

Министерство образования Республики Беларусь
Учреждение образования «Белорусский государственный университет
информатики и радиоэлектроники»

Факультет инженерно-экономический
Кафедра экономической информатики
Дисциплина «Программирование сетевых приложений»

«К ЗАЩИТЕ ДОПУСТИТЬ»
Руководитель курсового проекта
магистр. эконом. наук, старший
преподаватель

_____._____.2021 **А.А. Иванов**

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

К курсовому проекту

на тему:

«РАЗРАБОТКА СИСТЕМЫ УЧЕТА КНИГ В БИБЛИОТЕКЕ»

БГУИР КП 1-40 01 02-08 015 ПЗ

Выполнил студент группы 910101
Морозова Марина Евгеньевна

(подпись студента)

Курсовой проект представлен на
проверку_____._____.2021

(подпись студента)

Минск 2021

СОДЕРЖАНИЕ

Введение.....	5
1 Анализ работы библиотеки	7
1.1 Описание работы библиотеки.....	7
1.2 Основные понятия, связанные с библиотечным делом	8
2 Постановка задачи и обзор методов её решения.....	11
3 Функциональное моделирование на основе стандарта IDEF0	13
4 Информационная модель системы и её описание.....	18
5 Модели представления системы и их описание.....	21
5.1 Диаграмма вариантов использования	21
5.2 Диаграмма развёртывания.....	21
5.3 Диаграмма классов.....	22
5.4 Диаграмма компонентов.....	23
5.5 Диаграмма состояний процесса добавления акта	23
5.6 Диаграмма последовательности процесса добавления акта	24
6 Описание алгоритмов реализующих бизнес-логику проектируемой системы учета продаж компьютерной техники.....	25
6.1 Алгоритм работы магазина	25
6.2 Алгоритм просмотра списка книг	25
7 Руководство пользователя.....	26
8 Результаты тестирования разработанной системы.....	31
Заключение	32
Список использованных источников	33
ПРИЛОЖЕНИЕ А (обязательное) Блок-схемы алгоритма.....	34
ПРИЛОЖЕНИЕ Б (обязательное) Модели представления системы	36

ВВЕДЕНИЕ

Каждый из нас с детства слышит, как важен порядок. Однако, вырастая, многие оказываются буквально в плену у хаоса, который царит в их квартире, на офисном столе и, как следствие, в голове. Иметь чистый ум, находится в состоянии творческого вдохновения – так важно и так просто. Становясь во главе какой-либо организации, пусть и небольшой, люди, не привыкшие к систематичности, чёткому структурированию информации, наверняка не преуспеют в выбранной ими деятельности.

Какую организацию мы бы ни рассматривали, маленькую или крупную, существует проблема наиболее эффективной работы данными в этой организации. Ведь любая работа, связанная с обработкой некой информации, предполагает хранение, обработку, обеспечение её обмена.

Раньше люди использовали для хранения документов сейфы, шкафы, поиск зачастую вёлся вручную. Но большинство современных организаций предпочитают компьютеризированные базы данных и программы, позволяющие эффективно хранить, быстро извлекать нужную информацию и управлять большими объёмами данных.

Учет в любой организации или на предприятии служит для планирования, управления и контроля за всеми событиями процесса производства. Тем самым позволяя обеспечивать структуру и отслеживать любые изменения данных.

Для обеспечения порядка и для организованности в библиотечном деле нужна система учета книг в библиотеке, позволяющая частично автоматизировать процесс учета данных книг, хранить все сведения о них и полностью заменить устаревшие технологии приема документов в библиотечные фонды, которые использовались на протяжении долгого времени [1].

Таким образом, цель данного курсового проекта заключается в оптимизации системы учёта книг в библиотеке. Для её достижения сформулированы следующие задачи:

- Проанализировать заданную предметную область
- Реализовать клиент-серверное взаимодействие
- Создать базу данных
- Создать веб-приложение
- Создать простой и удобный пользовательский интерфейс.

1 АНАЛИЗ РАБОТЫ БИБЛИОТЕКИ

1.1 Значение библиотеки в современном мире

Библиотека, в переводе с греческого – «библио» - книга, «тека» - хранилище, то есть «хранилище книг».

О роли библиотек в жизни людей можно судить уже по тем образным наименованиям, которые им издавна присваивали. Их называли храмами мудрости, памятью человечества, хранилищами сокровищ цивилизации.

