

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

**Факультет информатики, математики и компьютерных наук
Образовательная программа «Программная инженерия»**

СОГЛАСОВАНО

Руководитель, к.ф.-м.н., доцент
кафедры информационных систем и
технологий факультета информатики,
математики и компьютерных наук

_____ Н.В. Асеева

«___» _____ 2024 г.

УТВЕРЖДАЮ

Академический руководитель
образовательной программы
«Программная инженерия»

_____ Н.В. Асеева

«___» _____ 2024 г.

Выпускная квалификационная работа – БАКАЛАВРСКАЯ РАБОТА
на тему «Разработка информационной системы для регистрации на мероприятия
НИУ ВШЭ - Нижний Новгород»

по направлению подготовки 09.03.04 «Программная инженерия»

Исполнитель
студент группы 20ПИ-1

_____ Н.Д. Титова

«___» _____ 2024 г.

Нижний Новгород, 2024

Аннотация

Данная выпускная квалификационная работа посвящена проекту, целью которого стало создание системы регистрации для мероприятий нижегородского кампуса НИУ ВШЭ. Актуальность проекта обусловлена отсутствием подобного приложения в рамках университета и отсутствием необходимого организаторам мероприятий кампуса функционала в сторонних сервисах. Приложение предоставляет участникам мероприятий возможность регистрироваться на мероприятия, настраивать свое расписание и скачивать QR-коды для входа на мероприятия и сертификаты о посещении мероприятий, а также упрощает работу организаторов по сбору данных участников, взаимодействию с ними и отслеживанию посещений. Это групповой проект. Среди задач автора данной работы, выполнение которых описано в представленной пояснительной записке, были сбор и анализ требований к системе, проектирование системы, frontend-разработка и тестирование системы. В данный момент система полностью реализована, ссылки на исходный код и работающее веб-приложение системы регистрации приводятся в Приложениях к работе.

Ключевые слова: система регистрации, организация мероприятий, веб-приложение, frontend.

Оглавление

Введение	
1. Проектирование системы	
1.1. Сбор требований.....	
1.2. Первичный анализ требований	
1.3. Выбор средств, методов и технологий разработки.....	
1.4. Проектирование пользовательского взаимодействия.....	
1.5. Написание спецификации.....	
1.6. Разработка дизайн-системы.....	
1.7. Проектирование интерфейсов и прототипов экранов	
2. Frontend-разработка	
2.1. Стили приложения	
2.2. Страницы приложения.....	
3. Тестирование системы	
Заключение	
Источники.....	
Приложение 1. Исходный код приложения.....	
Приложение 2. Разработанное приложение.....	
Приложение 3. Дизайн-система приложения и прототипы страниц	
Приложение 4. User Story	
Приложение 5. User Flow.....	
Приложение 6. Скриншоты некоторых экранов приложения	
Приложение 7. Матрица трассируемости	
Приложение 8. Отчет о финальном тестировании системы	
Приложение 9. Протоколы бета-тестирования	
Приложение 10. Спецификация приложения	

Введение

В последние десятилетия все больше и больше различных сфер жизни человека затрагивает цифровизация. Компьютеризированные решения активно используются даже теми людьми, чья профессиональная деятельность не связана напрямую со сферой информационных технологий. Любая работа, связанная со сбором и хранением информации, ее обработкой, анализом и передачей, существенно упрощается и многократно ускоряется, если подключить к ее выполнению информационные технологии. Компьютеризированные решения позволяют рационализировать различные процессы и затрачиваемые на них время и ресурсы как для пользователей-клиентов, желающих получить какую-либо услугу, так и для пользователей, эту услугу предоставляющих. И первые, и вторые благодаря информатизации избавляются от необходимости личного присутствия, от привязки к месту и времени и от необходимости ручной обработки данных [1]. Вследствие наличия этих преимуществ цифровые решения актуальны и в сфере менеджмента и организации мероприятий. Многие учреждения поручают сотрудникам работу по регистрации участников мероприятий, систематизации их данных, подсчету участников, и фиксации их посещения – работу, которую возможно автоматизировать. В настоящий момент все больше и больше организаций используют электронные ресурсы для регистрации на мероприятия.

Некоторые исследования [2] также подтверждают, что большинство людей, принадлежащих к младшим возрастным группам (в особенности люди до 20 лет) воспринимают цифровизацию различных процессов позитивно, в то время как люди старше 36 лет куда менее лояльны по отношению к ней. Из этого можно сделать вывод о том, что автоматизация процессов регистрации на мероприятия, ориентированные на аудиторию среднего и старшего школьного возраста, а также на студентов, востребована, и с большой вероятностью будет положительно принята пользователями полученной системы, чем автоматизация процессов, целевая аудитория которых существенно старше.

Нижегородский кампус Высшей школы экономики, как организация, регулярно проводящая мероприятия для школьников, также сталкивался с различными проблемами, обусловленными отсутствием автоматизированной системы регистрации. Существуют сторонние системы, которые можно использовать для регистрации, однако часто они не имеют всего необходимого функционала или требуют достаточно большой платы за использование, что невыгодно для организаторов некоммерческих мероприятий. Таким образом возникла идея о создании собственной регистрационной системы НИУ ВШЭ – Нижний Новгород. Наличие собственной системы позволило бы включить в нее все необходимые функции и полностью контролировать ее наполнение и использование.

Проект, выполнявшийся в рамках данной выпускной квалификационной работы, посвящен проектированию и разработке этой регистрационной системы. Цель работы – разработать веб-приложение, которое может использоваться как организаторами мероприятий в регулировании логистики, так и участниками мероприятий для регистрации. Работа над проектом была начата в рамках курсовой работы третьего курса [16][17], тогда же был реализован базовый прототип системы, включавший только функционал регистрации на мероприятия, без личного кабинета и других нужных функций. На основе этого прототипа велась дальнейшая работа.

В проектной группе участвовала также Лапшина Юлия Алексеевна. Все обязанности в рамках проекта были четко разделены: Юлия занималась исследованием рынка, backend-частью приложения и его деплоем, а также написанием основной части технической документации, автор данной работы – написанием спецификации, прототипированием интерфейсов, frontend-разработкой и тестированием.

В данной работе будет рассмотрено выполнение следующих задач, поставленных в рамках проекта:

- выявить требования к системе регистрации
- проанализировать полученные требования
- подобрать наиболее подходящие для разработки технологии
- составить спецификацию проекта
- разработать систему стилей для приложения
- спроектировать интерфейсы приложения и логику его взаимодействия с пользователем
- отрисовать макеты экранов приложения
- реализовать frontend-часть приложения на основе разработанных прототипов
- написать пользовательскую документацию
- провести тестирование приложения
- провести анализ проделанной работы.

В разделах ниже будет подробно поэтапно описано проектирование и создание системы.

1. Проектирование системы

1.1. Сбор требований

Для выявления требований к программному обеспечению, как правило, выделяются [3] следующие методы:

- интервьюирование
- проведение семинаров
- работа с фокус-группой
- наблюдение за рабочим процессом пользователей
- создание опросных листов
- анализ системных интерфейсов для уже существующих систем
- анализ пользовательских интерфейсов для уже существующих систем
- анализ документации для уже существующих систем

Карл Виггерт и Джой Битти в книге «Разработка требований к программному обеспечению» выделяют [3] для сбора требований при разработке полностью новых приложений такие методы, как интервьюирование, проведение семинаров и анализ системных интерфейсов, в качестве наиболее подходящих. С опорой на эту рекомендацию и в результате анализа возможностей проектной группы, как ключевой метод сбора требований было выбрано интервьюирование заказчика.

Поскольку заказчиком проекта выступала администрация факультета информатики, математики и компьютерных наук НИУ ВШЭ – Нижний Новгород, а сам проект ориентирован в том числе на организаторов различных университетских мероприятий, для интервьюирования было проведено несколько встреч с деканом факультета Асеевой Натальей Владимировной, начальником отдела по организации и сопровождению мероприятий Калинкиной Анной Сергеевной и начальником отдела по связям с общественностью и маркетингу Акимовой Анной Александровной.

На встречах был выявлен основной необходимый организаторам мероприятий нижегородского кампуса НИУ ВШЭ функционал системы регистрации, а также были обозначены примерные сроки разработки и ожидания от формы системы.

1.2. Первичный анализ требований

После проведения первых интервью были начаты выявление и анализ требований к системе. Сбор требований на интервью позволил выявить основные ожидания заказчиков от системы. В первую очередь была выделена цель разработки: создание единой системы для регистрации на мероприятия нижегородского кампуса НИУ ВШЭ, доступной людям без специфических знаний в сфере IT.

Следующим шагом было выделение целевой аудитории системы. Проект ориентирован на две группы пользователей: участников мероприятий и организаторов.

Участники - это пользователи, которые используют систему для регистрации. В большинстве своем это люди среднего и старшего школьного возраста, заинтересованные в посещении вузовских мероприятий для расширения кругозора,

развития различных навыков или для получения информации о поступлении.

Организаторы – это пользователи, которые используют систему для управления событиями и записями на них. Как правило, это сотрудники учебных офисов или центра довузовской подготовки, или же студенты-волонтеры.

Для каждой из пользовательских групп нужно было разработать отдельный интерфейс.

Участники должны иметь доступ к личному кабинету, который позволяет регистрироваться на мероприятия, настраивать и просматривать свое расписание, проходить на мероприятия и скачивать сертификаты о посещении.

Организаторы должны иметь возможность создавать, изменять и удалять события, настраивать их расписания и лейблы, а также просматривать и редактировать записи на события.

Кроме того, в системе должны существовать суперпользователи (администраторы системы), которым доступен весь функционал организаторов и у которых есть возможность настраивать права доступа всех пользователей системы.

Во время интервью также была обозначена ожидаемая форма системы – веб-приложение.

Поскольку было решено разрабатывать систему в формате веб-приложения, необходимо было также выделить браузеры, в которых она точно должна работать. Так как приложение ориентировано преимущественно на жителей России, с помощью сервиса Яндекс.Радар от компании Яндекс были выявлены самые популярные в этой стране браузеры. По метрикам сервиса, около 40% людей в России используют в качестве основного браузера Google Chrome, еще 30% – Яндекс Браузер. При работе над системой регистрации было принято решение ориентироваться на функционирование системы именно в этих двух браузерах.

На основании пожеланий заказчика, высказанных во время интервью, и базовых требований к системе были выбраны средства и технологии разработки.

1.3. Выбор средств, методов и технологий разработки

Для написания спецификации был выбран стандарт ISO/IEC/IEEE 29148:2011 [5] ввиду его широкой распространенности при написании требований к программному обеспечению и предусматриваемой им глубокой детализации различных аспектов программы.

Для разработки дизайн-системы, составления различных схем, проектирования пользовательского взаимодействия и интерфейсов системы использовалась Figma. Сравнительный анализ инструментов для прототипирования [6] показывает, что Figma предлагает наилучший баланс между функциональностью, удобством использования и производительностью, к тому же среди наиболее распространенного программного обеспечения такого рода это единственный продукт, предоставляющий возможность работы как в браузере, так и в виде десктопного приложения для любой операционной системы, что очень важно, учитывая, что с проектом будут работать разные люди, в том числе не связанные с дизайном и прототипированием – им не придется устанавливать ненужных программ для ознакомления с ходом и результатами работ.

Разработка самого веб-приложения велась на языке Python с использованием фреймворка Django [7]. Язык Python универсален и адаптивен, к тому же существует обширная база документации и библиотек для Python: он часто используется для

абсолютно различных целей. Для верстки интерфейсов приложения применялись язык разметки HTML в сочетании с CSS. Они были выбраны по причине их крайне широкого распространения и минимального количества популярных аналогов с тем же функционалом.

В качестве системы контроля версий использовался git, а в качестве онлайн-хранилища репозитория проекта и для отслеживания текущих задач – GitHub.

1.4. Проектирование пользовательского взаимодействия

Перед началом детальной проработки требований к системе на основании информации о нашей целевой аудитории были прописаны пользовательские истории, что позволило прояснить намерения пользователей при использовании системой и выделить необходимый им функционал. Были написаны пользовательские истории участников и организаторов мероприятий. С ними можно ознакомиться в Приложении 4.

