**Министерство науки и высшего образования Российской Федерации**

**федеральное государственное автономное образовательное учреждение высшего образования**

**“НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО”  
(УНИВЕРСИТЕТ ИТМО)**

**ИТОГОВАЯ АТТЕСТАЦИОННАЯ РАБОТА**

*АВТОМАТИЗАЦИЯ РАБОТЫ БИБЛИОТЕКИ (ВЫДАЧА КНИГ НА БУМАЖНЫХ НОСИТЕЛЯХ)*

Автор Бубырев Дмитрий Иванович \_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия Имя Отчество) (Подпись)

**Центр авторизованного обучения информационным технологиям**

Наименование программы «Разработчик профессионально

ориентированных компьютерных технологий»

Руководитель ст. преп. Смольянинов Антон Валентинович \_\_\_\_\_\_\_\_\_\_\_\_

(ФИО, ученая звание, степень) (Подпись)

**К защите допустить**

|  |  |  |
| --- | --- | --- |
| *Заместитель директора ЦАО ИТ,*  *к.т.н.* |  | / *Т.В. Зудилова*/ |

Санкт-Петербург, 2021г.

Обучающийся Бубырев Дмитрий Иванович Группа 124/4

(Фамилия, И. О.)

Работа принята «\_\_\_» марта 2021г.

Работа выполнена с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата защиты «\_\_\_» марта 2021г.

Секретарь ИАК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия, И. О.) (подпись)

Листов хранения \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Демонстрационных материалов \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ОГЛАВЛЕНИЕ**

[Введение 4](#_Toc65448943)

[Роли внутри системы 5](#_Toc65448944)

[Требования к ролям 5](#_Toc65448945)

[Общий иерархический список 8](#_Toc65448946)

[Объем реализации поставленных требований в патче v1.0 13](#_Toc65448947)

[Требования и функции системы 11](#_Toc65448948)

[Архитектура системы 14](#_Toc65448949)

[Модель БД 15](#_Toc65448950)

[Логика системы 17](#_Toc65448951)

[Пользовательский интерфейс 20](#_Toc65448952)

[Заключение 27](#_Toc65448953)

[Литература 28](#_Toc65448954)

# Введение

В XXI веке все большую популярность получает обмен информации на электронных носителях, это дешевле, экологичнее и доступнее. Тем не менее не теряю актуальность и библиотеки, осуществляющие выдачу бумажных книг, такие библиотеки часто встречаются в университетах, в научных учреждениях, в архивах. Выдача книг осуществляется по читательскому билету, и в присутствии читателя, зачастую данная процедура занимает много времени и неудобна как для читателя, так и для библиотекаря. Возможность подать заказ на выдачу книг заранее сильно бы упростила работу библиотеки, убрав или сильно сократив очереди из читателей, и именно для этой цели и разрабатывается наше приложение для автоматизации работы библиотек.

Разрабатываемое приложение будет реализовано по клиент-серверной схеме визуальная часть будет доступна для просмотра во всех современных браузерах и оптимизирована для экранов смартфонов и планшетов.

Основными функциями приложения будут:

1. Регистрация и авторизация пользователей.
2. Просмотр существующих авторов и книг.
3. Добавление новых авторов и книг.
4. Создание заказов на выдачу книг.
5. Обработку заказов на выдачу книг.

Для реализации серверной части приложения выбраны следующие технологии:

1. Язык – Java 1.8.
2. Фреймворки – Spring Boot, Maven.
3. Система защиты – Spring Security.
4. Средство работы с базой данных - PostgreSQL, Hibernate.

Для реализации клиентской части приложения выбраны следующие технологии:

1. Фреймворк - Bootstrap
2. Шаблонизатор - FreeMarker
3. Язык разметки Web страниц – HTML, стили - CSS.

Разрабатываемое приложение актуально для всех библиотек, еще не обладающих собственными системами автоматизации.

# Роли внутри системы

В сервисе предусматривается разделение пользователей на три группы ролей:

1. Пользователи.

2. Библиотекари.

3. Администраторы.

# Требования к ролям

**Посетители**(неавторизированные пользователи):

Функциональные требования:

1. Видит общую информацию на сайте.

2. Имеет доступ к страницам авторизации и регистрации).

Нефункциональные требования:

1. Проверка правильности указанной электронной почты (при регистрации) осуществляется с помощью письма с проверочным кодом.

2. Система позволяет запомнить личные данные пользователя для упрощения повторной авторизации.

3. Web интерфейс сервиса должен быть оптимизирован для чтения с устройств с различным разрешением.

**Пользователи**(авторизированные пользователи):

Функциональные требования:

1. Имеет доступ к просмотру книг в базе данных, может искать книги по названию.

2. Может просматривать страницы авторов и смотреть книги по авторам

3. Может сформировать заказ на выдачу книг.

4. Видит историю своих заказов.

5. Видит список книг, находящихся на руках.

6. Может отредактировать личные данные.

Нефункциональные требования:

1. Проверка правильности измененной электронной почты осуществляется с помощью письма с проверочным кодом.

2. При истечении времени (14 дней) хранения книги пользователем, система направляет письмо с напоминанием о необходимости вернуть книгу в библиотеку.

3. Личная страница пользователя должна хранить его личную информацию, а именно: логин, пароль (в зашифрованном виде), ФИО, дату рождения, электронную почту, мобильный телефон.

**Библиотекари.**

Функциональные требования:

1. Добавляет новые книги и новых авторов книг.

2. Имеет доступ к списку заказов книг от пользователей, может изменить статус заказа.

3. Может применять операции CRUD к книгам и авторам.

4. Видит сколько книг выдано и у кого из пользователей они находятся на руках.

5. Может отредактировать личные данные

Нефункциональные требования:

1. Информация о книге должна содержать: название, обложку(изображение), автора, год написания, жанр, страну написания/издания, количество данных книг в библиотеке.

2. Обновление базы данных при применении CRUD операций к книгам и авторам должно занимать не более 1 секунды.

3. Менеджер видит лист активных заказов пользователей отсортированным по дате заполнения, заказы по которым книги не выданы подсвечиваются для лучшей читаемости.

**Администраторы.**

Функциональные требования:

1. Видит полный список пользователей.

2. Может изменить статус пользователя, может удалить пользователя.

3. Добавляет новые книги и новых авторов книг.

4. Имеет доступ к списку заказов книг от пользователей, может изменить статус заказа.

