

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ  
ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (ФГБОУ ВО «ВГУ»)

# **Техническое задание**

## **Разработка web-приложения**

### **«GoldenFish»**

(в соответствии с ГОСТ 34)

Заказчик

Ассистент Тарасов В.С.

Исполнители

Никонова С.А., Скворцова Е.М., Шурыгина А.А.

Воронеж, 2020

## Оглавление

<b>1. Общие сведения .....</b>	<b>3</b>
1.1. Наименование заказчика .....	3
1.2. Наименование Исполнителя .....	3
1.3. Основание для разработки .....	3
1.4. Плановые сроки начала и окончания работ .....	3
1.5. Понятия и термины .....	3
<b>2. Назначение и цели создания системы .....</b>	<b>5</b>
2.1. Назначение системы .....	5
2.2. Цели создания .....	5
<b>3. Требования к приложению .....</b>	<b>6</b>
3.1. Требования к структуре приложения .....	6
3.2. Функциональные требования .....	6
3.2.1. Основные функциональные возможности системы .....	6
3.2.2. Диаграммы использования .....	7
3.2.3. Описание вариантов использования .....	8
3.3. Нефункциональные требования .....	14
3.3.1. Требования к пользовательскому интерфейсу .....	14
3.3.2. Требования к безопасности системы .....	14
3.3.3. Требования к лингвистическому обеспечению .....	14
3.4. Требования к видам обеспечения .....	14
3.4.1. Технический обзор .....	14
3.4.2. Требования к техническому обеспечению .....	15
<b>4. Порядок контроля и приемки .....</b>	<b>16</b>

## **1. Общие сведения**

### **1.1. Наименование Заказчика**

Ассистент Тарасов Вячеслав Сергеевич, кафедра программирования и информационных технологий.

### **1.2. Наименование Исполнителя**

Студентка Никонова Светлана Андреевна, кафедра технологий обработки и защиты информации.

Студентка Скворцова Екатерина Михайловна, кафедра информационных технологий управления.

Студентка Шурыгина Ангелина Александровна, кафедра технологий обработки и защиты информации.

### **1.3. Основание для разработки**

Необходимость заказчика в наличии веб-приложения, позволяющего вести учет своего списка желаний и списка подарков для друзей.

### **1.4. Плановые сроки начала и окончания работ**

Плановый срок начала работ – 15 марта 2020 г.

Плановый срок окончания работ – июнь 2020 г.

### **1.5. Понятия и термины**

В Техническом Задании используются следующие понятия, термины и аббревиатуры:

*ВИ* - вариант использования.

*Система* - краткое наименование разрабатываемой системы.

*Аутентификация* - прохождение проверки подлинности.

*Авторизация* - предоставление и проверка прав на совершение каких-либо действий в Системе.

*Пользователь* - авторизованный на сайте человек, пользующийся функционалом приложения.

*Желание* - стремление к осуществлению чего-либо или обладанию чем-либо.

*Подарок* - осуществление желания другого человека.

*Юзернейм* - псевдоним пользователя в Системе.

*Чекбокс* - элемент графического пользовательского интерфейса, позволяющий пользователю управлять параметром с двумя состояниями: включено и отключено.

*Фреймворк* - программная оболочка, позволяющая упростить и ускорить решение типовых задач, характерных для данного языка программирования.

*Библиотека* - сборник подпрограмм или объектов, используемых для разработки программного обеспечения.

*СУБД* - система управления базами данных.

*База данных* - хранилище данных.

*Модальное окно* в графическом интерфейсе пользователя - окно, которое блокирует работу пользователя с родительским приложением до тех пор, пока пользователь его не закроет.

*Хеширование* - преобразование произвольного массива данных в состоящую из букв и цифр строку фиксированной длины.

*SQL-инъекция* - внедрение в запрос произвольного SQL-кода для взлома сайта.

*Frontend* - пользовательский интерфейс, работающий на клиентской стороне приложения.

*Backend* - программно-аппаратная часть сервиса.

*REST API* - стиль архитектуры программного обеспечения для построения распределенных масштабируемых веб-сервисов.

## **2. Назначение и цели создания системы**

### **2.1. Назначение системы**

Назначением системы является ведение списка желаемых подарков, а также учет подарков, которые пользователь планирует подарить пользователям-друзьям.

### **2.2. Цели создания**

Облегчить пользователям выбор подарков для своих друзей.

