

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	3
1. ОБЩАЯ ЧАСТЬ .....	7
1.1. Цель разработки .....	7
1.2. Средства разработки .....	7
2. СПЕЦИАЛЬНАЯ ЧАСТЬ .....	9
2.1. Постановка задачи .....	9
2.1.1. Входные данные .....	9
2.1.2. Выходные данные .....	10
2.1.3. Подробные требования к проекту .....	11
2.2. Внешняя специфика .....	13
2.2.1. Описание задачи .....	13
2.2.2. Входные и выходные данные .....	15
2.2.3. Методы .....	19
2.2.4. Тесты .....	19
2.2.5. Контроль целостности данных .....	20
2.3. Проектирование .....	21
2.3.1. Схема архитектуры программы .....	21
2.3.2. Логическая модель данных .....	22
2.3.3. Физическая модель данных .....	23
2.3.4. Структурная схема программы .....	28
2.3.5. Функциональная схема .....	31
2.3.7. Схема пользовательского интерфейса .....	31
2.4. Результат работы программы .....	32
3. ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ .....	34

3.1. Инструментальная среда разработки.....	34
3.2. Отладка программы.....	34
3.3. Защитное программирование .....	34
3.3.2. Защита приложения .....	35
3.3.3. Защита данных.....	36
3.4. Характеристика программы.....	37
ЗАКЛЮЧЕНИЕ .....	38
СПИСОК ИСПОЛЬЗУЕМЫХ МАТЕРИАЛОВ .....	40
ПРИЛОЖЕНИЕ А. Текст программы	
ПРИЛОЖЕНИЕ Б. Сценарий и результаты тестовых испытаний	
ПРИЛОЖЕНИЕ В. Руководство пользователя	
ПРИЛОЖЕНИЕ Г. Скрипт базы данных	
ПРИЛОЖЕНИЕ Д. Диаграмма бизнес-процессов	
ПРИЛОЖЕНИЕ Е. Диаграмма классов	

## ВВЕДЕНИЕ

В современном информационном обществе, где технологический прогресс и доступ к интернету становятся все более распространенными, онлайн-обучение приобретает все большую популярность. Это новое направление в образовании предоставляет уникальные возможности для получения знаний и развития навыков без привязки к определенному месту и времени.

Целью данной дипломной работы является создание WEB-платформы для онлайн-обучения, специально адаптированной к потребностям студентов ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова». Платформа будет предоставлять студентам удобный и доступный способ получения образовательных материалов, необходимых для достижения желаемого уровня знаний и навыков.

Разработка данной WEB-платформы будет основываться на современных технологиях веб-разработки, а также учитывать специфические потребности и требования университета и его студентов. Она будет предоставлять разнообразные возможности для проведения онлайн-курсов, взаимодействия между преподавателями и студентами, оценивания успеваемости, доступа к обучающим материалам и другим ресурсам.

В результате реализации данной работы ожидается создание инновационной и удобной платформы, способствующей эффективному обучению и повышению качества образования в ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова». Это позволит студентам получать знания и умения в интерактивной форме, а преподавателям — эффективно вести образовательный процесс, отслеживать прогресс студентов и взаимодействовать с ними в режиме реального времени.

Данная дипломная работа состоит из нескольких разделов, включая анализ требований студентов и преподавателей ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова», а также описание выбранных технологий и инструментов для разработки WEB-платформы.

Кроме того, будут рассмотрены этапы проектирования, реализации и тестирования платформы, а также вопросы ее деплоя и поддержки.

Важным аспектом данной работы является анализ существующих проблем и недостатков в области онлайн-обучения, которые могут влиять на эффективность и результативность образовательного процесса. Это позволит определить основные требования к разрабатываемой WEB-платформе, чтобы она могла решить эти проблемы и предоставить студентам и преподавателям оптимальное обучающее окружение.

В итоге разработка WEB-платформы для онлайн-обучения на примере ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова» будет способствовать совершенствованию образовательного процесса и повышению качества обучения в университете. Студенты получают удобный и доступный инструмент для обучения и развития своих навыков, а преподаватели смогут эффективно проводить занятия, взаимодействовать со студентами и оценивать их успеваемость.

Данная работа представляет значимость разработки и внедрения современных WEB-платформ для онлайн-обучения в высшем образовании, особенно в контексте быстро меняющегося информационного общества и растущей роли онлайн-образования. Путем создания инновационной и удобной платформы, данная работа преследует цель улучшения образовательного процесса, повышения доступности образования и развития профессиональных навыков студентов ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова».

Разработка WEB-платформы для онлайн-обучения представляет собой актуальную и перспективную задачу, которая может принести значительную пользу образовательной системе. Онлайн-обучение имеет ряд преимуществ, таких как гибкость в организации образовательного процесса, доступность из любой точки мира, а также возможность индивидуализации обучения под потребности каждого студента.

Университеты все больше осознают важность и преимущества онлайн-обучения, поэтому разработка собственной WEB-платформы становится востребованной задачей. Использование современных технологий и инструментов веб-разработки позволяет создать интуитивно понятный и функциональный интерфейс, который удовлетворит требованиям и ожиданиям студентов и преподавателей.

Однако, разработка WEB-платформы для онлайн-обучения требует учета специфических потребностей учебного заведения. ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова» имеет свои особенности, связанные с профилем и специализацией университета, а также требованиями образовательных стандартов. Поэтому разработка платформы должна учитывать все эти аспекты и предоставить специализированные функции, которые помогут студентам и преподавателям достичь высоких результатов в учебном процессе.

В заключение, разработка WEB-платформы для онлайн-обучения на примере ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова» является актуальной и важной задачей, направленной на совершенствование образовательного процесса и повышение качества обучения студентов. Реализация этой работы позволит создать инновационную и эффективную платформу, способствующую развитию профессиональных навыков и успешной учебной деятельности студентов университета. Платформа будет предоставлять широкий спектр возможностей, включая проведение онлайн-курсов, доступ к обучающим материалам, взаимодействие между преподавателями и студентами, оценивание успеваемости, а также мониторинг и анализ прогресса обучения.

Таким образом, разработка WEB-платформы для онлайн-обучения на примере ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова» имеет огромный потенциал для улучшения образовательного процесса и обеспечения качественного обучения студентов. Платформа будет способствовать расширению доступа к образованию, повышению гибкости и

эффективности обучения, а также развитию современных технологий в образовании.

## 1. ОБЩАЯ ЧАСТЬ

### 1.1. Цель разработки

Целью данной дипломной работы является разработка WEB-платформы для онлайн-обучения, сфокусированной на улучшении образовательного процесса в ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова». Платформа будет создана с целью предоставления студентам и преподавателям инновационного и эффективного инструмента, который позволит улучшить доступность, гибкость и качество обучения. Основной акцент будет сделан на удобстве использования, взаимодействии между участниками образовательного процесса, доступе к обучающим материалам, оценивании успеваемости и мониторинге прогресса студентов. В результате разработки WEB-платформы ожидается повышение эффективности образовательного процесса и развитие профессиональных навыков студентов в рамках ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова».

В результате, разработка системы публикаций домашних заданий позволит упростить процесс назначения, опубликования и выполнения домашних заданий, повысить эффективность обучения и обеспечить удобный доступ к учебным материалам для всех пользователей системы.

### 1.2. Средства разработки

В качестве программных средств разработки использовалось ПО, представленное в таблице 1.

Таблица 1 – Программные средства разработки

№	Тип средства	Название средства	Назначение
1	2	3	4
1	Инструментальное средство разработки программных решений	Microsoft Visual Studio Code 1.66	Разработка клиентского приложения
2	Графическая оболочка	Supabase	Платформа для разработки приложений, которая предоставляет готовые инструменты и

№	Тип средства	Название средства	Назначение
1	2	3	4
			сервисы для работы с базами данных PostgreSQL

В качестве средств вычислительной техники используется ноутбук Honor Magicbook Pro. Характеристики представлены в таблице 2.

