|  |  |  |
| --- | --- | --- |
|  |  | УТВЕРЖДАЮ |
|  |  | Руководитель  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.  М.П. |

ТЕХНИЧЕСКИЙ ПРОЕКТ

Информационной системы для компании по обмену книг

2024

Содержание

[1 Уточнение структур данных 2](#_Toc182578365)

[2 Формы и их представления 4](#_Toc182578366)

[3 Разработка алгоритма решения задачи 7](#_Toc182578367)

[4 Определения языка разработки 11](#_Toc182578368)

[5 Структура программы и требования к техническим средствам 12](#_Toc182578369)

# Уточнение структур данных

Информационная система для компании по обмену книг должна иметь базу данных, подробности который представлены в виде логической и физической ER-диаграмм на рисунках 1 и 2.

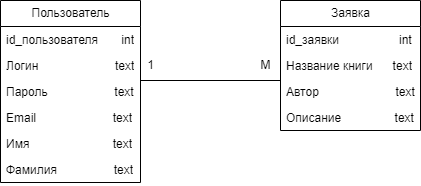


Рисунок 1 – Логическая ER диаграмма

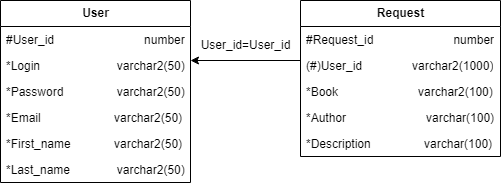


Рисунок 2 – Физическая ER диаграмма

В данной базе данные есть две таблицы: «Пользователь» и «Заявка».

Таблица «Пользователь» содержит следующие данные:

* id\_пользователя: идентификатор пользователя;
* Логин: логин, используемый для входа в систему;
* Пароль: пароль, используемый для входа в систему;
* Email: почта пользователя;
* Имя: имя пользователя, отображающееся в профиле;
* Фамилия: фамилия пользователя, отображающаяся в профиле.

Таблица «Заявка» содержит следующие данные:

* id\_заявки: идентификатор заявки;
* Название книги: название книги, отображающееся в заявке;
* Автор: автор книги, отображающийся в заявке;
* Описание: контактные данные пользователя и другие данные.

# Формы и их представления

Прототип главной страницы изображен на рисунке 3.

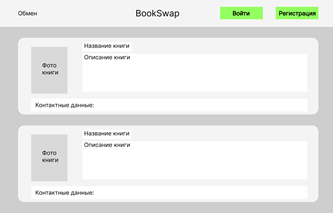


Рисунок 3 – Прототип главной страницы (обмен)

На данном прототипе показано, что пользователь может просматривать заявки на обмен книгами, войти в свой профиль или зарегистрироваться.

Прототип окна входа в профиль изображен на рисунке 4.

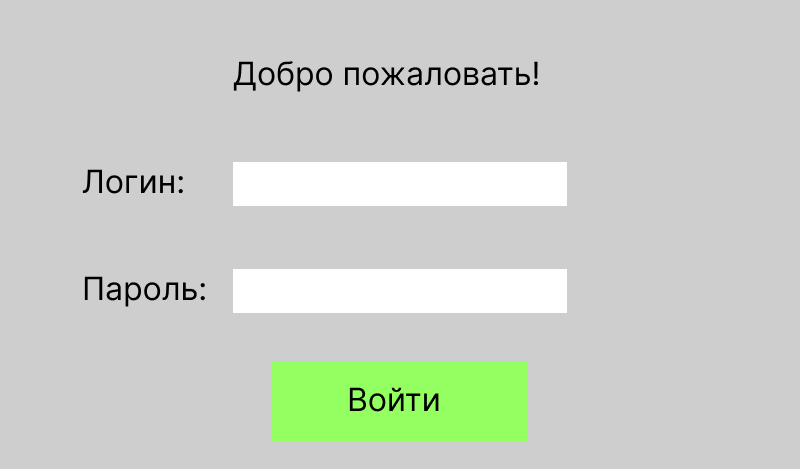


Рисунок 4 – Прототип окна входа в систему

На данном прототипе показано, как будет осуществляться вход пользователей в систему. Вводятся логин и пароль в соответствующие поля, а после нажатия кнопки «Войти» происходит вход в систему, если пользователь существует, а окно входа закрывается.

Окно регистрации в систему представлено на рисунке 5.

