

Министерство просвещения Республики Казахстан  
УО «Центральноазиатский технико-экономический колледж»

КУРСОВОЙ ПРОЕКТ  
Secondhand Book Marketplace «ReRead»  
ПОЯСНИТЕЛЬНАЯ ЗАПИСКА  
РК ЦАТЭК 4S06130103 КП ПЗ

Руководитель  
Нехорошев В.Д.  
«   » 2024 года

Выполнил  
И.В Федоров  
«   » 2024 года

Защита курсового проекта  
«   » 2024 года

Полученный балл

---

Алматы 2024

# ЦЕНТРАЛЬНОАЗИАТСКИЙ ТЕХНИКО-ЭКОНОМИЧЕСКИЙ КОЛЛЕДЖ

УТВЕРЖДАЮ

Заместитель директора по  
учебной работе

\_\_\_\_\_ Жаксыбаева Н.Н.

\_\_\_\_\_ 2024 года

## ЗАДАНИЕ ДЛЯ КУРСОВОГО ПРОЕКТИРОВАНИЯ

По дисциплине ПМ2 Составление алгоритма и создание блок-схемы на основе спецификации программного обеспечения

Студента 3 курса, группы ПЗА

УО «Центральноазиатского технико-экономического колледжа»

\_\_\_\_\_ Федоров Илья Владимирович \_\_\_\_\_  
(фамилия, имя, отчество)

Тема проекта \_\_\_\_\_  
Secondhand Book Marketplace «ReRead»

При выполнении курсового проекта на указанную тему должны быть представлены:

1. Пояснительная записка
  - 1) Общая часть
  - 2) Специальная часть
  - 3) Организация производства
2. Приложения
  - 1) Листинг программы
  - 2) Входные формы комплекса
  - 3) Выходные формы комплекса

Дата выдачи 10 апреля 2024 года

Срок окончания «\_\_\_\_\_» \_\_\_\_\_ 2024 года

Преподаватель – руководитель курсового проекта  
Нехорошев Владимир Дмитриевич

# ЦЕНТРАЛЬНОАЗИАТСКИЙ ТЕХНИКО-ЭКОНОМИЧЕСКИЙ КОЛЛЕДЖ

График работы \_\_\_\_\_ Федорова Ильи Владимировича \_\_\_\_\_

## КУРСОВОЕ ПРОЕКТИРОВАНИЕ

с 10 апреля 2024 г. по 14 мая 2024 г.

№ п/п	Раздел проекта	Дата выполнения	Подпись руководителя
1.	Общая часть, введение	13.04	
2.	Специальная часть	20.04	
3.	Организация производства	28.04	
4.	Заключение, список литературы	10.05	
5.	Приложения	12.05	

Срок защиты «\_\_\_» \_\_\_\_\_ 2024 г.

Объем выполнения

Курсового проекта: на 28 апреля 2024 г. 70%  
на 14 мая 2024г. 100%

Руководитель  
курсового проекта  
\_\_\_\_\_ Нехорошев В.Д.

# СОДЕРЖАНИЕ

ВВЕДЕНИЕ	стр. 8
1 ОБЩАЯ ЧАСТЬ	9
1.1 Актуальность проекта	9
1.2 Описание языка программирования	11
1.2.1 Общие сведения	11
1.2.2 Особенности языка	12
1.2.2.1 Типы данных	12
1.2.2.2 Агрегаты данных	13
1.2.2.3 Программирование разветвляющихся и циклических процессов	13
2 СПЕЦИАЛЬНАЯ ЧАСТЬ	14
2.1 Общая постановка задачи	14
2.1.1 Описание входных данных и выходных данных	15
2.1.2 Схема работы комплекса	16
2.1.3 Описание комплекса программ	17
2.1.4 Описание набора данных	17
2.3 Описание проблемных программ	18
2.3.1 Описание проблемной программы №1	18
2.3.1.1 Схема алгоритма проблемной программы №1	19
2.3.1.2 Таблица идентификаторов проблемной программы №1	19
2.3.2 Описание проблемной программы №2	20
2.3.1.1 Схема алгоритма проблемной программы №2	20
2.3.1.2 Таблица идентификаторов проблемной программы №2	20
3 ОРГАНИЗАЦИЯ ПРОИЗВОДСТВА	21
3.1 Условия выполнения проекта	21
3.2 Инструкция пользователю	21
3.3 Формы входных и выходных документов	22
3.4 Структура записи файлов	22
ЗАКЛЮЧЕНИЕ	23
СПИСОК ЛИТЕРАТУРЫ	24
Приложения	20
Приложение А – Листинг программы	21
Приложение Б – Входные данные	26
Приложение В – Выходные данные	27
Приложение Г – Схема работы комплекса	28
Приложение Д - Схема взаимодействия таблиц базы данных	29

					РК ЦАТЭК 4S0613010КП ПЗ									
Изм	Лис	№ докум	Подпи	Дата	Secondhand Book Marketplace “ReRead”					Лист		Лист	Листов	
Разраб.	Федоров И.В											4		
Провер.	Нехорошее В.Д.									ЦАТЭК гр.ПЗА				
Утверд.	Нехорошее В.Д.													

## ВВЕДЕНИЕ

В эпоху цифровой трансформации, стремительный рост интернет-технологий оказывает значительное влияние на все сферы жизни, включая торговлю. Онлайн-рынки становятся всё более популярными, позволяя пользователям не только покупать новые товары, но и обмениваться поддержанными. Одним из актуальных направлений в данной области является создание платформ для продажи и обмена поддержанных книг. Эти платформы предлагают удобный способ для людей найти нужную литературу по доступным ценам, а также избавляться от ненужных книг, предоставляя им вторую жизнь.

Данная курсовая работа посвящена разработке и реализации веб-приложения "ReRead" – маркетплейса поддержанных книг. Проект ReRead призван создать удобную и эффективную платформу для взаимодействия пользователей, желающих покупать, продавать или обменивать поддержанные книги. Основными целями проекта являются:

Создание интуитивно понятного пользовательского интерфейса, обеспечивающего удобный поиск и просмотр объявлений.