### **3. Требования к приложению**

#### **3.1. Требования к структуре приложения**

Архитектура приложения будет основана на шаблоне проектирования MVC (Model-View-Controller), который предполагает разделение кода на 3 части:

- Model хранит данные, необходимые для работы Системы и предоставляет методы для работы с ними.
- Controller обеспечивает связь между пользователем и Системой: обращается к Model для получения или изменения данных и передает их View.
- View отвечает за отображение переданных данных в графическом интерфейсе.

Также предполагается разделение на клиентскую (front-end) и серверную (back-end) части, взаимодействие между которыми будет организовано с помощью REST API.

Более подробно используемые технологии описаны в п. 3.4.1

#### **3.2. Функциональные требования**

##### **3.2.1. Основные функциональные возможности Системы**

Система предоставляет следующие возможности для пользователей:

- Зарегистрироваться/проходить аутентификацию, при наличии аккаунта в Системе
- Редактировать свой профиль (E-mail, Юзернейм, Имя, Фамилия, Дата рождения)
- Осуществлять поиск пользователя по юзернейму
- Управлять списком своих друзей (добавлять/удалять друзей)
- Управлять списком своих желаний (добавлять/удалять желания)
- Редактировать желание (добавлять более подробное описание, ссылки на сторонние ресурсы (например, карточку товара в интернет-магазине), изображение)
- Отмечать свои желания как исполненные (перемещать их в список «Исполненные желания»)
- Просматривать списки желаний своих друзей
- Отмечать желания друзей (добавлять их в свой список «Хочу подарить»)

Более подробно сценарии использования Системы представлены в п. 3.2.3.