5. Может применять операции CRUD к книгам и авторам

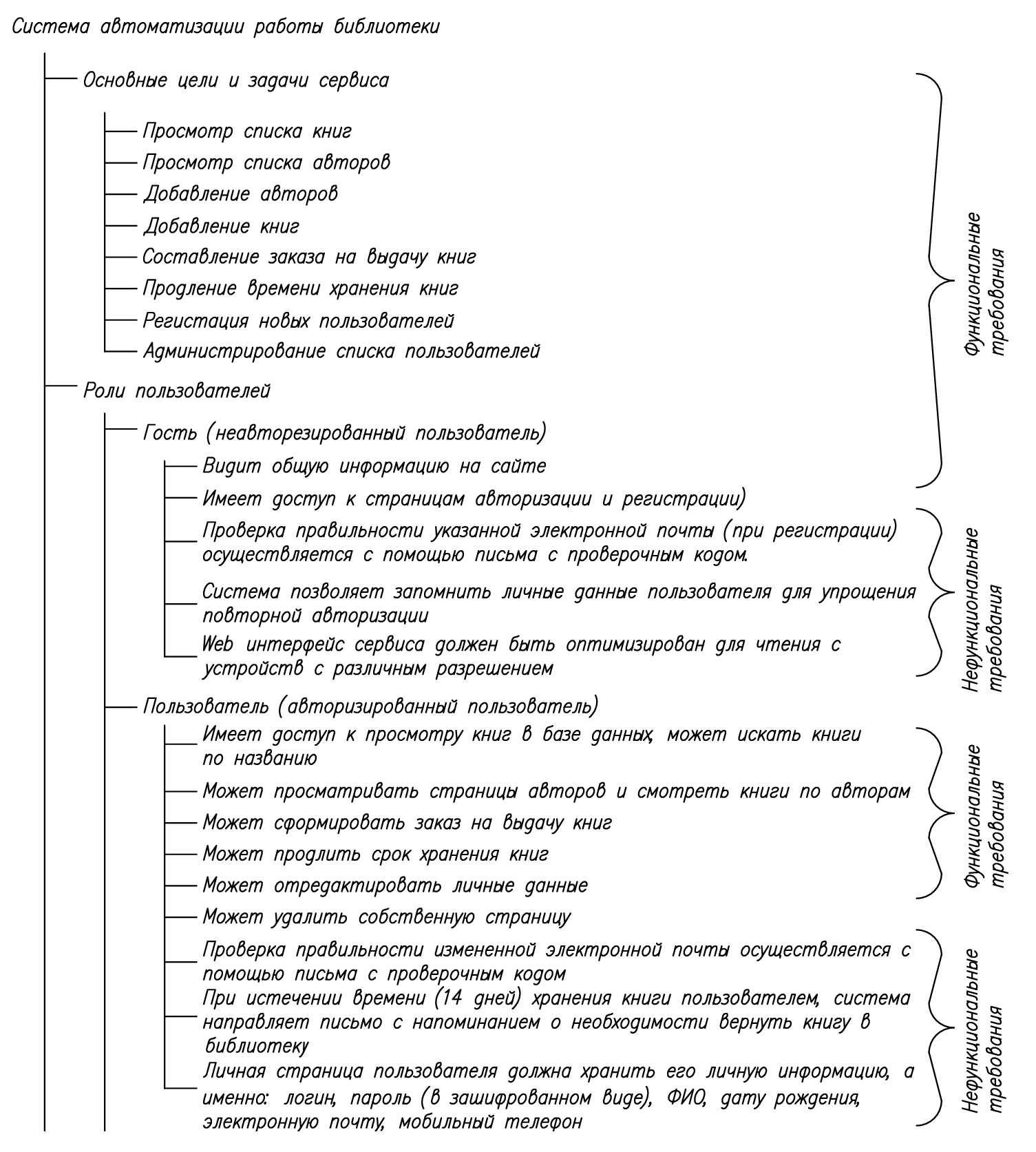
Нефункциональные требования:

1. Администратор должен иметь доступ к кабинету администратора, в который запрещен доступ остальным пользователям, более того пользователи и библиотекари не видят саму ссылку перехода в кабинет администратора

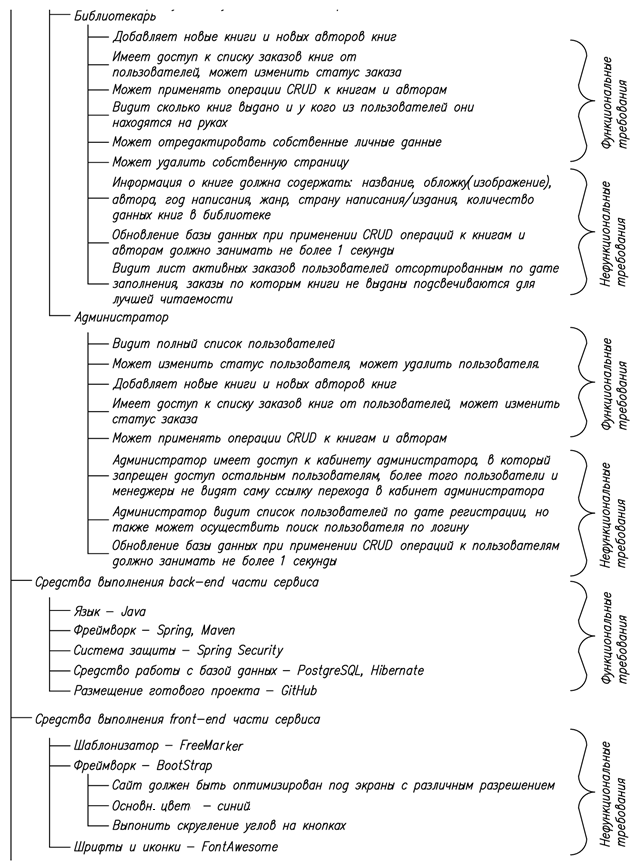
2. Администратор видит список пользователей по дате регистрации, но также может осуществить поиск пользователя по логину.

3. Обновление базы данных при применении CRUD операций к пользователям должно занимать не более 1 секунды.

