

Университет ИТМО
Факультет программной инженерии и компьютерной техники

Курсовая работа

тема “Система управления event”

1 часть

по дисциплине «Информационные системы»

Выполнили:
Дьячков Андрей Р3309
Захаркин Богдан Р3309

Преподаватель:
Райла Мартин

Санкт-Петербург
2024

Содержание

1. Введение

1.1 Цель

1.2 Область применения

1.3 Определения и аббревиатуры

1.4 Ссылки

1.5 Обзор документа

2. Общее описание

2.1 Функционал продукта

2.2 Описание пользователей

2.3 Влияющие факторы и зависимости

2.4 Ограничения

3. Спецификация требований

3.1 Функциональные требования

3.2 Требования к удобству использования

3.3 Требования к надежности

3.4 Требования к производительности

3.5 Ограничения разработки

4. Интерфейсы

4.1 Пользовательские интерфейсы

4.2 Аппаратные интерфейсы

4.3 Программные интерфейсы

4.4 Сетевые интерфейсы

5. Основные прецеденты

Прецедент: Создание аккаунта

Прецедент: Создание мероприятия

Прецедент: Регистрация на мероприятие

Прецедент: Участие в квизе

Прецедент: Оставить отзыв о мероприятии

Прецедент: Просмотр списка участников

6. Требования к лицензированию

Software Requirements Specification

1. Введение

1.1 Цель

Цель данного документа — описание функциональных и нефункциональных требований к системе. Разрабатываемая система представляет собой веб-приложение для создания карточек мероприятий, управления ими, а также возможности регистрации пользователей на эти мероприятия.

1.2 Область применения

Данный документ охватывает проект системы управления мероприятиями. В нём перечислены основные функциональные требования, а также требования к интерфейсам, надёжности, производительности и удобству использования системы. Он будет использоваться разработчиками, дизайнерами, тестировщиками и другими участниками процесса создания системы..

1.3 Определения и аббревиатуры

- Система – веб-приложение для управления мероприятиями.
- Клиент – пользователь, посещающий мероприятия.
- Организатор – пользователь, создающий и управляющий мероприятиями.
- API – интерфейс программирования приложений.
- UI – пользовательский интерфейс.
- UX — опыт взаимодействия пользователя с системой.

1.4 Ссылки

- Стандарт ISO/IEC 25010:2011 – Системная и программная инженерия – Модели качества систем и программных продуктов.

1.5 Обзор документа

Общее описание: данный раздел содержит описание факторов, влияющих на требования к продукту. Сами требования описываются в следующем разделе

Спецификация требований: Данный раздел содержит описание всех требований к разрабатываемой системе. Данное описание будет использоваться как разработчиками при разработке системы, так и тестировщиками в процессе проверки ее функционала

2. Общее описание

2.1 Функционал продукта

- Регистрация и авторизация пользователей.
- Создание, редактирование и удаление карточек мероприятий.

- Регистрация клиентов на мероприятия.
- Отправка уведомлений пользователям о событиях.
- Просмотр списка участников организатором.
- Интеграция с внешними API для оплаты билетов.
- Ведение статистики посещаемости.
- Возможность проведения квизов во время мероприятия
- Отправка отзывов на мероприятие

2.2 Описание пользователей

- **Клиент:** пользователь, который может просматривать мероприятия, регистрироваться на них и получать уведомления.
- **Организатор:** пользователь с правами на создание и управление событиями, просмотр статистики и управление регистрацией. Доступ к роли организатора предоставляется через настройки профиля.
- **Администратор:** пользователь с расширенными правами для управления учетными записями, карточками профилей и настройками системы.