Реализация эффективной системы управления объявлениями, позволяющей пользователям легко добавлять, редактировать и удалять свои объявления.

Внедрение функционала обмена сообщениями, обеспечивающего быструю и безопасную коммуникацию между покупателями и продавцами.

Разработка механизмов безопасности, включая защиту данных пользователей и предотвращение мошенничества.

Обеспечение возможности многопользовательского взаимодействия, поддерживающего различные роли пользователей, такие как покупатели и продавцы.

В процессе разработки были использованы современные технологии веб-разработки, включая Django для серверной части и Tailwind CSS для клиентской. Для хранения и управления данными была выбрана реляционная база данных, а для обеспечения безопасности – механизмы аутентификации и авторизации пользователей.

Введение описывает общую концепцию проекта и его значимость в современном мире. В следующих разделах курсовой работы будут подробно рассмотрены все этапы разработки, начиная от анализа требований и проектирования системы, до её реализации и тестирования. Особое внимание будет уделено архитектуре приложения, выбору технологий и инструментов, а также практическим аспектам их применения. Работа завершится обсуждением результатов, выводами и предложениями по дальнейшему развитию проекта.

# 1 ОБЩАЯ ЧАСТЬ

## 1.1 Актуальность проекта

В современном мире, где информация и знания становятся всё более доступными, книги продолжают оставаться важнейшим источником информации, вдохновения и культурного наследия. Однако, несмотря на развитие цифровых технологий и увеличение доли электронных книг, физические издания всё ещё пользуются значительным спросом среди читателей. Особенно это касается специализированной литературы, редких изданий и книг, которые могут быть недоступны в электронном формате.

Тем не менее, покупка новых книг может быть довольно затратной, особенно для студентов, молодых специалистов и людей с ограниченным бюджетом. В то же время у многих людей накапливается большое количество прочитанных книг, которые занимают место и не используются. Таким образом, создание платформы для продажи и обмена подержанными книгами становится крайне актуальным.

Основные факторы, подчеркивающие актуальность проекта ReRead, включают:

**Экономическая эффективность:** Покупка подержанных книг позволяет существенно сократить расходы на литературу. Маркетплейс ReRead предлагает пользователям возможность приобретать книги по значительно более низким ценам, чем в традиционных книжных магазинах.

**Экологическая значимость:** Повторное использование книг способствует сокращению количества бумажных отходов и уменьшению вырубки лесов. Проект ReRead способствует экологической устойчивости, поощряя повторное использование ресурсов.

**Доступность знаний:** Платформа предоставляет широкий доступ к разнообразной литературе, включая редкие и уникальные издания, которые могут быть недоступны в обычных магазинах или библиотеках. Это способствует распространению знаний и культуры.

**Социальное взаимодействие:** Проект ReRead стимулирует взаимодействие между пользователями, создавая сообщество книголюбов. Возможность обмена и продажи книг способствует формированию культурного обмена и социальной сплоченности.

**Технологическое развитие:** Разработка и внедрение современных веб-технологий для реализации платформы демонстрирует важность и потенциал цифровых инноваций в сфере электронной коммерции и обмена информацией.

Проект ReRead отвечает на актуальные вызовы и потребности общества, предлагая инновационное решение для эффективного и удобного обмена подержанными книгами. Внедрение такой платформы способствует экономической, экологической и социальной устойчивости, делая знания более доступными для всех слоев населения.

## 1.2 Описание языка программирования

### 1.2.1 Общие сведения

Проект ReRead разработан с использованием нескольких ключевых технологий, каждая из которых играет важную роль в создании функциональной и удобной платформы для обмена и продажи подержанных книг. В этом разделе мы рассмотрим основные аспекты Python, SQLite3, Tailwind CSS, Django, Django Rosetta, а также основы объектно-ориентированного программирования (ООП) и баз данных.

Python — это высокоуровневый язык программирования общего назначения, который был создан Гвидо ван Россумом и впервые выпущен в 1991 году. Он известен своей простой и понятной синтаксической структурой, что делает его доступным для начинающих программистов, и в то же время мощным инструментом для профессионалов. Python поддерживает множество парадигм программирования, включая объектно-ориентированное, императивное, функциональное и процедурное программирование.

#### Ключевые особенности

**Простота и читабельность:** Код на Python легко читать и писать благодаря его лаконичному синтаксису.

**Богатая стандартная библиотека:** Python предоставляет широкий спектр модулей и пакетов для выполнения разнообразных задач.

**Кроссплатформенность:** Программы на Python могут работать на различных операционных системах без необходимости значительных изменений в коде.

**Расширяемость:** Python легко интегрируется с другими языками программирования и может использоваться для создания расширений на C или C++.

**Сообщество и поддержка:** Широкое и активное сообщество разработчиков способствует постоянному улучшению и развитию языка.

SQLite3 — это легковесная реляционная система управления базами данных (СУБД), которая реализована в виде библиотеки и встраивается непосредственно в приложения. Она известна своей простотой и высокой производительностью, что делает её идеальной для мобильных и настольных приложений.

#### Ключевые особенности

**Встраиваемость:** SQLite3 не требует отдельного сервера для работы; база данных хранится в одном файле.

**Малый размер:** Библиотека SQLite3 занимает всего несколько сотен килобайт.

**Полное соответствие стандартам SQL:** Поддержка большинства функций SQL делает SQLite3 мощным инструментом для работы с данными.

**Транзакции и целостность данных:** SQLite3 поддерживает транзакции с атомарностью, целостностью, изоляцией и долговечностью (ACID).

Tailwind CSS — это утилитарный CSS-фреймворк, который позволяет быстро и легко создавать адаптивные и современные пользовательские интерфейсы. В отличие от традиционных CSS-фреймворков, Tailwind CSS предлагает готовые классы, которые можно комбинировать для создания уникального дизайна.

Ключевые особенности

Утилитарный подход: Большое количество предопределённых классов для управления стилями.