Таблица 2 – Технические средства

№	Тип оборудования	Наименование оборудования
1	2	3
Honor Magicbook Pro		
1	Центральный процессор	AMD Ryzen 5 5600H
2	Видеоадаптер	NVIDIA GeForce MX350
3	Оперативная память	2×8 ГБ DDR4-2666
4	Твердотельный накопитель	NVMe SSD M.2 512GB
5	Жёсткий диск	HLX-W19R 512 GB



## 2. СПЕЦИАЛЬНАЯ ЧАСТЬ

### 2.1. Постановка задачи

Задачи данной дипломной работы заключаются в разработке WEB-платформы для онлайн-обучения студентов ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова», которая позволит им получать доступ к качественным учебным материалам и курсам в любое время и из любой точки мира. Конкретные задачи включают в себя:

1. Изучение потребностей и ожиданий студентов в области онлайн-обучения, анализ существующих платформ и примеров лучших практик.
2. Проектирование и разработка базы данных для хранения курсов, материалов и информации о студентах, а также системы управления данными и администрирования.
3. Разработка пользовательского интерфейса и функциональных возможностей платформы, которые будут включать в себя возможность поиска, фильтрации и сортировки материалов, оценки курсов, обратной связи и т.д.
4. Реализация системы аутентификации и авторизации пользователей, обеспечивающей безопасность и конфиденциальность данных.
5. Тестирование и отладка платформы, а также ее развертывание на сервере для публичного доступа.
6. Написание документации и инструкций для пользователей и администраторов платформы.
7. Оценка эффективности и удобства использования платформы, а также предложение рекомендаций для ее дальнейшего развития и улучшения.

#### 2.1.1. Входные данные

В информационной системе входными данными является информация, предоставляемая пользователем во время различных действий, таких как регистрация, создание курсов, заданий и комментариев. Конкретные данные включают:

- Регистрация пользователя: фамилия, имя, отчество, адрес электронной почты, аватар и пароль.
- Создание курса: название курса, раздел, предмет, заголовок курса, аудитория и цветовая тема.
- Создание группы: название группы.
- Создание темы для добавления курсов, объединённых одной темой: название темы.
- Создание задания: тема задания, описание, срок сдачи, баллы для оценки и возможность прикрепления файлов.
- Добавление файлов: пользователи могут прикреплять различные файлы, такие как текстовые документы, изображения или видео, в рамках своих заданий или комментариев.
- Комментарии пользователей: пользователи могут оставлять комментарии и отзывы, предоставлять обратную связь по заданиям.
- Баллы для оценивания домашних заданий: пользователи могут присваивать баллы для оценки выполненных домашних заданий и устанавливать критерии оценки.

Все эти данные являются важными компонентами информационной системы и позволяют пользователям взаимодействовать с платформой, создавать и управлять содержимым, а также обмениваться информацией и получать обратную связь.

#### 2.1.2. Выходные данные

Выходными данными в информационной системе являются следующие информационные элементы:

- Информация о пользователе: включает данные о профиле пользователя, такие как фамилия, имя, адрес электронной почты, аватар и другие регистрационные данные.
- Информация о курсе: содержит данные о созданном курсе, включая его название, раздел, предмет, заголовок, аудиторию и цветовую тему.

- Информация о задании: включает данные о созданных заданиях, такие как тема, описание, срок сдачи и критерии оценки.
- Информация о группе: содержит данные о созданных группах, включая их название, участников и другую связанную информацию.
- Информация о темах: содержит данные о всех темах, включая их название и список заданий, объединённый одной темой.
- Уведомления по электронной почте: система может генерировать и отправлять письма с различными уведомлениями, такие как приглашения пользователей на курс, уведомления о регистрации пользователя в системе, ссылки для сброса пароля, приглашения в группу и уведомления о новых заданиях и уведомления о выставленных оценках.

Все эти выходные данные предоставляют информацию пользователям и позволяют им эффективно взаимодействовать с системой, получать уведомления и актуальную информацию о своей активности в рамках курсов и групп.

### 2.1.3. Подробные требования к проекту

Требования к проекту "Разработка WEB-платформы для онлайн-обучения (на примере ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова»)":

Функциональные требования:

Регистрация и аутентификация:

- Возможность регистрации новых пользователей.
- Система аутентификации для обеспечения безопасного доступа к платформе.
- Восстановление пароля и смена учетных записей пользователей.

Управление курсами и учебными материалами:

- Создание и редактирование курсов с указанием названия, описания, материалов и заданий.

- Возможность добавления видео лекций, презентаций, документов и других учебных материалов курса.

- Управление доступом к курсам для студентов и преподавателей.

- Управление заданиями, включая добавления их общую тему.

Взаимодействие между студентами и преподавателями:

- Возможность студентам задавать вопросы и получать ответы от преподавателей.

- Отслеживание прогресса студентов и предоставление обратной связи от преподавателей.

Оценка и проверка заданий:

- Возможность создания и назначения заданий студентам.

- Проверка заданий преподавателями.

- Выставление оценок и обратной связи по выполненным заданиям.

Нефункциональные требования:

Интерфейс и удобство использования:

- Интуитивно понятный и привлекательный пользовательский интерфейс, удобный для навигации и работы с учебными материалами.

- Адаптивный дизайн, поддерживающий различные устройства, включая компьютеры, планшеты и мобильные устройства.

Безопасность и конфиденциальность:

- Защита данных пользователей, включая персональные данные, с помощью шифрования и мер безопасности.

- Обеспечение безопасности платформы от вредоносных атак и несанкционированного доступа.

В заключение, эти требования предоставляют общую основу для разработки WEB-платформы для онлайн-обучения, ориентированной на ФГБОУ ВО "Российский экономический университет им. Г.В. Плеханова".

## 2.2. Внешняя специфика

### 2.2.1. Описание задачи

Основной задачей проекта является создание удобного и интуитивно понятного интерфейса для пользователей, где они смогут получать доступ к различным онлайн-курсам, сдавать домашние задания и получать обратную связь от преподавателей.

Основное требование к разрабатываемой платформе — это возможность интеграции с существующей системой управления образованием, используемой в университете. Также важно предусмотреть возможность добавления новых курсов и заданий, создание групп для студентов и обеспечение доступа к материалам курсов только авторизованным пользователям.

Разрабатываемая платформа должна быть адаптивной и удобной для использования на любых устройствах, включая мобильные устройства. Важно обеспечить безопасность пользователей, в том числе защиту персональных данных и контента курсов.

Итоговым результатом работы должна быть полнофункциональная платформа, позволяющая студентам получать качественное онлайн-обучение, а преподавателям - удобно создавать и проводить онлайн-курсы и оценивать успеваемость студентов.

На Рисунке 1 представлена диаграмма прецедентов, отображающая возможности пользователя в рамках системы манипуляции данными о контенте.

