7oa8y2yrgvt)[**8**](#_i7oa8y2yrgvt)

[3.1. Цели и задачи проекта](#_8yikfch3g9ce) [8](#_8yikfch3g9ce)

[**4. Описание требований к разрабатываемой системе**](#_lj9wiufo8ehv)[**9**](#_lj9wiufo8ehv)

[4.1. Функциональные требования](#_1x6dgssndxa8) [9](#_1x6dgssndxa8)

[4.2. Критерии качества](#_etxkso64bg1e) [11](#_etxkso64bg1e)

[4.3. Входные/выходные данные](#_w9ujdinzzaho) [11](#_w9ujdinzzaho)

[4.4. Требования к интерфейсу](#_idhgwkv8rq6z) [12](#_idhgwkv8rq6z)

[4.5. Средства реализации](#_7iil85q6hjei) [12](#_7iil85q6hjei)

[**5. Планирование работ**](#_9o481ifm9rn5)[**14**](#_9o481ifm9rn5)

[**6. Проектирование**](#_w2qimodgildp)[**16**](#_w2qimodgildp)

[6.1. Архитектура ПО](#_io5vbx51e4jn) [16](#_io5vbx51e4jn)

[6.2. Схема БД](#_939ka3l0dxfl) [17](#_939ka3l0dxfl)

[6.3. Диаграмма развертывания](#_kl18h4llyp9n) [18](#_kl18h4llyp9n)

[6.4. Диаграмма классов](#_knsxwzjh7fqd) [19](#_knsxwzjh7fqd)

[6.5. Диаграмма взаимодействия основного прецедента](#_e70p6llydkjg) [20](#_e70p6llydkjg)

[**7. Интерфейс пользователя**](#_q1a1w6l5ik5w)[**21**](#_q1a1w6l5ik5w)

[**8. Тестирование**](#_vune9cc7a5ko)[**28**](#_vune9cc7a5ko)

[8.1. Модульное тестирование](#_pkhm0trl1cnr) [28](#_pkhm0trl1cnr)

[8.2. Функциональное тестирование](#_2n0y4s7k073a) [28](#_2n0y4s7k073a)

[**9. Календарный план**](#_62pdpr6mccz4)[**30**](#_62pdpr6mccz4)

[**10. Заключение**](#_ahlnrd6sm3az)[**32**](#_ahlnrd6sm3az)

[Приложение 1. Руководство пользователя](#_8nspfxr1osdt) [33](#_8nspfxr1osdt)

# Введение

Важнейшим из ресурсов в наши дни стало время. Люди проводят большую часть дня на работе и в пробках. Отличительной чертой IT-технологий является направленность на упрощение жизни людей. Разного рода упрощения происходят и в сфере торговли. Так, у большинства крупных сетей появились интернет каталоги с подробным описанием товаров. А магазины бытовой техники и электроники предлагают и доставку товаров покупателю. Тем не менее, эти услуги работают не всегда. Во-первых, не все магазины могут позволить себе иметь интернет-магазин. Во-вторых, существует ряд товаров, которые не доставляют свои товары в принципе или доставляют, но редко. Разрабатываемый сервис призван решить именно подобные проблемы. Использование его дает ряд преимуществ:

* Вы можете указать товар и максимальную цену, которую вы готовы за него заплатить, не задумываясь о том, где данный товар будет приобретен;
* Вы можете в любой момент связаться с курьером, которому поручен Ваш заказ, и уточнить детали доставки;
* Вы можете заказывать вещи, которые не доставляются стандартными службами доставки.

Уже сейчас существуют сервисы со схожей функциональностью, но их работа основана на курьерах-пользователях, то есть любой пользователь может зарегистрироваться в приложении как курьер и доставлять заказы. Но даже такие сервисы представлены исключительно в крупных городах (Москва, Санкт-Петербург, Екатеринбург).

Данная курсовая работа посвящена технической реализации сервиса, решающего поставленные проблемы, и обладающего интуитивно-понятным интерфейсом, акцент в котором делается исключительно на заказ и доставку товара.

# Анализ предметной области

## 1.1. Глоссарий

**Доставка** — это услуга, при которой заказчику доставляют заранее выбранный им товар по указанному им адресу за определенный период времени по установленной между заказчиком и поставщиком стоимости.

**Заказ** — поручение от пользователя на доставку чего-либо.

**Курьер** — человек или компания, нанимаемые для доставки сообщений, писем, иной корреспонденции, а также небольших грузов.

## 1.2. Анализ существующих решений

[**http://courier.youdo.com**](http://courier.youdo.com) - Похожая функциональность, но сервис ориентирован по большей части на перевозку товаров из одной точки в другую. Также в этом примере нет возможности сделать заказ без указания места покупки, т.е. здесь клиент должен сам выбрать место покупки товара.

[**http://moskva.gdeuslugi.ru/uslugi\_kurera/**](http://moskva.gdeuslugi.ru/uslugi_kurera/) - Сайт ориентирован на предоставление разного рода услуг. В частности есть возможность нанять определённого курьера. Курьером может быть любой человек. В отличаем от этого мы планируем пользоваться исключительно услугами заранее нанятых курьеров.

На рынке подобных услуг представлены по большей части крупные города, такие как Москва, Санкт-Петербург, Екатеринбург, Казань. В то время, как в Воронеже нет ничего подобного.

## 1.3. Целевая аудитория

Целевой аудиторией являются люди старше 18 лет (для совершения платежей) и живущие в пределах зоны доставки.

# 2. Постановка задачи

Целью данной курсовой работы является реализация веб-приложения, которое:

* позволяет пользователю создавать заказ на доставку чего-либо;
* позволяет пользователю осуществлять отслеживание сделанного заказа;
* позволяет администратору отслеживать появление новых заказов.

Приложение должно:

* поддерживать авторизацию;
* обеспечивать защиту от несанкционированного доступа и SQL-инъекций;
* xранить данные в СУБД.

# 3. Общие сведения о проекте

## 3.1. Цели и задачи проекта

Проект будет использоваться как платформа для покупки и доставки товаров.Проект позволяет людям заказывать доставку различных товаров. Основная идея проекта - товар может быть как уже купленным (например, через Интернет), так и еще не купленным. Более того, пользователь может даже не искать нужный ему товар, а лишь заказать его - администрация системы сама выберет лучшее место для покупки. Также пользователям предоставляется возможность отслеживания своего заказа.