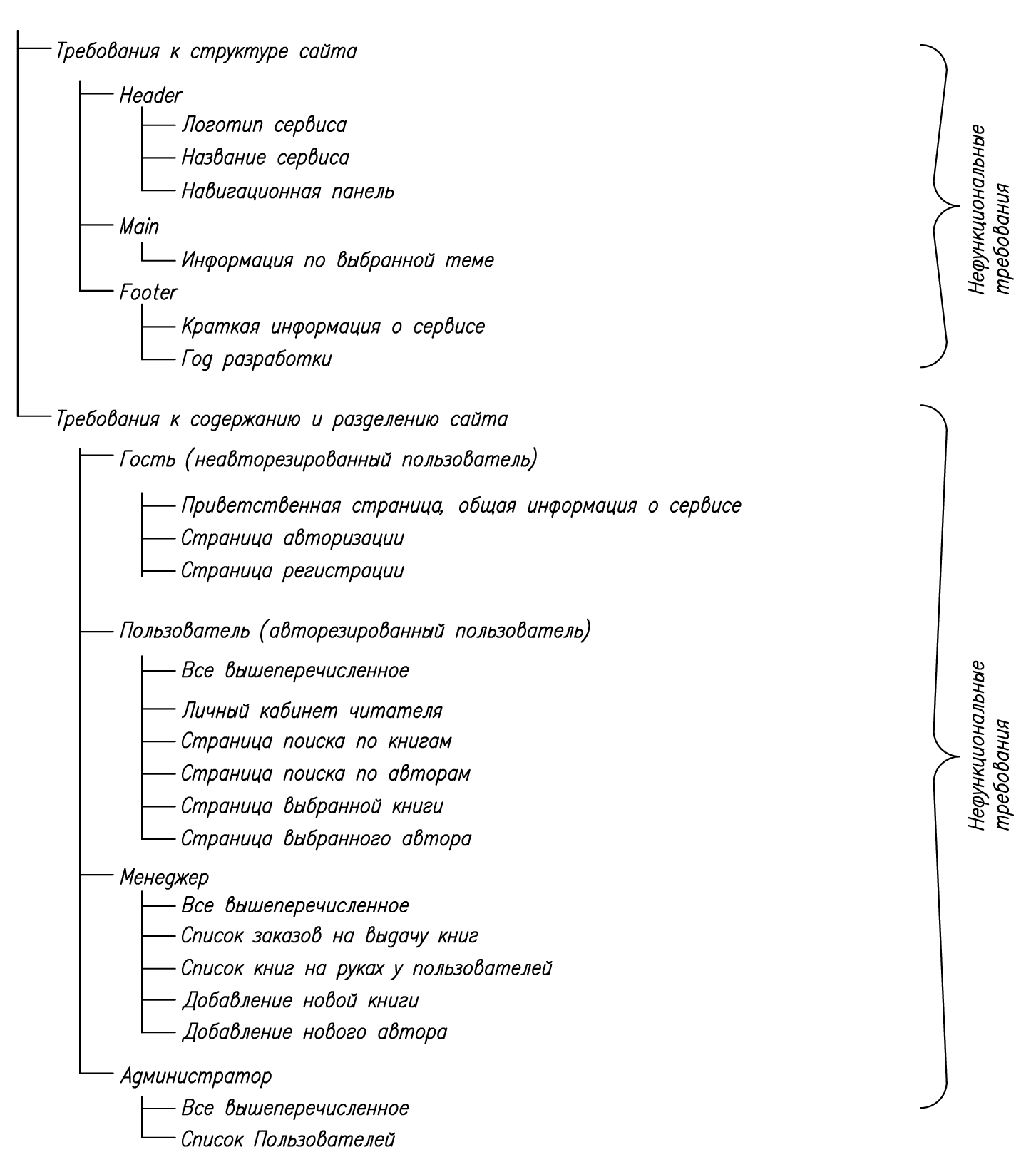
# Общий иерархический список



*Рис.1.1. Общая иерархическая схема, часть 1.*



*Рис.1.2. Общая иерархическая схема, часть 2.*



*Рис.1.3. Общая иерархическая схема, часть 3.*

# Требования и функции системы

| **URS**  **ID** | **Описание «рамки решения»** | **FRS**  **ID** | **Описание «рамки проекта»** |
| --- | --- | --- | --- |
| **Функция: *Название функции*** | | | |
| 1 | Система обеспечивает возможность пользователям зарегистрировать свой аккаунт | 101 | Создание меню регистрации с вводом: имени аккаунта, пароля, электронного адреса. |
| 102 | Подтверждение электронной почты. |
| 104 | Интеграция с сервисом reCaptcha. |
| 2 | Система обеспечивает возможность пользователям изменить свой аккаунт | 201 | Создание меню изменения данных аккаунта, ввода ФИО, электронной почте, телефона, загрузки фотографии пользователя. |
| 202 | Подтверждение номера телефона. |
| 3 | Система обеспечивает возможность пользователям просмотра авторов и книг | 301 | Создание страницы просмотра книг, создание фильтров поиска по авторам, по жанрам, по названию книги. |
| 302 | Создание страницы просмотра авторов, создание фильтра поиска по жанру, имени автора. |
| 303 | Создание страницы каждого отдельного автора, с перечислением всех написанных книг данного автора. |
| 4 | Система обеспечивает возможность пользователям заказа книг | 401 | Создание страницы оформления заказа, и отправки заказа в работу библиотекарю. |
| 5 | Система обеспечивает возможность своевременно напоминать пользователям о необходимости возврата книг | 501 | Система следит за временем нахождения книги на руках у пользователя. |
| 502 | Система осуществляет интеграцию с почтовым сервисом gMail, для отправки уведомлений пользователям о необходимости возврата книг. |
| 6 | Система обеспечивает возможность библиотекарям удобной работы с заказами книг от пользователей. | 603 | Создание страницы с выводом заказов всех пользователей, фильтров по пользователям, по состоянию заказов, по времени отправки. |
| 604 | Создание страницы работы с заказом, изменения его статуса. |
| 7 | Система обеспечивает возможность библиотекарям пополнять базу данных новыми книгами и новыми авторами. | 701 | Создание меню добавления новых авторов. |
| 702 | Создание меню добавления новых книг и привязки их к авторам. |
| 8 | Система обеспечивает администратору работать со списком пользователей. | 801 | Создание страницы со списком пользователей. |
| 802 | Создание страницы изменения данных каждого пользователя, присвоения пользователю ролей. |

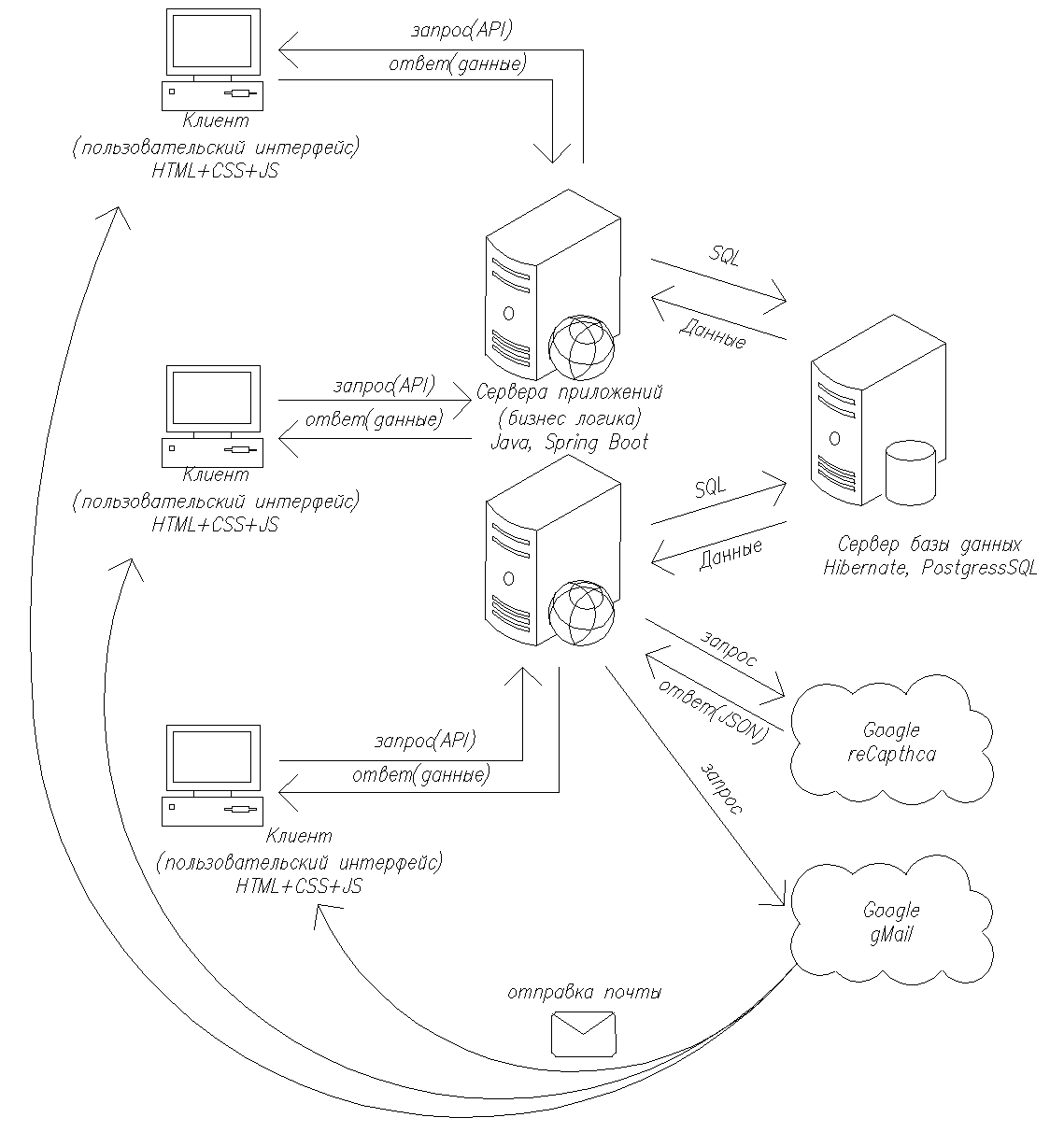
# Объем реализации поставленных требований в патче v1.0

В патче v1.0 будут реализованы не все требования, указанные в предыдущем пункте, приведем ввиде таблицы реализованные требования:

|  |  |  |
| --- | --- | --- |
| **URS**  **ID** | **FRS**  **ID** | **Описание «рамки проекта»** |
| 1 | 101 | Создание меню регистрации с вводом: имени аккаунта, пароля, электронного адреса. |
| 102 | Подтверждение электронной почты. |
| 104 | Интеграция с сервисом reCaptcha. |
| 2 | 201 | Создание меню изменения данных аккаунта, ввода ФИО, электронной почте, телефона, загрузки фотографии пользователя. |
| 3 | 301 | Создание страницы просмотра книг, создание фильтров поиска по авторам, по жанрам, по названию книги. |
| 302 | Создание страницы просмотра авторов, создание фильтра поиска по жанру, имени автора. |
| 303 | Создание страницы каждого отдельного автора, с перечислением всех написанных книг данного автора. |
| 4 | 401 | Создание страницы оформления заказа, и отправки заказа в работу библиотекарю. |
| 6 | 603 | Создание страницы с выводом заказов всех пользователей, фильтров по пользователям, по состоянию заказов, по времени отправки. |
| 604 | Создание страницы работы с заказом, изменения его статуса. |
| 7 | 701 | Создание меню добавления новых авторов. |
| 702 | Создание меню добавления новых книг и привязки их к авторам. |
| 8 | 801 | Создание страницы со списком пользователей. |
| 802 | Создание страницы изменения данных каждого пользователя, присвоения пользователю ролей. |