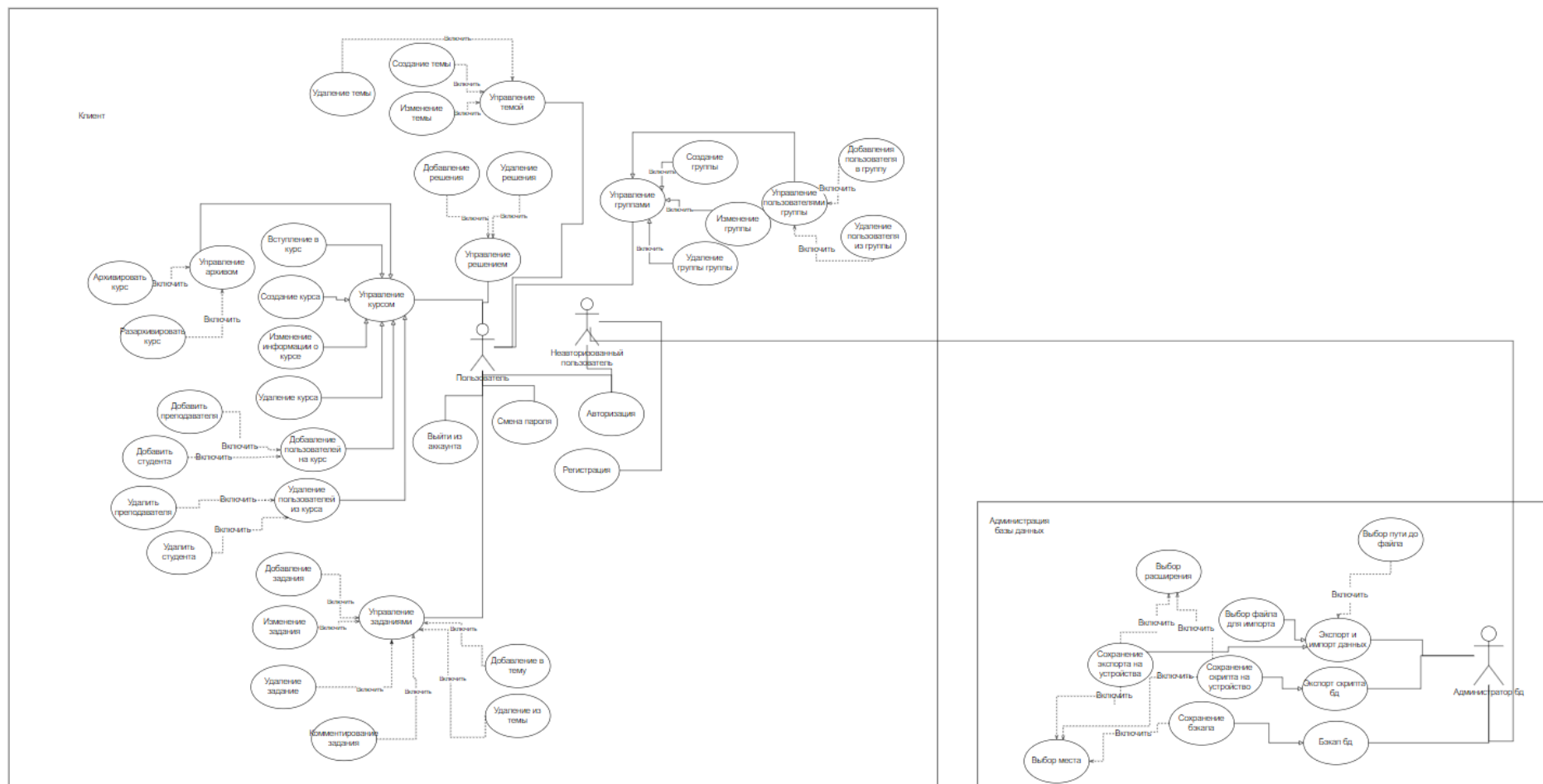


Рисунок 1 – Диаграмма прецедентов

По диаграмме видно, что в системе предусмотрено две роли, не считая неавторизованного пользователя: пользователь и администратор базы данных.

Приложение состоит из базы данных и самого WEB-сервиса. Реляционная база данных, с которой взаимодействует WEB-ресурс.

Диаграмма бизнес-процессов представлена в Приложение Д.

Диаграмма бизнес-процессов.

#### 2.2.2. Входные и выходные данные

В Таблице 3 представлены входные данные, которые вводит пользователь при взаимодействии с WEB-приложением.

Таблица 3 – Входные данные

Имя	Тип	Ограничение	Формат ввода	Описание
1	2	3	4	5
Аккаунт				
Email	String	[a-z, A-Z] {1,50}	Текстовое поле	Адрес электронной почты, используемый для авторизации, регистрации и рассылки
Surname	String	[a-я, А-Я] {1,50}	Текстовое поле	Фамилия
Name	String	[a-я, А-Я] {1,50}	Текстовое поле	Имя
Lastname	String	[a-я, А-Я] {1,50}	Текстовое поле	Отчество
Password	String	[a-z, A-Z, 0-9] {1,50}	Текстовое поле	Пароль пользователя для входа
File	String	[a-z, A-Z] {1,50}	Текстовое поле	Аватар пользователя
Курс				
Title	String	[a-я, А-Я, A-Z, a-z, 0-9] {1,50}	Текстовое поле	Название курса
Subject	String	[a-я, А-Я, A-Z, a-z, 0-9] {1,50}	Текстовое поле	Предмет
Section	String	[a-я, А-Я, a-z, A-Z, 0-9] {1,50}	Текстовое поле	Раздел
Cover	String	[a-z, A-Z, 0-9] {1,50}	Текстовое поле	Обложка курса
Decor	String	[a-z, A-Z, 0-9]	Поле выбора	Стиль курса

Имя	Тип	Ограничение	Формат ввода	Описание
1	2	3	4	5
		{1,50}		
Auditorium	String	[a-z, A-Z, 0-9] {1,50}	Текстовое поле	Аудитория, в которой проходят оффлайн занятия
Задание				
Title	String	[a-я, А-Я, a-z, A-Z, 0-9] {1,50}	Текстовое поле	Название задания
Description	String	[a-я, А-Я, a-z, A-Z, 0-9] {1,50}	Текстовое поле	Описание задания
Deadline_datetime	Date	[0-9] {1,50}	Текстовое поле	Дата и время сдачи
Points	Int	[0-9] {1,50}	Текстовое поле	Максимальный балл, который можно получить за выполнение этого задания
Files	String	[a-z, A-Z, 0-9] {1,50}	Текстовое поле	Дополнительные файлы с заданием
Решение				
Files	String	[a-z, A-Z, 0-9] {1,50}	Текстовое поле	Файлы с решением
Оценка				
Mark	Int	[0-9] {1,50}	Текстовое поле	Оценка решения, выставляемая преподавателем при проверке
Комментарий				
Text	String	[a-я, А-Я, a-z, A-Z, 0-9] {1,50}	Текстовое поле	Текст комментария, который пользователь оставляет под заданием
Группа				
Title	String	[a-я, А-Я, a-z, A-Z, 0-9] {1,50}	Текстовое поле	Наименование группы
Тема				
Title	String	[a-я, А-Я, a-z, A-Z, 0-9] {1,50}	Текстовое поле	Наименование темы

В Таблице 4 представлены выходные данные, которые предоставляет web - приложение.



Таблица 4 – Выходные данные

Поле	Ограничение	Тип	Описание
1	2	3	4
Информация о пользователе			
Email	[a-z, A-Z, 0-9, @] {1,50}	String	Адрес электронной почты
Surname	[a-я, А-Я] {1,50}	String	Фамилия пользователя
Name	[a-я, А-Я] {1,50}	String	Имя пользователя
Lastname	[a-я, А-Я] {1,50}	String	Отчество пользователя
File	[a-z, A-Z, 0-9] {1,50}	String	Аватар пользователя
Курс			
Title	[a-z, A-Z, 0-9, А-Я, а-я] {1,50}	String	Наименование курса
Subject	[a-z, A-Z, 0-9, А-Я, а-я] {1,50}	String	Предмет курса
Section	[a-z, A-Z, 0-9, А-Я, а-я] {1,50}	String	Раздел
Cover	[a-z, A-Z, 0-9] {1,50}	String	Обложка курса
Auditorium	[a-z, A-Z, 0-9, А-Я, а-я] {1,50}	String	Аудитория для оффлайн занятий
WelcomeLink	[a-z, A-Z, 0-9] {1,50}	String	Ссылка на курс
Задание			
Title	[a-z, A-Z, 0-9, А-Я, а-я] {1,50}	String	Название задания
Description	[a-z, A-Z, 0-9, А-Я, а-я] {1,50}	String	Описание задания
DeadlineDateTime	[0-9] {1,50}	DateTime	Дата сдачи
Status	[a-z, A-Z, 0-9, А-Я, а-я] {1,50}	String	Статус сдачи задания
DateTime	[0-9] {1,50}	DateTime	Дата создания задания
Points	[0-9] {1,50}	Int	Максимальный балл, который можно получить при сдаче задания
Files	[a-z, A-Z, 0-9] {1,50}	String	Дополнительные файлы с заданием

Поле	Ограничение	Тип	Описание
1	2	3	4
Письмо с приглашением на курс			
WelcomeLink	[a-z, A-Z, 0-9] {1,50}	String	Ссылка на курс
Письмо для подтверждения регистрации аккаунта			
WelcomeLink	[a-z, A-Z, 0-9] {1,50}	String	Ссылка для подтверждения
Письмо с приглашением в группу			
WelcomeLink	[a-z, A-Z, 0-9] {1,50}	String	Ссылка в группу
Письмо с уведомлением о новом задании			
WelcomeLink	[a-z, A-Z, 0-9] {1,50}	String	Ссылка на новое задание
Письмо с ссылкой для смены пароля			
PasswordLink	[a-z, A-Z, 0-9] {1,50}	String	Ссылка для смены пароля
Комментарии пользователя			
Text	[a-z, A-Z, 0-9, А-Я, а-я] {1,50}	String	Комментарий пользователя под заданием
DateTime	[0-9] {1,50}	DateTime	Дата, когда пользователь оставил комментарий
Пользователи на курсе			
Email	[a-z, A-Z, 0-9, @] {1,50}	String	Адрес электронной почты
Surname	[а-я, А-Я] {1,50}	String	Фамилия пользователя
Name	[а-я, А-Я] {1,50}	String	Имя пользователя
Lastname	[а-я, А-Я] {1,50}	String	Отчество пользователя
Пользователи в группе			
Email	[a-z, A-Z, 0-9, @] {1,50}	String	Адрес электронной почты
Surname	[а-я, А-Я] {1,50}	String	Фамилия пользователя
Name	[а-я, А-Я] {1,50}	String	Имя пользователя
Lastname	[а-я, А-Я] {1,50}	String	Отчество пользователя
Группа			
Title	[а-я, А-Я, а-з, А-З, 0-9] {1,50}	String	Наименование группы
Тема заданий			
Title	[а-я, А-Я, а-з, А-З, 0-9] {1,50}	String	Наименование темы