Высокая производительность: Снижение количества кода и улучшение производительности за счёт минимизации CSS.

Гибкость: Легко настраиваемый конфигурационный файл позволяет изменять дизайн под конкретные нужды проекта.

Модульность: Комбинируемые классы позволяют создавать сложные интерфейсы без необходимости написания дополнительного CSS-кода.

Django — это высокоуровневый веб-фреймворк на языке Python, предназначенный для ускорения разработки и упрощения создания сложных веб-приложений. Он следует архитектурному шаблону Model-View-Controller (MVC), который в Django реализован как Model-View-Template (MVT).

Ключевые особенности

Быстрая разработка: Набор встроенных инструментов и библиотек позволяет быстро создавать и развертывать веб-приложения.

Безопасность: Django включает в себя средства для защиты от распространённых веб-угроз, таких как SQL-инъекции и межсайтовый скриптинг (XSS).

Масштабируемость: Способность обрабатывать большое количество запросов и легко масштабироваться.

Поддержка ORM: Встроенная система Object-Relational Mapping (ORM) для работы с базами данных, которая упрощает взаимодействие между Python-кодом и реляционными базами данных.

Django Rosetta — это приложение для перевода, которое интегрируется с Django и упрощает процесс локализации и интернационализации веб-приложений. Оно предоставляет удобный интерфейс для работы с переводами и позволяет разработчикам и переводчикам легко управлять файлами перевода.

Ключевые особенности

Удобный интерфейс: Веб-интерфейс для добавления и редактирования переводов.

Поддержка различных форматов файлов: Поддержка файлов PO и других форматов для хранения переводов.

Интеграция с Django: Легкая интеграция с существующими проектами на Django.

Объектно-ориентированное программирование (ООП) — это парадигма программирования, которая основывается на концепции "объектов", которые могут содержать данные и код для работы с этими данными. ООП является



фундаментальной концепцией для многих современных языков программирования, включая Python.

Классы и объекты: Класс — это шаблон для создания объектов. Объект — это экземпляр класса.

Инкапсуляция: Процесс скрытия внутренних деталей работы объекта и предоставление доступа только через методы.

Наследование: Способность одного класса наследовать атрибуты и методы другого класса.

Полиморфизм: Способность использовать один и тот же интерфейс для разных типов объектов.

База данных (БД) — это организованная коллекция данных, которая хранится и управляется системой управления базами данных (СУБД). Реляционные базы данных (РБД) являются наиболее распространённым типом баз данных и организуют данные в таблицы с установленными связями между ними.

Таблицы: Основные структуры для хранения данных, состоящие из строк и столбцов.

Поля: Колонки таблицы, каждая из которых хранит определённый тип данных.

Записи: Строки таблицы, каждая из которых представляет один экземпляр данных.

Связи: Связи между таблицами, которые могут быть одного из трёх типов: один к одному, один ко многим, многие ко многим.

Ключи: Первичные ключи (PK) уникально идентифицируют записи в таблице, а внешние ключи (FK) используются для установления связей между таблицами.

Использование Python, SQLite3, Tailwind CSS, Django и Django Rosetta в проекте ReRead демонстрирует интеграцию мощных и современных технологий для создания эффективного и удобного веб-приложения. Эти инструменты и концепции обеспечивают высокую производительность, масштабируемость и безопасность, что делает их идеальным выбором для разработки современного маркетплейса для подержанных книг.

## 2 СПЕЦИАЛЬНАЯ ЧАСТЬ

### 2.1 Общая постановка задачи

Целью данного проекта является создание маркетплейса для обмена и продажи подержанных книг под названием ReRead. Этот маркетплейс должен предоставлять пользователям возможность размещать объявления о продаже или обмене книг, общаться с потенциальными покупателями и продавцами, а также управлять своими объявлениями. Проект должен быть интуитивно понятен, безопасен и масштабируем.

#### 2.1.1 Описание входных данных и выходных данных

##### **Входные данные:**

- Регистрация и аутентификация пользователя:
- Логин и пароль пользователя.
- Электронная почта.
- Имя пользователя.
- Данные книги:
- Название книги.
- Автор книги.
- Издательство.
- Язык книги.
- Состояние книги.
- Тип переплета.
- Описание книги.
- Цена книги.
- Изображения книги.
- Статус (продано/доступно).
- Возможность обмена.
- Сообщения и чаты:
- Содержание сообщения.
- Участники чата.

##### **Выходные данные:**

- Информация о книге:
- Название книги.
- Автор книги.
- Издательство.
- Язык книги.
- Состояние книги.

- Тип переплета.
- Описание книги.
- Цена книги.
- Изображения книги.
- Статус (продано/доступно).
- Возможность обмена.
- Информация о пользователе:
- Имя пользователя.
- Электронная почта.
- Логин.
- История сообщений и чатов:
- Содержание сообщений.
- Время отправки.
- Участники чата.

### 2.1.2 Схема работы комплекса

Схема работы приложений представлена в приложении Г.

### 2.1.3 Описание программы

Проект ReRead состоит из нескольких модулей и приложений, обеспечивающих функциональность маркетплейса:

1. Аутентификация и регистрация:
  - Модуль для управления пользователями, включая регистрацию, вход и выход из системы, восстановление пароля и т.д.
2. Управление объявлениями:
  - Модуль для создания, редактирования и удаления объявлений о продаже или обмене книг.
  - Управление изображениями книг.
3. Категории:
  - Модуль для управления категориями книг.
4. Чаты и сообщения:
  - Модуль для создания и управления чатами между пользователями.
  - Возможность отправки сообщений в рамках чатов.
5. Локализация и перевод:
  - Модуль для управления языковыми версиями интерфейса с использованием Django Rosetta.
6. Административный интерфейс:

- Модуль для администраторов сайта для управления пользователями, объявлениями и категориями.