Библиотека – это обыкновенное и в то же время удивительное место, потому что в этом помещении живут книги. Мы привыкли к книге, редко думаем о ней, как о чуде, как о сокровище, и бывает, что не всегда ценим и бережем её. Но вдумайтесь, ведь книга до недавнего времени была единственным средством передачи знаний от поколения к поколению. Как только люди изобрели письменность, появилась возможность собирать и накапливать знания.

А знаете ли вы, где появились первые библиотеки? Самые первые появились в Древнем Египте. Они назывались «домами папируса» и «домами жизни». По мнению древних египтян, книги можно сравнить с лекарством, которое делает сильным ум человека, облагораживает его душу [2].

Сегодня многие современные библиотеки мегаполисов значительно преобразились и модернизировались. Эти культурные места могут иметь даже интернет-кафе, где также с пользой можно провести свой досуг. Но в маленьких городах и в сельских глубинках эти учреждения работают еще по старой программе, ввиду чего, к сожалению, их функционирование сводится к минимуму.

В читальных залах и книжном фонде данных учреждений человек будет нуждаться до тех пор, пока существует часть духовной культуры народонаселения. Есть целая категория людей, которая ни за что не променяет шуршащие и пахнущие жизнью пожелтевшие страницы раритетных книг на холодный голубой экран дисплея ПК.

В нашей стране крупнейшей является Национальная библиотека Беларуси. В ней хранится более 8,9 миллионов хранения на различных носителях и самый большой набор баз данных в 110 единиц. Она была помещена на 11-е место в рейтинге фантастических архитектурных работ по версии американского издания Flavorwire (англ.) русск. в 2013 году. В 2011 году включена в российскую книгу «100 самых удивительных достижений современной архитектуры». Библиотека заняла 24-е место в списке 50 самых необычных сооружений мира по версии сайта Village of Joy, упоминается в берлинском издании «Лучшие библиотеки мира» [3].

1.2 Основные понятия, связанные с библиотечным делом

Библиотека – учреждение, собирающее и хранящее произведения печати и письменности для общественного пользования, а также осуществляющее справочно-библиографическую работу.

Библиотечный фонд – упорядоченное собрание документов, которое хранится в библиотеке, в которое включены произведения печати (книги, периодические издания), а также другие документы (диафильмы, микрофильмы, электронные документы).

Информационная система - содержащаяся в базах данных информация и обеспечивающих ее обработку информационных технологий и технических средств. Информационные системы предназначены для хранения, обработки, поиска, распространения, передачи и предоставления информации. Библиотека (или ее компоненты), обладая названными признаками, представляет собой информационную систему.

Документоборот - взаимный обмен документами между библиотеками или с другими учреждениями, осуществляемый без денежных расчетов (безвозмездно) или на заранее оговоренных условиях.

Информационный ресурс - организованная совокупность документированной информации, включающая базы данных, другие совокупности взаимосвязанной информации в информационных системах.

Учёт библиотечного фонда - комплекс операций, обеспечивающий регистрацию и обобщение сведений об объеме, составе, движении, его стоимости, являющихся основой финансовой отчетности о фонде, а также составной частью мероприятий по обеспечению его сохранности.

Фонд электронных документов -упорядоченный массив документов в цифровой форме, для использования которого необходимы средства вычислительной техники или иные специализированные устройства для воспроизведения текста, звука, изображения. К электронным документам относятся электронные документы на объемных носителях, сетевые документы локального доступа, сетевые документы удаленного доступа.

Формирование библиотечного фонда - создание, постоянное обновление, развитие и поддержание фонда в работоспособном состоянии; собирательное понятие для всех технологических процессов по моделированию библиотечного фонда, комплектованию библиотечного фонда, приему, учету библиотечного фонда, библиотечной обработке, 88 размещению фонда, расстановке фонда.

Хранение документов - процесс, обеспечивающий расположение документов в установленном месте хранилища, направленное на предохранение документов от порчи, повреждений, загрязнений с целью их последующего нахождения и использования.

Экземпляр документа как учетная единица библиотечного фонда - каждый документ библиотечного фонда, включаемый в фонд или выбывающий из него, считающийся отдельной единицей хранения.