# Архитектура системы

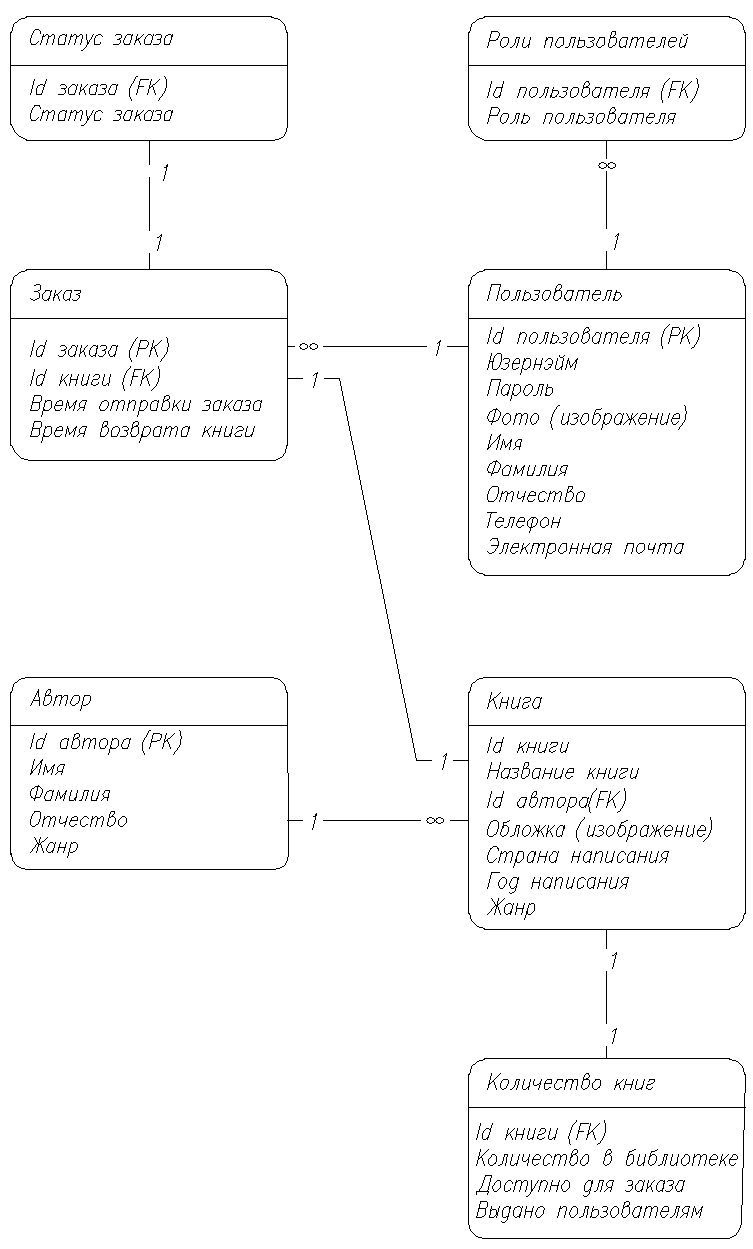
Ниже приведем архитектурную схему клиент-серверного приложения. Для надежной работы системы предусматривается установка двух серверов. Для проверки пользователей предусматривается использование сервиса Google reCaptcha, для отправки почты пользователям системы используется Google Gmail.



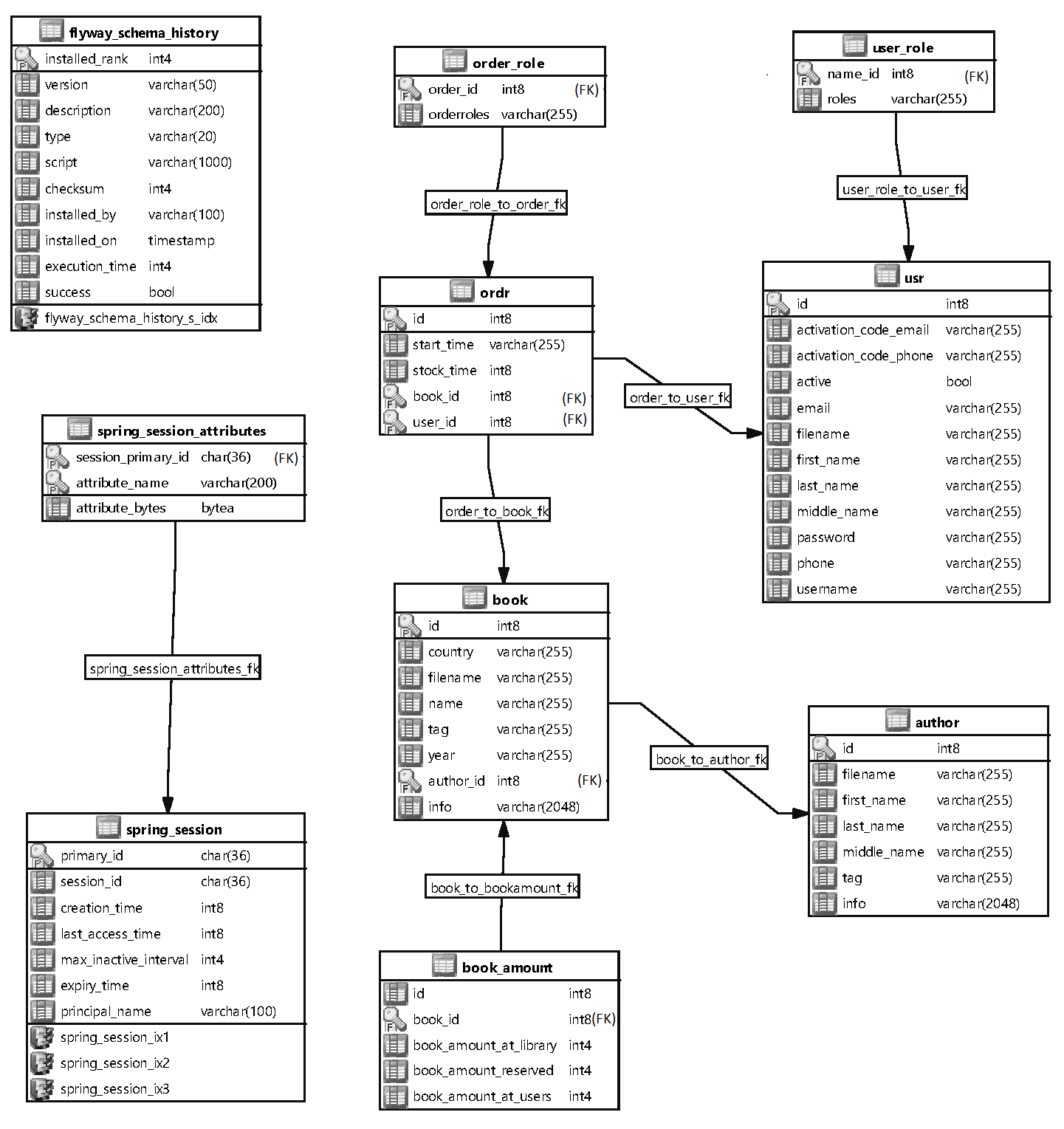
*Рис. 2. Архитектура системы.*

# Модель БД

Для хранения данных в разрабатываемом приложении используется СУБД PostgreSQL, приведем логическую и физическую диаграммы БД.



