

ФГАОУ ВО «САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ»

КАФЕДРА ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

КУРСОВАЯ РАБОТА

Программирование интернет-приложений

Саржевский Иван Анатольевич

Лабушев Тимофей Михайлович

Группа Р3202

Санкт-Петербург

2018 г.

Sumaju nikki

БРАУЗЕРНАЯ МНОГОПОЛЬЗОВАТЕЛЬСКАЯ ТЕКСТОВАЯ ИГРА
ЖАНРА ZERO-PLAYER GAME

Содержание

1	Введение	2
1.1	Цель создания	2
1.2	Целевая аудитория	2
2	Требования к аппаратно-программному обеспечению	2
2.1	Требования к серверному обеспечению	2
2.2	Требования к клиентскому обеспечению	2
3	Требования к архитектуре системы	3
3.1	Глоссарий	3
3.2	Уровень back-end	3
3.3	Уровень front-end	3

1 Введение

Предметом разработки является платформа для многопользовательской текстовой игры жанра *Zero-Player Game (ZPG)*, использующая браузерный интерфейс и сохраняющая прогресс игрока на игровом сервере.

В основе жанра лежит концепция протекания игрового процесса без вмешательства со стороны пользователя. При этом игроку предоставляются возможности ускорения своего игрового прогресса и взаимодействия с другими пользователями.

Предполагается, что взаимодействие с платформой будет представлять собой не длинные сессии, свойственные традиционным играм, а короткие, но частые посещения, характерные для социальных сетей.

1.1 Цель создания

Платформа разрабатывается с целью предоставления творческим молодым людям возможности самовыражения в игровой форме. Платформа предоставляет набор базовых игровых механик, в то время как содержательная часть создается самими пользователями, путём предоставления возможности формировать предложения для контента

1.2 Целевая аудитория

Творческие молодые люди, в возрасте 18-25 лет, знакомые с современными медиа, активные пользователи интернета. Не имеют времени/желания проводить много времени за игрой в компьютерные игры, ищут более подходящего под свой стиль жизни формата.

2 Требования к аппаратно-программному обеспечению

2.1 Требования к серверному обеспечению

Система, которая обеспечивает выполнение программных продуктов сервера приложений и хранение данных платформы, должна отвечать следующим требованиям:

- Поддержка операционной системой бинарного интерфейса приложений (ABI) Linux
- Наличие сервера баз данных PostgreSQL версии 9.6 и выше

2.2 Требования к клиентскому обеспечению

Браузерный интерфейс разрабатывается с учетом следующих требований к программному обеспечению на стороне пользователя:

- веб-браузер Google Chrome версии 67 и выше или Mozilla Firefox версии 61 и выше с включенным интерпретатором сценариев JavaScript,

- отсутствие запрета веб-страницам платформы доступа к внешним ресурсам, а именно изображениям, шрифтам, таблицам стилей CSS и сценариям JavaScript, в том числе блокировщиками рекламы.

3 Требования к архитектуре системы

3.1 Глоссарий

Уровень back-end — серверное приложение, с которым взаимодействует пользовательский интерфейс игры. Уровень back-end обеспечивает хранение данных, расчет игровых процессов, взаимодействие между пользователями.

Уровень front-end — браузерное приложение, которое реализует пользовательский интерфейс игры. Уровень front-end включает в себя HTML-страницы, сценарии JavaScript, таблицы стилей CSS.

3.2 Уровень back-end

1. Серверное приложение должно разрабатываться на платформе *JVM* с использованием фреймворков *Akka* и *Play*
2. Взаимодействие между уровнями front-end и back-end должно осуществляться посредством REST API; возможно использование протокола WebSockets для реализации обновлений в режиме реального времени
3. Серверное приложение должно реализовывать аутентификацию пользователей с поддержкой входа через социальные сети (OAuth), используя библиотеку *play-silhouette*. Пароли пользователей должны храниться как криптографический хэш
4. Серверное приложение должно публиковать push-уведомления об игровых событиях, используя сервис *Firebase Cloud Messaging*
5. Серверное приложение должно отправлять пользователям еженедельное новостное сообщение электронной почты, используя *JavaMail API*

3.3 Уровень front-end

1. Клиентское приложение должно разрабатываться с использованием фреймворка *Vue.js*
2. Веб-интерфейс должен быть адаптирован для отображения в трех режимах: *desktopном* (ширина экрана больше 1107px), *планшетном* (больше 889px) и *мобильном* (меньше 889px)
3. Веб-интерфейс должен предоставлять пользователю возможность подписаться на push-уведомления
4. Веб-интерфейс должен распознаваться в мобильной ОС Android как приложение, используя *Web App Manifest*