## 2.1.4 Описание набора данных

Для хранения данных проекта используется реляционная база данных SQLite3. Структура базы данных включает следующие таблицы:

1. Пользователи (users):
  - id: Уникальный идентификатор пользователя.
  - username: Имя пользователя.
  - email: Электронная почта.
  - password: Хэш пароля.
  - is\_active: Статус активности пользователя.
2. Категории (categories):
  - id: Уникальный идентификатор категории.
  - name: Название категории.
  - image: Изображение категории.
3. Книги (items):
  - id: Уникальный идентификатор книги.
  - name: Название книги.
  - author: Автор книги.
  - publisher: Издательство книги.
  - language: Язык книги.
  - condition: Состояние книги.
  - cover\_type: Тип переплета.
  - description: Описание книги.
  - price: Цена книги.
  - is\_swap: Флаг, указывающий на возможность обмена.
  - is\_sold: Флаг, указывающий на статус продажи.
  - image: Основное изображение книги.
  - image1, image2, image3: Дополнительные изображения книги.
  - category\_id: Внешний ключ на категорию.
  - created\_by\_id: Внешний ключ на пользователя, создавшего объявление.
  - created\_at: Дата и время создания объявления.
4. Чаты (conversations):
  - id: Уникальный идентификатор чата.
  - item\_id: Внешний ключ на книгу.
  - created\_at: Дата и время создания чата.
  - modified\_at: Дата и время последнего изменения чата.
5. Сообщения (conversation\_messages):
  - id: Уникальный идентификатор сообщения.
  - conversation\_id: Внешний ключ на чат.

- content: Содержание сообщения.
- created\_at: Дата и время создания сообщения.
- created\_by\_id: Внешний ключ на пользователя, создавшего сообщение.

Эта структура базы данных обеспечивает гибкость и эффективность при управлении данными маркетплейса ReRead, позволяя пользователям удобно взаимодействовать с системой и между собой.

## 2.2 Описание проблемных программ

### 2.2.1 Описание проблемной программы №1

Для маркетплейса "Reread" требуется реализация функции перевода сайта на три языка: русский, казахский и английский. Эта задача включает в себя возможность динамического переключения языков интерфейса и содержания сайта пользователем. Функция должна обеспечивать корректный перевод всех элементов интерфейса, включая текстовые поля, кнопки, заголовки и сообщения об ошибках.

#### 2.2.1.1 Схема алгоритма проблемной программы №1

1. Определение текущего языка:
  - Проверка языковых предпочтений пользователя.
  - Установка языка по умолчанию (русский) при отсутствии явного выбора.
2. Загрузка языковых файлов:
  - Загрузка соответствующих языковых файлов для выбранного языка.
3. Перевод элементов интерфейса:
  - Замена текста всех элементов интерфейса на выбранный язык.
4. Сохранение предпочтений пользователя:
  - Сохранение выбранного языка в настройках пользователя или в куки браузера.
5. Обновление интерфейса:
  - Обновление отображаемого интерфейса на основе выбранного языка.

#### 2.2.1.2 Таблица идентификаторов проблемной программы №1

Идентификатор	Смысловое содержание	Тип	Разрядность
lang	Выбранный язык	String	2
user_pref	Языковые предпочтения пользователя	JSON	-
lang_files	Языковые файлы	Dict	-
element_id	Идентификатор элемента интерфейса	String	255
translated_text	Переведенный текст	String	1024

## 2.2.2 Описание проблемной программы №2

Необходимо реализовать функциональность заказа и оплаты товара. Пользователь должен ввести свои данные и информацию о товаре, а система должна сохранить ID заказа для повторных покупок. Это обеспечит удобство для пользователей, позволяя быстро повторять заказы без повторного ввода всех данных.

### 2.2.2.1 Схема алгоритма проблемной программы №2

Схема алгоритма проблемной программы №2 представлена в приложении И.

### 2.2.2.2 Таблица идентификаторов проблемной программы №2

Идентификаторы проблемной программы №2 приведены в таблице 2.3.

Таблица 2.3 - Таблица идентификаторов проблемной программы №2.

Идентификатор	Смысловое содержание	Тип	Разрядность
ex	Документ Excel	Variant	255
rnd	Генератор чисел	Integer	4
idz	ID заказа	Integer	4
sum	Сумма заказа	Integer	4
card	Номер карты пользователя	String	255
Text	SQL запрос на добавление данных в таблицу	String	255

## 2.2.3 Описание проблемной программы №3

Третья проблема связана с необходимостью реализации системы уведомлений для пользователей. Система должна уведомлять пользователей о новых сообщениях в чатах, изменениях статуса их объявлений, а также о

завершении сделок. Это улучшит взаимодействие пользователей с платформой и повысит удобство использования сервиса.

### 2.2.3.1 Схема алгоритма проблемной программы №3

Схема алгоритма проблемной программы №3 представлена в приложении К.

### 2.2.3.2 Таблица идентификаторов проблемной программы №3

Идентификаторы проблемной программы №3 приведены в таблице 2.4.

Таблица 2.4 - Таблица идентификаторов проблемной программы №3.

Идентификатор	Смысловое содержание	Тип	Разрядность
user_id	Уникальный идентификатор пользователя	Integer	4
notification	Текст уведомления	String	255
status	Статус уведомления (новое, прочитанное и т.д.)	String	50
created_at	Дата и время создания уведомления	DateTime	8

## 3 ОРГАНИЗАЦИЯ ПРОИЗВОДСТВА

### 3.1 Условия выполнения проекта

Для работы программного комплекса по учету движения книг на маркетплейсе ReRead, необходим следующий набор технических средств:

- процессор с тактовой частотой не ниже 1,8 ГГц;
- оперативная память 512 Мбайт;
- 2 ГБ ОЗУ (рекомендуется 8 ГБ ОЗУ, минимум 2,5 ГБ при выполнении на виртуальной машине);
- стандартная клавиатура;
- мышь;
- место на жестком диске: до 210 ГБ (минимум 800 МБ) свободного места в зависимости от установленных компонентов;
- скорость жесткого диска: для повышения производительности рекомендуется использовать твердотельный накопитель (SSD);
- видеоадаптер с минимальным разрешением 720p (1280 на 720 пикселей); для оптимальной работы рекомендуется разрешение WXGA (1366 на 768 пикселей) или выше.