*Рис. 3. Логическая диаграмма БД.*

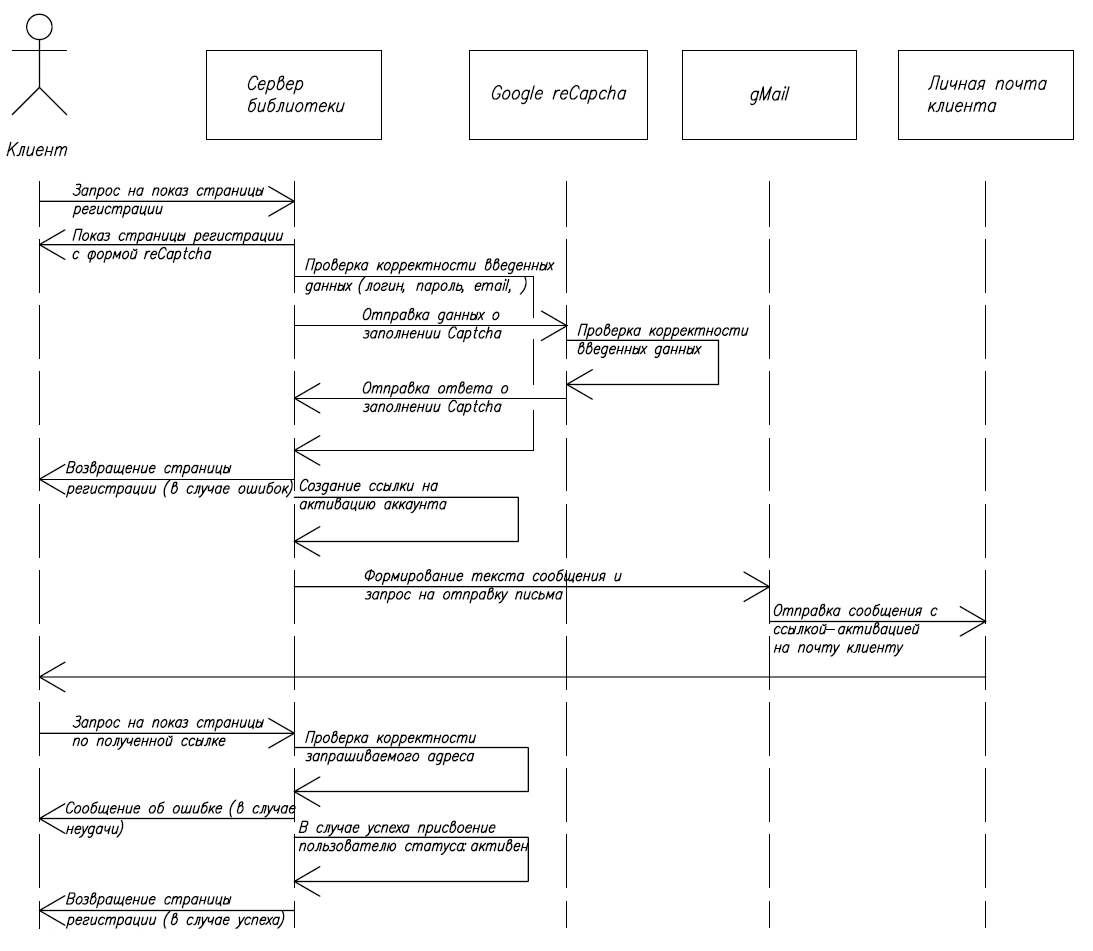
*Рис. 4. Физическая диаграмма БД. СУБД – PostgreSQL.*

# Логика системы

Необходимые для реализации функции системы были описаны выше. В этом пункте мы остановимся на логике работы некоторых частей разрабатываемого приложения.

Во время регистрации пользователь заполняет своё имя, электронную почту, вводит пароль (+повтор пароля для проверки). Система проверяет, записи ли уже пользователь с таким именем и если нет, система формирует уникальную ссылку для подтверждения аккаунта и высылает письмо с ним на указанную пользователем почту. Когда пользователь проходит по ссылке из письма программа считает почту верифицированной и регистрация пользователя на этом заканчивается.

В системе есть возможность изменять дополнять и изменять данные пользователя, а именно: заполнять ФИО, телефон, изменять пароль, изменять электронную почту, добавлять свою фотографию. При изменении электронной почты на новую почту так же направляется письмо для подтверждения новой электронной почты.



*Рис. 5.1. Диаграмма взаимодействий при регистрации пользователя.*

Система позволяет пользователям со статусом “Библиотекарь” добавлять в систему новые данные, а именно новых авторов и новые книги.

При добавлении автора нужно указать: имя(обязательное поле), фамилию(обязательное поле), отчество/второе имя(необязательное поле), жанр(на данный момент записывается 1 жанр, в дальнейшем планируется реализация данного поля в виде перечисления), а также загрузить файл изображения для портрета автора.

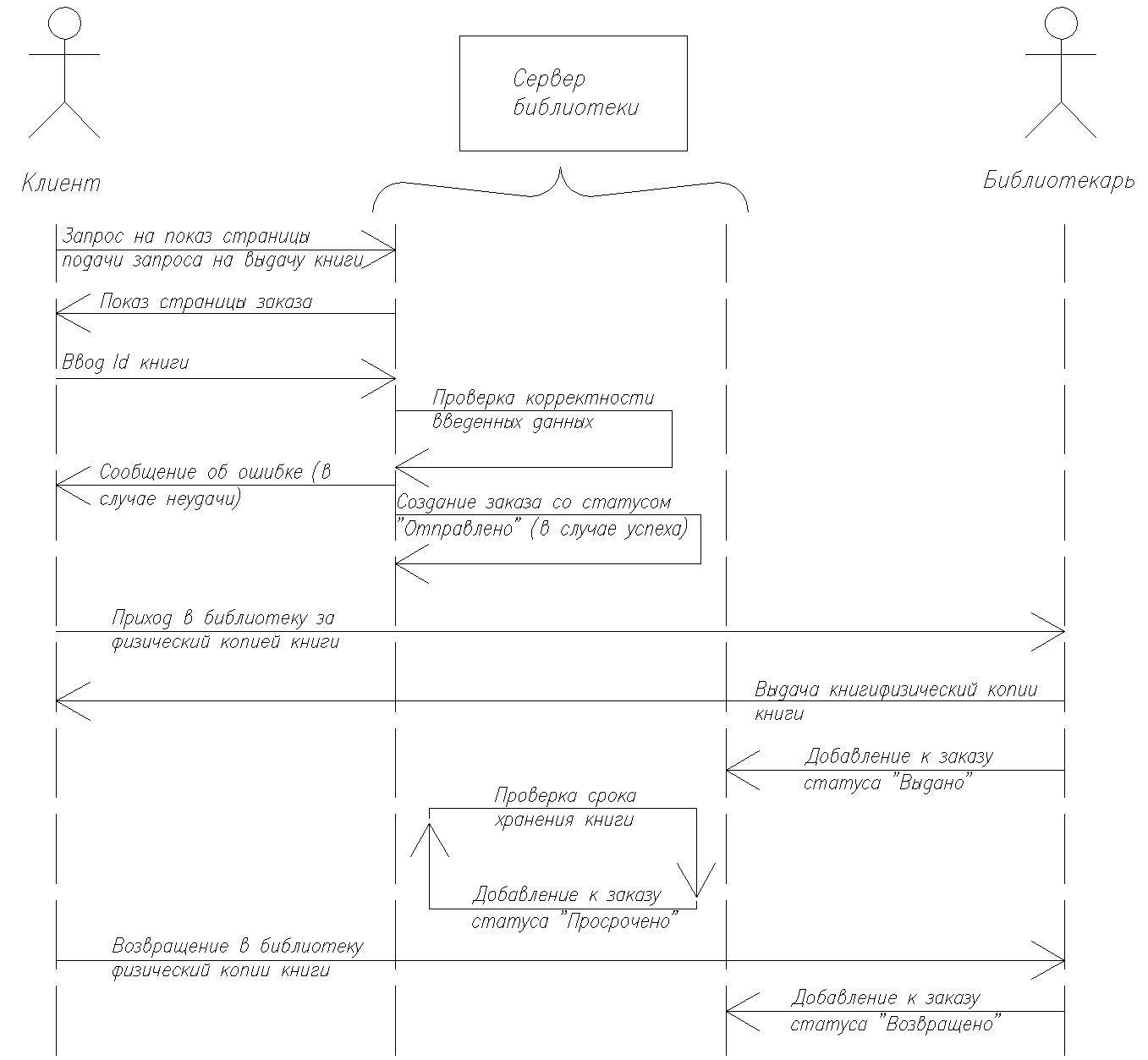
При добавлении книги нужно указать: название (обязательное поле), id автора (обязательное поле), год написания, страну написания, жанр (на данный момент записывается 1 жанр, в дальнейшем планируется реализация данного поля в виде перечисления), а также загрузить изображение для обложки книги.

При просмотре списка авторов и списка книг пользователь может фильтровать выборки пользуясь различными фильтрами: фильтр по имени автора, фильтр по названию книги, фильтр по жанрам.