# 4. Описание требований к разрабатываемой системе

## 4.1. Функциональные требования

1. Авторизация пользователя

Авторизация пользователя происходит с помощью протокола авторизации OAuth 2.0 через социальную сеть ВКонтакте (VK open API). Для пользователя это облегчает процедуру авторизации (гораздо проще авторизоваться нажатием одной кнопки, чем вводить логин и пароль) и упраздняет процедуру регистрации (для создания профиля пользователю не нужно придумывать новый логин и пароль, подтверждать свою почту и т.д.). Также благодаря этому отпадает необходимость хранить пароли и данные пользователей непосредственно в базе данных.

2. Создание заказа

Авторизовавшемуся пользователю предоставляется возможность сделать заказ. Для этого необходимо корректно заполнить форму заказа. Техническая реализация данной функции состоит в следующем:

* Пользователь заполняет необходимые поля
* Данные обрабатываются на сервере приложений
* Если все данные корректны, приложение отправляет запрос на создание заказа в базу данных.

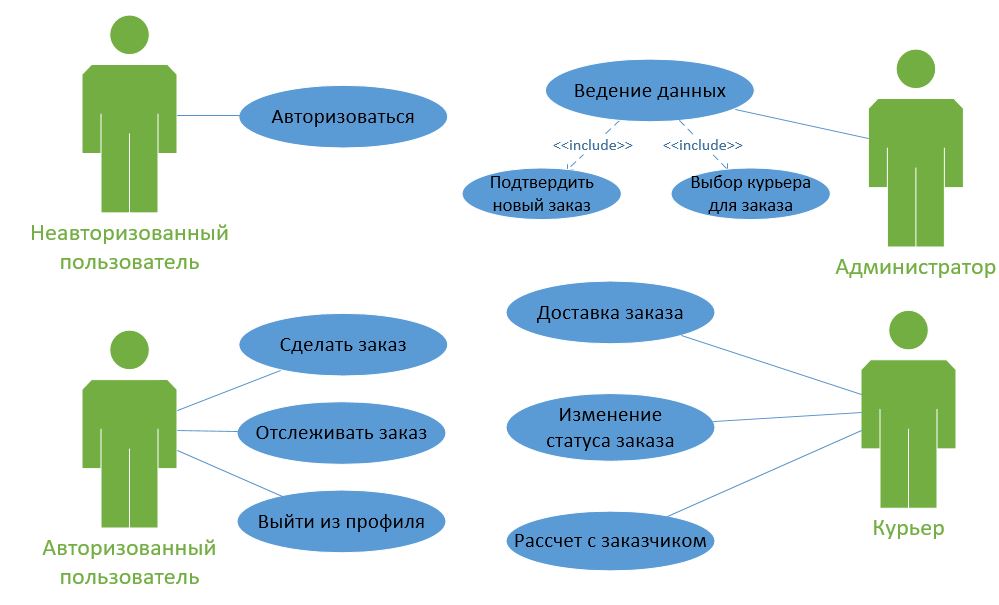
3. Отслеживание заказа

Пользователи могут отслеживать свои заказы. Для этого необходимо лишь ввести номер своего заказа.

Реализация данной функции также происходит в несколько этапов:

* Пользователь вводит в форму номер своего заказа
* Номер заказа отправляется в приложение
* Приложение отправляет запрос в базу данных
* База данных возвращает информацию о заказе

Подразумевается, что пользователю необходима информация о курьере, который выполняет его заказ, дате доставки и статусе заказа.



#### Рис. 1. Диаграмма вариантов использования

## 4.2. Критерии качества

1. Эргономичность. Так как для создания любого заказа необходимо определенное количество информации, пользователю придется вводить ее. Количество заполненных пользователем полей и усилия, затраченные на это, напрямую влияют на впечатления, которые система оставляет у пользователя. Поэтому данный критерий является наиболее приоритетным в системе.
2. Надежность. Критерий качества, который также имеет высокий приоритет. Система должна обеспечивать безотказную работу, потому что пользователь, у которого система хоть однажды даст сбой, в следующий раз может отказаться от использования системы.
3. Функциональность. Система должна выполнять набор функций, описанный выше, и соответствовать заявленным требованиям.
4. Эффективность. Проект предоставлять максимальное качество услуг и при этом задействовать минимальное количество ресурсов пользователя.
5. Мобильность. Система должна обладать возможностью переноса из одной среды в другую, так как пользователь может осуществлять создание заказа или проверку статуса созданного заказа с использованием разных сред.
6. Модифицируемость. При разработке системы алгоритмы и используемые программные средства подбирались с целью уменьшения сложности и усилий при модификации.

## 4.3. Входные/выходные данные

В зависимости от того, что в конкретное время нужно пользователю, входные/выходные данные будут разными

1. Авторизация. В этом случае входными данными будут данные профиля Вконтакте. Если пользователь на данный момент в данном браузере является неавторизованным, то ему будет предложено ввести свой логин и пароль. Выходными данными будут фото, фамилия и имя пользователя, которые указаны в его профиле.
2. Создание заказа. В этом случае входными данными будут данные о заказе, которые предоставит пользователь. Пользователь заполняет обязательные поля и по своему усмотрению-необязательные. Обязательными для заполнения являются поля: наименование товара, адрес, по которому нужно осуществить доставку, дата доставки, номер телефона и цена, которую пользователь предлагает за осуществление доставки. При создании заказа выходные данные для пользователя отсутствуют.
3. Проверка статуса заказа. Здесь входными данными будет лишь введенный пользователем номер заказа. Выходными данными является информация о заказе: дата осуществления доставки, имя и телефон курьера, ответственного за доставку, статус исполнения доставки.

## 4.4. Требования к интерфейсу

Пользовательский интерфейс должен обеспечивать максимальную надежность работы с приложением, чтобы пользователю при создании заказа или проверке статуса заказа не пришлось несколько раз вводить данные. Также пользовательский интерфейс должен быть предельно прост и ненавязчив.

## 4.5. Средства реализации

1. Операционная система: Windows/Linux
2. Используемые языки:

* HTML+CSS (разметка страниц сервиса);
* PHP (элементы интерфейса и взаимосвязь API с БД);
* Python + Flask (написание API);
* SQL (развертывание БД).

1. Система контроля версий: GitHub.
2. СУБД: MySQL.
3. Веб-сервер: Apache.

# 

# 5. Планирование работ

Основные виды работ

Разработка приложения должна быть проведена в четыре этапа:

1. Разработка технического задания.
2. Разработка приложения.
3. Тестирование приложения.
4. Создание пользовательской документации.

Рассмотри подробнее каждый из них.

