

**МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ**

**«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)**

**Факультет компьютерных наук
Кафедра информационных систем**

Техническое задание

«Paranoid Club»

Обучающийся _____ Андросов А.Д.

Обучающийся _____ Дудкин Н.А.

Обучающийся _____ Астрединов И.Г.

Руководитель _____ Тарасов В.С.

Воронеж 2022

1. Общие сведения

1.1. Наименование системы

1.1.1. Полное наименование: Paranoid Club.

1.1.2. Краткое наименование: РС.

1.2. Основания для проведения работ

Необходимость заказчика в веб-приложении для обмена мгновенными сообщениями с возможностью ограничить круг собеседников.

1.3. Наименование Заказчика и Исполнителя

1.3.1. Заказчик

Старший преподаватель Тарасов Вячеслав Сергеевич
кафедра программирования и информационных технологий

1.3.2. Исполнитель

Студент Андросов Алексей Дмитриевич, кафедра информационных систем

Студент Дудкин Николай Александрович, кафедра информационных систем

Студент Астрединов Илья Геннадьевич, кафедра информационных систем

1.4. Плановые сроки начала и окончания работы

Плановые сроки начала работы - февраль 2022г.

Плановые сроки окончания работы - июнь 2022г.

2. Назначение и цели создания системы

2.1. Назначение системы

2.1.1. РС предназначен для ограничения круга лиц участвующих в переписке. Основным назначением РС является коммуникация между пользователями продукта.

2.2. Цели создания системы

2.2.1. РС создается с целью:

- обеспечение обмена текстовыми сообщениями между пользователями в личной переписке или групповых чатах;
- создание системы для управления платформами для общения между пользователями;
- повышения уверенности пользователей в конфиденциальности личной переписки

3. Требования к системе

3.1. Требования к системе в целом

3.1.1. Требования к структуре и функционированию системы.

Продукт должен состоять из следующих приложений:

- веб-приложение для обмена сообщениями между пользователями (далее мессенджер);
- панель управления для команды разработки, позволяющая создавать платформы для общения пользователей в мессенджере.

Клиенты панели администрирования и мессенджера должны являться веб страницами.

Для мессенджера должны быть разработаны следующие конфигурации сборки:

- конфигурация сборки для команды разработки. После сборки в данной конфигурации в мессенджере должно быть доступно множество платформ для общения пользователей и доступ к информации об этих платформах при запросе по http протоколу к панели управления;
- пользовательская конфигурация сборки. Данная конфигурация предусматривает возможность для пользователя вести самостоятельно разработку мессенджера. Мессенджер собранный в данной конфигурации должен быть независим от панели управления. Внутри мессенджера будет доступна лишь одна платформа для общения.

Также должен быть разработан скрипт и файл конфигураций, позволяющий легко разместить мессенджер на сервере конечному пользователю.

Панель управления должна давать возможность участникам команды разработки создавать платформы для общения внутри мессенджера. Для регистрации новых пользователей в панели управления необходимо разработать скрипт, запускаемый из терминала. После регистрации в панели управления пользователь будет иметь возможность изменить пароль. При создании платформы внутри мессенджера также должен быть создан аккаунт первого пользователя на созданной платформе. У этого пользователя будет имя Shop и случайно сгенерированный пароль, состоящий из букв русского и латинского алфавита в верхнем и нижнем регистре, цифр и нижнего подчеркивания. Регистрационные данные пользователя и имя созданной платформы должны быть отправлены пользователю на email. В случае, если отправить регистрационные данные пользователю не получилось на экран администратора должно быть выведено предупреждение с причиной возможной ошибки.

Мессенджер должен иметь следующие страницы:

- страница авторизации;
- страница с переписками пользователя;
- страница с контактами всех пользователей на платформе;
- страница с настройками аккаунта пользователя.

На странице авторизации пользователь должен иметь возможность ввести свои авторизационные данные: логин, пароль, имя платформы, на которую он заходит. После успешной авторизации пользователь перенаправляется на страницу с переписками. При неудачной авторизации на экране пользователя появляется предупреждение с возможной причиной ошибки.

Если пользователь авторизован, но не изменил первоначальные регистрационные данные, то на странице с переписками должно открываться страница активации аккаунта, где пользователь должен поменять имя и пароль. Пользователь должен быть не способен перейти куда-либо кроме страницы логина до тех пор, пока не поменяет свои регистрационные данные. После успешной смены регистрационных данных он выйдет со страницы активации аккаунта. При неудачной смене регистрационных данных на экране пользователя должно отображаться предупреждение с описанием возможной причины ошибки.