### 3.2 Инструкция пользователю

#### 1. Регистрация и вход:

- Перейдите на главную страницу сайта.
- Если у вас уже есть учетная запись, введите свой логин и пароль в соответствующие поля и нажмите кнопку "Войти". Если нет, нажмите на ссылку "Регистрация" и следуйте инструкциям для создания нового аккаунта.

#### 2. Просмотр каталога товаров:

- После входа на сайт вы будете перенаправлены на страницу каталога товаров, где вы можете просматривать доступные товары.
- Используйте фильтры или поиск, чтобы быстро найти интересующие вас товары.

#### 3. Просмотр деталей товара:

- Щелкните на изображение или наименование товара, чтобы увидеть его подробное описание, цену и другую полезную информацию.

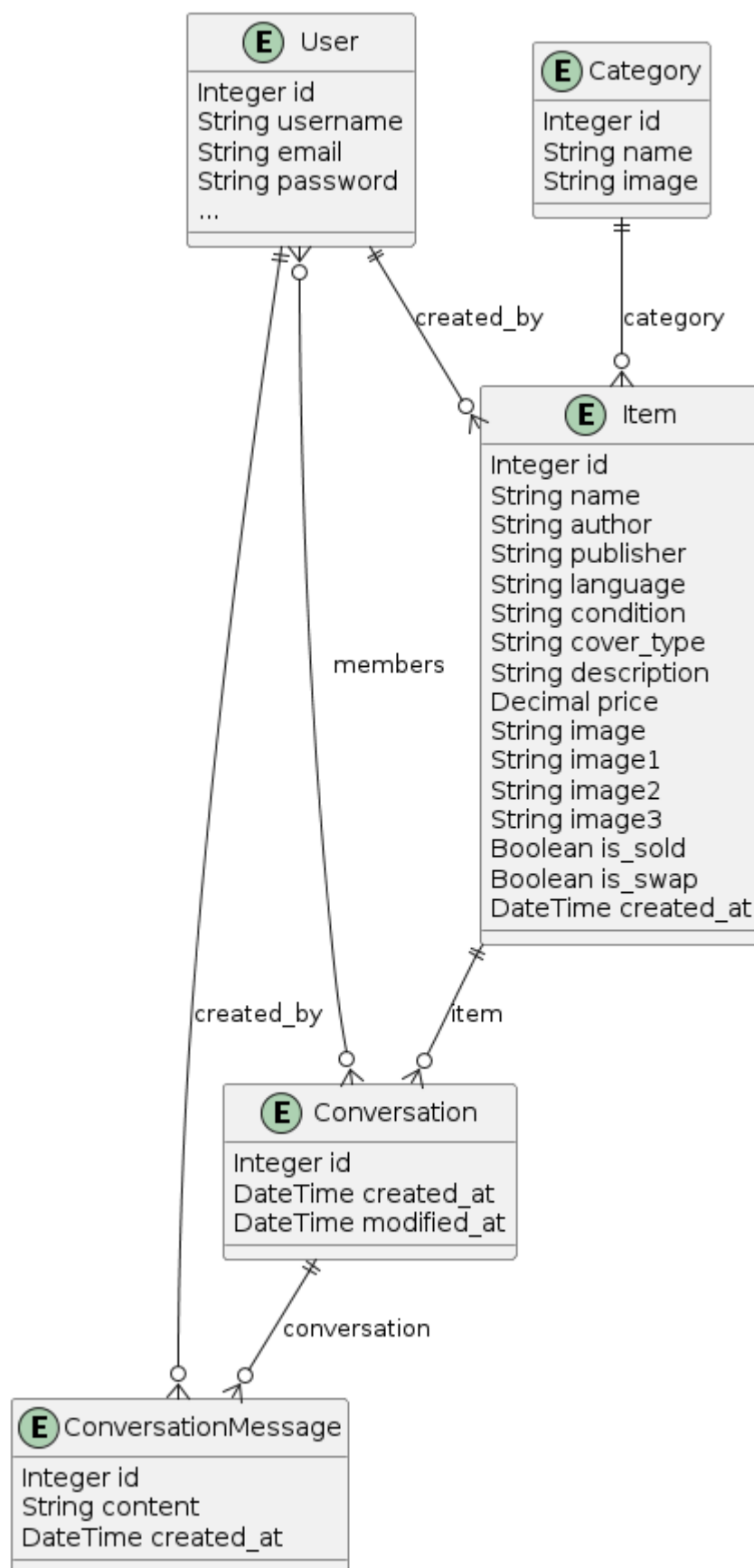
#### 4. Связь с продавцом:

- Если вы хотите приобрести товар, нажмите кнопку "Написать продавцу".
- После этого будет открыт чат с продавцом, который вы можете просмотреть, нажав на соответствующую ссылку в верхнем меню.

					РК ЦАТЭК 4S06130103 КП ПЗ	Лист
Изм.	Лист	№ докум	Подпись	Дата		16



Таблица 3.1 - Структура таблиц БД «reread.db»



## Заключение

В заключении можно подытожить основные результаты работы над проектом и подчеркнуть их значимость. Вот пример заключения:

В ходе разработки данного проекта был создан маркетплейс "Reread" для продажи букинистических книг. Благодаря использованию современных технологий, таких как Django, Python, SQLite3, Tailwind CSS и других, был реализован функционал, который позволяет пользователям удобно просматривать каталог книг, добавлять их в корзину, оформлять заказы и отслеживать их статус.

Основные достижения проекта включают в себя:

- Реализация полноценного функционала для управления каталогом книг, заказами и пользователями.
- Создание удобного и интуитивно понятного интерфейса для пользователей сайта.
- Обеспечение безопасности данных пользователей и защиты от несанкционированного доступа.
- Использование механизмов локализации для поддержки различных языков и культур пользователей.
- Оптимизация производительности и скорости работы сайта для обеспечения приятного пользовательского опыта.