На основе пользовательских историй было спроектировано взаимодействие пользователей с системой. Поскольку интерфейс организаторов было решено сделать максимально простым и лаконичным – взаимодействие с различными сущностями баз данных, их создание, редактирование и удаление – при проектировании взаимодействия основной фокус внимания был направлен на интерфейс участника. Были созданы схемы пользовательского пути (user flow) для различных аспектов взаимодействия пользователя с системой. Для примера здесь приведен user flow процесса входа в личный кабинет системы. Также были реализованы user flow процессов работы с регистрациями на мероприятия и страницей профиля пользователя, со всеми user flow в более крупном масштабе можно ознакомиться в Приложении 5.

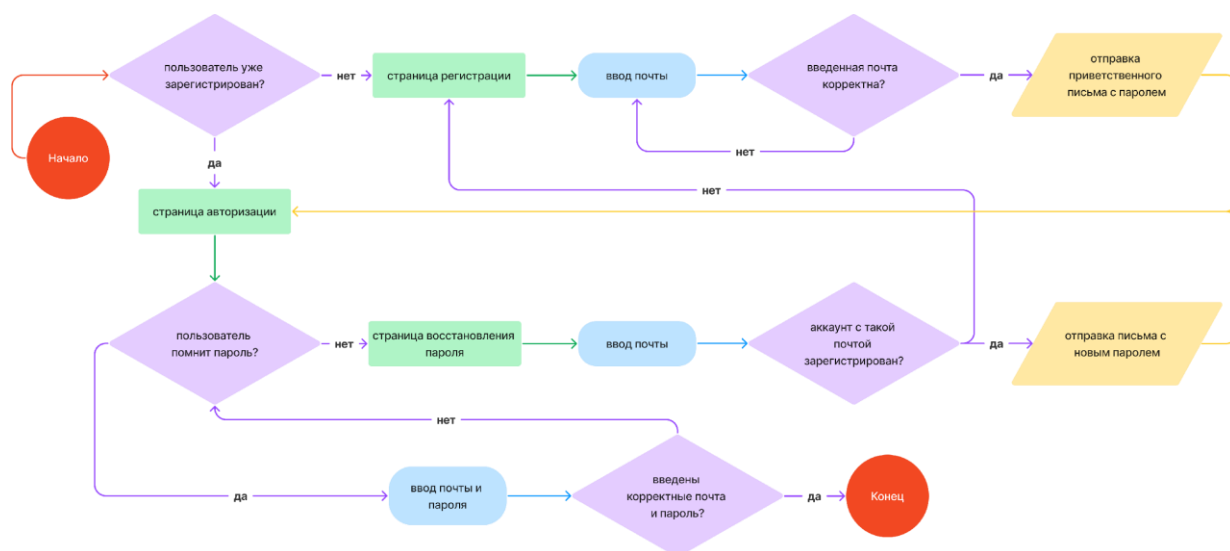


Рис.1 User Flow процесса регистрации

Также были выделены основные страницы системы и спроектирована логика их взаимодействия друг с другом. Для наглядности здесь приводится схема навигации пользователя по этим страницам.

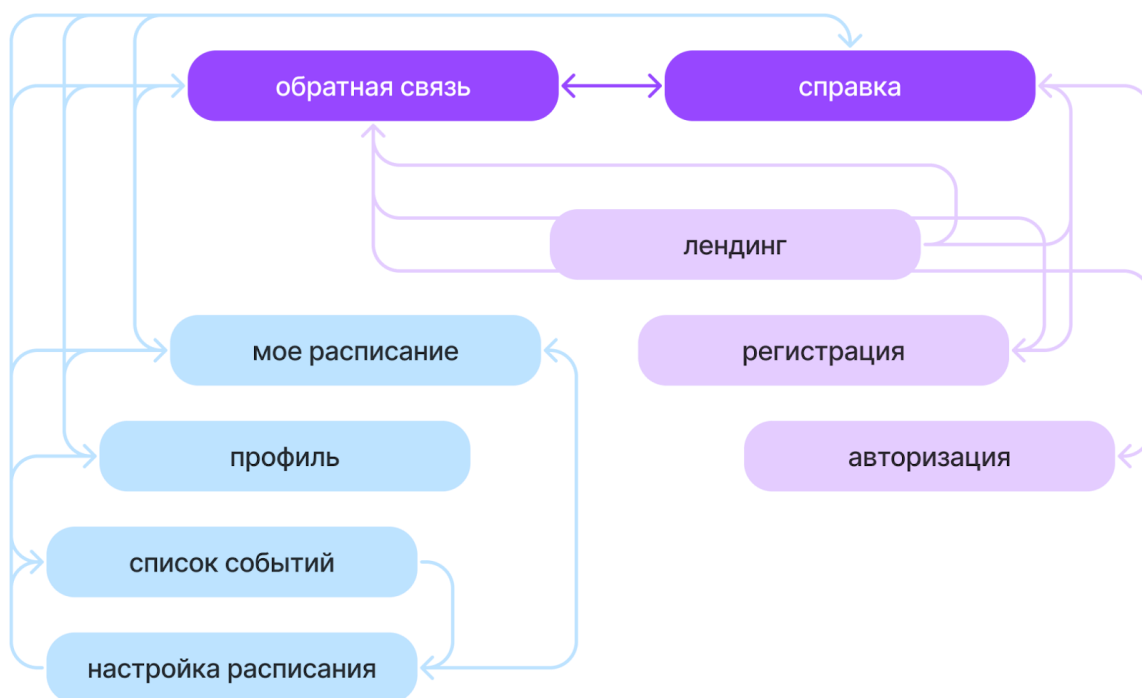


Рис.2 Схема навигации пользователя по страницам системы

На этом этапе работы над проектом также был проведен анализ рынка подобных приложений, что позволило выделить необходимый приложению функционал, перенять положительный опыт и убедиться в необходимости создания приложения. Подробнее с процессом проведенного анализа рынка можно ознакомиться в работе Юлии Лапшиной [18] или в документах, хранящихся в GitHub-репозитории проекта (см. Приложение 1).

1.5. Написание спецификации

Как было сказано ранее, написание спецификации строилось на стандарте ISO/IEC/IEEE 29148:2011. Впрочем, его использование в проекте носило скорее рекомендательный характер. Итоговая структура полученной спецификации следующая:

1. Введение

- 1.1. Назначение
- 1.2. Предполагаемая аудитория и рекомендации к прочтению
- 1.3. Ссылки

2. Общее описание

- 2.1. Классы и характеристики пользователей
- 2.2. Функциональность продукта
- 2.3. Среда функционирования продукта
- 2.4. Ограничения при проектировании и разработке
- 2.5. Пользовательская документация

3. Функциональность системы

- 3.1. Функционал участников
- 3.2. Функционал администраторов

4. Требования к внешним интерфейсам

- 4.1. Пользовательские интерфейсы
- 4.2. Программные интерфейсы
- 4.3. Интерфейсы оборудования
- 4.4. Интерфейсы связи

5. Нефункциональные требования

6. Глоссарий

Эта спецификация предназначена в основном для разработчиков, которые вели работу над проектом. В ней описаны основные требования к системе.

В спецификацию была внесена вся информация о назначении продукта, целевой аудитории и необходимом функционале, приведенная в предшествующих разделах.

На основании спроектированного пользовательского взаимодействия были написаны функциональные и нефункциональные требования к системе.

Чтобы проиллюстрировать полученный формат требований, здесь приводятся функциональные требования к процессу настройки расписания.

3.1.9 Настройка расписания

FR-9.1. Система должна предоставлять зарегистрированным пользователям возможность выбирать для посещения некоторые из мероприятий, идущих параллельно внутри одного события. Доступ осуществляется при нажатии кнопки регистрации на странице списка доступных для регистрации событий (FR-8.3) или через кнопку изменения расписания на странице Мое расписание (FR-7.3.4).

FR-9.2. На странице Настройки расписания пользователь видит название события и список доступных для регистрации мероприятий внутри события. Для каждого мероприятия указана информация о нем, описанная в пункте FR-7.3.1.

FR-9.2.1. Мероприятия на странице настройки расписания сгруппированы по дням и времени проведения.

FR-9.2.2. Пользователь должен выбрать для посещения ровно одно мероприятие из каждой временной категории.

FR-9.3. После настройки расписания пользователь должен завершить регистрацию и перейти страницу Мое расписание.

Кроме того, рамках данной работы кажется нужным прояснить значения некоторых терминов, которые широко используются как в спецификации, так и в описании работы системы. Их пояснения взяты из Глоссария спецификации.

Событие - масштабное мероприятие, включающее в себя различные активности (чаще всего указаны как мероприятия)

Мероприятие - активность, относящаяся к конкретному событию; в рамках данного документа мероприятиями иногда называются и события вообще.

Пример: мастер-класс по программированию на зимней школе ИМиКН; здесь зимняя школа - событие, мастер-класс - мероприятие.

Тег - категория, к которой относится событие. Есть два типа лейблов: теги аудитории и теги тематики.

Пример: теги аудитории - студенты, школьники; теги тематики: компьютерные науки, лингвистика, дизайн.

Тег события - связующее звено между тегом и событием, связывает многие ко

многим.

Запись - связующее звено между мероприятием и участником, пометка о записи участника на мероприятие со статусом посещения.

Участник - пользователь, регистрирующийся на мероприятия.

Администраторы и организаторы - пользователи, имеющие доступ к административной части сайта, в частности - к настройке и редактированию событий, мероприятий и записей.

Со всей спецификацией проекта можно ознакомиться в Приложении 10.

1.6. Разработка дизайн-системы

Следующим этапом проектирования системы стало проектирование пользовательского интерфейса.

За основу дизайн-системы был взят актуальный брендбук НИУ ВШЭ [8]. Данный брендбук был разработан Школой дизайна ВШЭ в 2021 году и включил в себя различные элементы, а также цветовые и стилевые решения, которые рекомендовано использовать в различных электронных ресурсах, связанных с НИУ ВШЭ, и в печатной продукции университета.

Одно из центральных нововведений данного брендбука – новые университетские шрифты HSE Sans и HSE Slab. HSE Slab – шрифт-антиква, то есть шрифт с засечками, который хорошо подходит для печатной продукции, однако может осложнить восприятие текста при использовании на электронных ресурсах. Для целей проекта больше подошел гротескный шрифт HSE Sans: он минималистичен и легок для чтения. Все текстовые стили системы были реализованы с использованием данного шрифта.



header black
Header Semibold
label
Regular
Caption

Рис.3 Текстовые стили проекта

Брендбук ВШЭ также предлагает минималистичную цветовую палитру, которая отталкивается от нового основного цвета – насыщенного темно-синего (#374b9b). На момент обращения к брендбуку (2023 год) помимо двух оттенков синего и трех – серого цветов он предоставлял только несколько палитр из трех цветов для отдельных направлений и подразделений, поэтому было принято решение разработать для проекта свое цветовое решение, взяв из брендбука только основной цвет и построив палитру вокруг него. К моменту написания данной работы брендбук был обновлен, теперь он предлагает единую и более широкую палитру различных

оттенков без деления на направления.

При разработке цветовой палитры в первую очередь было необходимо выделить оттенки красного, зеленого и желтого, сочетающиеся с основным цветом. Оттенки этих цветов нужны в системе для сообщений: красные – для сообщений об ошибках, желтые – для предупреждений, зеленые – для сообщений об успешном выполнении каких-либо действий. Для подбора нужных оттенков использовалась платформа Colors [9], позволяющая сгенерировать огромное количество палитр на основе заданных цветов. После выбора основных цветов с помощью сервиса по составлению палитр от Material Design [10] на основе выбранных оттенков синего, красного, зеленого, желтого и серого была построена палитра из десяти оттенков каждого цвета.

Ниже приведены полученные цветовые стили проекта: палитры серого (Base), синего (Primary), зеленого (Success), желтого (Warning) и красного (Error) цветов.



Рис.4 Цветовые стили проекта

Еще один элемент, который был взят из брендбука Высшей школы экономики – изображение маскота университета, вороны. Разработчики брендбука рекомендуют [11] использовать маскот в случаях, когда нужна символика ВШЭ с нейтральным посылом и нет возможности сделать свою иллюстрацию. Изображения ворон – единственные иллюстрации, используемые системой, это способствует ее лаконичности и единообразию. Изображения вороны использовались, в частности, в качестве основной иконки платформы (далее - favicon / фавиконка) и на страницах лендинга и ошибок.