На этапе *разработки технического задания* должны быть выполнены следующие работы:

* постановка задачи;
* анализ предметной области;
* анализ аналогов;
* создание глоссария проекта;
* определение функциональных требований;
* создание диаграммы вариантов использования;
* создание диаграммы взаимодействия для основного прецедента;
* создание физической схемы базы данных;
* создание эскиза пользовательского интерфейса
* выбор инструментальных средств;
* определение сроков работ;
* распределение обязанностей между членами команды.

На этапе *разработки приложения* должна быть выполнены следующие виды работ:

* создание репозитория в системе контроля версия GitHub;
* установка и настройка необходимых инструментальных средств;
* проектирование и создание БД;
* проектирование архитектуры приложения;
* разработка серверной части приложения, взаимодействующей с БД (язык программирования PHP);
* создание клиентской части приложения, выполняемой в браузере клиента (язык программирования PHP);
* создание пользовательского интерфейса;
* создание процедуры по авторизации пользователя с помощью профиля Вконтакте (язык программирования Python).

На этапе *тестирования приложения* должна быть выполнены следующие виды работ:

* тестирование отдельных модулей приложения;
* функциональное тестирование приложения (создание нескольких сценариев, соответствующих вариантам использования и проверка их работоспособности);
* исправление выявленных при тестировании ошибок;
* проверка соответствия разработанной системы заявленным требованиям.

На этапе *разработки документации* должно быть выполнено создание руководства пользователя с описанием функциональности и интерфейса созданного приложения.

Календарный план работ каждого из четырех этапов будет представлен позднее.

# 6. Проектирование

## 6.1. Архитектура ПО

В основе приложения лежит трехзвенная архитектура.

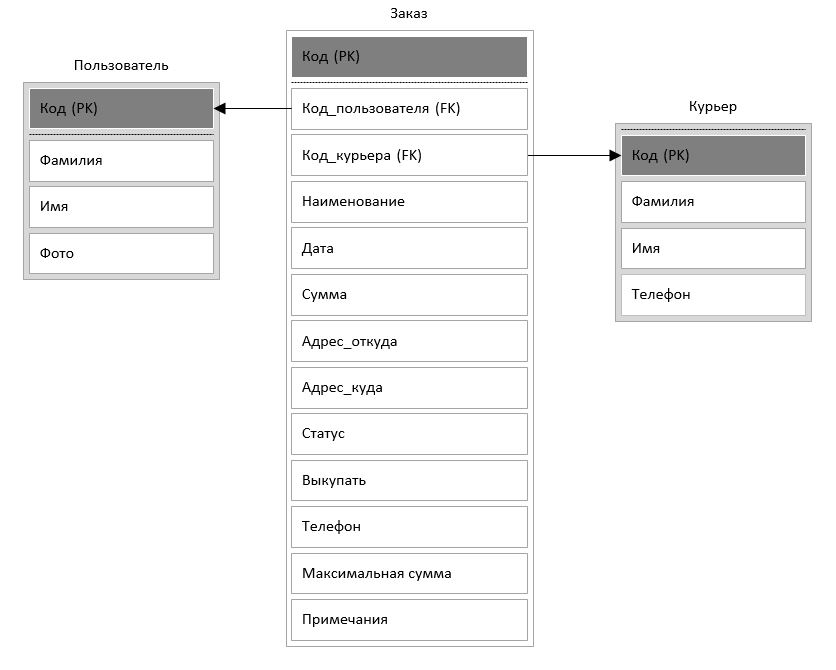
1. Сервер с базой данных

2. Сервер с приложениями

3. Пользовательский интерфейс

Пользователь непосредственно взаимодействует с интерфейсом, который обращается к приложениям. Приложения в свою очередь обращаются запросами к базе данных. Таким образом конечный пользователь не взаимодействует с базой данных напрямую, что обеспечивает достаточно высокий уровень безопасности и надёжности системы.