2.3 Влияющие факторы и зависимости

- **Зависимость от сторонних API:** система зависит от внешних API, а значит изменения в этих API могут повлиять на работу системы.
- **Зависимость от сервера электронной почты:** система отправляет уведомления по электронной почте, что требует стабильной работы почтового сервера и правильной настройки Java Mail API
- **Сетевые требования и стабильность интернета:** для корректной работы интеграций с API и отправки уведомлений требуется стабильное интернет-соединение

2.4 Ограничения

- **Ограничение на использование API:** каждый сторонний API может иметь свои ограничения по количеству запросов в сутки (например, лимиты Yandex Metrica и Uniqode QR Code API)
- **Совместимость с браузерами:** приложение должно корректно работать в последних версиях популярных браузеров (Chrome, Safari, Яндекс.Браузер и других)
- **Требования к безопасности:** система должна использовать HTTPS для передачи данных пользователей и обеспечивать шифрование паролей и другой конфиденциальной информации
- **Поддержка мультиязычности:** система должна поддерживать несколько языков интерфейса, что требует адаптации контента и элементов UI для различных языков.
- **Поддержка мобильных устройств:** система должна корректно отображаться на экранах с разными разрешениями, в том числе на мобильных устройствах с разрешением от 360x640 пикселей

3. Спецификация требований

3.1 Функциональные требования

<i>Код</i>	<i>Требование</i>	<i>Приоритет</i>	<i>Мин. оценка, чел.ч</i>	<i>Макс. оценка, чел.ч</i>
FR1	Система должна позволять пользователю создавать аккаунт	MUST	20	30
FR2	Система должна позволять пользователю авторизоваться в существующий аккаунт по корректным данным для входа	MUST	15	30
FR3	Система должна позволять пользователю изменять данные аккаунта (имя, номер телефона, почту, пароль)	MUST	20	30
FR4	Система должна позволять пользователю полностью удалять аккаунт, принадлежащий данному пользователю	MUST	10	15
FR5	Система должна предоставлять возможность администраторам удалять аккаунт, содержащий неправомерную для страны регистрации аккаунта информацию	MUST	20	30
FR6	Система должна предоставлять возможность организаторам создавать мероприятия.	MUST	15	30
FR7	Система должна предоставлять возможность организаторам редактировать мероприятия.	MUST	20	40
FR8	Система должна предоставлять возможность организаторам и администраторам удалять мероприятия	MUST	10	20
FR9	Система должна предоставлять возможность организаторам создавать квизы для мероприятий	MUST	30	50

FR10	Система должна предоставлять возможность организаторам делать рассылку зарегистрированным на мероприятие пользователям	MUST	10	15
FR11	Система должна предоставлять возможность организаторам просматривать список зарегистрированных на конкретное мероприятие, принадлежащее организатору, пользователей	MUST	10	20
FR12	Система должна предоставлять возможность организаторам просматривать статистику пользователей, зарегистрировавшихся и (или) пришедших на мероприятие, таких как возраст, пол, зарегистрировался/пришёл, участвовал в квизах или нет	SHOULD	20	30
FR13	Система должна предоставлять возможность пользователям регистрироваться на мероприятия	MUST	20	30
FR14	Система должна предоставлять возможность пользователям оставлять отзывы на посещённые мероприятия	MUST	15	20
FR15	Система должна предоставлять доступ к квизам во время мероприятий и отображать результаты по их завершению.	MUST	30	40
FR16	Система должна предоставлять пользователю покупать билеты на мероприятия за виртуальную валюту?			
FR17	Система должна поддерживать отправку уведомлений пользователям (например, подтверждения регистрации на мероприятие, напоминания) по электронной почте с использованием встроенного Java Mail API.	MUST	10	20
FR18	Система должна предоставлять возможность отображения местоположения мероприятия с использованием Yandex Maps API. Организаторы должны указывать адрес, который отображается на карте при просмотре карточки мероприятия.	SHOULD	15	25