Для заказа на выдачу книги пользователь вводит желаемый Id книги, далее осуществляется проверка сервером правильности введенного Id и наличия книги в библиотеке, и если обе проверки проходят успешно, то формируется новый заказ с Id выбранной книги, Id пользователя и статусом “Выбрано”. Далее пользователь отправляет заказ в работу, при этом к статусу заказа добавляется статус “Отправлено” и ставится дата отправки заказа. Далее библиотекарь при выдаче бумажной книги пользователю добавляет к заказу статус “Выдано” и ставится срок выдачи книги пользователю. Во время пребывания книги на руках у пользователя система производит проверку срока хранения книги, и в случае его истечения добавляет статус “Просрочено”. При возвращении пользователем книги в библиотек заказу устанавливается статус “Возвращено”.

Для удобной работы библиотекарям с запросами пользователей предусмотрена страница, отображающая запросы всех пользователей, показывающая Id заказов, Id книги, Id пользователя, статус заказа. Библиотекарь может менять статус или удалять заказы пользователей,

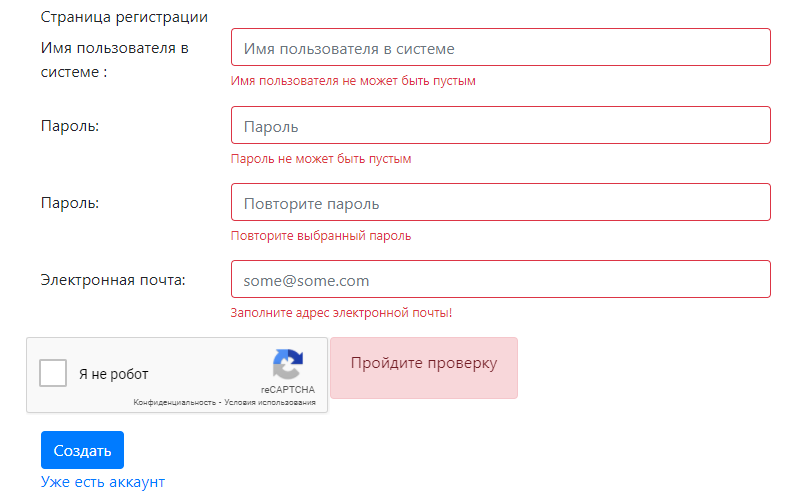
Для присвоения пользователю ролей в приложении предусмотрена роль администратора с отдельной вкладкой “Кабинет администратора”, на данной вкладке администратор видит список пользователей и может поменять данные пользователей, назначит или лишить пользователей их ролей, а также удалить пользователей.



*Рис. 5.2. Диаграмма взаимодействий при заказе книг.*

# Пользовательский интерфейс

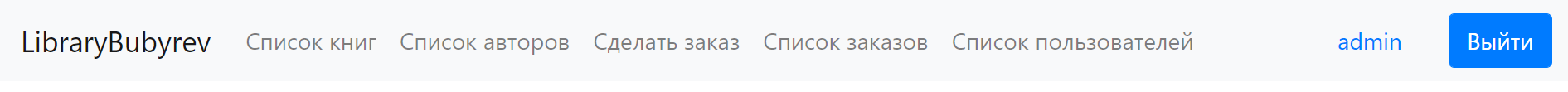
При первом входе в приложении пользователю следует зарегистрироваться, указав имя пользователя (в дальнейшем используется для авторизации пользователя), пароль, электронную почту и пройдя проверку.



*Рис. 6.1. Форма регистрации с валидацией данных.*

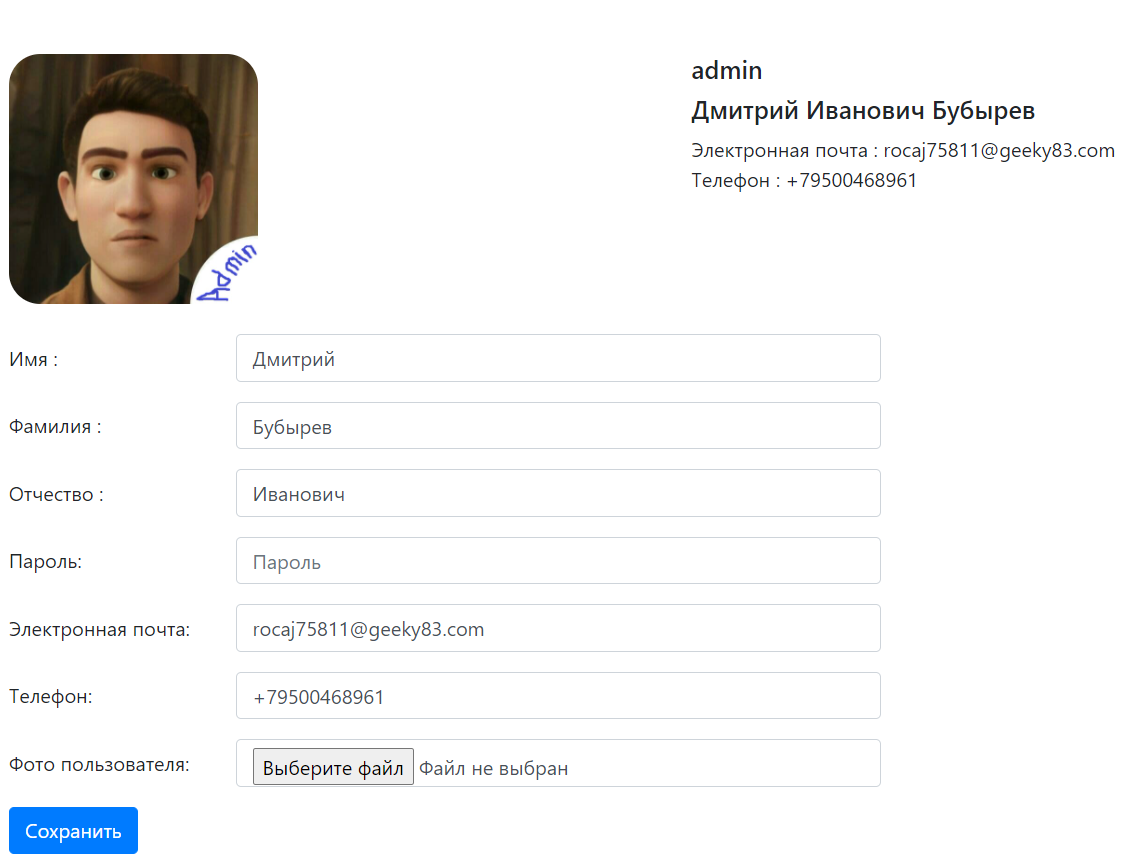
После заполнения данных в форме регистрации на указанную пользователем почту приходит письмо со ссылкой для активации аккаунта, пользователь переходит по ней и аккаунт считается успешно активированным.

Переход между разделами сервиса осуществляется с помощью навигационного меню в верхней части каждой страницы.



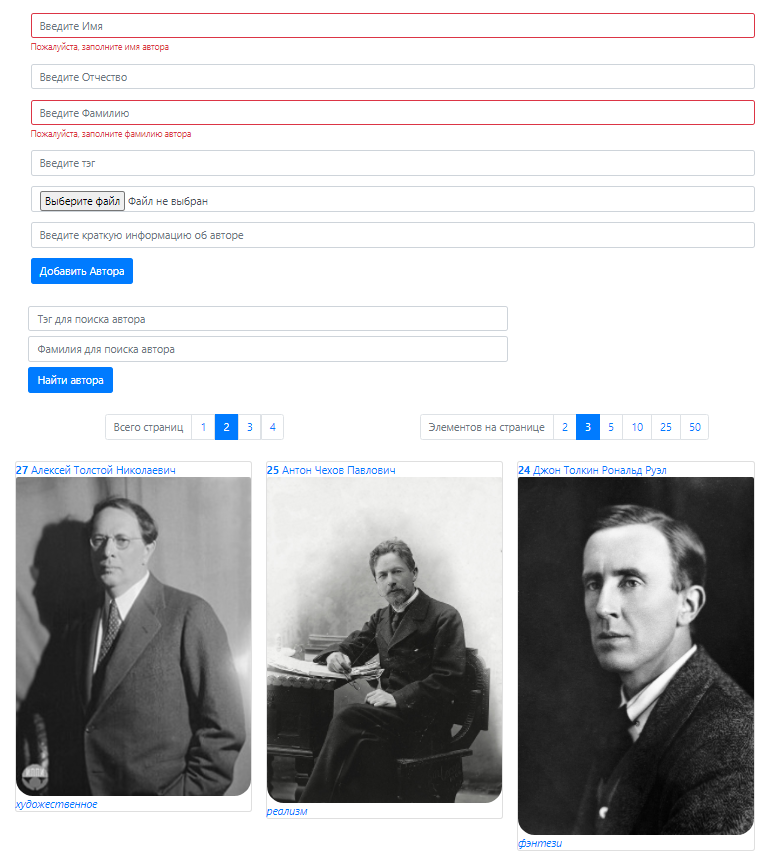
*Рис. 6.2. Меню навигации.*

В дальнейшем пользователь может дополнить/изменить информацию о себе в разделе редактирования профиля.