Таким образом, создание маркетплейса "Reread" является важным шагом в сфере электронной коммерции, обеспечивая пользователям удобный доступ к букинистическим книгам и способствуя их распространению и переиспользованию. Реализованный функционал позволяет пользователям не только приобретать книги, но и создавать собственные объявления о продаже, что делает сайт полноценным ресурсом для любителей чтения.

Благодаря усилиям всей команды проекта удалось достичь поставленных целей и создать продукт, который будет полезен и удобен для пользователей.

## СПИСОК ЛИТЕРАТУРЫ

1. Python  
[Python.org](https://python.org) - Официальный сайт языка программирования Python. Включает документацию, учебные пособия и новости. На этом ресурсе можно найти исчерпывающую информацию о синтаксисе языка, встроенных функциях, библиотеках и лучших практиках использования Python.
2. Django  
[Django.fun](https://djangofun.com) - Документация фреймворка Django, версия 5.0. Этот ресурс содержит полное руководство по использованию Django, включая установку, настройку, основные функции и модули, а также примеры кода и руководства по разработке веб-приложений на базе Django.
3. Tailwind CSS  
[Tailwind CSS](https://tailwindcss.com) - Официальный сайт CSS-фреймворка Tailwind. Tailwind CSS предоставляет утилитарные классы для быстрого и легкого создания пользовательских дизайнов. На сайте можно найти документацию, примеры использования, а также руководство по установке и настройке фреймворка.
4. MDN Web Docs  
[MDN Web Docs](https://mdn.com) - Обширная документация по веб-технологиям, включающая HTML, CSS, JavaScript и другие веб-стандарты. Этот ресурс поддерживается Mozilla и является одним из наиболее авторитетных источников информации для веб-разработчиков, предлагая подробные руководства, справочные материалы и примеры кода.

ПРИЛОЖЕНИЯ

					РК ЦАТЭК 4S06130103 КП ПЗ		
Изм	Лист	№ докум	Подпи	Дата	Secondhand Book Marketplace «ReRead»		
Разраб.	Федоров И.В.						
Провер.	Нехорошев В.Д.						
Утверд.	Нехорошев В.Д.				ЦАТЭК гр.ПЗА		
					Лит.	Лист	Листов
						20	

## ПРИЛОЖЕНИЕ А

### (обязательное)

Main project

Urls.py

```
from django.conf import settings
from django.conf.urls.static import static
from django.contrib import admin
from django.urls import path, include
```

```
from django.views.generic.base import RedirectView
```

```
urlpatterns = [
    path('favicon.ico/', RedirectView.as_view(url='/static/img/reread-logo.ico', permanent=True),
name='favicon'),
    path('i18n/', include('django.conf.urls.i18n')),
    path('rosetta/', include('rosetta.urls')),

    path("", include('core.urls')),
    path('items/', include('item.urls')),
    path('dashboard/', include('dashboard.urls')),
    path('inbox/', include('conversation.urls')),
    path('admin/', admin.site.urls),
```

```
] + static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

Core app

Urls.py

```
from django.contrib.auth import views as auth_views
from django.urls import path
```

```
from . import views
from .forms import LoginForm
```

```
app_name = 'core'
```

```
urlpatterns = [
    path("", views.index, name='index'),
    path('contact/', views.contact, name='contact'),
    path('signup/', views.signup, name='signup'),
    path('login/', auth_views.LoginView.as_view(template_name='core/login.html',
authentication_form=LoginForm), name='login'),
    path('logout/', auth_views.LogoutView.as_view(), name='logout'),
```

```
]
```

Views.py

```
from django.shortcuts import render, redirect
```

```
from item.models import Category, Item
```

```
from .forms import SignupForm
```

```

def index(request):
    items = Item.objects.filter(is_sold=False)[0:6]
    categories = Category.objects.all()

    return render(request, 'core/index.html', {
        'categories': categories,
        'items': items,
    })

def contact(request):
    return render(request, 'core/contact.html')

def signup(request):
    if request.method == 'POST':
        form = SignupForm(request.POST)

        if form.is_valid():
            form.save()

            return redirect('/login/')
    else:
        form = SignupForm()

    return render(request, 'core/signup.html', {
        'form': form
    })

```

## Index.html

```

{% extends 'core/base.html' %}

{% load static i18n %}
{% get_current_language as language_code %}

{% block title %}
    {% trans 'Welcome' %}
{% endblock %}

{% block content %}
    <div class="flex justify-center flex-col sm:md:flex-row mx-3 px-4 py-6 rounded-xl items-center">
        <div class="text-left sm:md:w-2/4">
            <h1 class="text-xl"><b>ReRead</b> - {% trans 'это ваш персональный маркетплейс книг,
предназначенный для любителей чтения и коллекционеров книг. Наша платформа предлагает
широкий выбор книг различных жанров, авторов и изданий по доступным ценам.' %} </h1>
            <br>
            <h2 class="text-base">{% trans 'На "Reread" вы можете как продавать свои книги, так и
приобретать новые для своей коллекции. Мы стремимся создать удобное и безопасное онлайн-
пространство для обмена литературой, где каждый пользователь может найти то, что ищет.'
%}</h2>
        </div>
        
    </div>

```

```

<div class="mt-6 px-4 py-12 bg-gray-100 rounded-xl">
  <h2 class="mb-12 text-2xl text-center">{% trans 'Newest items' %}</h2>

  <div class="grid xl:md:lg:grid-cols-4 md:grid-cols-3 sm:grid-cols-2 gap-3">
    {% for item in items %}
      <div class="relative m-5 flex w-full max-w-xs flex-col overflow-hidden rounded-lg border
border-gray-100 bg-white shadow-md">
        <a class="relative mx-3 mt-3 flex h-60 rounded-xl" href="{% url 'item:detail' item.id %}">
          