Учет библиотечного фонда является основой для государственного статистического учета, отчетности библиотеки, планирования ее деятельности, обеспечения сохранности фонда, контроля над наличием и

движением документов. К ведению учета библиотечного фонда предъявляются следующие требования:

- достоверность сведений о фонде;
- обязательная фиксация в соответствующих учетных формах каждого поступления и исключения из библиотечного фонда;
- надежность приемов и форм учета в условиях автоматизированного и ручного режимов работы;
- соответствие номенклатуры показателей первичного учета библиотечного фонда по объему и составу аналогичным показателям государственной статистики.

Неиспользуемый фонд - часть фонда, состоящая из документов, на которые не поступали запросы абонентов библиотеки. В составе неиспользуемой части различают пассивную часть фонда, обеспечивающую ему запас надежности, и балласт фонда. В соответствии с общеполорусским классификатором видов экономической деятельности интернет-магазин занимается торговлей. Учет поступлений документов в фонд осуществляется дифференцированно, путем их подразделения на документы постоянного, длительного и временного хранения.

Учету для временного хранения (без инвентарных номеров) подлежат документы, содержащие информацию краткосрочного значения, в связи с чем они исключаются из фонда библиотеки через непродолжительный период времени. Также учет фонда библиотеки включает оценку стоимости документов, составляющих фонд, исключение и подведение итогов движения фонда, проверку его целостности [1].

2 ПОСТАНОВКА ЗАДАЧИ И ОБЗОР МЕТОДОВ ЕЕ РЕШЕНИЯ

Основной целью данного курсового проекта является облегчение процесса учёта книг в библиотеке.

Задачей курсового проекта является разработка системы учета продаж компьютерной техники – создание веб-приложения с организацией клиент-серверного взаимодействия, взаимодействия с базой данных. Для ее достижения сформированы следующие задачи:

- Создать базу данных на PostgreSQL 12.0.
- Спроектировать и создать иерархии классов.
- Использовать инкапсуляцию, перегрузку методов, переопределение методов, статические методы, обработку исключительных ситуаций.
- Предусмотреть механизм авторизации.
- Создать простой и удобный пользовательский интерфейс.

Бизнес логика должна быть реализована только на серверной части приложения. Разработка системы учет книг предусмотрена для трех ролей: читатель, библиотекарь и администратор.

Для роли читателя предусмотреть следующие функции: просмотр списка книг, поиск книг, запрос на бронирование книги, возможность следить за текущим состоянием личной библиотеки и отслеживать даты возврата, просмотр контактов библиотек, смена локали.

Для роли библиотекаря предусмотреть следующие возможности: работа со списком книг (добавление, поиск, удаление, изменение), работа с читателями (блокировка), работа с взятыми книгами (закрытие заказа), смена локали, возможность сохранения заказов в файл.

Для роли администратора предусмотреть полный доступ ко всему: работа со списком книг (добавление, поиск, удаление, изменение), работа с читателями (блокировка, удаление, редактирование), работа с взятыми книгами (закрытие заказа), смена локали, возможность сохранения заказов в файл.

Важным аспектом системы учета книг является база данных, которая хранит в себе всю информацию. Поэтому, необходимо обеспечить удобное взаимодействие.

3 ФУНКЦИОНАЛЬНОЕ МОДЕЛИРОВАНИЕ НА ОСНОВЕ СТАНДАРТА IDEF0

Для более детального процесса рассмотрения учета книг в библиотеке была разработана функциональная модель по стандарту IDEF0. IDEF0 – методология функционального моделирования и графическая нотация, предназначенная для формализации и описания бизнес-процессов. На рисунке 3.1 представлена контекстная диаграмма «Поставить книгу на учёт».



Рисунок 3.1 – Контекстная диаграмма «Постановка книги на учёт»

В качестве входных данных выступает новое поступление документа. Управляющими факторами являются нормативные указания и ГОСТы а также порядок постановки на учёт. Механизмами являются библиотекарь, программное и техническое обеспечения. Выходными данными будет структурированная информация в виде акта поступления на учёт.

Проектирование производится посредством декомпозиции (decomposition) – деления моделируемой функции на функции-компоненты.

Функция (function) – деятельность, процесс или преобразование, идентифицируемое глаголом или глагольной формой, которая описывает, что должно быть выполнено.