Поле	Ограничение	Тип	Описание
1	2	3	4
	{1,50}		
Оценка			
Mark	[0-9] {1,50}	Integer	Оценка, выставляемая преподавателем
Письмо с уведомлением об изменённой оценке			
WelcomeLink	[a-z, A-Z, 0-9] {1,50}	String	Ссылка на задание, в котором преподаватель изменил оценку

### 2.2.3. Методы

База данных проекта была реализована с помощью СУБД PostgreSQL. База данных построена по реляционному принципу и доведена до 3 нормальной формы.

Для разработки использовались фреймворк React.js совместно с JavaScript. В качестве API был выбран фреймворк Node.js Express.

Express предоставляет удобные методы для обработки HTTP-запросов и ответов, а также многочисленные функции маршрутизации и обработки ошибок. Он также позволяет использовать различные middleware, чтобы обрабатывать запросы до того, как они достигнут обработчика маршрута.

Express не является полным стеком веб-разработки, но он предоставляет достаточно функциональности для создания веб-приложений любой сложности. Он также имеет большое сообщество разработчиков и множество сторонних модулей, которые могут быть использованы для расширения его возможностей.

В целом, Express — это мощный инструмент для создания веб-приложений и API на Node.js, который является одним из самых популярных фреймворков для Node.js.

### 2.2.4. Тесты

С тестовыми сценариями и результатами тестовых испытаний можно ознакомиться в приложении Б. Сценарий и результат тестовых испытаний.

### 2.2.5. Контроль целостности данных

В Таблице 5 представлен контроль целостности данных, описывающий ситуации и реакцию приложения на выполнение задач, связанных с сохранением, выводом, изменением или удалением данных.

Таблица 5 – Контроль целостности данных

№	Ситуация	Аномалия	Реакция программы	Пояснение
1	2	3	4	5
1	Попытка регистрации пользователя	Не заполнены обязательные поля	Вывод сообщения об ошибке пустых полей	Информация о пользователе должна быть корректно заполнена
2	Попытка регистрации пользователя	Введение уже существующего в системе адреса электронной почты	Вывод сообщения об ошибке с указанием, что пользователь уже существует	Email пользователей в системе не должен повторяться
3	Попытка входа в аккаунт приложения	Не заполнены обязательные поля	Вывод сообщения об ошибке пустых полей	Чтобы идентифицировать пользователя нужно корректно ввести его email и пароль
4	Попытка входа в аккаунт приложения	Введён несуществующий пользователь	Вывод сообщения об ошибке неверного адреса электронной почты	Чтобы идентифицировать пользователя нужно корректно ввести его email и пароль
5	Попытка входа в аккаунт приложения	Введён неверный адрес электронной почты	Вывод сообщения об ошибке неверного адреса электронной почты	Чтобы идентифицировать пользователя нужно корректно ввести его email и пароль
6	Попытка входа в аккаунт приложения	Введён неверный пароль	Вывод сообщения об ошибке неверного пароля	Чтобы идентифицировать пользователя нужно корректно ввести его email и пароль
7	Попытка входа в курс	Такого курса не существует	Вывод ошибки, что курса не существует	Чтобы увидеть курс для начала требуется его создать
8	Изменение информации об аккаунте	Изменение адреса электронной почты на уже существующий	Отмена изменения аккаунта	Email пользователей в системе не должен повторяться

№	Ситуация	Аномалия	Реакция программы	Пояснение
1	2	3	4	5
9	Изменение информации об аккаунте	Количество символов в изменённом пароле меньше шести	Отмена изменения аккаунта	Согласно правилу авторизации минимальное количество символов в пароле: 6
10	Изменение информации об аккаунте	Не заполнены обязательные поля	Информация об аккаунте останется прежней	Информация о пользователе должна быть корректно заполнена
11	Добавление данных	Не заполнены обязательные поля	Вывод предупреждения, о том, что все поля должны быть заполнены	Информация о данных должна быть корректно заполнена
12	Изменение данных	Поля не заполнены	Информация о данных останется прежней	Информация о данных должна быть корректно заполнена

### 2.3. Проектирование

#### 2.3.1. Схема архитектуры программы

На Рисунке 2 представлена схема архитектуры программы использованная в WEB – приложении.

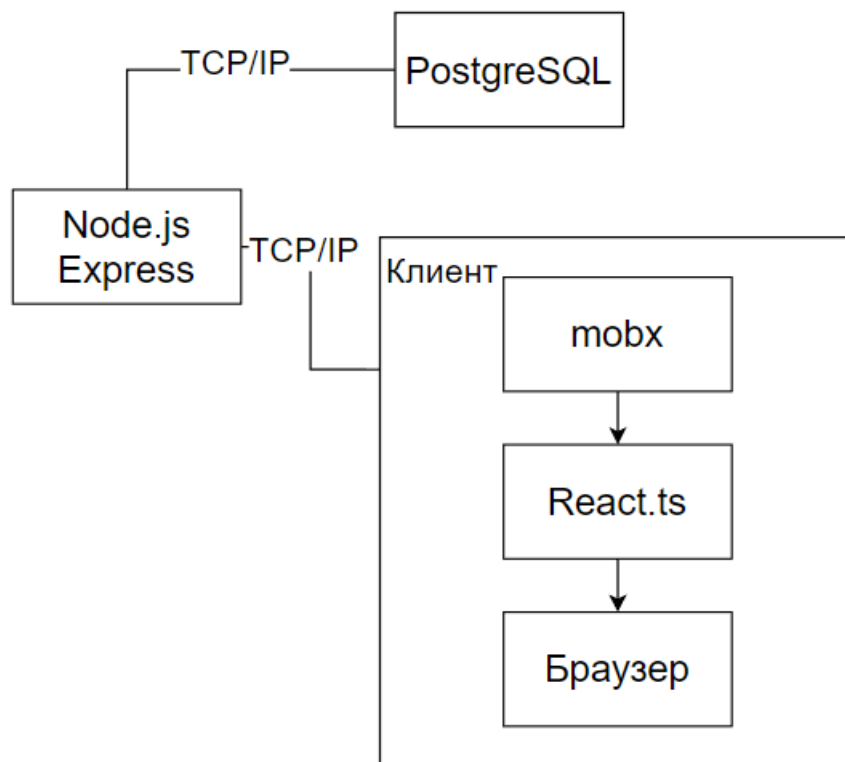


Рисунок 2 – Схема архитектуры программы

В Таблице 6 представлено описание схемы архитектуры программы.

Таблица 6 – Описание схемы архитектуры программы

Наименование модуля	Функция модуля	Окружение модуля
База данных (PostgreSQL)	Хранит данные вводимые и обрабатываемые пользователем	Передаёт данные из бд на сервер
Сервер приложения (Node.js Express)	Обслуживает веб-запросы и возвращает ответы на эти запросы	Получает взаимодействие из базы данных и отправляет все остальные действия на клиентскую часть
Клиент (React.ts)	Клиентская часть приложения, которая обрабатывает запросы с сервера и представляет собой тело приложения	Создаёт компоненты для пользовательского вида

### 2.3.2. Логическая модель данных

На Рисунке 3 представлена логическая модель данных.