## 6.2. Схема БД

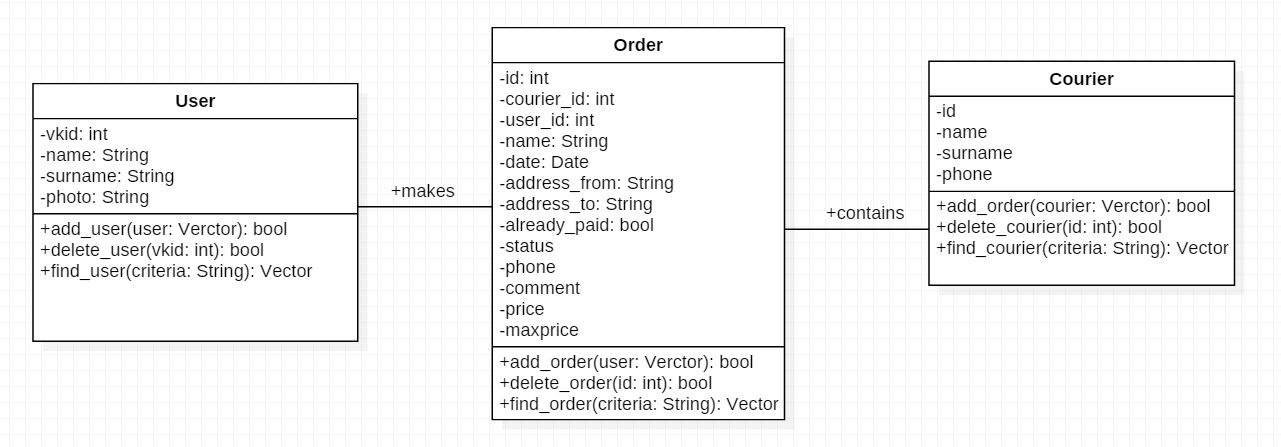


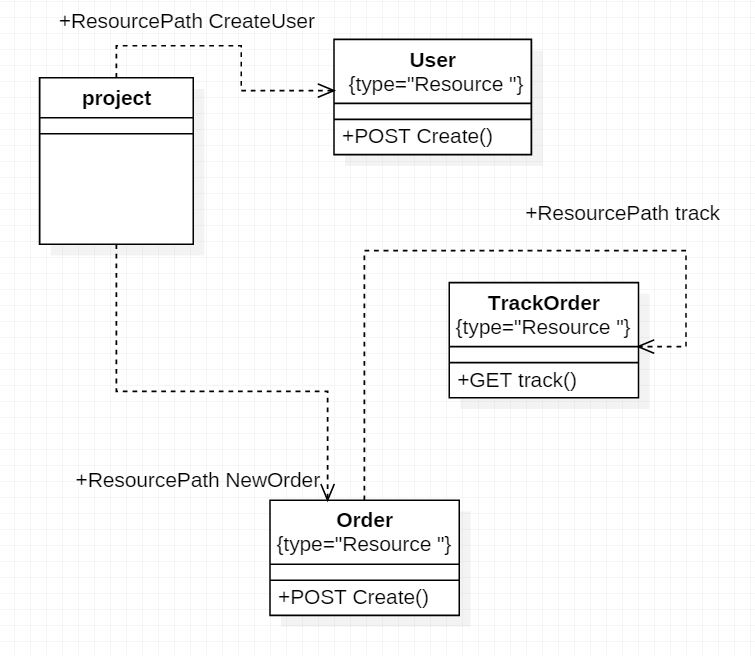
## 6.3. Диаграмма развертывания

## 

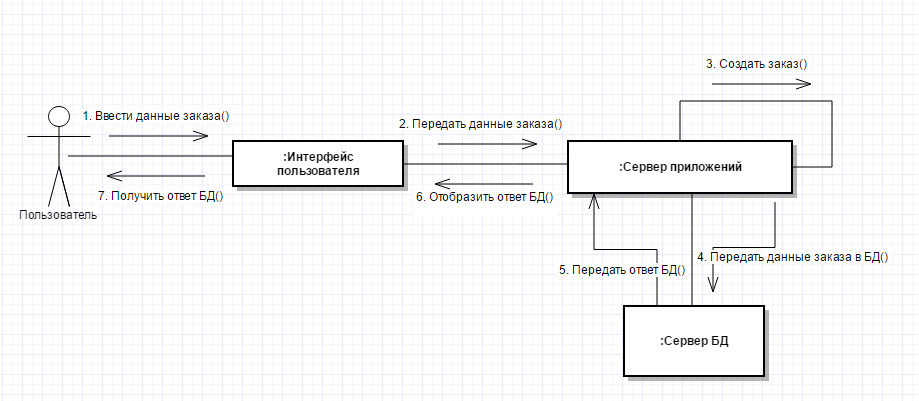
### 

## 6.4. Диаграмма классов





## 6.5. Диаграмма взаимодействия основного прецедента



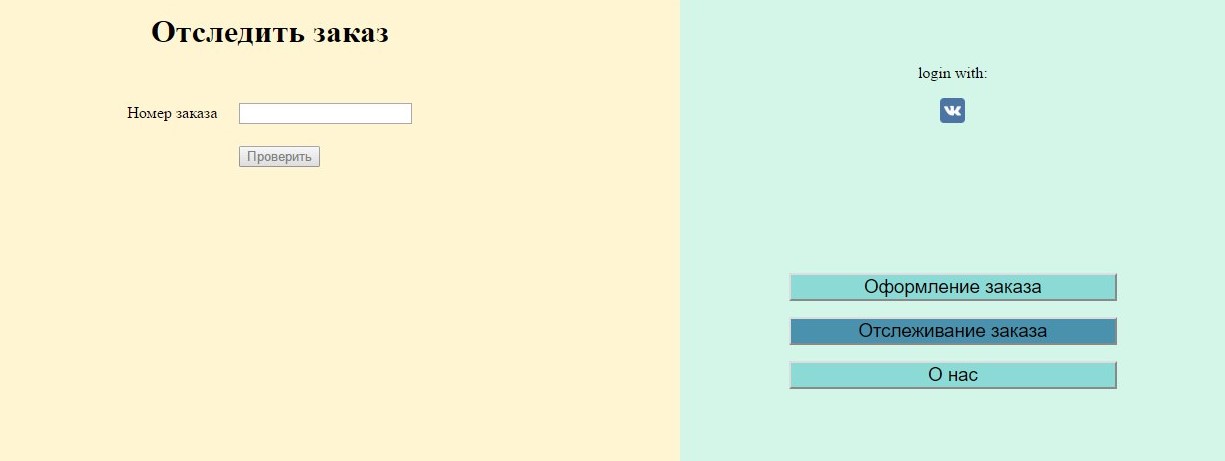
# 7. Интерфейс пользователя

### 

#### Рис.2. Стартовая страница приложения неавторизованного пользователя.

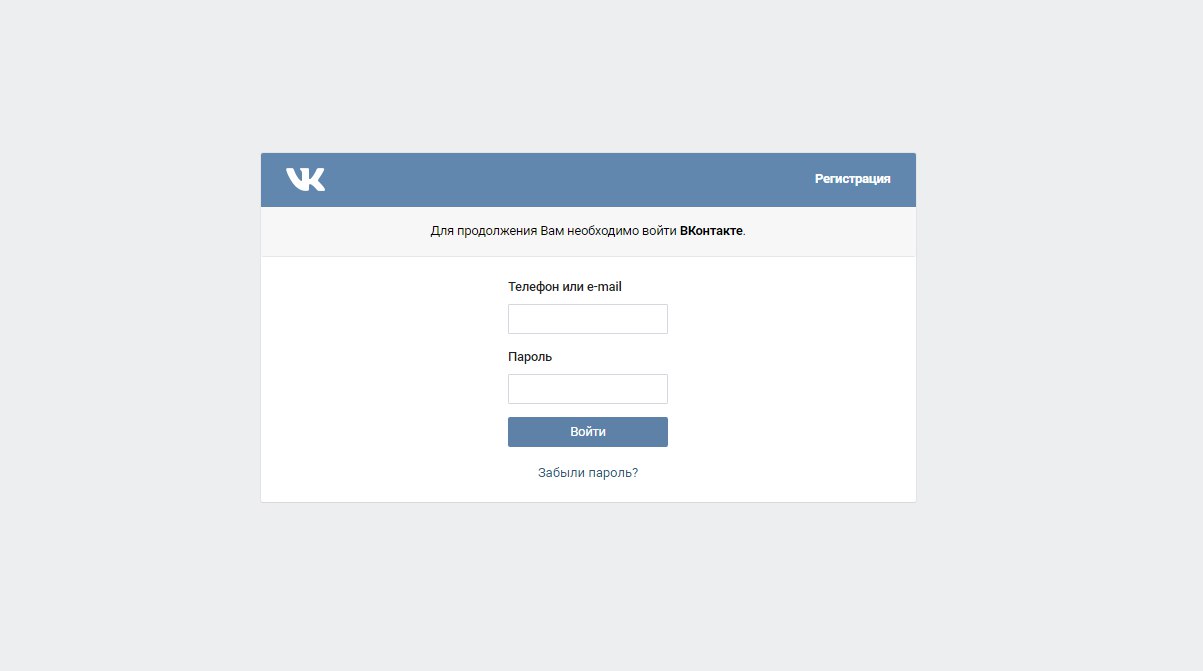
На данном рисунке представлена стартовая страница для пользователя, который еще не прошел процедуру авторизации. Он может сделать это лишь нажав на логотип социальной сети “Вконтакте” в правой части экрана. Если в данном браузере в данный момент пользователь авторизован, то авторизация произойдет автоматически (благодаря VK API), и пользователю не придется заполнять какие-либо поля с логином, паролем и т.п.