### 3.2.2. Диаграммы использования

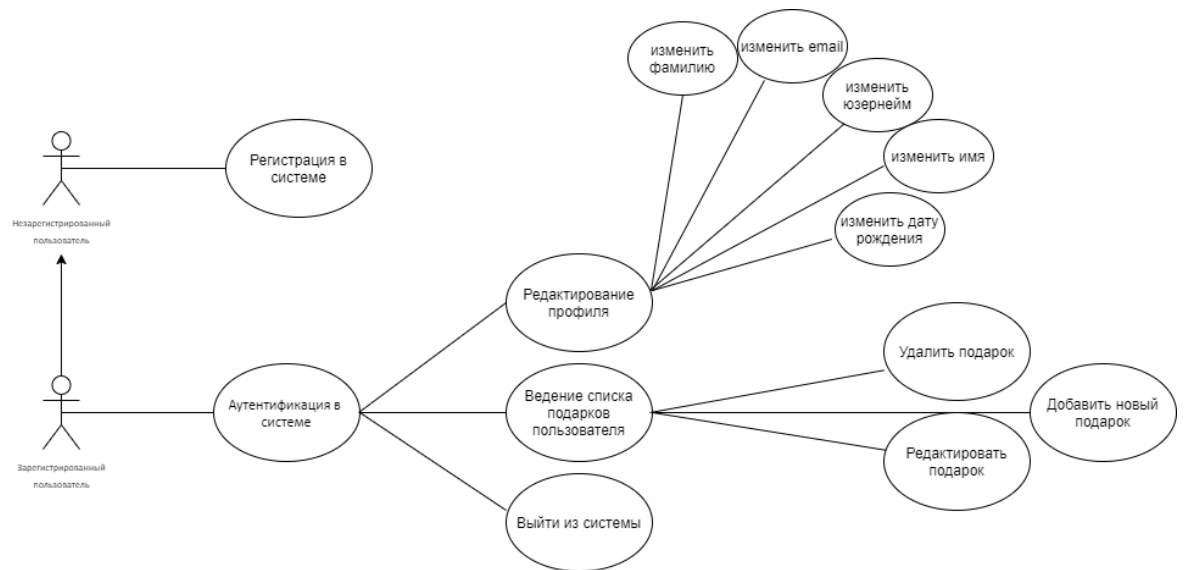


Диаграмма 1. Регистрация и редактирование профиля

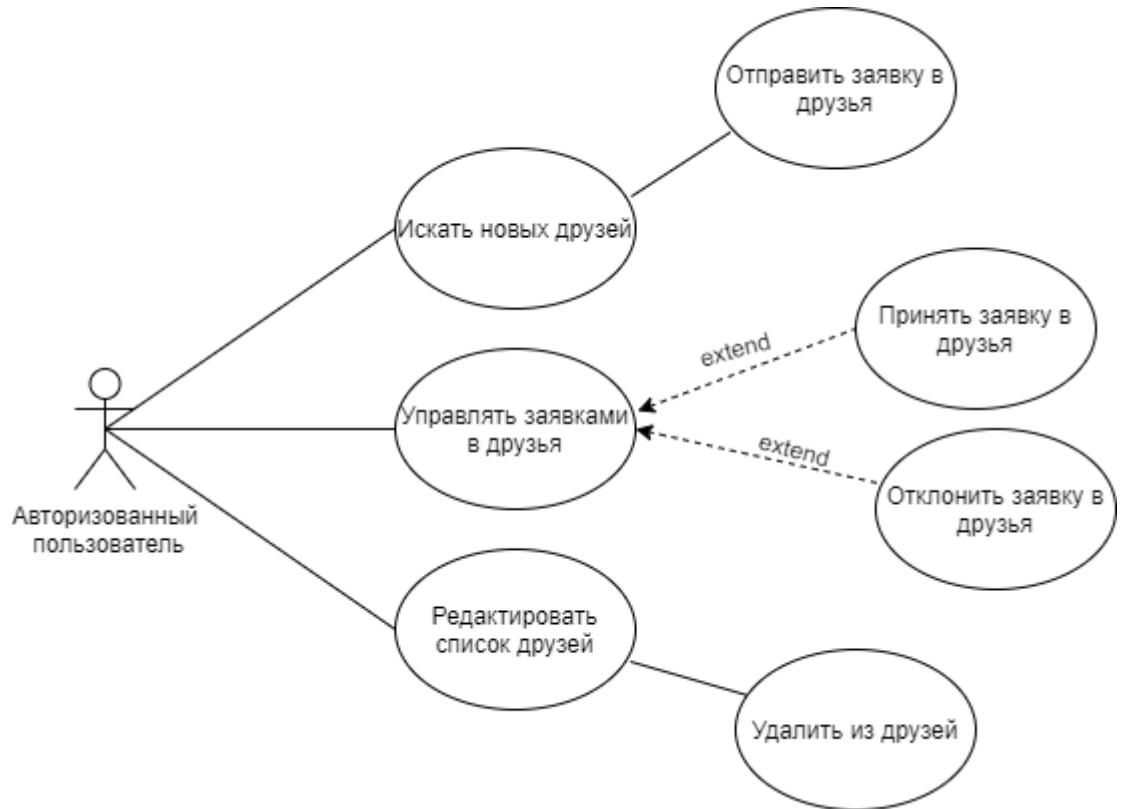


Диаграмма 2. Управление списком друзей



Диаграмма 3. Управление списком желаний

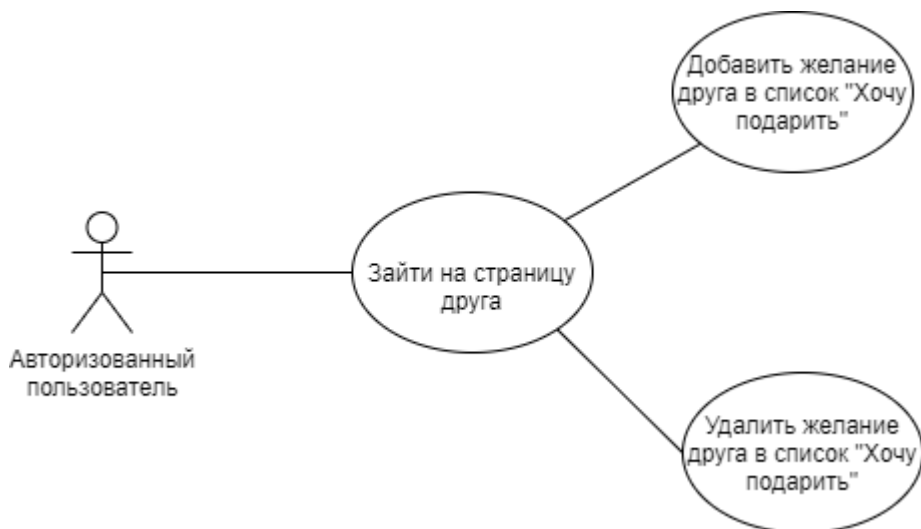


Диаграмма 4. Управление списком желаний друзей

### 3.2.3. Описание вариантов использования

#### 3.2.3.1. ВИ «Зарегистрироваться в Системе»

##### 3.2.3.1.1. Описание ВИ

Незарегистрированный пользователь должен иметь возможность зарегистрироваться в Системе