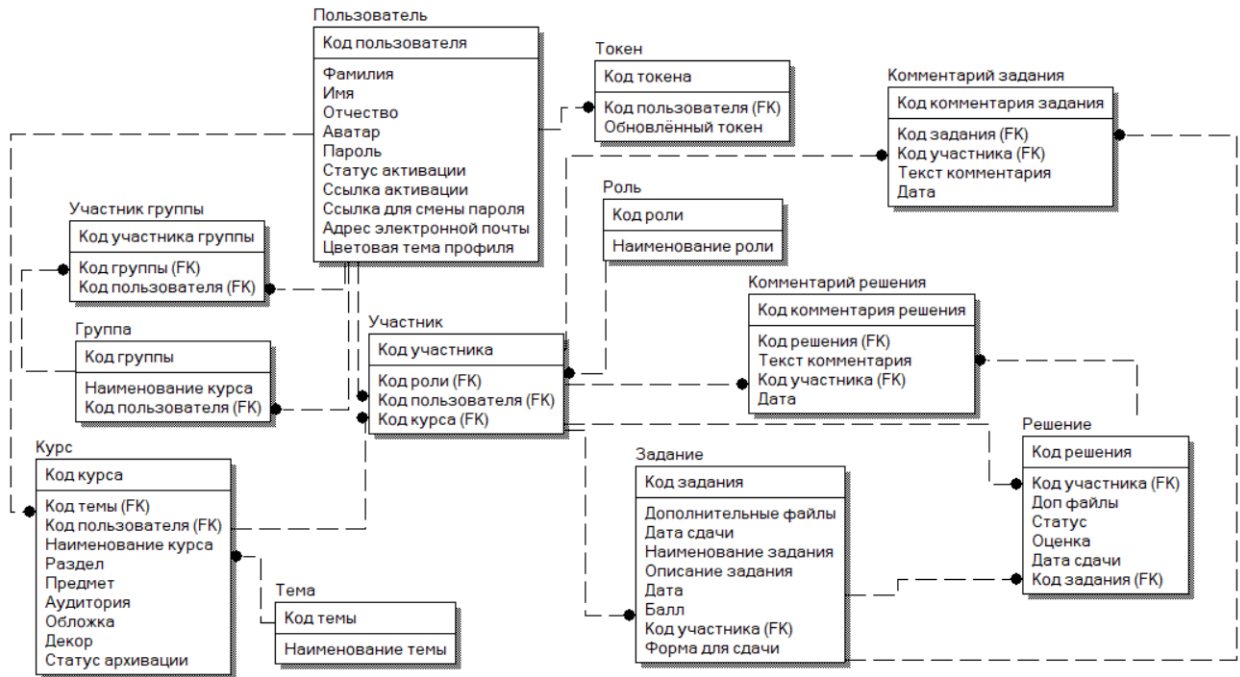


Рисунок 3 – Логическая модель данных

### 2.3.3. Физическая модель данных

На рисунке 4 представлена физическая модель данных.

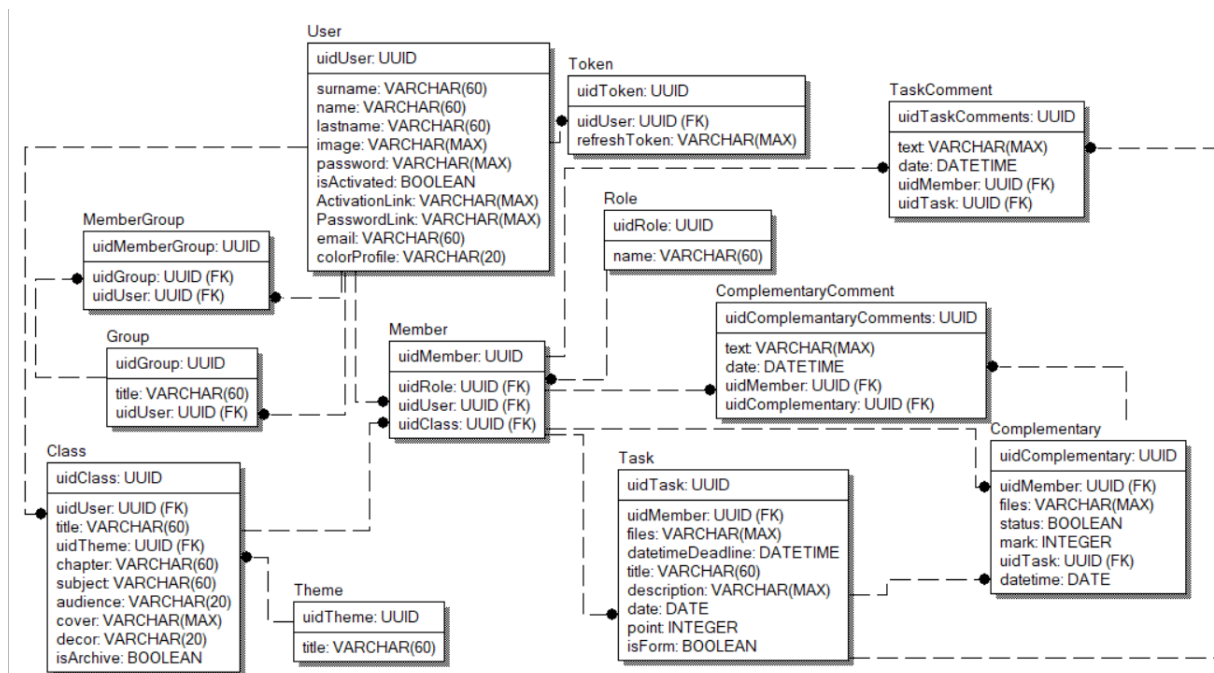


Рисунок 4 – Физическая модель данных

В Таблице 7 представлен словарь данных, использованные при создании базы данных.

Таблица 7 – Словарь данных

Ключ	Поле	Тип данных	Обязательное	Описание
1	2	3	4	5
Role				
PK	uidRole	UUID	Да	Первичный ключ таблицы роль
	Name	VARCHAR (60)	Да	Наименование роли
Token				
PK	uidToken	UUID	Да	Первичный ключ таблицу токен
	refreshToken	VARCHAR (MAX)	Да	Обновляемый токен
FK	uidUser	UUID	Да	Внешний ключ, ссылающийся на таблицу пользователь
User				
PK	uidUser	UUID	Да	Первичный ключ таблицы пользователя
	Surname	VARCHAR (60)	Да	Фамилия пользователя
	Name	VARCHAR (60)	Да	Имя пользователя
	Lastname	VARCHAR (60)	Нет	Отчество пользователя
	Password	VARCHAR (MAX)	Да	Пароль
	Email	VARCHAR (60)	Да	Адрес электронной почты
	isActivated	BOOLEAN	Да	Указывает активирован аккаунт или нет
	PasswordLink	VARCHAR (MAX)	Нет	Ссылка для сброса пароля
	ActivationLink	VARCHAR (MAX)	Да	Ссылка для активации пароля
	Image	VARCHAR (MAX)	Нет	Аватар пользователя
	ColorProfile	VARCHAR (20)	Да	Цветовая тема профиля
Class				
PK	uidClass	UUID	Да	Первичный ключ таблицы курс
	Title	VARCHAR (60)	Да	Наименование курса
	Chapter	VARCHAR (60)	Да	Раздел курса



Ключ	Поле	Тип данных	Обязательное	Описание
1	2	3	4	5
	Subject	VARCHAR (60)	Да	Предмет
	Audience	VARCHAR (60)	Нет	Аудитория
	Cover	VARCHAR (MAX)	Нет	Обложка курса
	Decor	VARCHAR (20)	Да	Цветовая стилистика курса
	isArchive	BOOLEAN	Да	Означает находиться ли курс в архиве
FK	uidTheme	UUID	Да	Внешний ключ, ссылающийся на таблицу темы
FK	uidUser	UUID	Да	Внешний ключ, ссылающийся на таблицу пользователя, обозначающий автора курса
Member				
PK	uidMember	UUID	Да	Первичный ключ таблицы участник
FK	uidUser	UUID	Да	Внешний ключ, ссылающийся на таблицу пользователя
FK	uidClass	UUID	Да	Внешний ключ, ссылающийся на таблицу курса
FK	uidRole	UUID	Да	Внешний ключ, ссылающийся на таблицу роли
Task				
PK	uidTask	UUID	Да	Первичный ключ таблицы задание
	Title	VARCHAR (60)	Да	Наименование задания
	Description	VARCHAR (MAX)	Да	Описание задания
	File	VARCHAR (MAX)	Нет	Дополнительные файлы
	DeadlineDateTime	DATETIME	Нет	Дата и время сдачи
	Date	DATE	Да	Дата выгрузки задания
	Point	INTEGER	Нет	Максимальный балл, который