В процессе разработки дизайн-системы для приложения было необходимо, помимо прочего, создать стилизованные варианты нужных нам элементов. Все эти компоненты выполнялись с использованием подобранных ранее цветовых и текстовых стилей, ниже представлены некоторые из разработанных элементов: кнопки множественного выбора (далее – checkbox / чекбокс) и единичного выбора (далее – radiobutton/ радиобаттон), системные сообщения, кнопки и инпуты.



Рис.5 Стилизованные элементы системы

1.7. Проектирование интерфейсов и прототипов экранов

После создания UI-кита системы необходимо было создать прототипы интерфейсов системы, чтобы отталкиваться от них при frontend-разработке. Были созданы прототипы 16 страниц системы (интерфейса участника). Это следующие экраны:

- экраны ошибок 400, 403, 404 и 500
- экран лендинга
- экраны обратной связи для авторизованных и неавторизованных пользователей
- экраны справки для авторизованных и неавторизованных пользователей
- экраны регистрации, авторизации и восстановления пароля
- экран профиля пользователя
- экраны списка событий, настройки расписания и просмотра расписания.

Все эти страницы были спроектированы и отрисованы в Figma, ссылку на документ с прототипами можно найти в Приложении 3.

Прототипирование экранов организаторского интерфейса не проводилось, поскольку было принято решение создавать интерфейс организаторов на базе панели администрирования сайтов Django.

2. Frontend-разработка

По завершении написания спецификации системы и ее прототипирования стала возможна реализация страниц системы в рамках frontend-разработки. Как было сказано выше, технологии, выбранные здесь для разработки – Python, HTML, CSS.

Ссылка на исходный код проекта находится в Приложении 1. Наиболее важная часть кода с точки зрения frontend-разработки хранится в директориях *regsys/templates/* и *regsys/static/regsys/*. Также важен файл *views.py* директории *regsys/*.

2.1. Стили приложения

В директории *regsys/static/regsys/* хранятся стили проекта, а также некоторые важные pdf-шаблоны. Стили проекта настраиваются преимущественно в файлах *fonts.css*, *colors.css* и *elements.css*. Для их систематизации и удобного подключения к проекту создан файл *style.css* – именно на него ссылаются все HTML-файлы.

В файле *fonts.css* подключаются шрифты, хранящиеся в директории *regsys/static/regsys/fonts*, и настраиваются текстовые стили проекта. В директории шрифтов хранятся начертания HSE Sans (Black, Bold, SemiBold, Regular, Italic, Thin) в форматах *.eot*, *.otf*, *.woff* и *.woff2*. Такое многообразие форматов необходимо, чтобы шрифты поддерживались в большем количестве браузеров, в том числе в не слишком новых их версиях. По умолчанию (если, конечно, шрифт уже не установлен локально) система пытается использовать формат *.woff2*: он имеет наилучшее среди представленных форматов сжатие и поддерживается большинством современных браузеров. Однако некоторые браузеры, например, Internet Explorer, этот формат не обрабатывают [12]. Именно для таких случаев при невозможности подключения шрифта в формате *.woff* система последовательно проверяет, подойдут ли для браузера следующие форматы: *.woff*, *.eot*, *.otf* (отсортированы по степени сжатия). Система подключает большее количество начертаний, нежели используют текстовые стили: это делается ввиду нескольких модификаций дизайн системы, имевших место во время работы над проектом. Поскольку текстовые стили менялись и, вероятно, будут продолжать меняться, было решено сохранить самые популярные начертания для простоты адаптации системы к новым их новым вариациям. Таким образом, подключаются следующие начертания HSE Sans: Black, Bold, Semibold, Regular, Italic, Thin.

После подключения шрифтов (выполняется с помощью правила *@font-face*) выполняется настройка текстовых стилей. Эти стили:

- *text-header-black* (HSE Sans, normal, 900, 32px, lowercase) - для центральных (основных) заголовков страниц
- *text-header-semibold* (HSE Sans, normal, 600, 22px) - для подзаголовков и неосновных заголовков
- *text-label* (HSE Sans, normal, 600, 18px, lowercase) - для кнопок, ссылок панели навигации (далее – navbar / навбар) и выделений
- *text-regular* (HSE Sans, normal, 400, 18px) - для обычного текста
- *text-caption* (HSE Sans, normal, 400, 16px) - для подсказок, аннотаций, второстепенных текстов

Поскольку даже при использовании для загрузки четырех различных форматов шрифта могут возникнуть непредвиденные сложности и затруднения, при возникновении ошибок при подгрузке HSE Sans в текстовых стилях используется шрифт Roboto.

В файле *colors.css* формируются цветовые стили приложения. Это стили *col-base-900 – col-base-50* для палитры серых оттенков (используются в текстах и в качестве фона), *col-primary-900 – col-primary-50* (используются в качестве цветовых акцентов: в заголовках, навбаре, кнопках и т.д.), *col-success-900 – col-success-50* (используются для сообщений об успешном выполнении пользователем каких-либо действий), *col-warning-900 – col-warning-50* (для сообщений с предупреждением) и *col-error-900 – col-error-50* (для сообщений об ошибках и в качестве акцентного красного).

Стилизация всех основных элементов происходит в файле *elements.css*. Здесь настраиваются общие стили страниц (выравнивание их содержимого, отступы, промежутки и т.п.). Здесь же настраиваются навбар системы, поле фильтрации, формат и цвета системных сообщений.

Далее стилизуются такие системные элементы, как кнопки и поля ввода. Их стилизация достаточно простая – настраивается внешний вид (цветовые решения, отступы, отсутствие контура) для дефолтного состояния, активного состояния (и схожие стили при нажатии или наведении курсора), заблокированного состояния элемента.

Некоторую сложность в процессе разработки представляла стилизация полей загрузки файла. Авторы многих статей (например, [13]) утверждают, что попытки изменить внешний вид базовой формы загрузки файла не ведут ни к какому приемлемому результату, и предлагают решать проблему следующим путем: скрыть целиком родительский элемент и создать новое поле ввода с кастомным дизайном, не включая в него ничего ненужного. Однако другие разработчики [14] высказываются более оптимистично и находят несложное решение проблемы. В результате удалось вполне успешно кастомизировать данные поля.

Затем стилизуются такие компоненты системы, как чекбоксы и радиобаттоны. Их кастомизация проста: родительский компонент скрывается, на его место подгружаются стилизованные изображения компонента в различных состояниях (*empty*, *empty & hover*, *filled*, *filled & hover*). Эти изображения были созданы в Figma во время прототипирования и выгружены оттуда в формате *.png*, система подтягивает их из директории *regsys/static/regsys/images/*.

В оставшейся части кода файла настраиваются таблицы и *fieldset*, нужные для корректного отображения расписаний, и подсказки (далее – *tooltips / тултипы*) – для вывода аннотаций событий и советов.

Также, как упоминалось выше, в статической папке системы хранятся pdf-шаблоны и некоторые изображения. Здесь находятся шаблон сертификата о посещении, соглашения на обработку персональных данных и на создание фотографий с участником, а также документ по пользованию системой для организаторов. В директории с изображениями (*regsys/static/regsys/images/*) располагаются, помимо изображений стилизованных чекбоксов и радиобаттонов, изображения маскота ВШЭ - фавиконка и изображения для страниц лендинга и ошибок.

2.2. Страницы приложения

В разделе “Проектирование интерфейсов и прототипов экранов” главы, посвященной проектированию системы, приводится список всех страниц интерфейса участника, над которыми велась работа в процессе разработки. В данном разделе будет подробнее описана структура данных страниц с упоминаниями их специфики с

точки зрения разработки.

Страницы пользовательского интерфейса хранятся в директории *regsys/templates/regsys*. Первая страница, с которой сталкивается пользователь во время взаимодействия с приложением – страница лендинга (*landing.html*). Переход на эту страницу осуществляется с внешних ресурсов, например, по ссылке из анонса мероприятия в соцсетях. На этой странице – описание системы и призыв начать ее использовать. Вверху страницы – навбар, общий для всех страниц, доступных неавторизованному пользователю, на нем находятся ссылки на страницы регистрации, авторизации, справку и форму обратной связи. Страница справки для неавторизованных пользователей (*help_anon.html*) содержит инструкции по регистрации и входу в аккаунт и справочную информацию о системе. Через форму обратной связи (*feedback_anon.html*) можно отправить вопросы и комментарии по работе системы, указав предварительно свою почту, если пользователь хочет получить ответ. К обращению в форме обратной связи можно прикрепить файл - изображение или pdf-документ.

Логика процессов регистрации, авторизации и восстановления пароля, указана в схеме user flow из Приложения 5. Страница регистрации (*signup.html*) содержит форму ввода электронной почты. На эту почту направляется автоматически генерируемый пароль, с которым пользователь сможет войти в систему (при необходимости после авторизации пароль можно изменить на более легкий в запоминании). Перед регистрацией в системе пользователь также должен подтвердить (обязательные чекбоксы), что он согласен на обработку персональных данных и на то, что на мероприятиях его могут фотографировать. На странице доступно скачивание документов с обоими соглашениями. На странице авторизации (*signin.html*) пользователь вводит электронную почту и свой пароль, если они корректны – он переходит на страницу своего расписания. Также со страницы авторизации можно перейти на страницу восстановления пароля (*forgot.html*). Процесс восстановления пароля аналогичен процессу регистрации: пользователь вводит адрес электронной почты, и если пользователь с такой почтой действительно был зарегистрирован, на почту направляется новый сгенерированный пароль.

После авторизации пользователь оказывается на странице своего расписания (*mylist.html*). На странице - расписание грядущих событий, на которые пользователь был зарегистрирован, с кнопками скачивания QR-кодов (автоматически генерирующиеся QR-коды, с помощью которых отслеживается посещение участником каждого мероприятия: QR-код сканирует организатор при входе, после сканирования кода организатором отметка о посещении у участника сменяется на “Посещено”; только посещенные мероприятия включаются в сертификат о посещении события) и скачивания, изменения или удаления расписания. При скачивании расписания происходит загрузка на устройство пользователя pdf-документа, в который записывается актуальное расписание пользователя. При нажатии на кнопку изменения расписания пользователь переходит на страницу настройки расписания, о которой будет рассказано ниже. Ниже расписания предстоящих событий находится список событий, которые пользователь уже посетил. Пользователь может скачать сертификат о посещении любого из событий: автоматически генерируемый (на основе шаблона) сертификат включает информацию о событии, имя участника и список посещенных им мероприятий.

На всех страницах, доступных пользователю-участнику после авторизации, также

находится навбар, включающий ссылки для перехода к страницам профиля пользователя, его расписания, регистрации на события, справки и формы обратной связи. Также в навбаре находится кнопка выхода из аккаунта. Форма обратной связи авторизованного пользователя (*feedback.html*) аналогична форме для неавторизованного, однако не включает в себя поле ввода электронной почты: в системе уже сохранена почта, через которую с пользователем можно связаться. Страница справки (*help.html*) также устроена аналогично справке для неавторизованного пользователя, однако содержит не ответы на вопросы о процессе авторизации, а ответы на вопросы о процессе регистрации на мероприятия, их посещении и другом функционале для участников мероприятий.

С процессами регистрации на мероприятия, их посещения и работы с расписанием можно ознакомиться в Приложении 5. На странице регистрации на мероприятия (*register.html*) приводится список предстоящих событий, на которые открыта регистрация. Их можно сортировать с помощью находящихся в левой части страницы чекбоксов с фильтрами. Существуют фильтры двух типов: аудитория и категория. Они соответствуют существующим в системе лейблам. Рядом с каждым из доступных для регистрации событий есть кнопка регистрации, с помощью которой осуществляется переход к странице настройки расписания (*timetable.html*). На эту страницу также можно перейти по кнопке изменения расписания на странице пользовательского расписания. На этой странице - список всех мероприятий, проходящих в рамках события - лекций, мастерклассов и тому подобного. Они отсортированы по времени проведения: в каждой временной категории можно выбрать ровно одно мероприятие (выбор реализован с использованием радиобаттонов). После того, как участник выбирает мероприятие из каждой временной категории, по нажатию кнопки завершения регистрации он переходит к странице своего расписания, описанной выше.