          <span
            class="absolute top-0 left-0 m-2 rounded-full px-3 py-1 text-center text-sm font-medium
text-white
            {% if item.is_swap %}
              bg-orange-600
            {% endif %}
            ">
            {% if item.is_swap %}
              {% trans 'Обмен' %}
            {% endif %}
          </span>
          <span
            class="absolute top-0 {% if item.is_swap %} left-20 {% endif %} m-2 rounded-full px-3
py-1 text-center text-sm font-medium text-white
            {% if item.price == 0 %}
              bg-green-600
            {% else %}
              {% if item.price == None %}
                bg-green-600
              {% endif %}
            {% endif %}
            ">
            {% if item.price == 0 %}
              {% trans 'Даром' %}
            {% else %}
              {% if item.price == None %}
                {% trans 'Даром' %}
              {% endif %}
            {% endif %}
          </span>
        </a>
      <div class="mt-4 px-5 pb-5">
        <h5 class="text-xl tracking-tight text-slate-900">
          {% if item.name|length > 50 %}
            {{ item.name|slice:"":50" }}...
          {% else %}
            {{ item.name }}
          {% endif %}

          {% if item.author|length > 50 %}
            , {{ item.author|slice:"":50" }}...
          {% else %}
            , {{ item.author }}
          {% endif %}

```

```

</h5>
<div class="mt-2 mb-5 flex items-center justify-between">
  <p>
    <span class="text-3xl font-bold text-slate-900">
      {% if item.price %}
        {% if item.is_swap %}
          {% if item.price %}
            {{ item.price }} ₾
          {% else %}
            {% trans 'Обмен' %}
          {% endif %}
        {% else %}
          {{ item.price }} ₾
        {% endif %}
      {% else %}
        {% if item.price == 0 %}
          {% trans 'Даром' %}
        {% else %}
          {% trans 'Даром' %}
        {% endif %}
      {% endif %}
    </span>
  </p>

  <span class="mr-2 ml-3 rounded bg-blue-200 px-2.5 py-0.5 text-xs font-semibold">{{
item.get_language_display }}</span>
  <span class="mr-2 ml-3 rounded bg-purple-200 px-2.5 py-0.5 text-xs font-semibold">{{
item.publisher }}</span>

</div>
  <span class="mr-2 rounded bg-yellow-200 px-2.5 py-0.5 text-xs font-semibold">{{
item.get_cover_type_display }}</span>
  <span class="mr-2 rounded bg-green-200 px-2.5 py-0.5 text-xs font-semibold">{{
item.get_condition_display }}</span>
</div>
</div>

  {% endfor %}
</div>
</div>

<div class="mt-6 px-6 py-12 bg-gray-100 rounded-xl">
  <h2 class="mb-12 text-2xl text-center">{% trans 'Categories' %}</h2>

  <div class="grid grid-cols-3 gap-3">
    {% for category in categories %}
      <div>
        <a href="{% url 'item:items' %}?category={{ category.id }}">
          <div>
            
          </div>
          <div class="p-6 bg-white rounded-b-xl">
            <h2 class="text-2xl">
              {% for lang_code, translation in category.name_translations.items %}

```