##### 3.2.3.1.2. Предусловия

Пользователь не зарегистрирован в Системе

##### 3.2.3.1.3. Основной поток действий пользователя



**3.2.3.1.3.1.** Незарегистрированный пользователь заходит на главную страницу сайта

**3.2.3.1.3.2.** Незарегистрированный пользователь переходит по ссылке «Зарегистрироваться» на страницу регистрации.

**3.2.3.1.3.3.** Страница регистрации содержит форму со следующими обязательными полями для заполнения: E-mail, Юзернейм, Пароль, Имя, Фамилия, и необязательными: Дата рождения, а также кнопку «Зарегистрироваться».

**3.2.3.1.3.4.** Незарегистрированный пользователь заполняет предложенные поля и нажимает на кнопку «Зарегистрироваться».

**3.2.3.1.3.5.** Система проверяет:

- все ли обязательные поля заполнены
- поле E-mail на корректность
- поле Имя на минимальную длину (2 символа), которое должно состоять из символов латинского или русского алфавита
- поле Фамилия на минимальную длину (2 символа), которое должно состоять из символов латинского или русского алфавита
- поле Юзернейм на минимальную длину (3 символа), которое также не должно начинаться с цифры или специального символа
- поле Пароль на минимальную длину (5 символов), которое также не должно начинаться с цифры или специального символа
- уникальность Юзернейма и E-mail (не зарегистрирован ли в Системе пользователь с такими же данными)

Если проверка успешно пройдена, поток продолжается, иначе рядом с формой появляется сообщение об ошибке и сценарий переходит в п. 3.2.2.1.3.3, сами поля остаются заполненными введенными ранее данными.

**3.2.3.1.3.6.** Система регистрирует нового пользователя с введенными данными.

**3.2.3.1.3.7.** Система перенаправляет авторизованного пользователя на страницу желаний.

### **3.2.3.2. ВИ «Пройти аутентификацию»**

#### **3.2.3.2.1. Описание ВИ**

Зарегистрированный пользователь должен иметь возможность пройти аутентификацию в Системе

#### **3.2.3.2.2. Предусловия**

Пользователь не авторизован в Системе

#### **3.2.3.2.3. Основной поток действий пользователя**

**3.2.3.2.3.1.** Неавторизованный пользователь заходит на главную страницу сайта.

**3.2.3.2.3.2.** Неавторизованный пользователь переходит по ссылке “Войти” на страницу аутентификации.

**3.2.3.2.3.3.** Страница аутентификации содержит форму с полями для ввода E-mail и Пароля, а также кнопку «Войти»

**3.2.3.2.3.4.** Пользователь заполняет предложенные поля

**3.2.3.2.3.5.** Система проверяет:

- все ли поля заполнены
- существует ли пользователь с таким E –mail
- совпадает ли введенный пароль с тем, что связан с данным аккаунтом

Если проверки пройдены успешно, сценарий продолжается, иначе Система отображает сообщение о неправильно введенном логине и/или пароле и сценарий переходит к п. 4.2.2.3.3, при этом только поле E-mail содержит введенные данные

**3.2.3.2.3.6.** Система авторизует пользователя и перенаправляет его на личную страницу

#### **3.2.3.3. ВИ «Отправить заявку ‘в друзья’»**

##### **3.2.3.3.1. Описание ВИ**

Авторизованный пользователь может иметь возможность добавлять других пользователей в друзья.

##### **3.2.3.3.2. Предусловия**

Пользователь авторизован в Системе.

##### **3.2.3.3.3. Основной поток действий пользователя**

**3.2.3.3.3.1.** Пользователь делает активной строку поиска друзей в шапке приложения.

**3.2.3.3.3.2.** Вводит юзернейм и нажимает на кнопку поиска.

**3.2.3.3.3.3.** Система выводит список результатов.

**3.2.3.3.3.4.** Если пользователи не найдены, система оповещает об этом надписью: «Поиск не дал результатов. Измените параметры поиска». Пользователь переходит к пункту 3.2.2.3.3.1, иначе поток продолжается.

**3.2.3.3.3.5.** Пользователь нажимает на кнопку Добавить в друзья рядом с найденным пользователем. Последнему приходит заявка в друзья.