Функции на диаграммах отображаются в виде блоков (box) – прямоугольников, содержащих имя и номер. Блоки диаграмм связываются стрелками. Стрелка (arrow) – направленная линия, состоящая из одного или нескольких сегментов, которая моделирует канал, передающий данные от источника (начальная точка стрелки) к потребителю (конечная точка с «наконечником»). Имеется 4 класса стрелок: входная стрелка (input arrow) – класс стрелок, которые отображают вход функционального блока, то есть данные, которые преобразуются функцией. Входные стрелки связываются с левой стороной блока; выходная стрелка (output arrow) – класс стрелок, которые отображают выход функционального блока, то есть данные, произведенные функцией. Выходные стрелки связываются с правой стороной блока; управляющая стрелка (control arrow) – класс стрелок, отображающих управление, то есть условия, при которых выход блока будет правильным. Управляющие стрелки связываются с верхней стороной блока; стрелка механизма (mechanism arrow) – класс стрелок, которые отображают средства, используемые для выполнения функции. Стрелки механизма связываются с нижней стороной блока.

В ходе декомпозиции блока «Поставить книгу на учёт» он был разбит на 4 подпроцесса:

- получить книгу;
- оформить учётную карточку;
- передать документ на хранение;
- Добавить учётную карточку в регистрационную книгу

Результат декомпозиции представлен на рисунке 3.2.