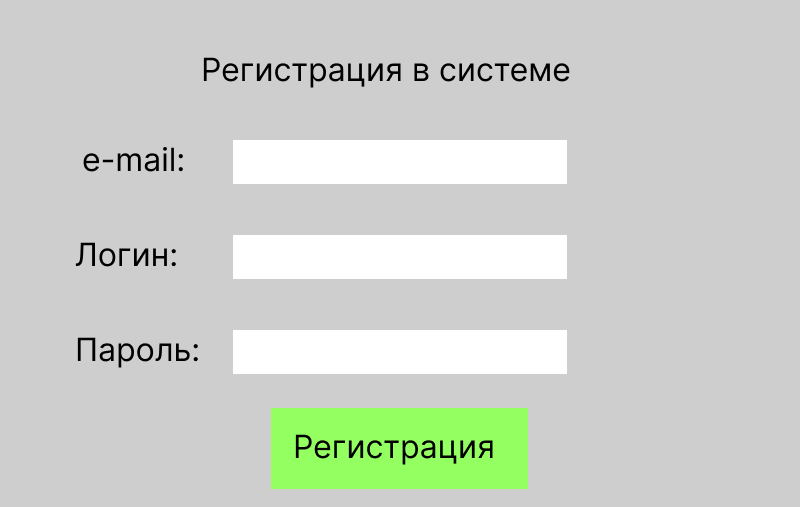


Рисунок 5 – Прототип окна регистрации в системе

На окне представлены: три поля для ввода (e-mail, логин, пароль), функциональная кнопка «Регистрация». При нажатии на кнопку происходит регистрация в системе, а окно регистрации закрывается.

Окно при выполненном входе в систему представлено на рисунке 6.

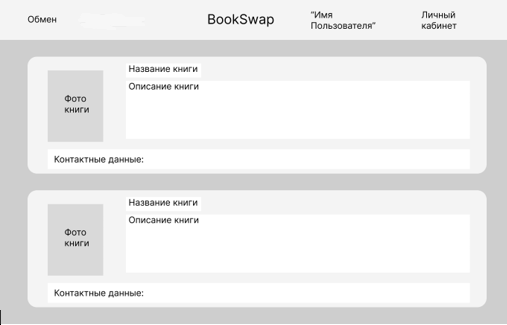


Рисунок 6 — Прототип главного окна при выполненном входе

На окне расположены: функциональные кнопки «Обмен», «Заявки», «Личный кабинет». Также после входа в систему появляется надпись с именем пользователя.

Окно личного кабинета пользователя представлено на рисунке 7.