Также авторизованному пользователю доступна страница профиля (*profile.html*). На этой странице пользователь может изменить свой пароль (для этого необходимо указать старый пароль и ввести новый пароль, соответствующий требованиям: только латиница и цифры, как минимум одна заглавная буква, как минимум одна цифра; после смены пароля осуществляется выход из аккаунта и переход к странице авторизации), а также ввести или изменить свои личные данные. Система хранит следующую личную информацию пользователя: ФИО, место обучения, контактный телефон и telegram. Эта информация используется для связи и для генерации сертификата.

HTML-файлы всех страниц системы организованы следующим образом. В начале подключаются CSS-файлы (подключается файл *style.css*, ссылающийся на все прочие). Затем задается заголовок страницы, подключается фавиконка. Устанавливается задний фон страницы (через CSS-класс, цвет - Base 50). Далее при наличии системных сообщений происходит их отображение. Настраивается навбар – ссылки связываются с CSS-элементом, делятся на выводимые в левой и правой части экрана. Оставшаяся часть страницы вариативна. Здесь могут использоваться поля ввода, кнопки, таблицы, чекбоксы и радиобаттоны, и так далее.

В рамках frontend-разработки также были кастомизированы экраны ошибок для ошибок 400, 403, 404 и 500. На страницах этих ошибок можно увидеть краткую информацию об ошибке и инструкцию для дальнейших действий. Также на экране ошибки есть кнопка перехода к форме обратной связи (чтобы пользователь мог

рассказать о проблеме или задать вопрос) и кнопка возвращения к системе.

Скриншоты некоторых из страниц системы приводятся в Приложении 6. Увидеть все страницы системы можно в самой системе регистрации, ссылка на которую находится в Приложении 2.

3. Тестирование системы

На всех этапах работы над проектом отдельное внимание уделялось тестированию системы. В начале работы над проектом проводилось тестирование документации. Основная работа здесь проводилась со спецификацией приложения, в особенности – с функциональными требованиями. Оценивались такие характеристики написанных требований, как последовательность и непротиворечивость, атомарность, корректность и возможность тестирования. Проведенная работа не позволила написать абсолютно идеальные требования, однако она позволила существенно улучшить их качество. Также при анализе спецификации оценивалась ее полнота, структурированность и простота навигации по ней.

В процессе написания кода приложения (как frontend-, так и backend-части) проводилось модульное тестирование всех разрабатывавшихся блоков системы. Тестирование проводилось каждым из разработчиков самостоятельно ввиду лучшей информированности о специфике написанного кода и возможных ошибок. Во время модульного тестирования проверялись корректность работы кода, соответствие поведения системы ожидавшемуся, наличие популярных для такого рода блоков ошибок.

Тестирование системы проводилось по методу “черного ящика” (black box testing): во время тестирования не уделялось внимания архитектуре системы, специфике ее реализации, структуре – только доступным пользователю внешним интерфейсам. Этот способ тестирования был выбран по той причине, что при постановке задачи разработки отдельное внимание было уделено удобству использования системы абсолютно разными пользователями, в том числе не имеющими знаний в компьютерных науках. С помощью данного вида тестирования можно имитировать взаимодействие пользователя с системой, проверяя, что UX приложения именно такой, как предполагалось.

В рамках тестирования способом “черного ящика” проводилось функциональное и нефункциональное тестирование приложения. На основании функциональных требований к системе были написаны тесты для пользовательского интерфейса, в результате тестами было покрыто 96,8% функциональных требований (с матрицей трассируемости требований можно ознакомиться в Приложении 7). Тесты функционального тестирования проверяли корректность навигации между страницами, поведение полей ввода при корректном и некорректном заполнении, корректность сохранения различных введенных данных, правильность отображения страниц.

Функциональное тестирование позволило выявить определенные недостатки системы и недоработки. Так, во время тестирования интерфейса участника мероприятий было обнаружено, к примеру, что при возникновении ошибки 400 кастомизированный экран не отображается, хотя страницы других ошибок работают корректно; при тестировании формы обратной связи стало понятно, что поле загрузки файла отслеживает формат файла при загрузке через кнопку, однако не отслеживает при перетаскивании файла в поле ввода, и таким образом можно прикрепить любой файл, а не только изображения и pdf. Во время тестирования интерфейса организатора было обнаружено, что при отправке массовых почтовых рассылок участникам теряется прикрепленное к письму изображение. Во время тестирования были выявлены и другие дефекты приложения, некоторые, особенно важные, исправлялись в первую очередь, исправление менее значимых и приоритетных

дефектов было отложено. Поскольку над проектом работали всего два человека, было решено не использовать для отслеживания дефектов дополнительных инструментов – дефекты, требующие исправления, помечались в Github Issues, а иногда исправлялись на уровне прямой коммуникации между разработчиками, не фиксируясь дополнительно. Также в Github Issues отслеживались все задачи, ставившиеся перед разработчиками: на номера этих задач ссылаются коммиты с их решениями, что упрощает восприятия содержания коммита.

Функциональное тестирование также дало разработчикам возможность взглянуть на приложение глазами пользователя и найти способы улучшить пользовательское взаимодействие: поскольку часть функционала внедрялась в систему на этапе разработки, ее проектированию не всегда уделялось должное внимание. Были выявлены некоторые недоработки с точки зрения проектирования. По итогу тестирования были приняты решения о изменении способа регистрации в системе, переносе кнопки выхода из аккаунта со страницы профиля в навбар, добавлении функции восстановления пароля, изменении функционала настройки числа свободных мест на мероприятие в административной части платформы.

В настоящий момент 96,91% проведенных 97 функциональных тестов выполняются в точности так, как это предполагалось. Результаты функционального тестирования и список проведенных тестов приводятся в Отчете по тестированию в Приложении 8.

На данный момент работы над проектом не проводилось особенно тщательного тестирования производительности системы и нагрузочного тестирования. Однако для нашей системы важна возможность одновременной работы многих пользователей: в требованиях к производительности – до 500 человек. Проведение нагрузочного тестирования планируется в ближайшее время. К данному же моменту было проведено базовое тестирование производительности с использованием инструмента PageSpeed Insights от Google [15]. Благодаря этому сервису были выявлены возможности для улучшения системы: настройка эффективной работы с кэшем для статических элементов системы и настройка адаптивности для мобильных устройств. Также отчет о проведенном анализе страниц показал недостаточно высокий показатель контрастности полей ввода. В целом же тестирование производительности показало очень хорошие результаты с точки зрения времени загрузки страниц.

Позднее было принято решение о проведении бета-тестирования системы. Бета-тестирование проводилось при участии пятерых людей, вызвавшихся помочь. К сожалению, из-за сжатых сроков не удалось найти добровольцев, максимально отражающих целевую аудиторию продукта. Однако четверо из пяти участников тестирования – никак не связанные со сферой IT люди, а удобство в использовании людьми без специфических знаний в компьютерных науках было одним из ключевых требований к программе. Участники тестирования – студенты младших курсов нижегородских вузов, как люди, наиболее приближенные к возрасту участников мероприятий, а также недавние выпускники и совсем взрослые люди, как люди, близкие к возрасту организаторов.

Бета-тестирование приложения проводилось на личных встречах с участниками тестирования. Перед участниками ставилась определенная задача, для выполнения которой требовалось воспользоваться интерфейсом участника или интерфейсом организатора. При необходимости участникам давался также документ с данными для тестирования, в частности, одной из популярных задач при тестировании

функционала администратора было создать событие для регистрации, а зачастую для такой задачи организаторы используют уже готовый документ с расписанием, информация из которого лишь переносится в систему. Во время выполнения участниками поставленных задач за их работой велось пристальное наблюдение, фиксировались все затруднения в работе с системой, отслеживались пути, которые пользователи выбирают для решения своих задач.

Бета-тестирование также помогло выявить некоторые ошибки системы и части функционала, которые наименее понятны пользователю. Во время функционального тестирования системы не было уделено достаточного внимания тестированию граничных значений. Однако при бета-тестировании участник, обучающийся в Институте международных отношений и мировой истории Нижегородского государственного университета имени Н. И. Лобачевского, заполнил поле ввода места обучения полным наименованием института, и в результате стало ясно, что ввод значений, превышающих заданную длину поля ввода, не контролируется, и в результате система выдает очень сложную для восприятия пользователем ошибку. После того, как ошибка была выявлена, дефект был быстро исправлен. Также на бета-тестировании стали заметны все недоработки проектирования пользовательского взаимодействия организаторской части приложения. В частности, выбранные слова для наименования некоторых сущностей воспринимались как контринтуитивные, из-за чего участник тестирования не мог понять, как ему поступать дальше для решения своей задачи. По завершении этого тестирования были пересмотрены названия сущностей, а также была расширена справочная документация для организаторов, и следующее бета-тестирование интерфейса организаторов показало куда лучшие результаты.

С протоколами проведенных встреч с участниками бета-тестирования можно ознакомиться в Приложении 9.

Заключение

В рамках данной выпускной квалификационной работы была разработана система регистрации для мероприятий НИУ ВШЭ – Нижний Новгород. На данный момент работа над проектом еще ведется, поскольку всегда есть пути для улучшения пользовательского взаимодействия, а проект приложения планируется использовать в реальной жизни. Однако основные цели разработки достигнуты: реализовано вполне готовое, целостное и работоспособное приложение, произведен тестовый деплой системы на реальный сервер (ссылка на работающую платформу находится в Приложении 2). В ближайшие недели планируется развертывание системы на одном из серверов НИУ ВШЭ.

Выполнены все задачи, ставившиеся в начале работы: проведены сбор и анализ требований, спроектировано пользовательское взаимодействие, написана спецификация приложения, проработаны стили приложения и проведено прототипирование экранов, весь спроектированный интерфейс реализован в рамках frontend-разработки, приложение протестировано, в том числе на пользователях.

Поскольку цель достигнута и задачи выполнены, проведенную работу можно считать успешной. Конечно, полученная система не идеальна – недостатки есть как в спецификации, так и в проектировании и разработке – и, кроме того, абсолютно успешно протестировать систему невозможно: всегда будут ошибки, которые не удалось покрыть тестами. Однако несовершенства системы объяснимы недостаточно большим жизненным опытом авторов, который не позволял выбирать оптимальные пути решения некоторых задач. Впрочем, многие несовершенства приложения были исправлены на разных этапах работы над проектом благодаря ознакомлению с различной литературой, советам более опытных людей из разных сфер и своевременному тестированию. Можно отметить, что за время работы над проектом профессиональные навыки авторов существенно улучшились, кроме того, развились насмотренность и некоторые мягкие навыки, что еще раз говорит об успешности проведенной работы.