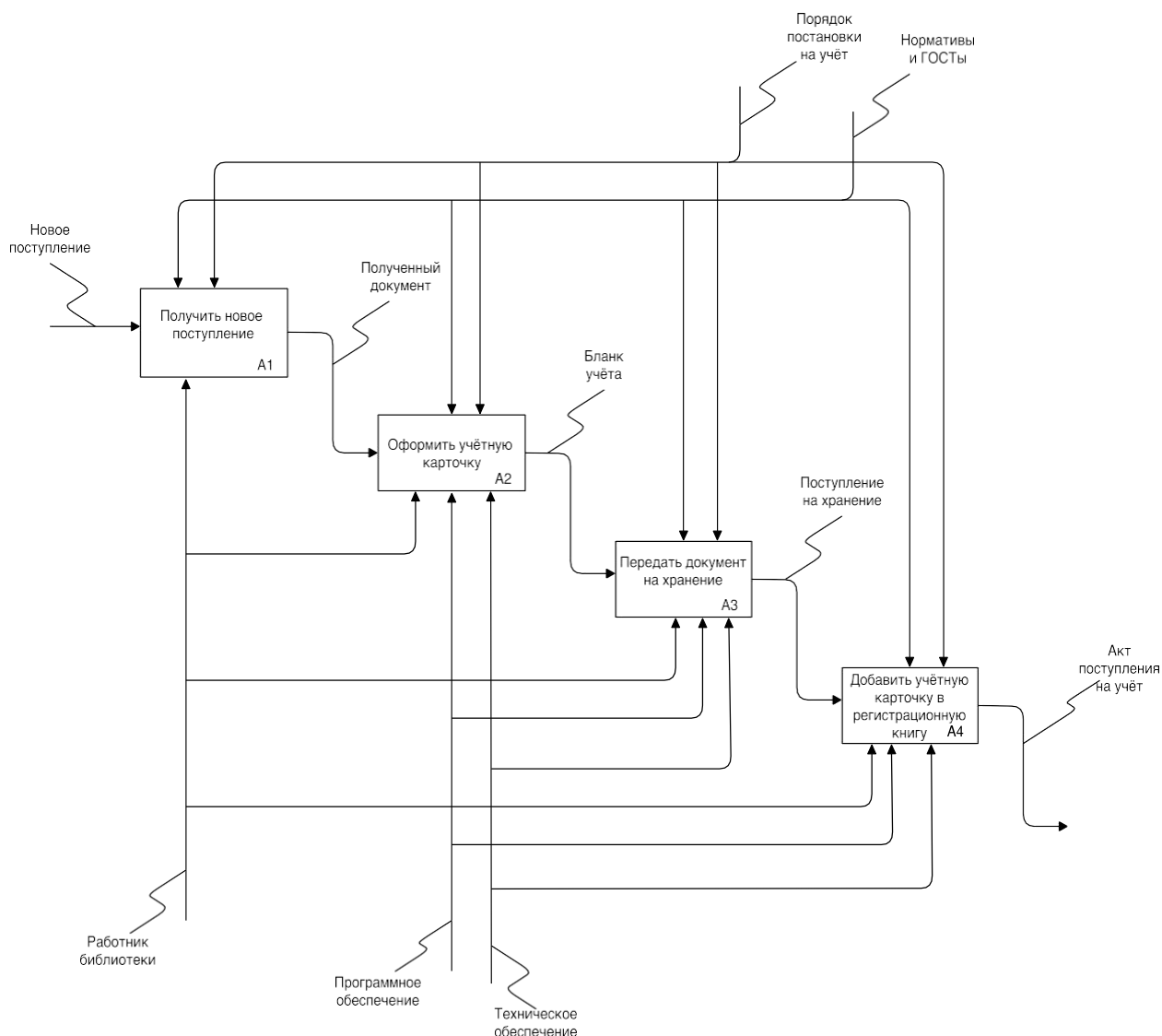


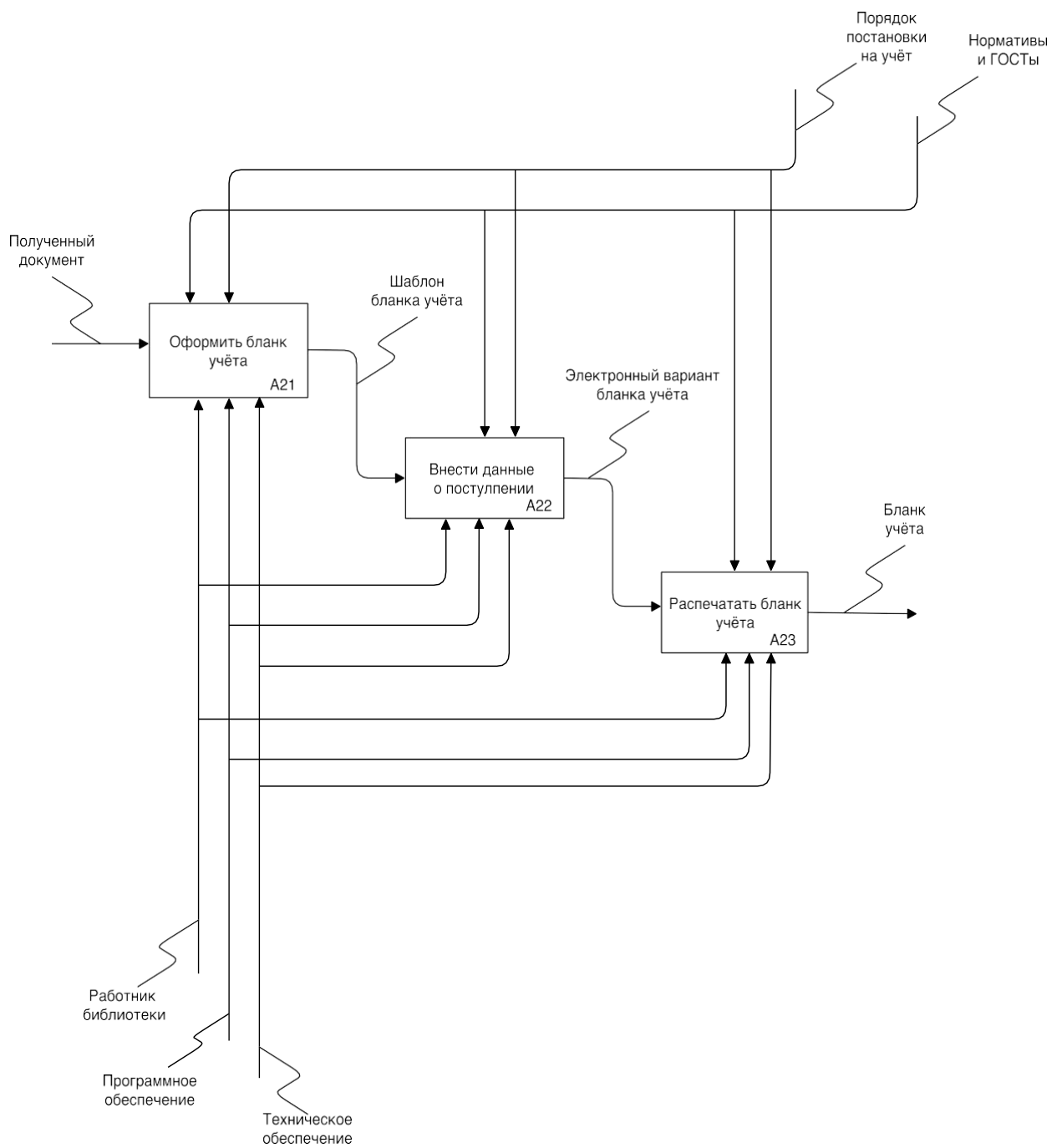
Рисунок 3.2 – Процесс оформления акта поступления на учёт

Была произведена декомпозиция блока «Оформить учётную карточку» (Рисунок 3.3). В результате были выделены следующие блоки:

- оформить бланк учёта;
- внести данные о книге;
- распечатать бланк учёта.