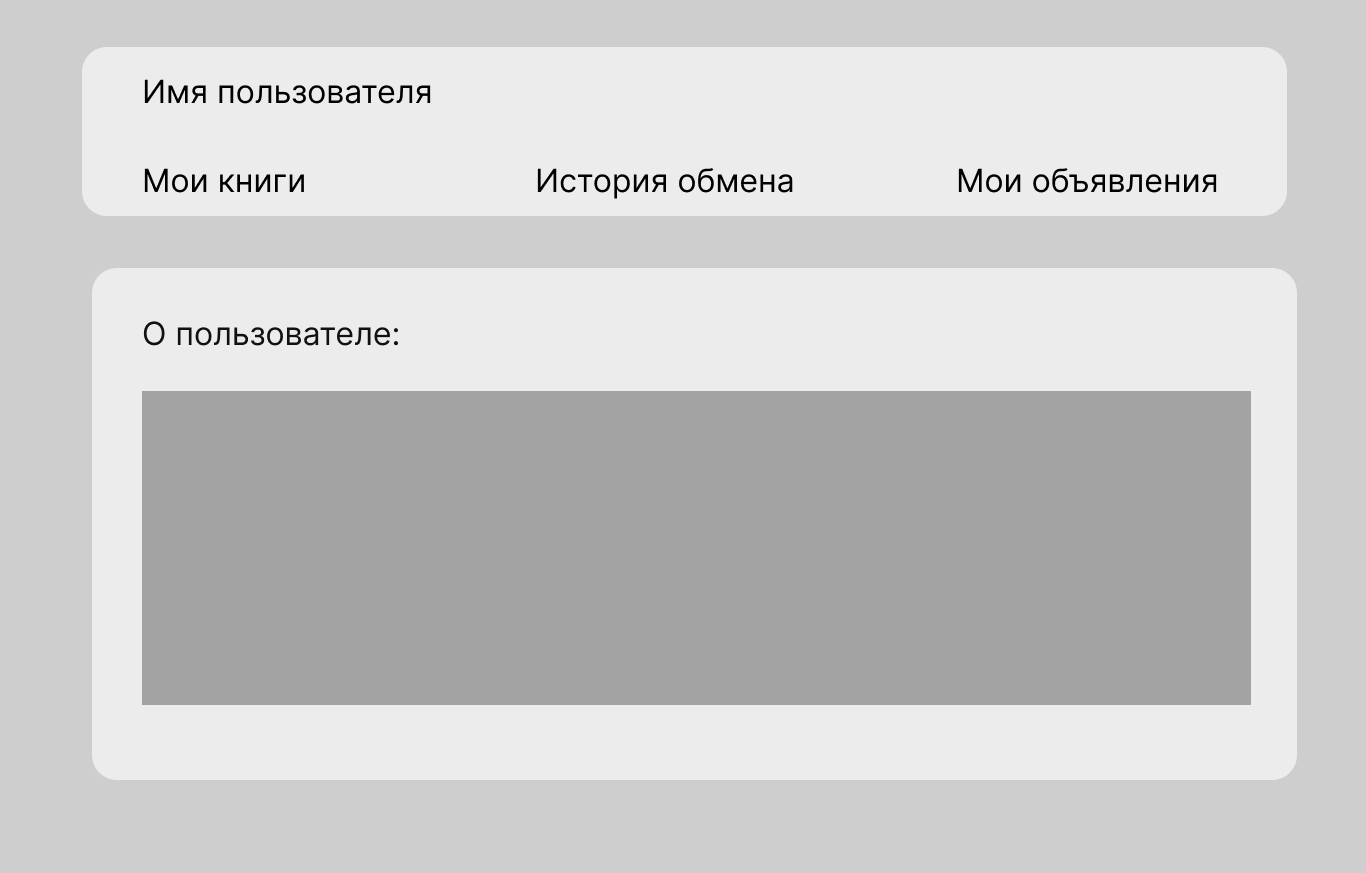


Рисунок 7 — Личный кабинет пользователя

На него можно перейти при нажатии на функциональную кнопку «Личный кабинет», которая расположена на главном окне при выполненном входе в систему. На окне расположены: имя пользователя, информация о пользователе, а также кнопки для просмотра книг, истории обмена и объявлений.

# Разработка алгоритма решения задачи

Схема алгоритма регистрации в системе представлена на рисунке 8.



Рисунок 8 – Схема алгоритма регистрации в системе

Схема алгоритма авторизации в системе представлена на рисунке 8.



Рисунок 8 – Схема алгоритма авторизации в системе

Схема алгоритма добавления заявки на обмен представлена на рисунке 9.

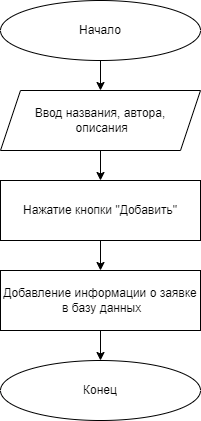


Рисунок 9 – Схема алгоритма добавления заявки на обмен

Схема алгоритма изменения личных данных пользователя представлена на рисунке 10.

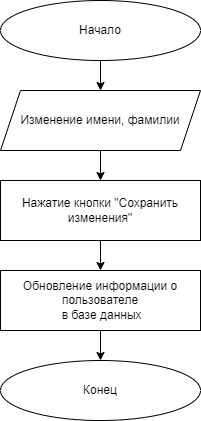


Рисунок 10 – Схема алгоритма изменения личных данных пользователя

Пользовательские сценарии указаны в таблице 1.

Таблица 1 – Пользовательские сценарий

|  |  |  |
| --- | --- | --- |
| № функции | Пользователь | Система |
| 1 | Авторизация | 1. Проверка введенных данных, обработка запроса на вход  2. Отображение страницы пользователя |
| 2 | Регистрация | 1. Проверка корректности введённых данных (валидность e-mail, уникальность логина, безопасность пароля)  2. Если данные корректны, сохранение информации в базе данных 3. Переход в окно авторизации |
| 3 | Оставление заявки на обмен книгами | 1. Проверка введенных данных  2. Сохранение заявки на обмен в базе данных  3. Отображение страницы со всеми заявками |
| 4 | Изменение данных в профиле | 1. Проверка введенных данных  2. Сохранение изменений в базе данных 3. Отображение обновлённых данных на странице профиля |

# Определения языка разработки

Для реализации информационной системы, будут использоваться следующие языки программирования и технологии:

* Python – мультипарадигмальный высокоуровневый язык программирования общего назначения с динамической строгой типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ.
* CSS – формальный язык декодирования и описания внешнего вида документа (веб-страницы), написанного с использованием языка разметки (чаще всего HTML или XHTML). Также может применяться к любым XML-документам, например, к SVG или XUL.
* PostgreSQL – СУБД для работы с базой данных системы.
* Django – свободный фреймворк для веб-приложений на языке Python, использующий шаблон проектирования MVC.

# Структура программы и требования к техническим средствам

Функциональные требования:

* Регистрация и авторизация пользователей
* Оставление заявок на обмен книгами
* Просмотр заявок на обмен книгами
* Изменение личных данных пользователей

Нефункциональные требования:

* Удобство использования: интерфейс должен быть интуитивно понятным, с понятной навигацией и инструкциями;
* Совместимость: система должна корректно отображаться в современных версиях популярных браузеров.