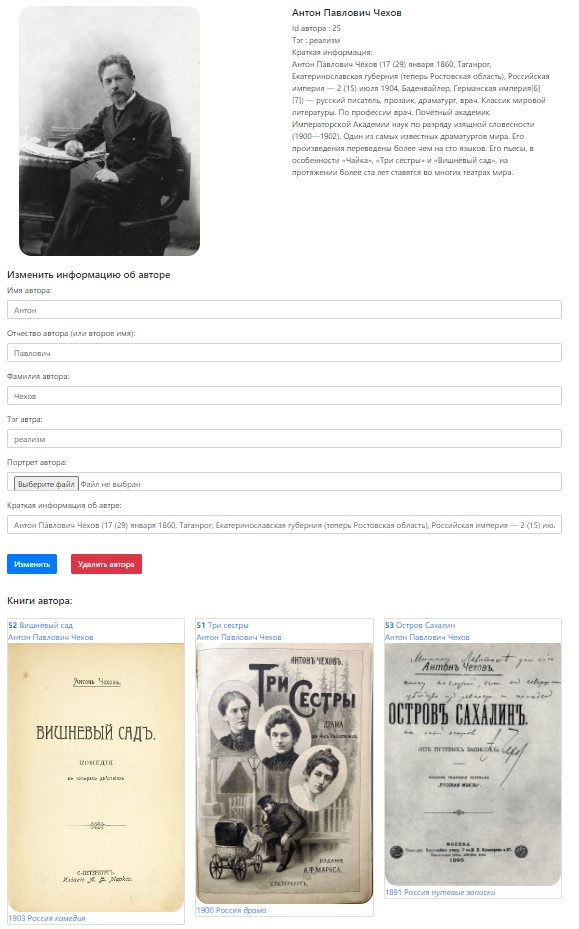


*Рис. 6.3. Форма изменения личных данных.*

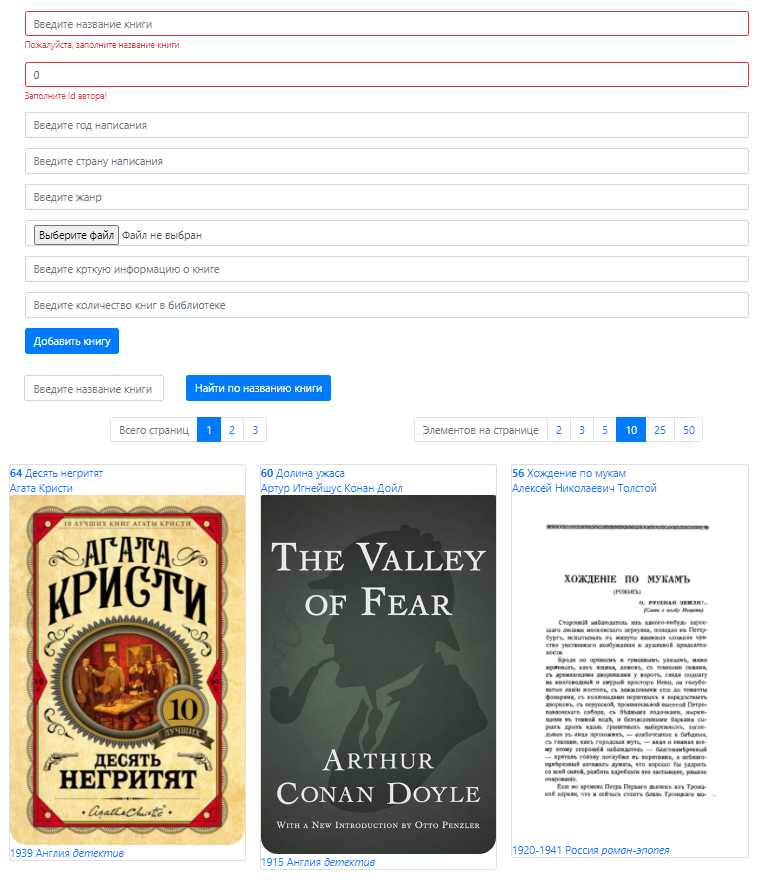
Показ базы данных авторов и книг реализована в виде карточек, возможностью добавлять новые книги и новых авторов обладают пользователи с ролью “Библиотекарь”. Пользователь может перейти на страницу автора/книги для просмотра подробной информации.



*Рис. 6.4. Страница добавления новых авторов (с валидацией) и отображения существующих.*

**

*Рис. 6.5. Страница отображения информации о выбранном авторе, правами изменения/удаления информации обладает пользователь с ролью “Библиотекарь”.*

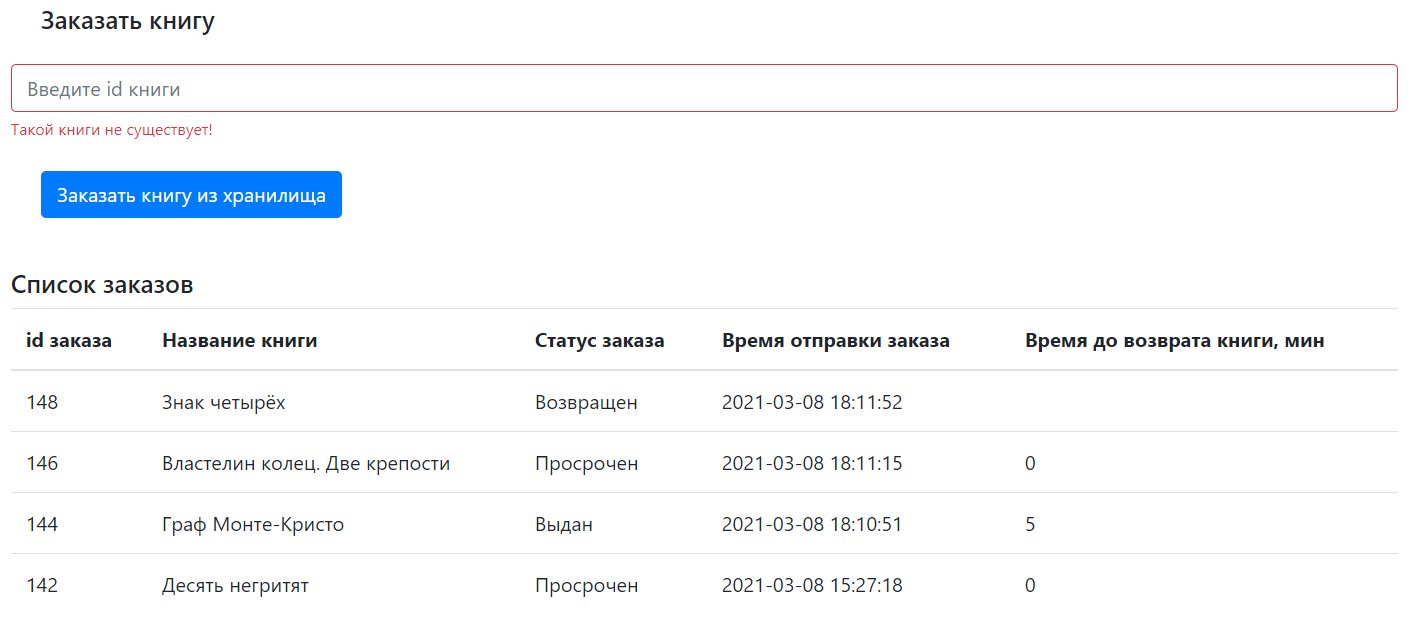


*Рис. 6.6. Страница добавления новых книг (с валидацией) и отображения существующих.*