В свою очередь блок «Внести данные о книге» был разделён на следующие процессы (графическое представление представлено на рисунке 3.4):

- внести информацию об авторе;
- внести название документа;
- внести дату издания.



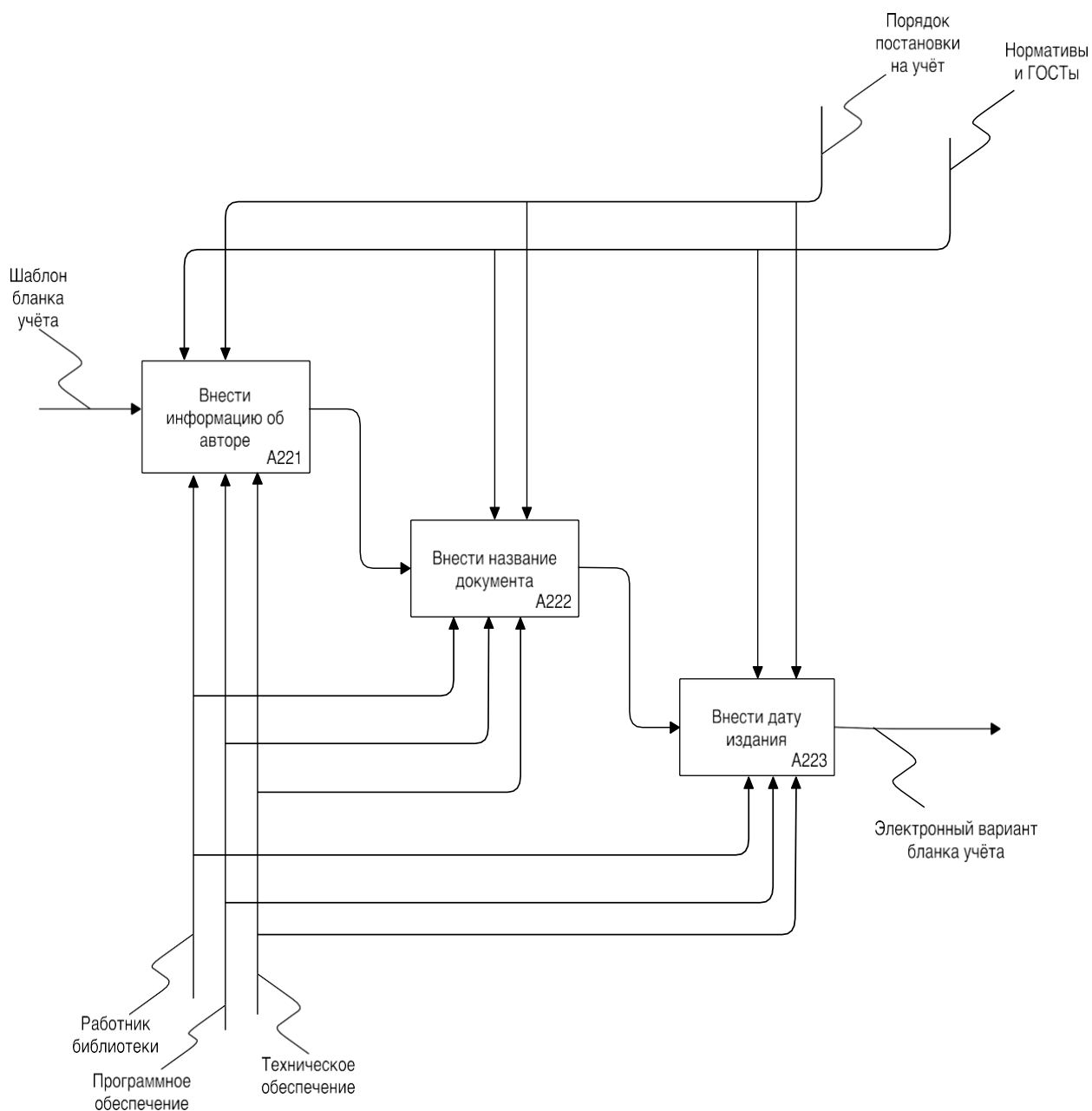


Рисунок 3.4 – Внесение данных об издании

В данном разделе была построена модель процесса становления нового поступления на учёт.