#### **3.2.3.4. ВИ «Принять заявку ‘в друзья’»**

##### **3.2.3.4.1. Описание ВИ**

Чтобы добавить друга, пользователь может принять его заявку в друзья.

##### **3.2.3.4.2. Предусловия**

Пользователь авторизован в Системе, в списке заявок в друзья есть актуальная заявка.

##### **3.2.3.4.3. Основной поток действий пользователя**

**3.2.3.4.3.1.** Пользователь переходит в раздел «Друзья».

**3.2.3.4.3.2.** Во вкладке «Заявки» находит нужную заявку.

**3.2.3.4.3.3.** Нажимает кнопку «Принять» рядом с ней.

#### **3.2.3.5. ВИ «Отклонить заявку ‘в друзья’»**

##### **3.2.3.5.1. Описание ВИ**

Пользователь имеет возможность отклонить заявку в друзья.

##### **3.2.3.5.2. Предусловия**

Пользователь авторизован в Системе, в списке заявок в друзья есть актуальная заявка.

##### **3.2.3.5.3. Основной поток действий пользователя**

**3.2.3.5.3.1.** Пользователь переходит в раздел «Друзья».

**3.2.3.5.3.2.** Во вкладке «Заявки» находит нужную заявку.

**3.2.3.5.3.3.** Нажимает кнопку «Отклонить» рядом с ней.

#### **3.2.3.6. ВИ «Удалить пользователя из друзей»**

##### **3.2.3.6.1. Описание ВИ**

Пользователь имеет возможность удалить другого пользователя из друзей

##### **3.2.3.6.2. Предусловия**

Пользователь авторизован в системе. Пользователь имеет в списке друзей пользователя, которого нужно удалить.

##### **3.2.3.6.3. Основной поток действий пользователя**

**3.2.3.6.3.1.** Пользователь переходит в раздел «Друзья»

**3.2.3.6.3.2.** Во вкладке «Друзья» находит нужного пользователя.

**3.2.3.6.3.3.** Нажимает рядом с именем пользователя кнопку «Удалить из друзей».

**3.2.3.6.3.4.** Удаленный пользователь исчезает из списка друзей удалившего, а удаливший исчезает из списка друзей удаленного

### **3.2.3.7. ВИ «Добавить желание»**

#### **3.2.3.7.1. Описание ВИ**

Пользователь имеет возможность добавить актуальное желание в свой список желаний.

#### **3.2.3.7.2. Предусловия**

Пользователь авторизован в системе

#### **3.2.3.7.3. Основной поток действий пользователя**

**3.2.3.7.3.1.** Пользователь переходит в раздел «Мои желания», где находится список его желаний.

**3.2.3.7.3.2.** Пользователь нажимает на кнопку «Добавить желание», появляется модальное окно, где нужно ввести название желания и можно добавить описание, ссылку на магазин, ссылку на изображение по желанию.

**3.2.3.7.3.3.** Пользователь вводит необходимые данные, нажимает на кнопку «Сохранить», после чего введенное желание добавляется в общий список желаний. Пока пользователь не введет название, кнопка «Сохранить» будет недоступна.

### **3.2.3.8. ВИ «Редактировать желание»**

#### **3.2.3.8.1. Описание ВИ**

Пользователь имеет возможность редактировать название желания, описание, ссылку на магазин и изображение.

#### **3.2.3.8.2. Предусловия**

В списке желаний пользователя есть хотя бы одно желание.

#### **3.2.3.8.3. Основной поток действий пользователя**

**3.2.3.8.3.1.** Пользователь заходит в раздел «Мои желания»

**3.2.3.8.3.2.** Пользователь нажимает на любое желание из списка.

**3.2.3.8.3.3.** Появляется модальное окно-форма для редактирования желания с полями: название, описание, ссылка на магазин, ссылка на изображение.

**3.2.3.8.3.4.** Пользователь может изменить имеющиеся значения и нажать на кнопку «Сохранить» для подтверждения изменений.

### **3.2.3.9. ВИ «Удалить желание»**

#### **3.2.3.9.1. Описание ВИ**

Пользователь имеет возможность удалить желание из общего списка.

#### **3.2.3.9.2. Предусловия**

Пользователь авторизован в Системе, в общем списке желаний есть хотя бы одно желание.