**

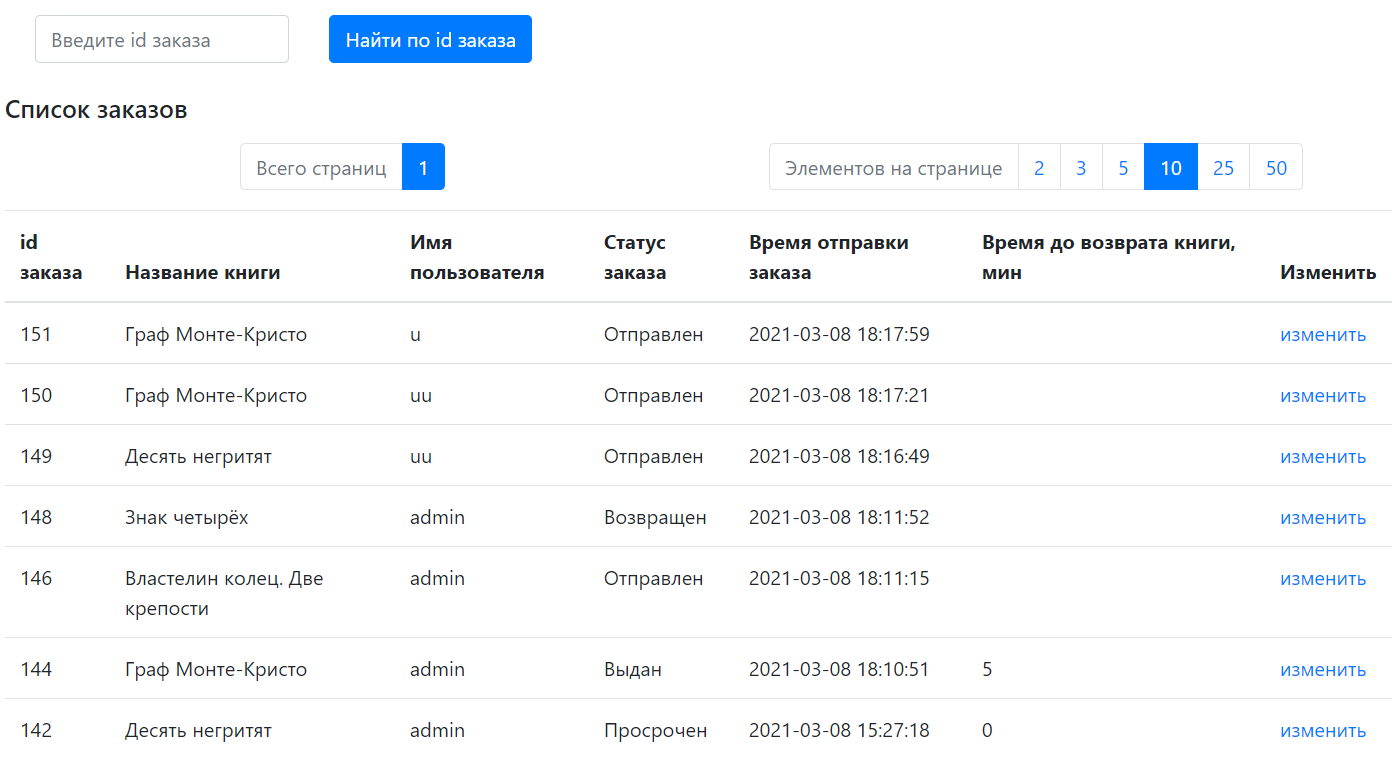
*Рис. 6.7. Страница отображения информации о выбранной книге, правами изменения/удаления информации обладает пользователь с ролью “Библиотекарь”.*

В форме добавления заказа производится проверка наличия в базе данных книги с таким Id, в дальнейшем будет реализован и запрос на проверку количества книг в библиотеке. Форма добавления заказа:



*Рис. 6.8. Страница обработки заказов пользователей.*

Показ списка заказов осуществляется в отдельной вкладке, доступной только библиотекарям, для редактирования заказов необходимо пройти по ссылке “изменить”. “Столбец время до возврата книги” будет доработан в будущем.

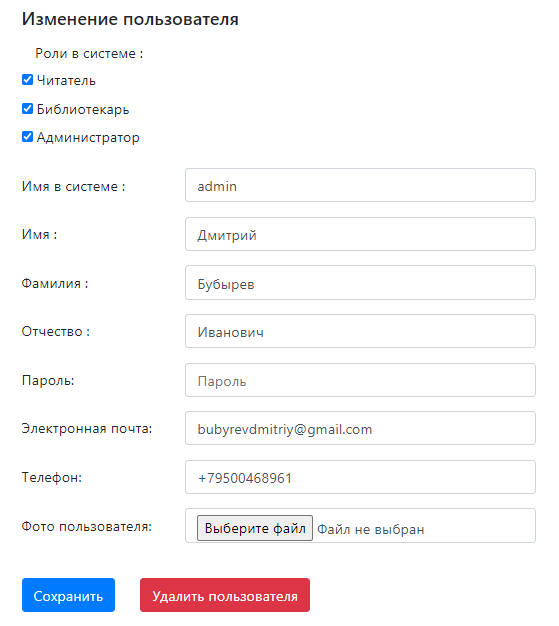


*Рис. 6.9. Страница обработки заказов пользователей.*

**

*Рис. 6.10. Страница обработки заказов пользователей (доступна библиотекарям).*

Показ списка пользователей осуществляется в отдельной вкладке, доступной только администратору, для редактирования пользователя необходимо пройти по ссылке “изменить”. Администратор может изменять информацию о пользователе, присваивать ему или лишать его ролей, а также может удалить пользователя



*Рис. 6.11. Страница редактирования пользователей (доступна администраторам).*

# Заключение

В ходе работы было успешно запроектировано клиент-серверное приложение по автоматизации работы библиотеки.

Выпущенное приложение демонстрирует работоспособное приложение, хоть и с немного ограниченным функционалом.

Были успешно решены все поставленные задачи, а именно:

1. Регистрация и авторизация пользователей.
2. Просмотр существующих авторов и книг.
3. Добавление новых авторов и книг.
4. Создание заказов на выдачу книг.
5. Обработку заказов на выдачу книг.

В дальнейшем планируется развивать сервис добавляя новые возможности и развивая уже имеющиеся. Для последующих патчей остаются такие важнейшие элементы:

1. Реализация жанра/тэга в сущностях “Автор” и “Книга” как список значений (сейчас реализован как одно единственное значение).
2. Интеграция с телеграмм-ботом для проверки указанного пользователем телефона.
3. Рассылка в телеграмме напоминаний о необходимости о возвращения книг.
4. Создание отметок “нравится” для книг.
5. Создание подписок пользователей на авторов.

# Литература

1. Вигерс Карл, Битти Джой В41 Разработка требований к программному обеспечению. 3-е изд., дополненное /Пер. с англ. — М. : Издательство «Русская редакция» ; СПб. : БХВ-Петербург, 2014. — 736 стр. : ил.
2. Java 8. Руководство для начинающих / Герберт Шилдт. - М.: Вильямс, 2015. - 720 c.
3. Лонг Джош. Java в облаке. Spring Boot, Spring Cloud, Cloud Foundry. – СПб.: Питер, 2019 – 624 с.: ил. – (Серия “Бестселлеры” O’Relly).
4. Бауэр К., Кинг Г., Грегори Г, Java Persistance API и Hibernate/пер. с англ. Д. А. Зинкевича; под науч. Ред. А. Н. Киселева. М.: ДМК Пресс, 2017. 632с.:ил.
5. Моргунов, Е. П. Язык SQL. Базовый курс: учеб.-практ. пособие / Е. П. Моргунов; под ред. Е. В. Рогова, П. В. Лузанова; Postgres Professional. — М., 2017. — 257 с.
6. Робсон Э., Фримен Э. Изучаем HTML, XHTML и CSS. 2-е изд. — СПб.: Питер, 2014. — 720 с.: ил. — (Серия «Head First O’Reilly»). ISBN 978-5-496-00653-8