Источники

1. Горбунова О.Н. Информатизация общества и формирование трудового ресурса: проблемы, пути решения // Социально-экономические явления и процессы. - 2012. - №5-6. - С. 32-38.
2. Луков В.А., Луков С.В, Цифровизация в России: человеческое измерение // Знание. Понимание. Умение. - 2020. - №1. - С. 92-100.
3. Виггерс К., Битти Дж. Разработка требований к программному обеспечению. - 3-е изд., дополненное изд. - СПб.: Русская редакция, БХВ-Петербург, 2014. - 736 с.
4. Браузеры в России // Яндекс Радар URL: https://radar.yandex.ru/browsers?selected_rows=Ct58LP%252CRysHuf%252CnmpVtr (дата обращения: 25.11.2023).
5. ISO/IEC/IEEE International Standard - Systems and software engineering -- Life cycle processes -- Requirements engineering // IEEE Standards Associations URL: <https://standards.ieee.org/ieee/29148/5289/> (дата обращения: 20.11.2023).
6. Соловьева А.А. Сравнение программного обеспечения для разработки пользовательских интерфейсов и их прототипирования // Наука без границ. - 2020. - №4 (44). - С. 55-60.
7. Django overview // Django URL: <https://www.djangoproject.com/start/overview/> (дата обращения: 30.11.2023).
8. Обновленный стиль Вышки // Национальный исследовательский университет "Высшая школа экономики" URL: <https://www.hse.ru/info/brandbook/> (дата обращения: 15.12.2023).
9. Coolors - The super fast color palettes generator // Coolors URL: <https://coolors.co/> (дата обращения: 18.12.2023).
10. The color system: Tools for picking colors // Material Design URL: <https://m2.material.io/design/color/the-color-system.html#tools-for-picking-colors> (дата обращения: 18.12.2023).
11. Руководство по использованию фирменного стиля на русском языке // Национальный исследовательский университет "Высшая школа экономики" URL: <https://www.hse.ru/info/brandbook#guide> (дата обращения: 08.02.2024).
12. Подключение шрифтов в CSS // Snipp.ru URL: <https://snipp.ru/html-css/font-face> (дата обращения: 10.01.2024).
13. Поле загрузки файлов, которое мы заслужили // Хабр URL: <https://habr.com/ru/articles/423035/> (дата обращения: 20.02.2024).
14. Custom styled input type file upload button with pure CSS // Nikita Hlopov: Frontend Dev Blog URL: <https://nikitahl.com/custom-styled-input-type-file> (дата обращения: 20.02.2024).
15. PageSpeed Insights // Google for developers URL: <https://pagespeed.web.dev/> (дата обращения: 20.04.2024).
16. Лапшина Ю.А. Разработка информационной системы для регистрации на мероприятия НИУ ВШЭ - Нижний Новгород - 2023 URL: https://lms.hse.ru/ap_service.php?getwork&guid=97ECA030-62FD-4A6A-9687-5CD30F6BDD3C.
17. Титова Н.Д. Разработка информационной системы для регистрации на мероприятия НИУ ВШЭ - Нижний Новгород - 2023 URL:

https://lms.hse.ru/ap_service.php?getwork&guid=442CB6A1-01BA-4813-83A9-0C5CF1DE9845.

18. Лапшина Ю.А. Разработка информационной системы для регистрации на мероприятия НИУ ВШЭ - Нижний Новгород - 2024 URL: https://lms.hse.ru/ap_service.php?getwork&guid=00E1BCE6-E5DD-40A3-BE45-62A8FF0B7207.

Приложение 1. Исходный код приложения

Весь исходный код приложения, а также вся документация к нему хранятся в GitHub-репозитории, доступном по данной ссылке:



<https://github.com/yualapshina/registration-system-coursework>.

Приложение 2. Разработанное приложение

Результат разработки, приложение, развернутое на сервере, доступно по данной ссылке:



<https://hse-reg-sys.dns-dynamic.net/>.

Приложение 3. Дизайн-система приложения и прототипы страниц

Дизайн система приложения, все разработанные стили, спроектированные компоненты и отрисованные экраны:



<https://www.figma.com/design/5eO3mF7nryTVAC1sMNwzQV/Regsys-Main?node-id=1%3A5&t=M5q1kHZ9AnDbAPIq-1>.

Приложение 4. User Story

Пользовательские истории участника мероприятий:

Вася Червяков, ученик 10 класса общеобразовательной школы, рассеянный, но очень активный и мотивированный, озабочен своим портфолио и поступлением в университет, мечтает стать IT-инженером и хочет знать все об этой сфере

Как рассеянный человек, я хотел бы иметь в постоянном доступе все свои регистрации, чтобы точно ничего не забыть

Как рассеянный человек, я хотел бы заходить в систему с помощью электронной почты, чтобы ускорить процесс входа и не запоминать еще и логин

Как рассеянный человек, я хотел бы иметь возможность восстановить пароль с помощью электронной почты, чтобы не терять свои регистрации и сертификаты, если я забуду пароль

Как школьник, мотивированный поступлением, я хотел бы скачивать сертификаты о посещениях мероприятий, чтобы затем включать их в портфолио

Как человек с проблемами с мобильным интернетом, я хотел бы скачивать расписания мероприятий, чтобы иметь к ним доступ в любой момент времени

Как школьник, уже наметивший свой жизненный путь, я хотел бы выбирать только самые любопытные мне лекции и мастер-классы, чтобы узнавать больше об интересных мне вещах и специальностях

Пользовательские истории организатора мероприятий:

Элеонора Рафаэлевна, молодой сотрудник учебного офиса программы "филология", филолог по образованию и призванию, организующий, в числе прочего, мероприятия для школьников, очень загруженный работой

Как менеджер с образованием филолога, я хотела бы иметь понятную и простую систему, чтобы разбираться в ней, не имея специфических знаний в IT

Как человек, которому требуется время для освоения новых технологий, я хотела бы иметь четкую и понятную инструкцию по пользованию системой, чтобы быстро освоить ее

Как человек с загруженным графиком, я хотела бы быстро создавать события и мероприятия для регистрации, чтобы освободить время для других дел

Как организатор множества мероприятий для школьников, я хотела бы иметь возможность отправить рассылку о новом мероприятии сразу многим участникам предыдущих, чтобы приглашать на мероприятия заинтересованных в этом людей

Как человек с большим количеством бумажной работы, я хотела бы, чтобы система сама генерировала сертификаты о посещениях по шаблонам, чтобы не вваливать на себя еще больше бумажной работы

Как организатор мероприятий для школьников, я хотела бы иметь возможность просматривать записи на мероприятия, чтобы отслеживать число записей и масштабировать мероприятия при слишком большом или малом числе записей

Как организатор мероприятий для школьников, я хотела бы простую систему отслеживания посещения мероприятий, чтобы избежать необходимости проводить переключку на каждом мастер-классе и при этом выдавать сертификаты только тем школьникам, которые действительно пришли

Как адресат вопросов не подкованных в информационных технологиях возрастных педагогов школ, я хотела бы иметь возможность вручную добавлять регистрации участников, чтобы регистрировать школьников, которые приходят не по своей инициативе, а от школы

Приложение 5. User Flow

Диаграммы различных процессов взаимодействия пользователя с системой находятся в FigJam-файле по ссылке



<https://www.figma.com/board/qPkYyhK69EOTEppWrA6YyW/User-Flow-%26-User-Story?node-id=0%3A1&t=Jmyp4RHQaJFyt2bt-1>.

На следующих страницах приводятся диаграммы процессов.

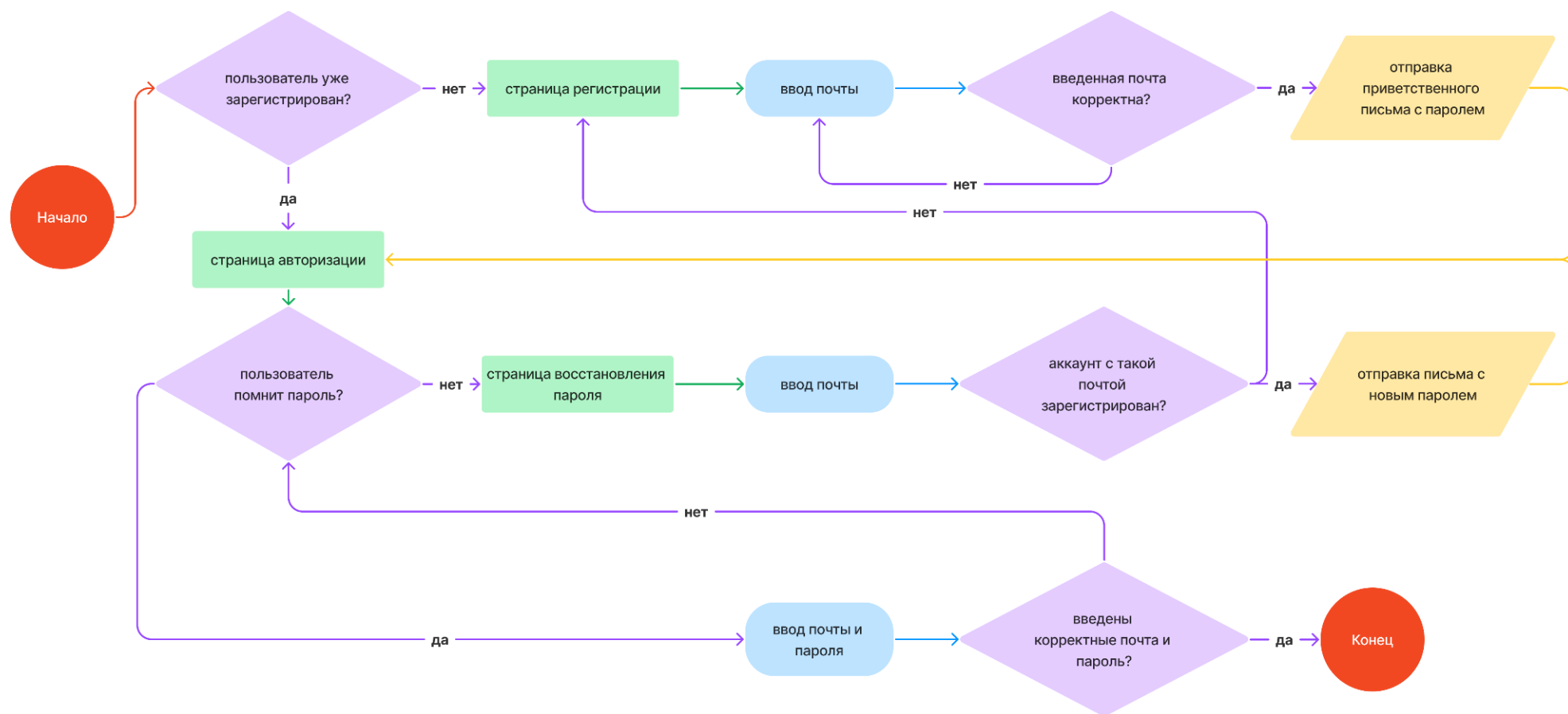


Диаграмма процесса авторизации пользователя (участника мероприятий) в системе

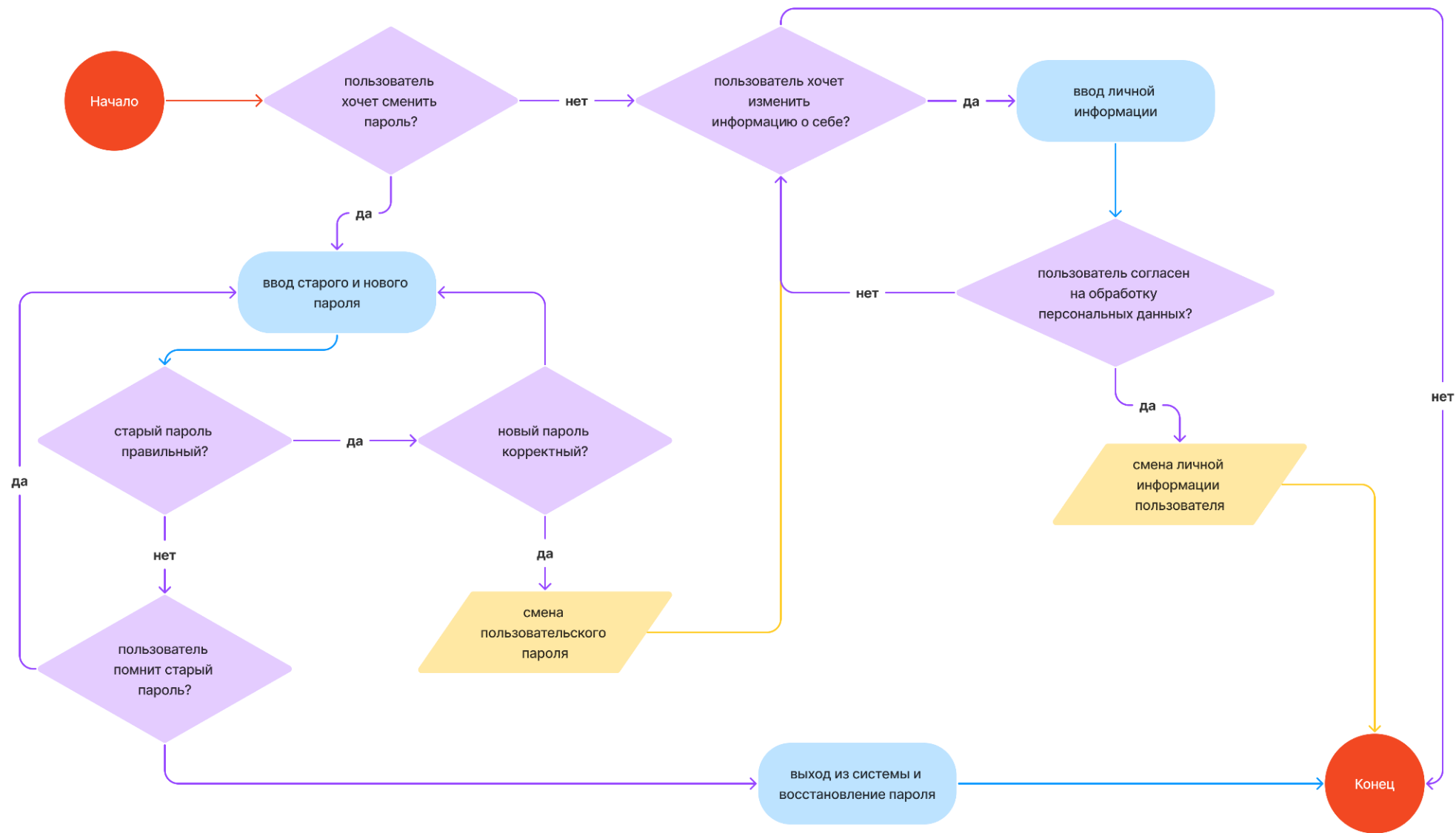


Диаграмма процесса работы со страницей профиля пользователя (участника мероприятий)