FR19	Система должна собирать и отправлять данные об активности пользователей на платформе (например, посещения страниц мероприятий, регистрации) в Yandex Metrica для анализа статистики.	SHOULD	12	20
FR20	Система должна генерировать уникальный QR-код для каждого билета с использованием Uniqode QR Code API. Пользователь получает QR-код для сканирования на мероприятии.	MUST	10	15
FR21	Система должна поддерживать авторизацию через социальные сети (Google, Facebook и др.) с использованием OAuth 2.0 для упрощённой регистрации и входа пользователей	COULD	20	30
FR22	Система должна поддерживать интеграцию с Typeform для проведения квизов, доступных пользователям во время мероприятий. Организаторы могут создавать опросы в Typeform и привязывать их к конкретным событиям	SHOULD	15	25
SEC1	Система должна разделять пользователей на клиента, организатора и неавторизованного пользователей.	MUST	5	20
SEC2	Система должна хранить пароли в зашифрованном виде	MUST	3	10
SEC3	Система должна использовать HTTPS для передачи данных	MUST	5	15

3.2 Требования к удобству использования

<i>Код</i>	<i>Требование</i>	<i>Приоритет</i>	<i>Мин. оценка, чел.ч</i>	<i>Макс. оценка, чел.ч</i>
U1	Время обучения обычного пользователя не должно превышать 30 минут для освоения основных функций системы.	MUST	8	16

U2	Опытный пользователь должен выполнять типовые задачи (создание мероприятия, оставление отзыва) за 5 минут.	SHOULD	4	8
----	--	--------	---	---

3.3 Требования к надежности

<i>Код</i>	<i>Требование</i>	<i>Приоритет</i>	<i>Мин. оценка, чел.ч</i>	<i>Макс. оценка, чел.ч</i>
R1	Доступность системы должна составлять не менее 99,5% в течение месяца.	MUST	12	24
R2	Среднее время восстановления работоспособности системы не должно превышать 1 часа.	SHOULD	4	8

3.4 Требования к производительности

<i>Код</i>	<i>Требование</i>	<i>Приоритет</i>	<i>Мин. оценка, чел.ч</i>	<i>Макс. оценка, чел.ч</i>
P1	Время отклика системы при выполнении основных функций не должно превышать 3 секунд.	MUST	12	24
P2	Система должна поддерживать работу не менее 50 одновременно работающих пользователей.	SHOULD	10	20

3.5 Ограничения разработки

- Серверная часть должна быть написана на Java (Spring Boot).
- Клиентская часть — на Angular.
- Система контроля версий — Git.

4. Интерфейсы

4.1 Пользовательские интерфейсы

- Интерфейс должен быть минималистичным, с понятным пользовательским опытом (UX), поддерживая несколько языков.

4.2 Аппаратные интерфейсы

- **Сервер:** Бэкэнд должен быть развёрнут на сервере VPS/VDS. Сервер должен иметь достаточные ресурсы для обработки запросов 50 пользователей одновременно, обеспечивая стабильную работу приложения. Фронтэнд должен быть развёрнут также на VPS/VDS.
- **Ограничения по дисплею:** Система должна поддерживать экраны с различным разрешением, в том числе адаптивно подстраиваться под мобильные устройства с разрешением от 360x640 пикселей до 1920x1080 и выше

4.3 Программные интерфейсы

- API для взаимодействия между frontend и backend
- Система должна поддерживать REST API для интеграции с внешними сервисами оплаты и отправки уведомлений.
- Приложение должно корректно работать в последних версиях браузеров Chrome, Safari, Яндекс.Браузер, поддерживая HTML5, CSS3 и JavaScript.

4.4 Сетевые интерфейсы

- Система должна поддерживать протокол HTTPS для защиты данных пользователей при передаче по сети.

5. Основные прецеденты

Прецедент: Создание аккаунта
ID: 1
Краткое описание: Неавторизованный пользователь регистрируется в системе.
Главный актёр: неавторизированный пользователь
Второстепенные: нет
Предусловия: Пользователь не авторизован в системе. Пользователь находится на главной странице.
Основной поток: <ol style="list-style-type: none"> 1. Пользователь открывает страницу регистрации. 2. Пользователь вводит имя, email и пароль. 3. Пользователь нажимает кнопку "Зарегистрироваться". 4. Система отправляет письмо с подтверждением на указанный email.