4 ИНФОРМАЦИОННАЯ МОДЕЛЬ СИСТЕМЫ И ЕЕ ОПИСАНИЕ

Что такое информационная модель? Это набор различных диаграмм типа "сущность – связь", которые определяют организацию и связи используемой в системе информации, полноценно отображающей процесс её работы.

Суть существования информационной модели заключается в том, что данные и связи между ними интерпретируются без каких-либо противоречий, что необходимо для совместного использования и управления целостностью данных.

Информационные модели типа "сущность - связь" - логические модели, которые не зависят от реализации баз данных и методов доступа. Для правильного построения модели необходима информация, причём как можно более точная. Чем больше предоставленные данные отвечают реальным показателем, тем эффективней применяется модель на практике. Процесс выполнения моделирования состоит из этапов, каждый из которых должен заканчиваться конкретным измеряемым результатом:

- определение сущностей;
- определение зависимостей между сущностями;
- задание первичных и альтернативных ключей;
- определение атрибутов сущностей;
- приведение модели к требуемому уровню нормальной формы;
- переход к физическому описанию модели: назначение соответствий
имя сущности - имя таблицы, атрибут сущности - атрибут
таблицы;
- генерация базы данных.

Этап за этапом в результате была спроектирована информационная модель системы учёта новых поступлений в библиотеку.

Одним из требований курсового проекта является приведение базы данных к третьей нормальной форме, которая подразумевает задание вместе с отношением системы функциональных зависимостей. Данная нормальная форма включает в себя то, что любое поле любой записи хранит только одно значение, не ключевое поле полностью зависит от ключа, отсутствует транзитивная функциональная зависимость.

Информационная модель системы представлена на рисунке 4.1.