Приложение 6. Скриншоты некоторых экранов приложения

The screenshot shows the registration page of an application. At the top, there is a navigation bar with links: "войти" (login), "зарегистрироваться" (register), "обратная связь" (feedback), and "справка" (help). The main heading is "регистрация" (registration). Below it, there is a section for "электронная почта" (email) with a text input field containing the placeholder "Введи твой адрес электронной почты". A note states: "На введённый тобой адрес будет направлен пароль, который ты сможешь изменить после входа в профиль". There are two checkboxes: "Я даю согласие на обработку персональных данных" and "Я даю согласие на фотографирование". At the bottom, there is a blue button labeled "создать аккаунт" (create account).

Страница регистрации

The screenshot shows the "предстоящие мероприятия" (upcoming events) page. The top navigation bar includes "профиль" (profile), "моё расписание" (my schedule), "регистрация" (registration), "обратная связь" (feedback), "справка" (help), and "выйти" (logout). On the left, there is a sidebar with filters: "Аудитория" (Audience) with checkboxes for "школьники" (schoolchildren) and "студенты" (students), and "Направление" (Direction) with checkboxes for "математика" (mathematics), "компьютерные науки" (computer science), "филология" (philology), "экономика" (economics), "лингвистика" (linguistics), and "маркетинг" (marketing). A "фильтровать" (filter) button is at the bottom of the sidebar. The main content area features the heading "предстоящие мероприятия" and a specific event titled "День открытых дверей НИУ ВШЭ" (Open House of HSE University). Below the title, it lists categories: "компьютерные науки", "школьники", "экономика", "лингвистика", and "маркетинг", followed by the location "ул. Родионова, 136" and the date "25 мая 2024 г.". A blue button labeled "хочу сюда" (I want to go here) is positioned below the event details.

Страница предстоящих мероприятий

профиль

моё расписание

регистрация

обратная связь

справка

выйти

твое расписание

День открытых дверей НИУ ВШЭ

25 мая 2024 г.

11.00-12.00

Открытие для открытых дверей

QR

Еще не посещено

ул. Родионова, 136, ауд. 102

Анна Бляхман

12.00-13.00

О программе Программная инженерия

QR

Еще не посещено

ул. Родионова, 136, ауд. 104

Наталья Асеева

скачать

изменить

удалить

прошедшие события

Регистрация успешно обновлена

127.0.0.1:8000/register

День открытых дверей ИМиКН

Страница расписания пользователя

профиль

моё расписание

мероприятия

обратная связь

справка

выйти

данные для входа:

электронная почта

dm1tryy4sh1n@yandex.ru

старый пароль

Введи старый пароль

новый пароль

Придумай новый пароль

новый пароль (снова)

Твой новый пароль - ещё раз

Пароль должен содержать как минимум 8 символов, одну заглавную букву и одну цифру

После изменения пароля потребуются заново зайти в профиль

изменить пароль

личная информация:

фамилия

Яшин

имя

Дмитрий

отчество

Евгеньевич

место обучения

Институт международных отношений и мировой истории

телефон

+79200196694

телеграм

@ArthurTheBox

сохранить

Страница профиля пользователя

Приложение 7. Матрица трассируемости

С полной матрицей трассируемости функциональных требований можно ознакомиться в таблице по ссылке:



<https://github.com/yualapshina/registration-system-coursework/blob/master/documents/side%20files/traceability-matrix.xlsx>.

Ниже приводится фрагмент матрицы для наглядности ее структуры.

Тесты		Требования																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																								</
-------	--	------------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	----

Приложение 8. Отчет о финальном тестировании системы

Полный отчет о тестировании приложения со списком проведенных тестов и найденных дефектов:



<https://github.com/yualapshina/registration-system-coursework/blob/master/documents/side%20files/final-test-report.xlsx>.

Краткие результаты тестирования:

Общее число проведенных тестов: 97.

Число успешных тестов: 94.

Процент успешности тестирования: 96,98%.

Найдено дефектов: 4 (1 с высоким приоритетом, 1 со средним, 2 с низким).

Приложение 9. Протоколы бета-тестирования

Все протоколы бета-тестирования находятся в таблице по ссылке:



<https://github.com/yualapshina/registration-system-coursework/blob/master/documents/side%20files/beta-testing-protocol.xlsx>.

Приложение 10. Спецификация приложения

Ссылка на спецификацию системы:



<https://github.com/yualapshina/registration-system-coursework/blob/master/documents/side%20files/requirements.docx>.

Текст спецификации – на следующих страницах.

Спецификация к проекту “Разработка информационной системы для регистрации на мероприятия НИУ ВШЭ – Нижний Новгород”

1. Введение

1.1 Назначение

1.2 Предполагаемая аудитория и рекомендации к прочтению

1.3 Ссылки

2. Общее описание

2.1 Классы и характеристики пользователей

2.2 Функциональность продукта

2.3 Среда функционирования продукта

2.4 Ограничения при проектировании и разработке

2.5 Пользовательская документация

3. Функциональность системы

3.1 Функционал участников

3.2 Функционал администраторов

4. Требования к внешним интерфейсам

4.1 Пользовательские интерфейсы

4.2 Программные интерфейсы

4.3 Интерфейсы оборудования

4.3 Интерфейсы связи

5. Нефункциональные требования

6. Глоссарий

1. Введение

1.1 Назначение

В данной спецификации описывается веб-приложение для регистрации на мероприятия НИУ-ВШЭ - Нижний Новгород. Цель разработки - создание единой системы для регистрации на мероприятия нижегородского кампуса НИУ ВШЭ, доступной в использовании людям без специфических знаний в программировании.

1.2 Предполагаемая аудитория и рекомендации к прочтению

Данная спецификация ориентирована на разработчиков, работающих над проектом. Рекомендуется последовательное чтение спецификации для обеспечения полного ее понимания. В конце спецификации есть Глоссарий - есть смысл ознакомиться с ним перед чтением для конкретизации значения некоторых понятий.

1.3 Ссылки

- Брендбук НИУ ВШЭ - [на сайте](#) и [в Figma](#)

- [Репозиторий проекта](#)
- [Основной Figma-файл](#)
- [User Flow](#)

2. Общее описание

2.1 Классы и характеристики пользователей

Проект ориентирован на две группы пользователей: участников мероприятий и организаторов.

Участники - пользователи, которые используют систему для регистрации. В большинстве своем это люди среднего и старшего школьного возраста, заинтересованные в посещении вузовских мероприятий для расширения кругозора, развития различных навыков или для получения информации о поступлении.

Организаторы - пользователи, которые используют систему для управления событиями и записями на них. Как правило, это сотрудники учебных офисов или центра довузовской подготовки, или же студенты-волонтеры.

2.2 Функциональность продукта

Для каждой из пользовательских групп должен существовать свой функционал.

Участники должны иметь доступ к личному кабинету, который позволяет регистрироваться на мероприятия, настраивать и просматривать свое расписание, проходить на мероприятия и скачивать сертификаты о посещении.

Организаторы должны иметь возможность создавать, изменять и удалять события, настраивать их расписания и теги, а также просматривать и редактировать записи на события.

Суперпользователи системы должны обладать всем функционалом организаторов, а также иметь возможность настраивать права доступа всех пользователей системы.

2.3 Среда функционирования продукта

Система должна быть размещена на сервере НИУ ВШЭ. Тестовый деплой нужно разместить на общедоступном сервере.

2.4 Ограничения при проектировании и разработке

Язык разработки: Python

Фреймворк: Django

База данных: PostgreSQL

Инструмент проектирования: Figma

Система контроля версий: Git

Онлайн-хранилище для репозитория: GitHub

2.5 Пользовательская документация

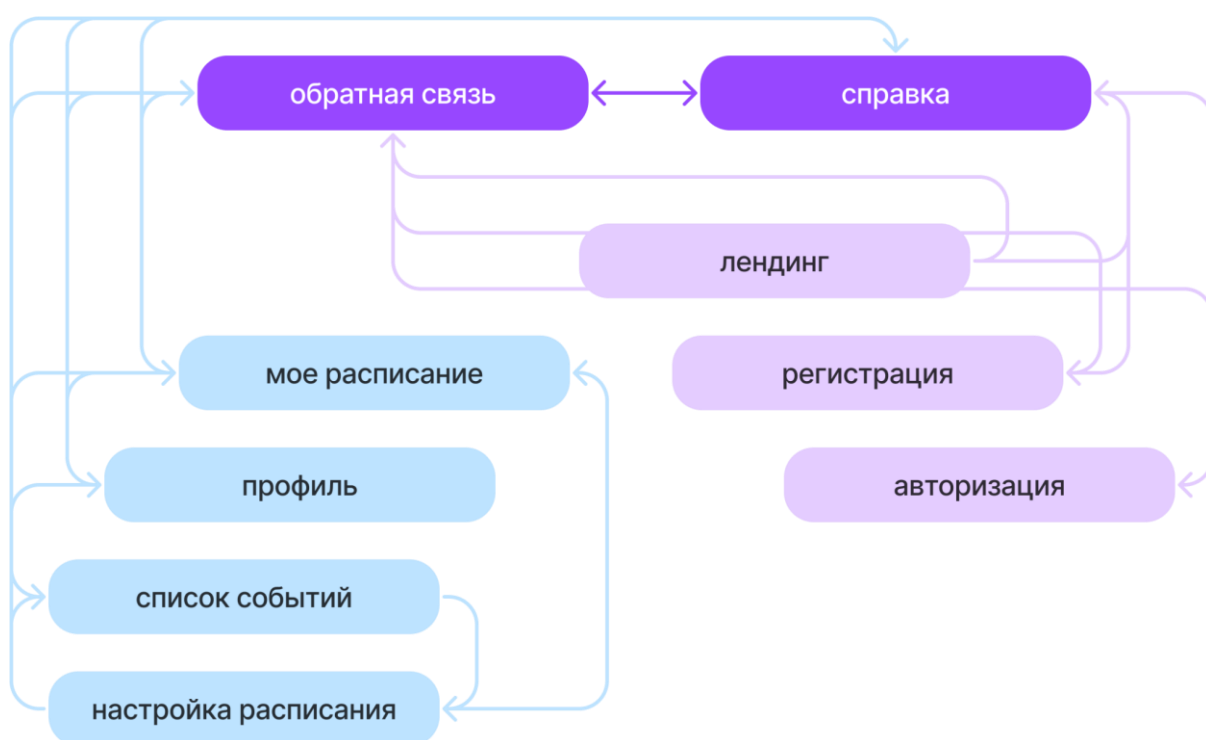
Система должна предоставлять интерактивные справки для пользователей-участников и документ-руководство для пользователей-администраторов.

3. Функциональность системы

Пользовательский интерфейс должен включать в себя интерфейсы участников и администраторов.

3.1 Функционал участников

Страницы интерфейса участника должны взаимодействовать друг с другом в соответствии со следующей схемой:



Функциональные требования к этим страницам изложены ниже.