5. Пользователь переходит по ссылке в письме для подтверждения регистрации. 6. Система подтверждает успешную регистрацию и предоставляет доступ к системе.
Альтернативный поток: <ul style="list-style-type: none"> Если email уже зарегистрирован, система уведомляет пользователя об ошибке и предлагает авторизоваться.

Прецедент: Создание мероприятия
ID: 2
Краткое описание: Организатор создаёт новое мероприятие через систему.
Главный актёр: Организатор
Второстепенные: нет
Предусловия: Организатор авторизован в системе.
Основной поток: <ol style="list-style-type: none"> Организатор заходит в личный кабинет и выбирает "Создать новое мероприятие". Организатор вводит название, описание, дату, время и место проведения мероприятия. Организатор добавляет квиз для участников мероприятия (опционально). Организатор сохраняет мероприятие. Система создаёт карточку мероприятия и делает её доступной для регистрации.
Альтернативный поток: <ul style="list-style-type: none"> Если обязательные поля не заполнены, система уведомляет организатора об ошибке и требует заполнить недостающие данные.

Прецедент: Регистрация на мероприятие
ID: 3
Краткое описание: Пользователь регистрируется на выбранное мероприятие.
Главный актёр: Клиент (пользователь)
Второстепенные: нет
Предусловия: Пользователь авторизован в системе. Мероприятие доступно для регистрации.
Основной поток: <ol style="list-style-type: none"> Пользователь выбирает мероприятие из списка доступных. Пользователь нажимает на кнопку "Зарегистрироваться". Система запрашивает подтверждение регистрации. Пользователь подтверждает участие. Система отправляет пользователю email с подтверждением и билет.

Альтернативный поток:

- Если мероприятие уже заполнено, система уведомляет пользователя о невозможности регистрации.

Прецедент: Участие в квизе**ID:** 4**Краткое описание:** Участник мероприятия отвечает на вопросы квиза во время мероприятия.**Главный актёр:** Клиент (пользователь)**Второстепенные:** Организатор**Предусловия:** Пользователь зарегистрирован на мероприятие. Мероприятие активно. Квиз добавлен организатором.**Основной поток:**

1. Во время мероприятия пользователь получает уведомление о доступности квиза.
2. Пользователь открывает квиз в приложении.
3. Пользователь отвечает на вопросы квиза.
4. По завершении квиза система отображает результаты для пользователя.
5. Организатор видит обобщённую статистику результатов квиза.

Альтернативный поток:

- Если пользователь не успевает завершить квиз до окончания времени, система закрывает доступ и фиксирует результат по завершённым вопросам.

Прецедент: Оставить отзыв о мероприятии**ID:** 5**Краткое описание:** Пользователь оставляет отзыв после участия в мероприятии.**Главный актёр:** Клиент (пользователь)**Второстепенные:** Организатор**Предусловия:** Мероприятие завершено. Пользователь был зарегистрирован на мероприятие.**Основной поток:**

1. Пользователь получает уведомление о возможности оставить отзыв после завершения мероприятия.
2. Пользователь переходит в форму оставления отзыва.
3. Пользователь заполняет поля с оценкой мероприятия и текстовым комментарием.
4. Пользователь отправляет отзыв.
5. Система сохраняет отзыв и уведомляет организатора.

Альтернативный поток:

- Если пользователь не заполнил обязательные поля, система уведомляет его об ошибке и просит заполнить все поля.

Прецедент: Просмотр списка участников**ID:** 6**Краткое описание:** Пользователь оставляет отзыв после участия в мероприятии.**Главный актёр:** Организатор**Второстепенные:** Клиент (пользователь)**Предусловия:** Организатор авторизован и имеет созданное мероприятие с зарегистрированными участниками.**Основной поток:**

1. Организатор заходит на страницу мероприятия.
2. Организатор открывает список участников мероприятия.
3. Система отображает список зарегистрированных участников с их именами и контактной информацией.

6. Требования к лицензированию

Продукт должен распространяться по лицензии MIT.