```

        {% if lang_code == LANGUAGE_CODE %}
            {{ translation }}
        {% endif %}
        {% empty %}
            {{ category.name }}
        {% endfor %}
    </h2>
    <p class="text-gray-500">{{ category.items.count }} {% trans 'items' %}</p>
</div>
</a>
</div>
    {% endfor %}
</div>
</div>
{% endblock %}

```

Полный код доступен по адресу <https://github.com/lyarov22/reread-reloaded>

ПРИЛОЖЕНИЕ Б  
(рекомендуемое)

Входные данные

Входные данные:

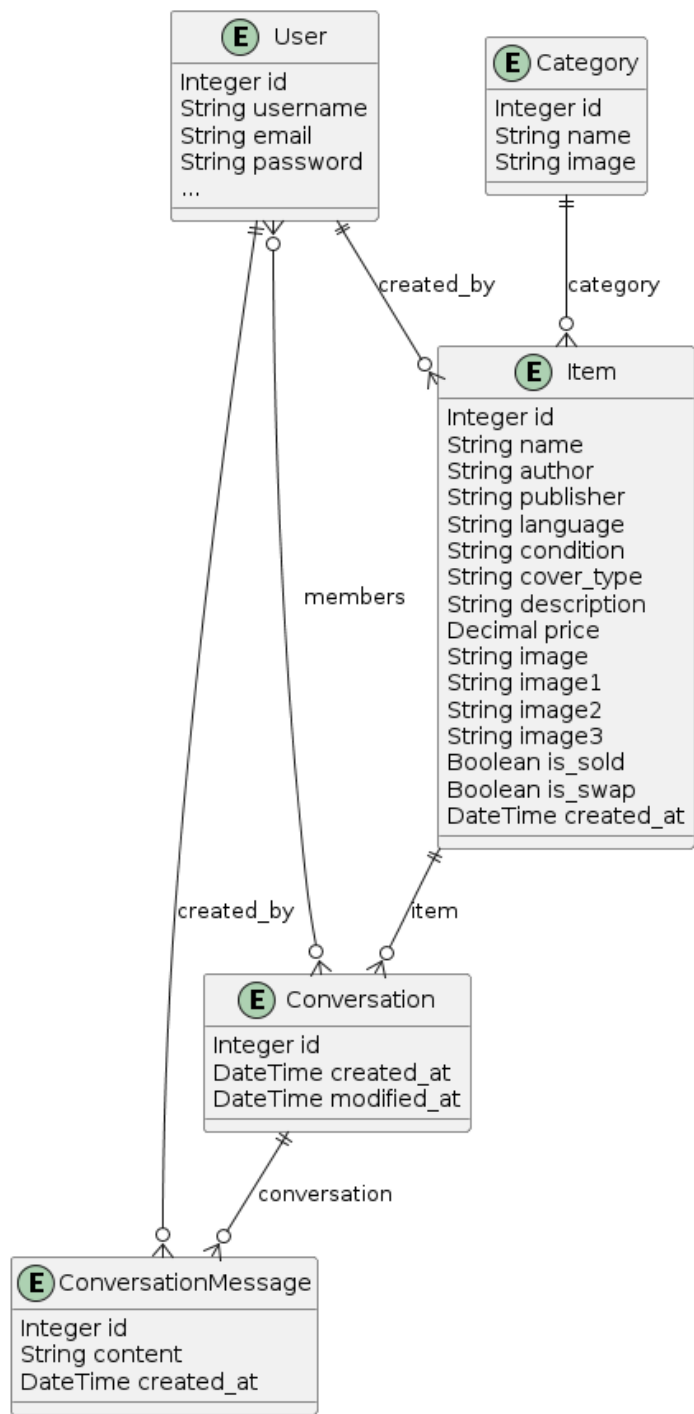
1. Регистрация и аутентификация пользователя:
  - Логин и пароль пользователя.
  - Электронная почта.
  - Имя пользователя.
2. Данные книги:
  - Название книги.
  - Издательство.
  - Язык книги.
  - Состояние книги.
  - Тип переплета.
  - Цена книги.
  - Изображения книги.
  - Статус (продано/доступно).
  - Возможность обмена.
3. Сообщения и чаты:
  - Содержание сообщения.
  - Участники чата.

Выходные данные:

1. Информация о книге:
  - Название книги.
  - Автор книги.
  - Издательство.
  - Язык книги.
  - Состояние книги.
  - Тип переплета.
  - Описание книги.
  - Цена книги.
  - Изображения книги.
  - Возможность обмена.
2. Информация о пользователе:
  - Имя пользователя.
  - Электронная почта.
3. История сообщений и чатов:
  - Содержание сообщений.
  - Время отправки.
  - Участники чата.

ПРИЛОЖЕНИЕ В  
(рекомендуемое)

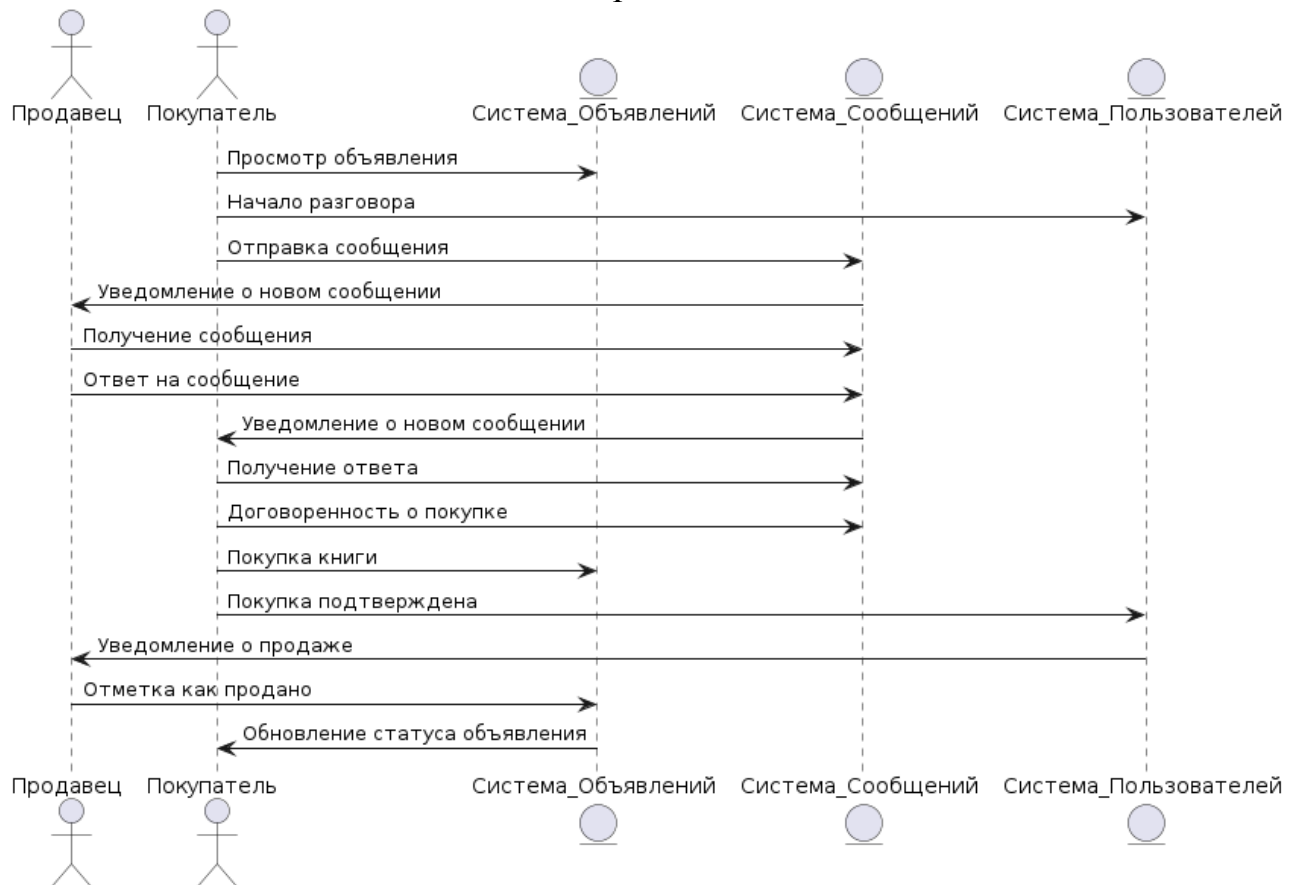
Диаграмма БД



					ПК ЦАТЭК 4S06130103 КП ПЗ		
Изм	Лист	№ докум	Подпись	Дата			
Разраб.	Федоров И.В.						
Провер.	ЖаксыбаеваНН						
Утверд.	ЖаксыбаеваНН						
					Лист	Лист	Листов
						27	
					ЦАТЭК зр.ПЗА		

# ПРИЛОЖЕНИЕ Г (обязательное)

## Схема работы комплекса



ПРИЛОЖЕНИЕ 3  
(рекомендуемое)