Ключ	Поле	Тип данных	Обязательное	Описание
1	2	3	4	5
				можно получить за это задание
	isForm	BOOLEAN	Да	Указывает есть ли форма для сдачи
FK	uidMember	UUID	Да	Внешний ключ, ссылающийся на таблицу участника
Complementary				
PK	uidComplementary	UUID	Да	Первичный ключ таблицы решение
	File	VARCHAR (MAX)	Нет	Файл с решением
	Status	BOOLEAN	Да	Статус решения
	Mark	INTEGER	Нет	Оценка
	DateTime	DATETIME	Нет	Дата и время сдачи
FK	uidTask	UUID	Да	Внешний ключ, ссылающийся на таблицу задания
FK	uidMember	UUID	Да	Внешний ключ, ссылающийся на таблицу участник
Group				
PK	uidGroup	UUID	Да	Первичный ключ таблицы группа
	Title	VARCHAR (60)	Да	Название группы
FK	uidUser	UUID	Да	Внешний ключ, ссылающийся на таблицу пользователя
Theme				
PK	uidTheme	UUID	Да	Первичный ключ таблицы тема
	Title	VARCHAR (60)	Да	Название темы
MemberGroup				
PK	uidMemberGroup	UUID	Да	Первичный ключ таблицы участник группы
FK	uidUser	UUID	Да	Внешний ключ, ссылающийся на таблицу пользователя

Ключ	Поле	Тип данных	Обязательное	Описание
1	2	3	4	5
FK	uidGroup	UUID	Да	Внешний ключ, ссылающийся на таблицу группы
TaskComment				
PK	uidTaskComment	UUID	Да	Первичный ключ таблицы комментариев задания
	Text	VARCHAR (MAX)	Да	Текст комментария
	Date	DATETIME	Да	Дата отправки комментария
FK	uidMember	UUID	Да	Внешний ключ, ссылающийся на таблицу пользователя, который отправил сообщение
FK	uidTask	UUID	Да	Внешний ключ, ссылающийся на таблицу задания, под которым оставили сообщение
ComplementaryComment				
PK	uidComplementaryComment	UUID	Да	Первичный ключ таблицы комментариев решения
	Text	VARCHAR (MAX)	Да	Текст комментария
	Date	DATETIME	Да	Дата отправки комментария
FK	uidMember	UUID	Да	Внешний ключ, ссылающийся на таблицу пользователя, который отправил сообщение
FK	uidComplementary	UUID	Да	Внешний ключ, ссылающийся на таблицу решение, под которым оставили сообщение

### 2.3.4. Структурная схема программы

На Рисунке 5 представлена структурная схема программы. Основными классами интерфейса являются Account, Classroom, Home, Login, Registration, Task, ChangePassword, ResetPassword, Archive. Дочерние классы от них представлены ниже по иерархии. Класс App взаимодействует между всеми вышеперечисленными классами и косвенно их соединяет. Классы config соединяют всю программу с базой данных.

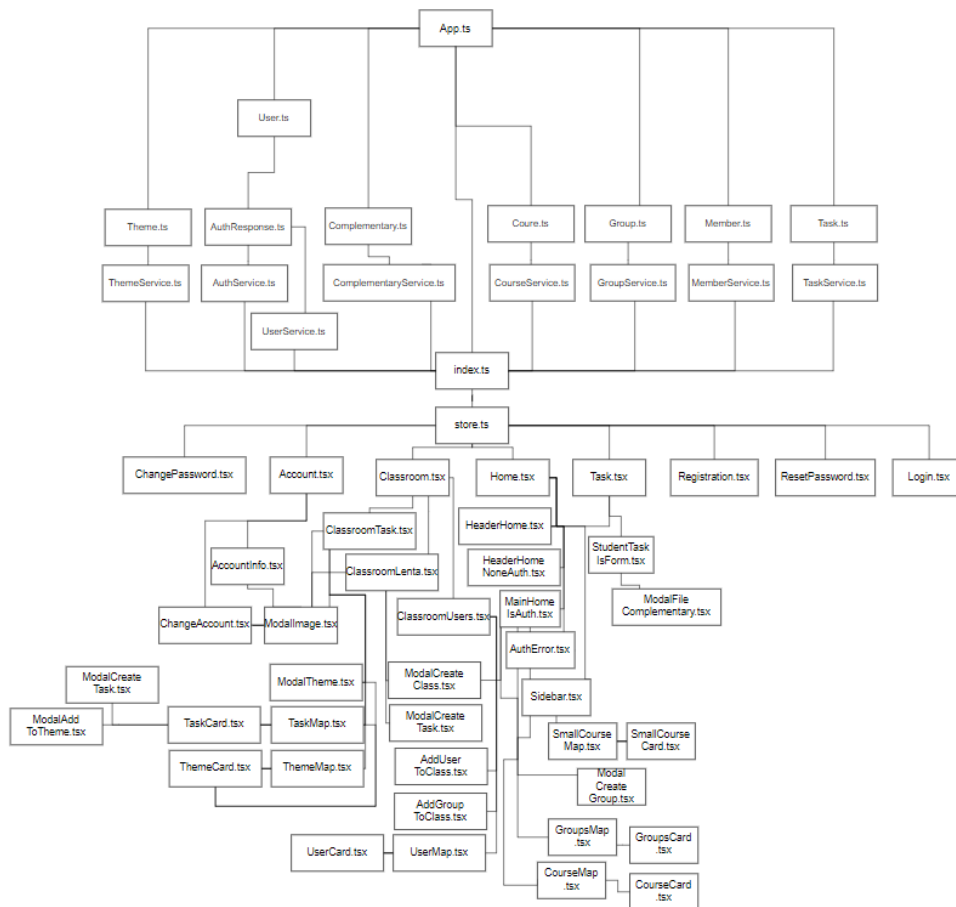


Рисунок 5 – Структурная схема

В Таблице 8 описаны модули указанные в структурной схеме программы.

Таблица 8 – Описание модулей

№	Название модуля	Описание модуля
1	2	3
1	App.ts	Является роутингом для всего WEB-приложения
2	Store.ts	Объект, который хранит дерево состояний приложения
3	Task.tsx	Одиночная страница с заданием
4	Home.tsx	Домашняя страница

№	Название модуля	Описание модуля
1	2	3
5	Account.tsx	Страница аккаунта пользователя
6	Login.tsx	Страница авторизации
7	Registration.tsx	Страница регистрации
8	ResetPassword.tsx	Страница для сброса пароля по email
9	ChangePassword.tsx	Страница смены пароля
10	HeaderHome.tsx	Заголовок для домашней страницы
11	HeaderHomeNoneAuth.tsx	Заголовок для домашней страницы не авторизованного пользователя
12	Classroom.tsx	Страница курса
13	ClassroomLenta.tsx	Вывод ленты курса
14	ClassroomUsers.tsx	Вывод списка пользователей курса
15	ClassroomTask.tsx	Вывод списка заданий курса
16	AccountInfo.tsx	Вывод информации об аккаунте пользователя
17	ChangeAccount.tsx	Компонент с изменением информации о пользователе
18	Index.ts	Компонент, связывающий frontend и backend
19	User.ts	Модель с данными о пользователе
20	AuthResponse.ts	Модель, содержащая данные для авторизации пользователя
21	Theme.ts	Модель с данными о теме
22	Complementary.ts	Модель, содержащая информацию о решении пользователя
23	Course.ts	Модель с данными о курсе
24	Group.ts	Модель с данными о группе
25	Member.ts	Модель с данными об участниках групп и курсов
26	Task.ts	Модель с данными о задании
27	ThemeService.ts	Файл, содержащий логику и функциональность, связанную с темой задания в приложении
28	AuthService.ts	Файл, содержащий логику и функциональность, связанную с авторизацией в приложении
29	UserService.ts	Файл, содержащий логику и функциональность, связанную с пользовательским сервисом в приложении
30	ComplementaryService.ts	Файл, содержащий логику и функциональность, связанную с решением задания в приложении
31	CourseService.ts	Файл, содержащий логику и функциональность, связанную с курсом в приложении
32	GroupService.ts	Файл, содержащий логику и функциональность, связанную с группой в приложении
33	MemberService.ts	Файл, содержащий логику и функциональность, связанную с участниками курсов и групп в приложении
34	TaskService.ts	Файл, содержащий логику и функциональность, связанную с заданием в приложении
35	ModalImage.tsx	Модальное окно, которое помогает загружать картинки в приложение
36	MainHomeIsAuth.tsx	Страница «Главная» для авторизованного пользователя, у которого есть курсы или группы