Пользователь еще не авторизован и поэтому не может добавлять заказ.



#### Рис.3. Страница проверки статуса заказа для неавторизованного пользователя

Проверка статуса заказа также недоступна неавторизованному пользователю.



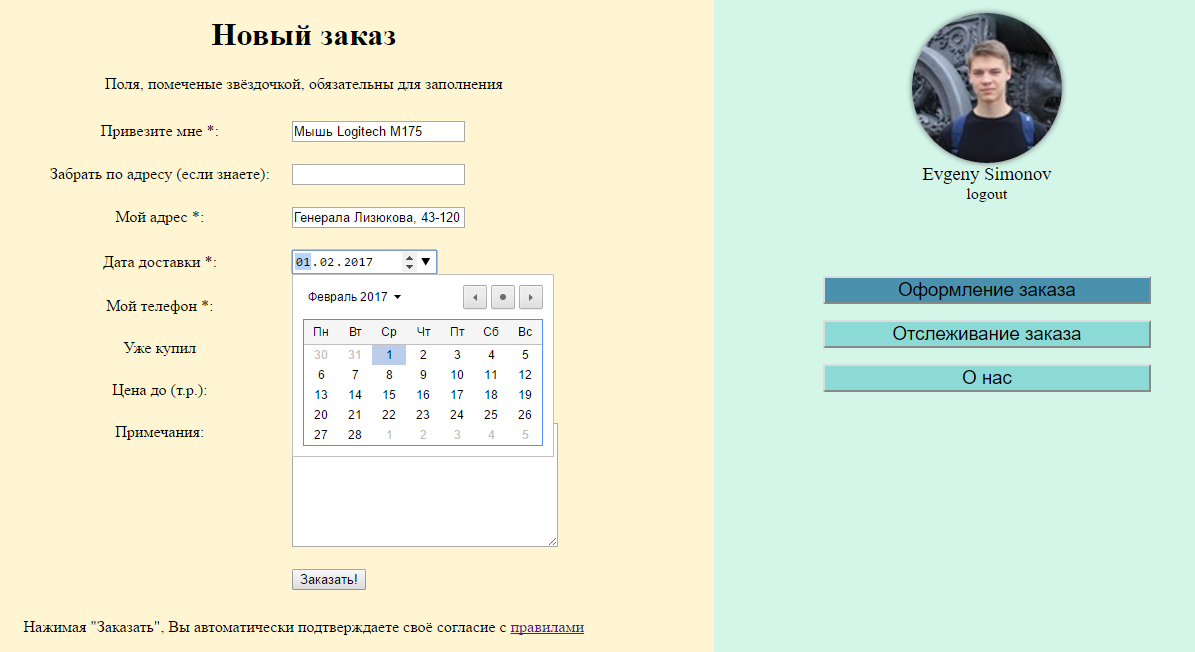
#### Рис.4. Окно для авторизации пользователя, который не авторизован в данный момент в данном браузере в социальной сети Вконтакте

На данном изображении представлена форма для авторизации пользователя, который не авторизован Вконтакте. После ввода своего логина и пароля пользователь станет авторизованным, и у него появится возможность добавления нового заказа и проверки сделанного заказа.