3.1.1 Регистрация

FR-1.1. Система должна предоставлять пользователю возможность зарегистрироваться с помощью электронной почты.

FR-1.2. Рядом с полем ввода адреса электронной почты должна располагаться кнопка регистрации.

FR-1.2.1. При нажатии на кнопку регистрации система должна вывести сообщение об ошибке, если поле ввода почты не было заполнено.

FR-1.2.2. При нажатии на кнопку регистрации система должна вывести сообщение об ошибке, если в системе уже зарегистрирован пользователь с введенной почтой.

FR-1.3. При нажатии на кнопку регистрации на указанную почту должно отправляться письмо, включающее приветствие, автоматически сгенерированный пароль и сообщение о возможности изменить его в личном кабинете.

FR-1.3.1. При успешной отправке письма пользователь должен получить сообщение об этом.

FR-1.3.2. При возникновении ошибки при отправке письма пользователь должен получить сообщение об ошибке.

3.1.2 Авторизация

FR-2.1. Система должна предоставлять пользователю возможность авторизоваться с помощью адреса электронной почты и пароля.

FR-2.2. Рядом с полями ввода почты и пароля должна располагаться кнопка авторизации.

FR-2.2.1 При попытке нажать на кнопку авторизации без заполнения одного из полей система должна вывести сообщение об ошибке.

FR-2.2.2 Если оба поля были заполнены, при нажатии на кнопку авторизации выполняется поиск почты среди почт зарегистрированных в системе пользователей.

FR-2.2.2.1 Если пользователь с соответствующей почтой не был зарегистрирован, система должна вывести соответствующее сообщение об ошибке.

FR-2.2.2.2 Если пользователь с соответствующей почтой был зарегистрирован, система должна сравнить введенный пользователем пароль с сохраненным в системе.

FR-2.2.2.2.1 Если введенный пользователем пароль и сохраненный в системе совпадают, пользователь должен успешно авторизоваться в системе и перейти на страницу своего расписания.

FR-2.2.2.2.2 Если введенный пользователем пароль и сохраненный в системе не совпадают, пользователь должен получить сообщение об ошибке.

FR-2.3. Рядом с кнопкой авторизации должна располагаться кнопка восстановления пароля.

FR-2.3.1. При нажатии на кнопку восстановления пароля пользователь переходит на страницу Восстановления пароля.

3.1.3 Восстановление пароля

FR-3.1. Система должна предоставлять возможность восстановления пароля через ввод адреса электронной почты, связанного с учетной записью. Переход пользователя на эту страницу описан в пункте FR-2.3.1 данных требований.

FR-3.2. Рядом с полем ввода адреса электронной почты должна располагаться кнопка восстановления пароля.

FR-3.2.1. При нажатии на кнопку восстановления система должна вывести сообщение об ошибке, если поле ввода почты не было заполнено.

FR-3.2.2. При нажатии на кнопку восстановления система должна вывести сообщение об ошибке, если в системе не был зарегистрирован пользователь с введенной почтой.

FR-3.3. При успешном нажатии на кнопку восстановления пароля на указанную почту должно отправляться письмо, включающее автоматически сгенерированный пароль и сообщение о возможности изменить его в личном кабинете.

FR-3.3.1. При успешной отправке письма пользователь должен получить сообщение об этом.

FR-3.3.2. При возникновении ошибки при отправке письма пользователь должен получить сообщение об ошибке.

3.1.4 Навбар

FR-4.1. В любой момент взаимодействия с системой пользователю должна быть доступна панель навигации (навбар). Навбар включает в себя ссылки на различные страницы системы, при нажатии на эти ссылки осуществляется переход к страницам.

FR-4.1.1. Если пользователь не авторизован в системе, навбар включает в себя ссылки на страницы регистрации, авторизации, обратной связи, справки.

FR-4.1.2. Если пользователь авторизован в системе, навбар включает в себя ссылки на страницы профиля, расписания пользователя, регистрации на мероприятия, обратной связи, справки.

3.1.5 Обратная связь

FR-5.1. Система должна предоставлять и зарегистрированным, и незарегистрированным пользователям доступ к форме обратной связи. Доступ осуществляется через навбар (FR-4.1).

FR-5.2. Форма обратной связи должна включать в себя поля ввода темы обращения и текста сообщения, заполнение этих полей обязательно.

FR-5.3. Форма обратной связи должна включать в себя функционал прикрепления файла, пользователь имеет возможность прикрепить ровно один файл - изображение или pdf - или не прикреплять ничего.

FR-5.4. Форма обратной связи для незарегистрированных пользователей также должна включать поле ввода электронной почты, на которую пользователь получит ответ. Это поле обязательно для заполнения.

FR-5.5. Ниже всех полей обратной связи находится кнопка отправки сообщения.

FR-5.5.1. При попытке нажать на кнопку отправки без заполнения обязательных полей система должна показывать пользователю сообщение об ошибке.

FR-5.6. После отправки сообщения письмо с ним направляется с почтового бота системы на почту для отправки и получения обратной связи.

FR-5.6.1 После отправки письма пользователь получает сообщение об успешной отправке письма.

FR-5.7. После отправки сообщения копия письма с ним направляется с почтового бота системы на указанную в аккаунте пользователя / введенную в соответствующем поле почту.

FR-5.7.1. При ошибке при отправке пользователю копии письма система должна выводить сообщение с предупреждением о том, что ответ на письмо скорее всего не дойдет до пользователя из-за некорректности введенной почты.

3.1.6 Профиль

FR-6.1. Система должна предоставлять зарегистрированным пользователям доступ к странице профиля. Доступ осуществляется через навбар (FR-4.1).

FR-6.2. Страница профиля включает в себя два блока: блок данных для входа и блок личных данных.

FR-6.3. В блоке данных для входа отображаются электронная почта пользователя (редактирование недоступно) и поля для изменения пароля: для ввода старого и нового пароля, а также кнопка изменения пароля.

FR-6.3.1 Текст в полях ввода пароля маскируется.

FR-6.3.2. Также блок содержит информацию о требованиях к паролю (не менее 8 символов) и о том, что после смены пароля придется войти в аккаунт заново.

FR-6.3.3. После нажатия на кнопку изменения происходит сравнение сохраненного в системе пароля пользователя с введенным им старым паролем. Если пароли не совпадают, пользователь получает сообщение об ошибке, смена пароля не производится.

FR-6.3.4. После нажатия на кнопку изменения происходит проверка введенного нового пароля на корректность (не менее 8 символов). Если пароль некорректен, пользователь получает сообщение об ошибке, смена пароля не производится.

FR-6.3.5. После нажатия на кнопку изменения и всех проверок пароль пользователя обновляется, пользователь оказывается на странице авторизации.

FR-6.4. В блоке личной информации пользователь может ввести или изменить информацию о себе.

FR-6.4.1. Блок личной информации содержит поля ввода имени, фамилии, отчества, места обучения, телефона и телеграма. В качестве плейсхолдеров используются сохраненные в системе значения этих полей.

FR-6.4.2. Перед сохранением информации пользователь должен подтвердить свое согласие на обработку персональных данных, пользователю доступно скачивание согласия

FR-6.4.3. Блок личной информации также включает в себя кнопку сохранения данных. Она доступна всегда, при нажатии на нее введенные в любое из полей данные сохраняются в системе.

FR-6.4.4. После изменения данных пользователь получает сообщение о том, что данные успешно изменены

3.1.7 Мое расписание

FR-7.1. Система должна предоставлять зарегистрированным пользователям доступ к странице расписания пользователя. Доступ осуществляется через навбар (FR-4.1) или автоматически при авторизации/регистрации на мероприятие.

FR-7.2. Страница “Мое расписание” включает два раздела: Текущие мероприятия и Прошедшие мероприятия.

FR-7.3. Раздел Текущих мероприятий содержит информацию о грядущих мероприятиях, на которые пользователь был зарегистрирован.

FR-7.3.1. В разделе текущих мероприятий мероприятия сгруппированы по объединяющим их событиям. Отображаются название, место и время проведения, ведущие мероприятия.

FR-7.3.2. Пользователь может скачивать QR-коды, по которым можно посетить мероприятия.

FR-7.3.3. Пользователь может скачивать свое расписание в формате pdf.

FR-7.3.4. Около каждого из мероприятий есть кнопка изменения расписания, при нажатии на которую осуществляется переход на страницу настройки расписания (FR-9.).

FR-7.3.5. Пользователь может удалить свое расписание.

FR-7.4 Раздел Прошедших мероприятий включает список уже прошедших событий, на которые пользователь был зарегистрирован.

FR-7.4.1 О каждом из прошедших событий отображается следующая информация: название события, место и даты проведения.

FR-7.4.2 Пользователь может скачать сертификат о посещении любого из прошедших событий в формате pdf.

3.1.8 Доступные для регистрации события

FR-8.1. Система должна предоставлять зарегистрированным пользователям доступ к странице со списком доступных для регистрации событий. Доступ осуществляется через навбар (FR-4.1).

FR-8.2. Пользователь может фильтровать события с помощью соответствующих им тегов аудитории и тематики.

FR-8.3. Пользователь видит список доступных для регистрации событий с датами, местом проведения и списком лейблов, а также с кнопками регистрации.

FR-8.3.1. При наведении мыши на название события пользователь может увидеть его аннотацию.

FR-8.3.2. При нажатии кнопки регистрации пользователь переходит на страницу Настройки расписания.

3.1.9 Настройка расписания

FR-9.1. Система должна предоставлять зарегистрированным пользователям возможность выбирать для посещения некоторые из мероприятий, идущих параллельно внутри одного события. Доступ осуществляется при нажатии кнопки регистрации на странице списка доступных для регистрации событий (FR-8.3) или через кнопку изменения расписания на странице Мое расписание (FR-7.3.4).

FR-9.2. На странице Настройки расписания пользователь видит название события и список доступных для регистрации мероприятий внутри события. Для каждого мероприятия указана информация о нем, описанная в пункте 7.3.1.

FR-9.2.1. Мероприятия на странице настройки расписания сгруппированы по дням и времени проведения.

FR-9.2.2. Пользователь должен выбрать для посещения ровно одно мероприятие из каждой временной категории.

FR-9.3. После настройки расписания пользователь нажимает на кнопку завершения регистрации и переходит на страницу Мое расписание.

3.1.10 Справка

FR-10.1. Система должна предоставлять пользователям инструкцию по пользованию системой. Доступ к странице инструкции (Справки) предоставляется через навбар (FR-4.1).

FR-10.2. Разделы справок различаются для авторизованных и неавторизованных пользователей.

FR-10.2.1. Неавторизованный в системе пользователь в разделе справки видит описание системы и инструкцию по регистрации, авторизации и смене пароля, а также контактные данные для обращения при проблемах с использованием системы.

FR-10.2.2. Авторизованный в системе пользователь в разделе справки видит описание системы и инструкцию по использованию различных разделов, а также контактные данные для обращения при проблемах с использованием системы.

3.1.11 Экраны ошибок

FR-11.1. При возникновении различных (нижеописанных) ошибок в работе системы пользователь должен переходить на страницу с уведомлением об ошибке и ее кратким описанием.

FR-11.2. Должны быть реализованы экраны для следующих ошибок:

FR-11.2.1. Экран ошибки 400 - неверный запрос;

FR-11.2.2. Экран ошибки 403 - нет доступа к странице;

FR-11.2.3. Экран ошибки 404 - страница не найдена;

FR-11.2.4. Экран ошибки 500 - внутренняя ошибка сервера.

FR-11.3. На каждом из экранов ошибок должна быть кнопка возвращения в систему.

3.1.12 Лендинг

FR-12.1. Система должна предоставлять пользователю доступ к странице лендинга. Переход на эту страницу осуществляется с внешних ресурсов.

FR-12.2. На странице лендинга содержится описание системы и кнопка, ведущая к регистрации.

3.2 Функционал администраторов

Административная часть системы должна предоставлять пользователям с правами администраторов возможность работать с аккаунтами пользователей (с настройкой их прав), с мероприятиями для регистрации и с записями на эти мероприятия.

Функциональные требования к этим разделам описаны ниже.

3.2.1 Работа с системой регистрации

При работе с системой регистрации администратор (организатор) взаимодействует с такими сущностями, как События, Элементы расписания, Теги, Теги событий, Участники и Записи. Подробнее об этих сущностях можно прочитать в Глоссарии.