#### **3.2.3.9.3. Основной поток действий пользователя**

**3.2.3.9.3.1.** Пользователь заходит в раздел «Мои желания».

**3.2.3.9.3.2.** Пользователь нажимает на кнопку «Удалить», находящуюся напротив того желания, которое нужно удалить.

**3.2.3.9.3.3.** Желание исчезает из общего списка желаний.

#### **3.2.3.10. ВИ «Добавить желание в список исполненных желаний»**

##### **3.2.3.10.1. Описание ВИ**

Авторизованный пользователь имеет возможность добавить одно из своих желаний в список исполненных.

##### **3.2.3.10.2. Предусловия**

Пользователь авторизован в Системе, в списке актуальных желаний есть хотя бы одно желание.

##### **3.2.3.10.3. Основной поток действий пользователя**

**3.2.3.10.3.1.** Пользователь заходит в раздел «Мои желания».

**3.2.3.10.3.2.** Пользователь выбирает из своего списка желаний то, которое он считает исполненным и нажимает на кнопку «Исполнено» напротив желания.

**3.2.3.10.3.3.** Желание исчезает из общего списка и появляется в списке исполненных желаний.

#### **3.2.3.11. ВИ «Добавить желание друга в список 'Хочу подарить'»**

##### **3.2.3.11.1. Описание ВИ**

Пользователь имеет возможность добавить желание друга в свой список «Хочу подарить».

##### **3.2.3.11.2. Предусловия**

Пользователь авторизован в Системе, у него есть хотя бы один друг, в списке желаний друга есть неотмеченные другими пользователями желания.

##### **3.2.3.11.3. Основной поток действий пользователя**

**3.2.3.11.3.1.** Пользователь переходит в раздел «Друзья».

**3.2.3.11.3.2.** Пользователь находит во вкладке «Друзья» имя нужного пользователя, нажимает на него.

**3.2.3.11.3.3.** Система перенаправляет пользователя на страницу выбранного друга, где находится список его желаний.

**3.2.3.11.3.4.** Пользователь выбирает желание из списка друга, которое еще не отмечено другим пользователем и нажимает на кнопку «Хочу подарить». Если желание уже отмечено, кнопка становится недоступна.

**3.2.3.11.3.5.** Отмеченное желание появляется в списке пользователя «Хочу подарить». В списке друга данного пользователя напротив отмеченного желания появляется юзернейм пользователя, отметившего данное желание.

### **3.2.3.12. ВИ «Удалить желание друга из списка 'Хочу подарить'»**

#### **3.2.3.12.1. Описание ВИ**

Пользователь имеет возможность отказаться от подарка, который он запланировал подарить.

#### **3.2.3.12.2. Предусловия**

Пользователь авторизован в Системе, в его списке «Хочу подарить» есть хотя бы одно желание.

#### **3.2.3.12.3. Основной поток действий пользователя**

**3.2.3.12.3.1.** Пользователь переходит в раздел «Хочу подарить».

**3.2.3.12.3.2.** Пользователь нажимает на кнопку «Удалить» рядом с нужным желанием, которое исчезает из списка «Хочу подарить». У пользователя – владельца данного желания – желание становится неотмеченным (исчезает юзернейм пользователя, отметившего данное желание).

### **3.3. Нефункциональные требования**

#### **3.3.1. Требования к пользовательскому интерфейсу**

Пользовательский интерфейс приложения должен удовлетворять следующим основным требованиям:

- Отсутствие нагромождений;
- Использование высококонтрастных шрифтов, размера не менее 15px;
- Единый стиль всех страниц приложения.

#### **3.3.2. Требования к безопасности системы**

##### **3.3.2.1. Требования к защите от несанкционированного доступа**

Приложение должно предусматривать хеширование паролей и защиту от SQL-инъекций.

#### **3.3.3. Требования к лингвистическому обеспечению**

Приложение должно поддерживать русский язык.

### **3.4. Требования к видам обеспечения**

#### **3.4.1. Технический обзор**

Для создания приложения должны быть использованы следующие инструменты:

- Язык программирования Python, с установленным фреймворком Flask и библиотекой SQLAlchemy. Возможно использование других сторонних библиотек.
- СУБД PostgreSQL.

#### **3.4.2. Требования к техническому обеспечению**

Приложение должно корректно работать в браузере Google Chrome v.80.0.3987.

#### **4. Порядок контроля и приемки**

Заказчик принимает разработанное приложение при выполнении следующих пунктов:

- Успешное прохождение всех функциональных тестов, основанных на сценариях, подробно описанных в п. 3.2.3.;
- Приложение соответствует заявленным требованиям.