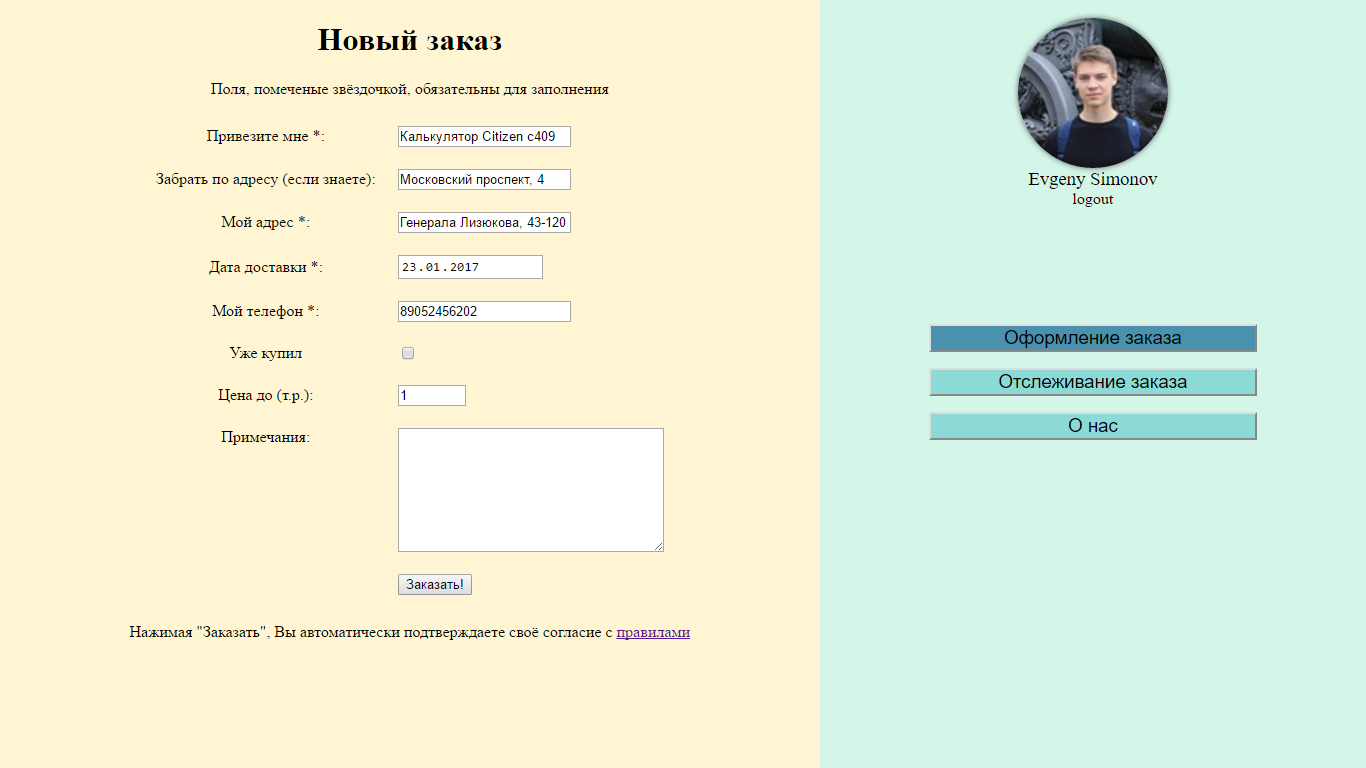


#### Рис. 5. Стартовая страница приложения после процедуры авторизации.

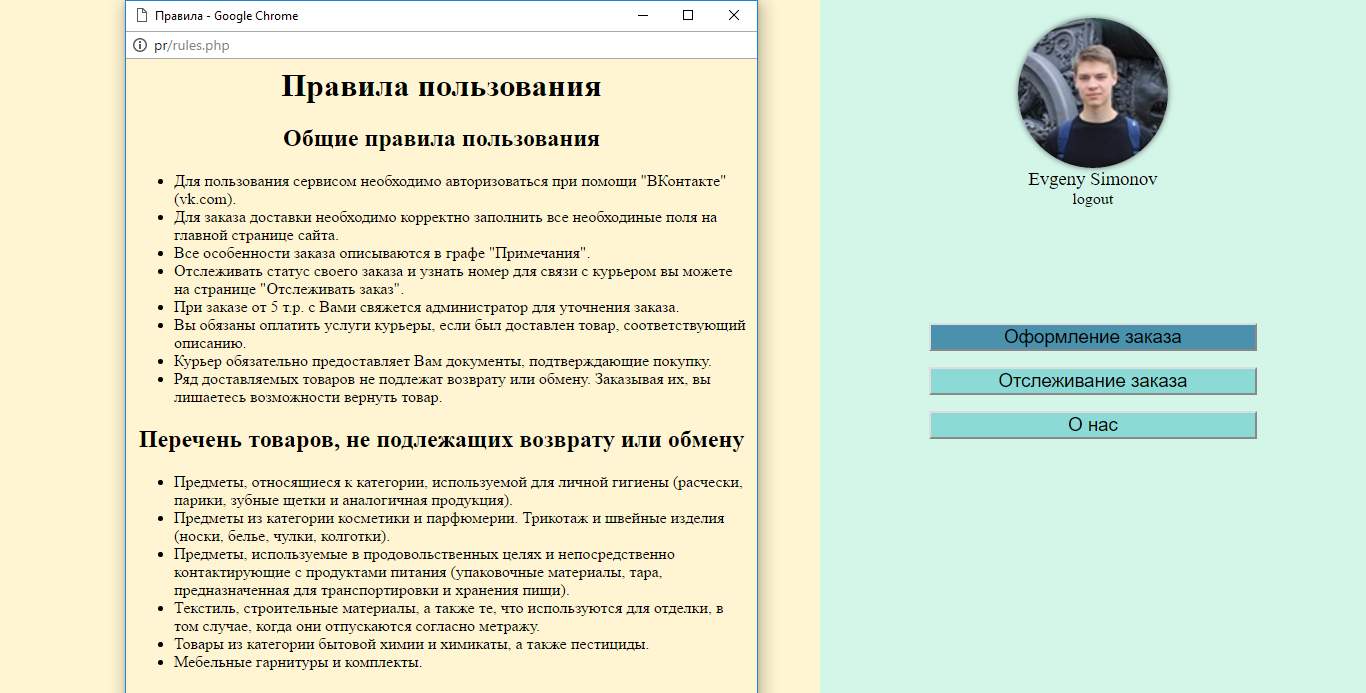
Пользователь авторизован и теперь может добавлять заказ или проверять статус своего заказа.