3.2.1.1 События

FR-13.1. Система должна предоставлять администратору доступ к странице списка событий.

FR-13.2. О каждом событии из списка должна быть представлена следующая информация: название, даты проведения, место проведения, количество мероприятий, относящихся к событию.

FR-13.2.1. При нажатии на название события администратор должен переходить на страницу события.

FR-13.2.2. При нажатии на количество относящихся к событию мероприятий администратор должен переходить к расписанию события.

FR-13.3. У администратора должна быть возможность фильтровать события по датам и месту проведения.

FR-13.4. Администратор должен иметь возможность создавать, изменять и удалять события.

FR-13.4.1. При создании или изменении события администратор должен ввести следующую информацию о событии: название, первый день, последний день, место проведения, аннотация. Все поля обязательны к заполнению.

FR-13.4.2. При создании или изменении события система должна проверять корректность введенных дат (дата начала должна быть не позже даты окончания).

FR-13.4.3. Система должна предоставлять администратору возможность просмотреть историю изменений события.

FR-13.4.4. При удалении события система должна также удалять все привязанные к нему мероприятия и записи.

FR-13.5. У администратора должна быть возможность искать события по названию.

3.2.1.2 Элементы расписания

FR-14.1. Система должна предоставлять администратору доступ к странице списка элементов расписания.

FR-14.2. О каждом элементе расписания (мероприятии) из списка должна быть представлена следующая информация: название, событие, к которому относится мероприятие, дата проведения, категория, место проведения, ведущий, количество занятых мест, количество записей на мероприятие.

FR-14.2.1. При нажатии на название мероприятия администратор должен переходить на страницу редактирования мероприятия.

FR-14.2.2. При нажатии на количество записей на мероприятие администратор должен переходить к записям на мероприятие.

FR-14.3. У администратора должна быть возможность фильтровать элементы расписания по событиям, к которым они относятся, дате, категории и повторяемости.

FR-14.4. Администратор должен иметь возможность создавать, изменять и удалять мероприятия.

FR-14.4.1. При создании или изменении мероприятия администратор должен ввести следующую информацию о мероприятии: название, событие, к которому относится мероприятие, дата проведения, категория, место проведения, ведущий, количество свободных мест, аннотация, наличие повтора. Все поля кроме наличия повтора обязательны к заполнению.

FR-14.4.2. При создании или изменении события система должна проверять корректность введенной даты (дата должна относиться к временному промежутку, в который проходит событие).

FR-14.4.3. Система должна предоставлять администратору возможность просмотреть историю изменений мероприятия.

FR-14.4.4. При удалении мероприятия система должна также удалять все записи на него.

FR-14.5. У администратора должна быть возможность искать мероприятия по названию.

3.2.1.3 Теги

FR-15.1. Система должна предоставлять администратору доступ к странице списка тегов.

FR-15.2. О каждом событии из списка должна быть представлена следующая информация: название, тип, число относящихся к нему событий.

FR-15.2.1. При нажатии на название тега администратор должен переходить на страницу редактирования тега.

FR-15.2.2. При нажатии на количество событий с тегом администратор должен переходить к списку этих событий.

FR-15.3. У администратора должна быть возможность фильтровать теги по типу.

FR-15.4. Администратор должен иметь возможность создавать, изменять и удалять теги.

FR-15.4.1. При создании или изменении тега администратор должен ввести следующую информацию о теге: название и тип.

FR-15.4.2. Система должна предоставлять администратору возможность просмотреть историю изменений тега.

FR-15.5. У администратора должна быть возможность искать теги по названию.

3.2.1.4 Теги событий

FR-16.1. Система должна предоставлять администратору доступ к странице списка тегов событий.

FR-16.2. Теги событий должны представлять собой связку тег-событие. Это сочетание должно быть уникальным.

FR-16.2.1. При нажатии на тег события администратор должен переходить на страницу изменения тега события.

FR-16.3. У администратора должна быть возможность фильтровать теги событий по типу и событию.

FR-16.4. Администратор должен иметь возможность создавать, изменять и удалять теги событий.

FR-16.4.1. При создании или изменении тегов событий администратор должен ввести следующую информацию о теге события: тег и событие.

FR-16.4.2. Система должна предоставлять администратору возможность просмотреть историю изменений тега события.

FR-16.5. У администратора должна быть возможность искать тег события по названию.

3.2.1.5 Участники

FR-17.1. Система должна предоставлять администратору доступ к странице списка участников мероприятий.

FR-17.2. О каждом из участников должна быть представлена следующая информация: адрес электронной почты, ФИО, номер телефона, telegram, место обучения, количество записей на мероприятия.

FR-17.2.1. При нажатии на адрес почты участника администратор должен переходить на страницу участника.

FR-17.2.2. При нажатии на количество записей на мероприятия администратор должен переходить к списку записей участника.

FR-17.3. Администратор должен иметь возможность создавать, изменять и удалять участников.

FR-17.3.1. При создании или изменении участника администратор должен ввести следующую информацию о нем: адрес почты, ФИО, место обучения, номер телефона, telegram.

FR-17.3.2. Система должна предоставлять администратору возможность просмотреть историю изменений участника.

FR-17.3.3. При удалении участника система должна также удалять все его записи.

FR-17.4. У администратора должна быть возможность искать участников по имени или почте.

3.2.1.6 Записи

FR-18.1. Система должна предоставлять администратору доступ к странице списка записей.

FR-18.2. Запись должна представлять собой связку участник-мероприятие. Это сочетание должно быть уникальным.

FR-18.2.1. Для каждой из записей должен быть указан статус посещения мероприятия.

FR-18.2.2. При нажатии на запись администратор должен переходить на страницу записи.

FR-18.3. У администратора должна быть возможность фильтровать записи событию и статусу посещения.

FR-18.4. Администратор должен иметь возможность создавать, изменять и удалять записи.

FR-18.4.1. При создании или изменении записи администратор должен ввести следующую информацию о записи: участник, мероприятие, статус посещения.

FR-18.4.2. Система должна предоставлять администратору возможность просмотреть историю изменений записи.

FR-18.5. У администратора должна быть возможность искать запись по названию.

3.2.2 Работа с пользователями

3.2.2.1 Пользователи

FR-19.1. Система должна предоставлять администратору доступ к странице списка пользователей.

FR-19.2. О каждом из пользователей должна быть представлена следующая информация: логин, адрес электронной почты, имя, наличие статуса администратора.

FR-19.2.1. При нажатии на логин пользователя администратор должен переходить на страницу редактирования пользователя.

FR-19.3. Администратор должен иметь возможность создавать, изменять и удалять пользователей.

FR-19.3.1. При создании или изменении пользователя администратор должен ввести следующую информацию о нем: адрес почты, имя, наличие статуса персонала и/или суперпользователя, список прав пользователя, принадлежность пользователя к группе.

FR-19.3.2. Система должна предоставлять администратору возможность просмотреть историю изменений пользователя.

FR-19.4. У администратора должна быть возможность искать пользователей по имени или почте.

FR-19.5. У администратора должна быть возможность фильтровать пользователей по статусу или принадлежности к группе.

3.2.2.1 Группы

FR-20.1. Система должна предоставлять администратору доступ к странице списка групп пользователей.

FR-20.2. При нажатии на название группы администратор должен переходить на страницу изменения группы.

FR-20.3. Администратор должен иметь возможность создавать, изменять и удалять группы.

FR-20.3.1. При создании или изменении группы администратор должен ввести следующую информацию о ней: название группы, список прав пользователей, принадлежащих к группе.

FR-20.3.2. Система должна предоставлять администратору возможность просмотреть историю изменений группы.

FR-20.4. У администратора должна быть возможность искать группы по названию.

3.2.3 Справка

FR-21. Администратор должен иметь возможность скачать документацию с руководством по работе с системой, находясь на любой из страниц административной части системы.

4. Требования к внешним интерфейсам

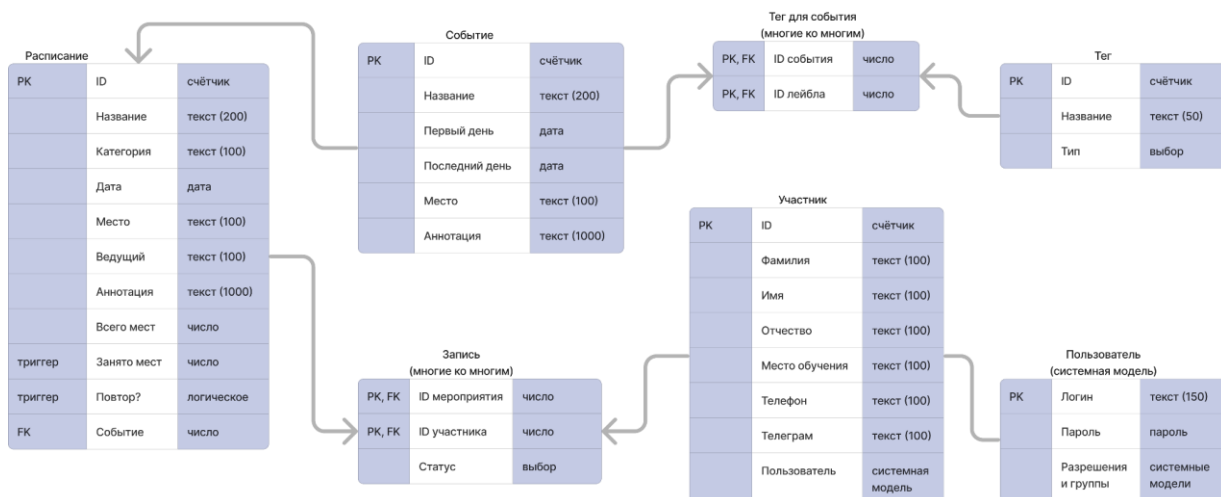
4.1 Пользовательские интерфейсы

Интерфейсы системы должны быть реализованы в соответствии с дизайн системой (см. ссылки). Дизайн система опирается на брендбук ВШЭ, любые изменения в ней не должны отходить от стилистики. В системе используются шрифты HSE Sans.

Взаимодействие пользователя с системой определяется user flow системы (см. ссылки).

4.2 Программные интерфейсы

Механизм обмена данными в системе и сама база данных должны быть реализованы следующим образом:



4.3 Интерфейсы оборудования

Пользователи будут использовать систему в браузерах как на десктопных, так и на мобильных устройствах. Разработка ведется с прицелом на использование на компьютерах, однако важно тестировать и корректность отображения интерфейсов на мобильных экранах.

4.3 Интерфейсы связи

Система должна поддерживать работу с почтовыми рассылками и уведомлениями. Схема осуществления рассылок следующая (есть две системных почты - почтовый бот и почта для отправки обратной связи):



Также возможны массовые рассылки участникам событий от организаторам.

5. Нефункциональные требования

NFR-1. Локализация

Поддерживаемые языки: русский

Формат дат ДД.ММ.ГГГГ

Формат времени ЧЧ:ММ:СС (24-часовой формат)

NFR-2. Совместимость

Поддерживаемые браузеры: Google Chrome, Yandex Browser.

Последняя и предыдущая версия для каждого

NFR-3. Производительность

Поддерживаемое число одновременно использующих систему пользователей: 500.

6. Глоссарий

Событие - масштабное мероприятие, включающее в себя различные активности (чаще всего указаны как мероприятия)

Элемент расписания (Мероприятие) - активность, относящаяся к конкретному событию; в рамках данного документа мероприятиями иногда называются и события вообще

Пример: мастер-класс по программированию на зимней школе ИМиКН; здесь зимняя школа - событие, мастер-класс - элемент расписания, мероприятие

Тег - категория, к которой относится событие. Есть два типа тегов: теги аудитории и теги тематики

Пример: теги аудитории - студенты, школьники; теги тематики: компьютерные науки, лингвистика, дизайн

Тег события - связующее звено между тегом и событием, связывает многие ко многим

Запись - связующее звено между мероприятием и участником, пометка о записи участника на мероприятие со статусом посещения

Участник - пользователь, регистрирующийся на мероприятия

Администраторы и организаторы - пользователи, имеющие доступ к административной части сайта, в частности - к настройке и редактированию событий, мероприятий и записей