№	Название модуля	Описание модуля
1	2	3
37	AuthError.tsx	Страница «Главная» если пользователь не авторизован или у него нет добавленных курсов и групп
38	StudentTaskIsForm.tsx	Страница задания с формой для сдачи
39	ModalFileComplementary.tsx	Модальное окно, позволяющее добавить файл с решением пользователя
40	Sidebar.tsx	Боковая панель, содержащая все курсы, в которых состоит пользователь, а также ссылку на архивированные курсы, в которых пользователь состоял
41	ModalCreateClass.tsx	Модальное окно для создания и изменения курса
42	ModalCreateTask.tsx	Модальное окно для создания и изменения задания
43	ModalAddUserToClass.tsx	Модальное окно с пользователями в системе, которых можно добавить на курс
44	ModalAddGroupToClass.tsx	Модальное окно с группами, которые создал авторизованный пользователь, которых можно добавить на курс
45	UserMap.tsx	Список пользователей
46	UserCard.tsx	Карточки, содержащие информацию о пользователях
47	ModalTheme.tsx	Модальное окно для создания и изменения темы заданий
48	TaskMap.tsx	Список заданий
49	TaskCard.tsx	Карточка, содержащая информацию о задании
50	ModalCreateTask.tsx	Модальное окно для создания и изменения задания
60	ModalAddToTheme.tsx	Модальное окно, позволяющее добавить задание в существующую тему
61	ThemeMap.tsx	Список тем
62	ThemeCard.tsx	Карточка, содержащая информацию о теме и о заданиях, добавленных в неё
63	SmallCourseMap.tsx	Список курсов, которые будут выводиться в боковой панели
64	SmallCourseCard.tsx	Карточка с информацией о курсе, которая выводиться в боковой панели
65	ModalCreateGroup.tsx	Модальное окно для создания и изменения группы
66	GroupMap.tsx	Список групп
67	GroupCard.tsx	Карточка группы, содержащая информацию о группе
68	CourseMap.tsx	Список курсов
69	CourseGroup.tsx	Карточка, содержащая всю информацию о курсе

### 2.3.5. Функциональная схема

На Рисунке 6 представлена функциональная схема программы, показывающая какие действия и функции осуществляются в приложении.

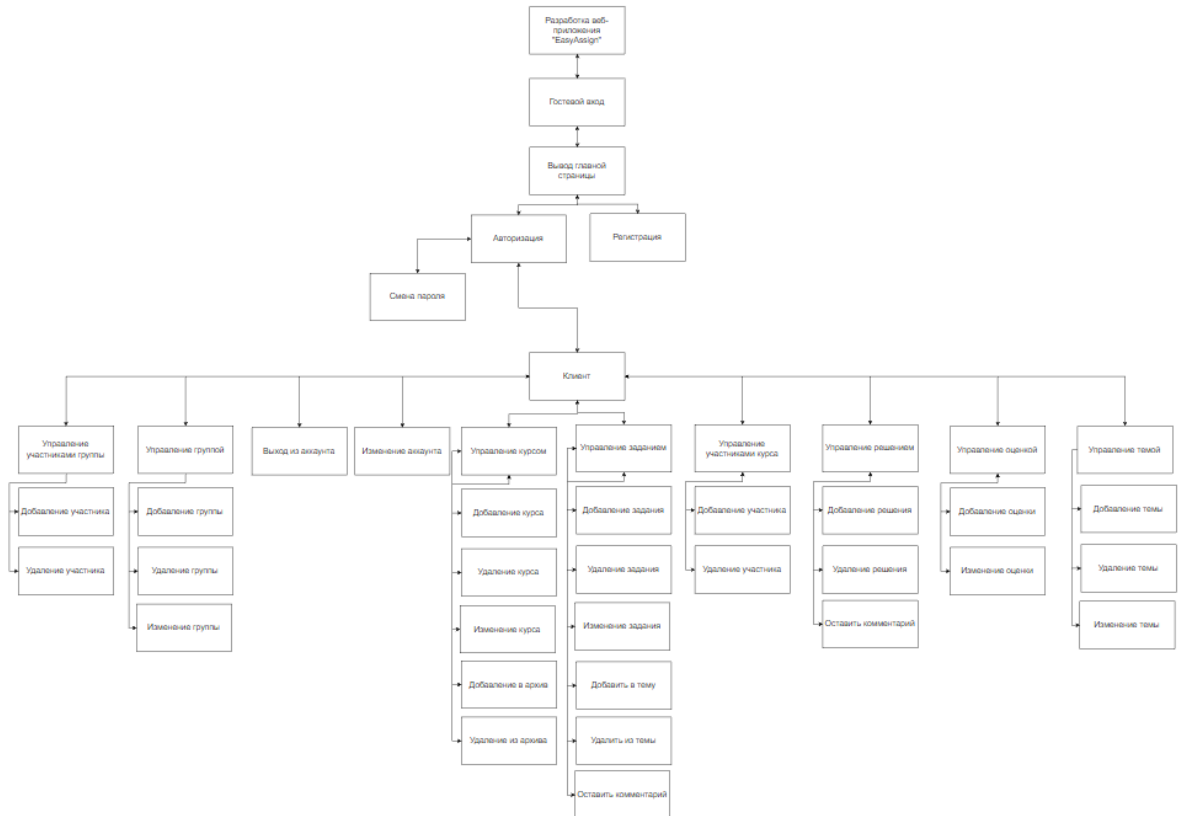


Рисунок 6 – Функциональная схема

### 2.3.6. Диаграмма классов

Диаграмма классов представлена в документе в Приложение Е. Диаграмма классов.

### 2.3.7. Схема пользовательского интерфейса

На Рисунке 7 представлена схема пользовательского интерфейса программы, показывающие все страницы, представленные пользователю.



Рисунок 7 – Схема пользовательского интерфейса

## 2.4. Результат работы программы

В результате выполнения поставленной задачи, было разработано требуемое программное решение, которое является удобным видеосервисом для просмотра аниме.

Данное решение является клиент-серверным.

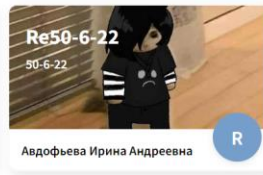
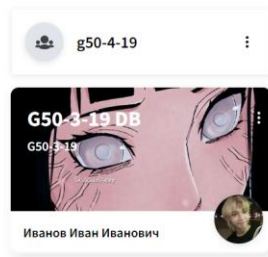
Информационная система состоит из трех основных частей, а именно:


- Из реляционной базы данных, которая размещается на сервере, рассмотреть состав и структуру которой можно в пункте 2.3.2;
- Из информационной системы, являющийся веб-ресурсом;
- Из WEB-API, размещаемого на сервере.

Узнать о взаимодействии пользователя с информационной системой можно из Приложения В. Руководство пользователя.

На рисунках 8 – 9 представлена первоначальная страница, открываемая при запуске сайта.





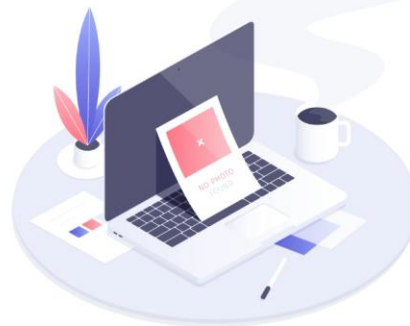
Обучение с  EasyAssign становится более интересным и увлекательным, а также более эффективным и доступным для всех ;)




  
 ©EasyAssign2023

Рисунок 8 – Начальная страница для авторизованного пользователя

**Ни одного курса**



Обучение с  EasyAssign становится более интересным и увлекательным, а также более эффективным и доступным для всех ;)



  
 ©EasyAssign2023

Рисунок 9 – Начальная страница для неавторизованного пользователя

### 3. ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ

#### 3.1. Инструментальная среда разработки

Для разработки примера была использована облачная реляционная база данных PostgreSQL. Преимущественно, она была выбрана за счёт поддержки БД неограниченного размера, наследования и лёгкости расширяемости.