#### Рис.6. Процесс добавления нового заказа (пользователь на знает адрес покупки товара)

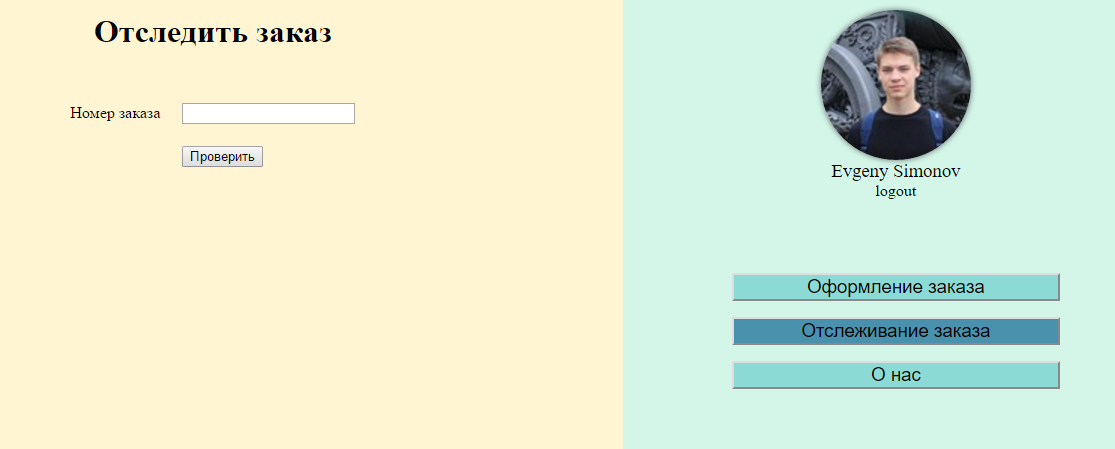


#### Рис. 7. Процесс добавления нового заказа (пользователь знает адрес, по которому будет куплен товар)



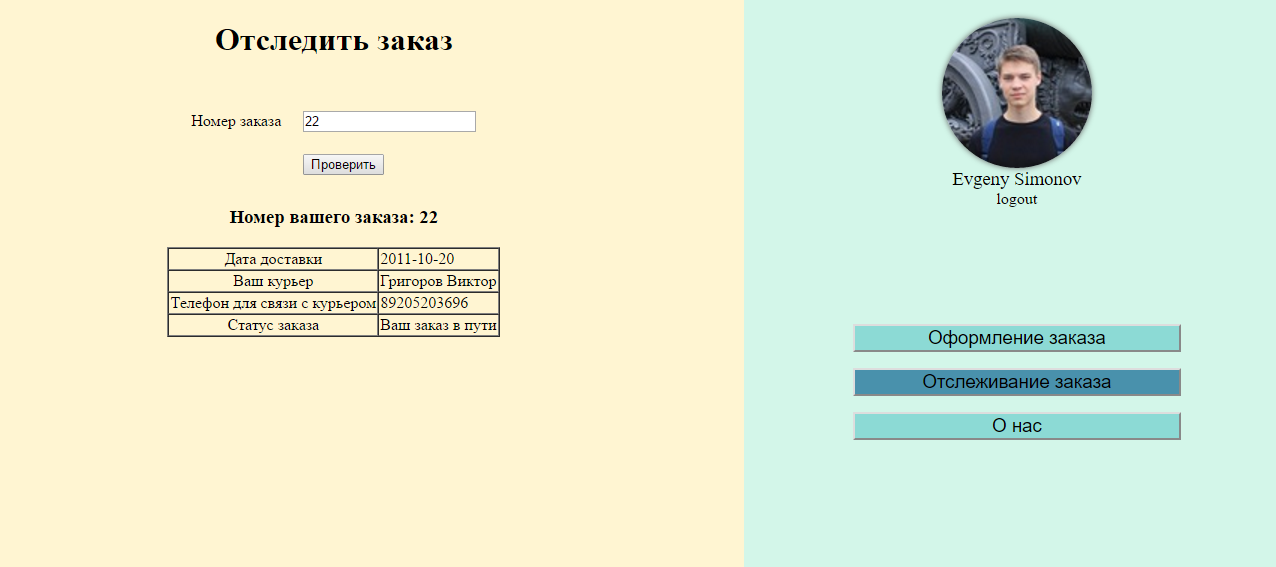
#### Рис. 8. Правила пользования системой.

При нажатии на кнопку “Заказать!” пользователь подтверждает факт согласия с правилами использования системы (о чем говорится рядом с кнопкой “Заказать!”). На данном изображении приведены эти правила.



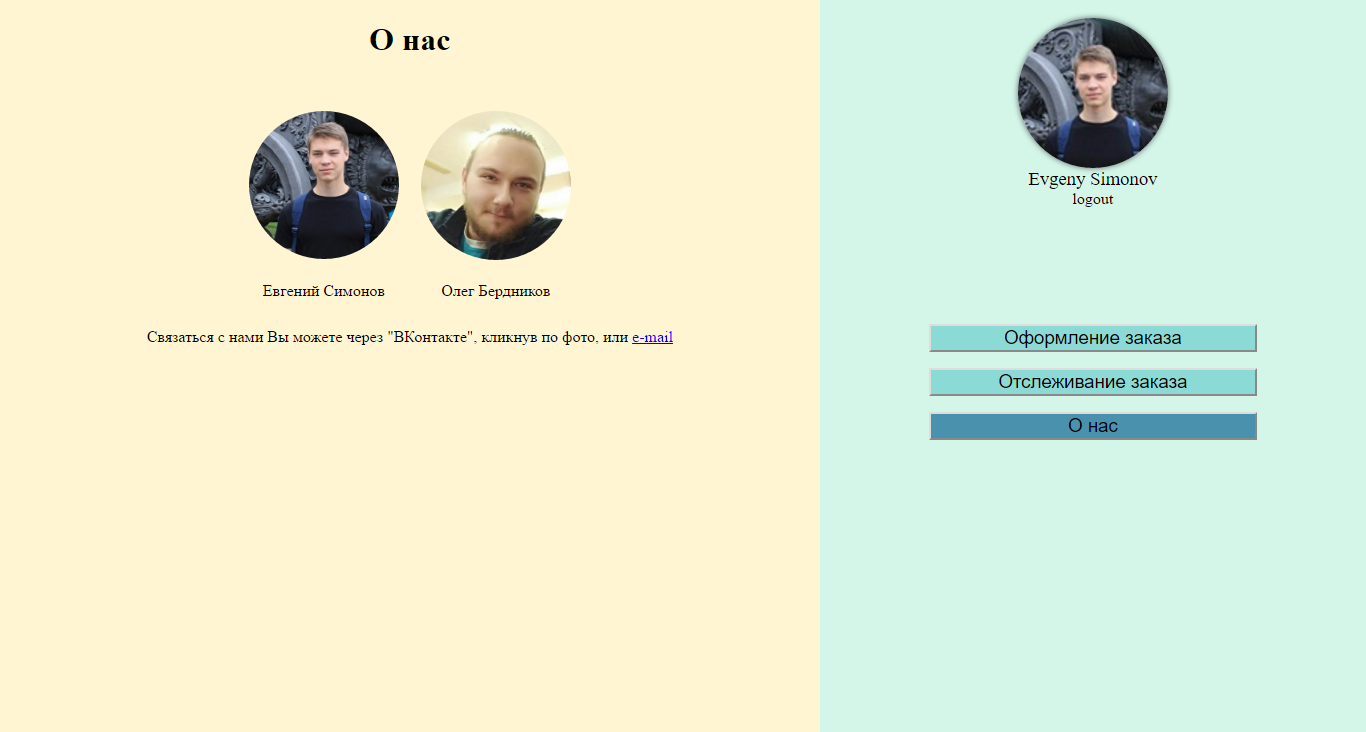
#### Рис. 9. Страница отслеживания заказа до ввода данных

После добавления нового заказа система переходит на страницу оформления заказа (изображена выше). Пользователь может перейти на нее и самостоятельно, выбрав соответствующий пункт меню в правой части экрана.



#### Рис. 10. Страница отслеживания заказа после ввода номера заказа

На данном изображении показано как пользователь может отслеживать свой заказ: после ввода номера заказа на странице появляется информация о заказе и курьере, работающим с ним. При желании, пользователь может связаться с курьером по телефону.



#### Рис. 11. Изображение вкладки “О нас”

На изображении представлена страница “О нас”, на которой пользователь может найти ссылки на страницы социальной сети Вконтакте разработчиков системы. Связаться с разработчиками пользователь может как с помощью социальной сети Вконтакте, так и с помощью электронной почты.

# 8. Тестирование

## 8.1. Модульное тестирование

Во время модульного тестирования было необходимо проверить правильность работы отдельных модулей.

Были протестированы модули:

1. Авторизация пользователя;
2. Обмен запросами с базой данных;
3. Защита от SQL-инъекций.

## 8.2. Функциональное тестирование

Во время функционального тестирования было необходимо вручную проверить все варианты использования системы различными группами пользователей.