На странице с переписками у пользователя должно быть 2 панели. На первой панели отображается информация о пользователях, с которыми он вел переписку, и групповых чатах, в которых состоит пользователь. Эта информация включает в себя: имя группового чата или пользователя, с которым велась переписка, текст последнего сообщения, дата и время последнего сообщения. На второй панели отображаются личные сообщения из переписки, которую выбрал пользователь.

Пользователь должен иметь возможность создать групповой чат с другими пользователями на текущей платформе и отправить личное сообщение в переписке с другим пользователем или групповом чате. Если другие пользователи отправили текущему пользователю сообщение в личную переписку или групповой чат, то это сообщение должно отобразиться на экране пользователя, если у него открыта личная переписка или групповой чат, в которые пришло сообщение. Если у пользователя открыта личная переписка с другим пользователем, то после клика по имени собеседника должно открываться модальное окно в котором должна отображаться информация о собеседнике. Эта информация включает в себя имя собеседника, описание профиля собеседника, имя пользователя, который пригласил собеседника на платформу или информация о том, что собеседник является первым пользователем на текущей платформе. Пользователь должен иметь доступ со страницы с переписками на страницу с настройками, страницу с контактами всех пользователей на платформе.

На странице с настройками пользователь должен иметь возможность поменять пароль от аккаунта, поменять описание своего аккаунта, выйти из аккаунта и вернуться на страницу с переписками.

На странице с контактами всех пользователей на платформе будут отображаться имена всех пользователей зарегистрированных на текущей платформе в алфавитном порядке. При клике на имя пользователя из списка будет открываться окно переписки с ним. В модальном окне с контактами всех пользователей должна быть предусмотрена возможность вернуться на страницу с переписками.

Также должна быть предусмотрена возможность сгенерировать ссылку-приглашение. После генерации ссылки приглашения на платформе должен быть создан аккаунт нового пользователя с именем Shon и порядковым номером аккаунта в системе (например: Shon1, Shon2 и т.д.) и случайно сгенерированный пароль, состоящий из букв русского и латинского алфавите в верхнем и нижнем регистре, цифр и нижнего подчеркивания. Регистрационные данные будут отправлены новому пользователю по email.

Для активации аккаунта пользователь должен перейти по ссылке-приглашению. До тех пор, пока пользователь не активировал аккаунт он не может зайти в систему.

Если приложение было собрано с помощью пользовательской конфигурации сборки, то при логине не нужно имя платформы.

Пользователи должны иметь возможность самостоятельно разрабатывать, собирать и разворачивать мессенджер. Для этого необходимо разработать соответствующие скрипты, задокументировать информацию о том, как самостоятельно добавить модуль в систему.

3.1.2. Требования к эргономике и технической эстетике

В части внешнего оформления:

- интерфейсы подсистем должен быть типизированы;
- должно быть обеспечено наличие локализованного (русскоязычного) интерфейса пользователя.

3.1.3. Требования к защите информации от несанкционированного доступа:

- пароли пользователей и администраторов должны быть зашифрованы с помощью функции формирования ключа bcrypt;
- при авторизации на клиенте должен сохраняться AccessToken и отправляться на сервер в http заголовке Authorization;
- AccessToken должен быть активным в течении 30 минут;
- при авторизации на сервере должны устанавливаться cookie с флагом HttpOnly, в которых будет храниться RefreshToken для обновления AccessToken;
- RefreshToken должен быть активным в течении 30 дней, по прошествию этого срока пользователь должен заново ввести регистрационные данные.

3.1.4. Требования к платформам на которых будет работать система.

Система должна работать в следующих браузерах: Google Chrome v. 99, Mozilla Firefox v. 99

Система должна отображаться корректно при разрешениях экрана 768px - 1440px.

3.1.5. Требования к производительности приложения:

- время загрузки первого байта при запросах к серверу не должно превышать 500мс
- время отображения первого элемента на экране пользователя не должно превышать 1.8с

3.2. Требования к информационному обеспечению.

При разработке системы должны применяться следующие технологии: NodeJs, NestJS, MongoDB, TypeScript.

Для документирования API необходимо использовать Swagger.

Для управления версиями проекта должен использоваться Git.

Проект должен храниться на GitHub.