За основной текстовый редактор был взят Microsoft Word 2016 года, являющийся лучшим текстовым редактором для создания документов. Word позволяет просто и быстро создавать, редактировать и систематизировать документы, а также имеет функцию печати и просмотра, которыми может воспользоваться каждый пользователь.

За инструментальное средство разработки программных решений было взято Microsoft Visual Studio Code, являющееся лучшим решение для написания вэб приложений за счёт своего удобного интерфейса и простоты использования.

#### 3.2. Отладка программы

Отладка программы выполнена за счёт встроенных средств отладки Visual Studio Code. Отладка работает при включённом приложении и останавливается, доходя до точки останова. Приложение тестируется по методике белого ящика.

При отладке программа позволяет проверить данные из приложения, выводя их в консоль.

При незнании способа или решения исправления ошибки, которая возникла при разработке новостного портала, использовались дополнительная информация, взятая из специализированной литературы, профильных сайтов и официальных мануалов. С открытыми источниками и литературой можно ознакомиться в разделе Список используемых материалов.

#### 3.3. Защитное программирование

В коде предусмотрена защита от ошибок, путём установки условий, циклов и проверок на пустоту с помощью ненулевых операторов, представленных на Рисунке 10.

```
if(id.id)
```

Рисунок 10 – Защита от пустоты

Также ошибки могут возникнуть при использовании ссылочных данных на интерфейс. Для таких случаев используется защита в виде поиска объекта по его ID. Пример представлен на Рисунке 11.

```
const classData = await prisma.class.findFirst({
  where: {
    id: class_id
  }
})
```

Рисунок 11 – Защита от пустых ссылочных типов

### 3.3.2. Защита приложения

Приложение обеспечивает защиту от пустого ввода текста, пример представлен на Рисунке 12.

```
let codeString = ''
if(task){
  codeString = decode(extractCode(task.description)); // Decode the HTML entities
}
```

Рисунок 12 – Защита от пустого ввода текста

Также приложение обеспечивает защиту от пустых файлов.

```

{selectedFile}&&( //проверка на то пуст ли файл
  <div style={{width:'100%'}}>
    {selectedFile.map((t:any)=>{
      <div className="body-file"
        style={{display:'flex',
          border:'2px solid #AEB5C8',
          marginTop:'10px',
          width:'100%',
          borderRadius:'5px',
          background:'#AEB5C8',
          color:'white'}}>
        <AiOutlineFileText className='file' style={{fontSize:'20px',
          marginTop:'16px', marginRight:'10px'}}/>
        <p style={{width:'90%'}}>{t.name}</p>
        <RxCross2 className='cross'
          style={{marginTop:'20px'}}
          onClick={()=>removeItemByName(t.name)}>
      </div>
    })}
  </div>
)}

```

Рисунок 13 – Защита от пустых файлов

### 3.3.3. Защита данных

Защита также обеспечивается и для базы данных. Она представлена в виде вывода ошибок из системы, если таковые присутствуют.

```

if(store.error){
  toast.error(store.error, {
    position: "top-right",
    autoClose: 5000,
    hideProgressBar: false,
    closeOnClick: true,
    pauseOnHover: true,
    draggable: true,
    progress: undefined,
    theme: "light",
  });
}

```

Рисунок 14 – Вывод ошибки из базы данных

Также, чтобы защитить базу данных от взломов, используется специальный .env файл, не отображающийся нигде.

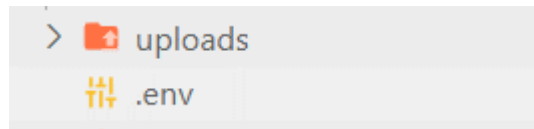


Рисунок 15 – Файл шифрования ключей базы данных

На Рисунке 16 показан способ как можно вытащить ключи из .env файла.

```
generateTokens(payload:any)(){
  let accessToken: any
  let refreshToken: any
  if(process.env.JWT_ACCESS_SECRET&&process.env.JWT_REFRESH_SECRET){
    accessToken= jwt.sign(payload,process.env.JWT_ACCESS_SECRET, {expiresIn:'30s'})
    refreshToken = jwt.sign(payload, process.env.JWT_REFRESH_SECRET, {expiresIn:'30d'})
  }
  return {
    accessToken,
    refreshToken
  }
}
```

Рисунок 16 – Способ получения ключей из зашифрованного файла

### 3.4. Характеристика программы

С характеристиками программы можно ознакомиться в приложении А. Текст Программы.

## ЗАКЛЮЧЕНИЕ

В ходе выполнения данной дипломной работы была разработана и реализована WEB-платформа для онлайн-обучения, сфокусированная на потребностях студентов ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова». Основной целью работы было создание удобного и эффективного инструмента, который позволит студентам получать образование в онлайн-формате, расширяя возможности дистанционного обучения.

В процессе разработки платформы были изучены современные технологии и подходы в области онлайн-обучения, а также проведен анализ потребностей и требований студентов ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова». Были проведены исследования существующих WEB-платформ для обучения и анализ их функциональности, чтобы определить основные характеристики и требования к нашей платформе.

В результате работы была разработана и реализована функциональная WEB-платформа, которая обладает следующими основными возможностями: доступ к учебным материалам, система оценки и обратной связи, форум для общения и сотрудничества студентов. Платформа была разработана с учетом современных стандартов и принципов дизайна, обеспечивая удобный интерфейс и интуитивно понятную навигацию для пользователей.

Основные преимущества разработанной WEB-платформы заключаются в возможности гибкого и доступного обучения для студентов, независимо от места и времени, а также обеспечении взаимодействия и обратной связи между преподавателями и студентами. Разработанная платформа имеет потенциал для дальнейшего расширения и улучшения, включая добавление новых функциональных возможностей и интеграцию с современными технологиями.

В заключение, разработка WEB-платформы для онлайн-обучения на примере ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова» представляет собой важный шаг в развитии образовательного

процесса и адаптации к современным технологиям. Созданная платформа предоставляет студентам удобное и эффективное средство для получения знаний и развития навыков в онлайн-формате. Дальнейшее развитие и улучшение платформы могут способствовать улучшению качества образования и расширению возможностей для обучения на ФГБОУ ВО «Российский экономический университет им. Г.В. Плеханова» и в других образовательных учреждениях.

## СПИСОК ИСПОЛЬЗУЕМЫХ МАТЕРИАЛОВ

1. ГОСТ Р – 7.0.5-2008 Система стандартов по информации, библиотечному и издательскому делу.
2. ГОСТ 19.101-77 Единая система программной документации виды программ и программных документов.
3. Дэвид Флэнаган. "JavaScript: The Definitive Guide". O'Reilly Media, 2020 (Дата обращения: 10.01.2023).
4. Эрик Эванс. "Domain-Driven Design: Tackling Complexity in the Heart of Software". Addison-Wesley Professional, 2003 (Дата обращения: 17.03.2023).
5. Кайл Симпсон. "You Don't Know JS" (серия книг). O'Reilly Media.
6. Энтони Алиджери. "React и Redux: функциональная веб-разработка". ООО "ДМК Пресс", 2017 (Дата обращения: 20.01.2023).
7. Дэн Абрамов, Андрей Хорев. "Программирование React". ООО "ДМК Пресс", 2020 (Дата обращения: 18.02.2023).
8. Сандерсон Джейсон. "React и TypeScript. Разработка компонентов для современных пользовательских интерфейсов". Питер, 2021 (Дата обращения: 27.01.2023).
9. Марк Фостер, Люк Гибсон. "Node.js в действии". ДМК Пресс, 2019 (Дата обращения: 24.12.2022).
10. Майкл Хартл. "Express в действии: веб-разработка на Node.js с использованием Express и Jade". ДМК Пресс, 2016 (Дата обращения: 02.01.2023).
11. Дэвид Ханеман, Ремо Яблонски. "TypeScript в действии". ДМК Пресс, 2020 (Дата обращения: 05.02.2023).
12. Документация Supabase: <https://supabase.io/docs/> (Дата обращения: 15.01.2023).
13. Документация Prisma: <https://www.prisma.io/docs/> (Дата обращения: 26.05.2023).



14. Документация Express.js: <https://expressjs.com/> (Дата обращения: 12.02.2023).
15. Документация React: <https://reactjs.org/> (Дата обращения: 10.01.2023).
16. Документация TypeScript: <https://www.typescriptlang.org/> (Дата обращения: 11.01.2023).