Варианты использования:

1. Авторизация пользователя, который неавторизован “Вконтакте”. При вводе неверной комбинации логина и пароля система сообщает об ошибке и предлагает пользователю повторить ввод. При правильном вводе пользователь становится авторизованным.
2. Выход из системы. При нажатии на кнопку “logout” из системы сессия пользователя уничтожается, открывается главная страница для неавторизованного пользователя.
3. Добавление нового заказа. Для заполнения полей добавления заказа пользователю необходимо пройти процедуру авторизации. Теперь после того как пользователь заполнит поля и нажмет кнопку “Заказать!” система проверит все введенные им данные на корректность. Если какие-то поля будут заполнены неверно, пользователю будет предложено перезаполнить их. Если же ошибок ввода не будет, пользователь будет переведен на страницу проверки заказа.
4. Проверка заказа. ля заполнения полей добавления заказа пользователю необходимо пройти процедуру авторизации. Теперь пользователь вводит номер своего заказа и нажимает “Проверить”. Если был введен некорректный номер заказа, появится сообщение об ошибке. Если номер заказа введен корректно, появится информация о заказе.

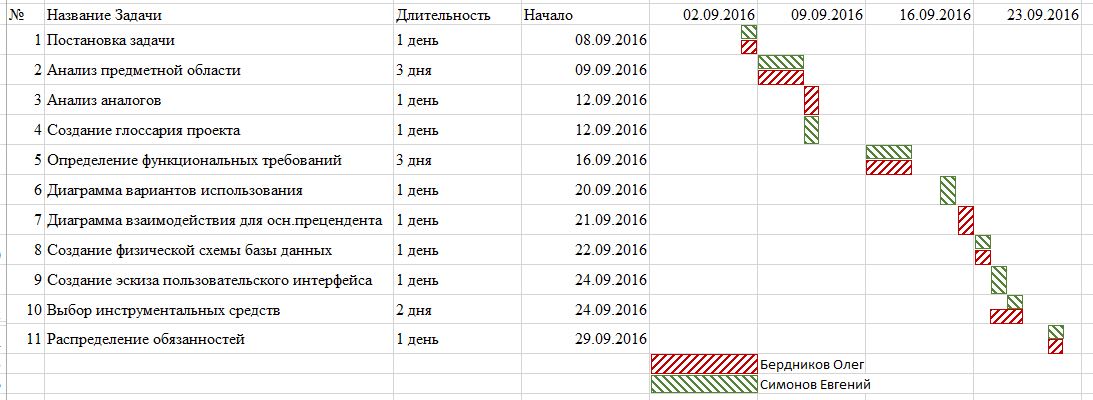
Просмотр информации о разработчиках. При нажатии на фотографии разработчиков открывается профиль одного из них в социальной сети Вконтакте.

# 9. Календарный план

В состав команды разработчиков вошли:

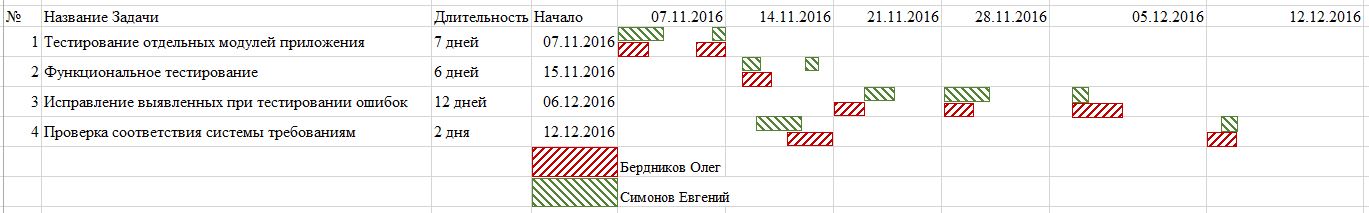
1. Симонов Евгений
2. Бердников Олег

Далее будет представлено расписание работ, которые вели разработчики, на 3 этапа, описанных в пункте “Планирование работ”.



#### Рис.12 Диаграмма Ганта на этапе разработки технического задания

#### Рис. 13. Диаграмма Ганта на этапе проектирования



#### Рис. 14. Диаграмма Ганта на этапе тестирования

# 10. Заключение

Целью работы было упрощение доставки и покупки любых товаров. Целью данной курсовой работы является реализация веб-приложения, которое:

* позволяет пользователю создавать заказ на доставку чего-либо;
* позволяет пользователю осуществлять отслеживание сделанного заказа;
* позволяет администратору отслеживать появление новых заказов.

В результате работы над проектом было создано веб-приложение, которое решает следующие задачи:

* создание заказа на доставку чего угодно;
* просмотр сведений о том, на какой стадии находится доставка заказа;
* предоставляет заказчику информацию о его курьере;
* предоставляет пользователю информацию о разработчиках;
* поддерживает авторизацию пользователей (благодаря социальной сети Вконтакте);
* имеет защиту от несанкционированного доступа и SQL-инъекций;
* имеет понятный интерфейс.

## Приложение 1. Руководство пользователя

1. В адресной строке браузера набрать адрес сервиса.
2. Если Вам интересна информация о сервисе, то Вы можете перейти на вкладку “О нас” для просмотра информации о сервисе и его разработчиках. Данная информация доступна без авторизации.
3. Если Вы хотите оформить заказ, то необходимо пройти авторизацию. *С инструкцией о том, как пройти авторизацию, вы можете ознакомиться ниже.*

Чтобы оформить новый заказ, заполните поля формы. Обязательными для заполнения являются поля названия товара, Вашего адреса, даты доставки и номера телефона. После заполнения нажмите кнопку “Заказать!”, и Ваш заказ будет оформлен. Нажимая кнопку “Заказать!” Вы подтверждаете согласие с правилами использования системы.

1. Если Вы хотите узнать статус своего заказа, пройдите процедуру авторизации.

*С инструкцией о том, как пройти авторизацию, вы можете ознакомиться ниже.*

После авторизации перейдите на вкладку “Отслеживание заказа”. Отобразится страница, в которой необходимо ввести номер Вашего заказа и нажать кнопку “Проверить”. После этого на отобразится информация о Вашем заказе.

Процедура авторизации:

Для авторизации необходимо нажать на кнопку  в правой части страницы. Если вы являетесь авторизованным пользователем на сайте vk.com, то авторизация на нашем сервисе пройдет автоматически. В противном случае Вам будет предложено ввести логин и пароль своей страницы Вконтакте. После авторизации на странице сервиса будут отображены Ваше имя, фамилия и фотография.