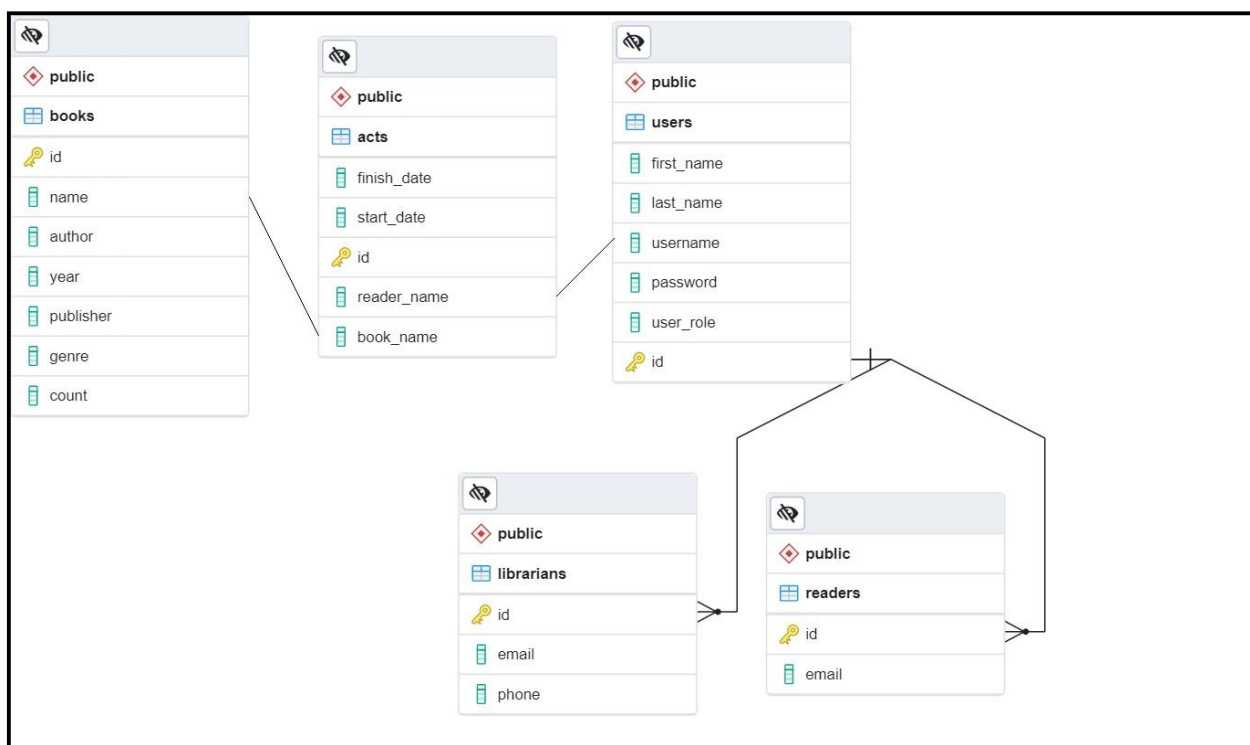


Рисунок 4.1 – Физическая модель базы данных

Учитывая архитектуру разрабатываемого приложения, выделено 6 сущностей: пользователи, читатели, библиотекари, работники, акты и книги, которые отображены на диаграмме.

Сущность Пользователь хранит в себе номер пользователя, информацию о ФИО, логине, пароле и роли пользователя. Используя логин и пароль, пользователь может авторизоваться. Пользователь может иметь одну из ролей: администратор, библиотекарь или читатель, в зависимости от которой определяются доступные функции.

Пользователи и читатели, пользователи и библиотекари связаны между собой как один к одному. Т.е. одному пользователю соответствует либо один работник, либо один покупатель.

Сущность Библиотекарь, помимо информации о пользователе, дополнительно хранит номер телефона и email библиотекаря.

Сущность Читатель, помимо информации о пользователе, дополнительно хранит email читателя. Связана с сущностью Акт как один ко многим.

Сущность Книга хранит в себе название книги, ее автора, год издания, издательство, жанр и количество экземпляров. Связана с сущностью Акт как один ко многим.

Сущность Акт является одной из главных в базе данных, и хранит в себе информацию о id взятой книги, имя читателя, дату взятия книги и дату требуемого возврата.

Данная модель находится в третьей нормальной форме, так как она находится во второй нормальной форме и не один не ключевой атрибут не находится в транзитивной функциональной зависимости от потенциального ключа, также отсутствует дублирование информации (сущности ссылаются друг на друг при помощи внешних ключей).

5 МОДЕЛИ ПРЕДСТАВЛЕНИЯ СИСТЕМЫ И ИХ ОПИСАНИЕ

5.1 Диаграмма вариантов использования

Диаграмма вариантов использования (use case) описывает отношения между системой и пользователями. Отражает, что система может делать и с кем будет взаимодействовать.

Диаграмма вариантов использования представлена в приложении Б.

На рисунке А.1 приведена диаграмма вариантов использования. Действующими лицами в ней являются пользователи с ролями «Читатель», «Библиотекарь», «Администратор».

Пользователь, авторизовавшийся как читатель, имеет самый ограниченный функционал: просмотр списка книг, поиск книг, запрос на бронирование книги, возможность следить за текущим состоянием личной библиотеки и отслеживать даты возврата, просмотр контактов библиотек, смена локали.

Для библиотекаря существуют следующие возможности: работа со списком книг, работа с читателями, работа с взятыми книгами (закрытие заказа), смена локали, возможность сохранения заказов в файл.

Для роли администратора предусмотреть полный доступ ко всему: работа со списком книг (добавление, поиск, удаление, изменение), работа с читателями (блокировка, удаление, редактирование), работа с взятыми книгами (закрытие заказа), смена локали, возможность сохранения заказов в файл.

5.2 Диаграмма состояний

Диаграмма состояний предназначена для отображения состояний объектов, имеющих сложную модель поведения. Она отражает все возможные состояния системы в порядке их следования.

На рисунке 5.2 представлена диаграмма состояний авторизации работника. В данном случае объектом является «Процесс авторизации в системе», который в результате работы системы может находиться в

различных состояниях, переходя из одного в другое, вследствие совершения над объектом каких-либо действий.

Начальным состоянием данного процесса является запуск программы, после чего будет выполнено действие открытие окна авторизации. Затем ожидается ввод данных: логина и пароля. После нажатия кнопки «Войти» система проверяет правильность данных: все поля должны существовать в базе данных. При неправильном вводе информации система выдает соответствующее сообщение и возвращает на форму авторизации. При правильном вводе всех данных осуществляется открытие меню, что является конечным состоянием данного процесса.

В том случае, если пользователь решит не авторизоваться в системе, то будет выполнено действие перенаправления на главную страницу, что также является конечным состоянием данного процесса.

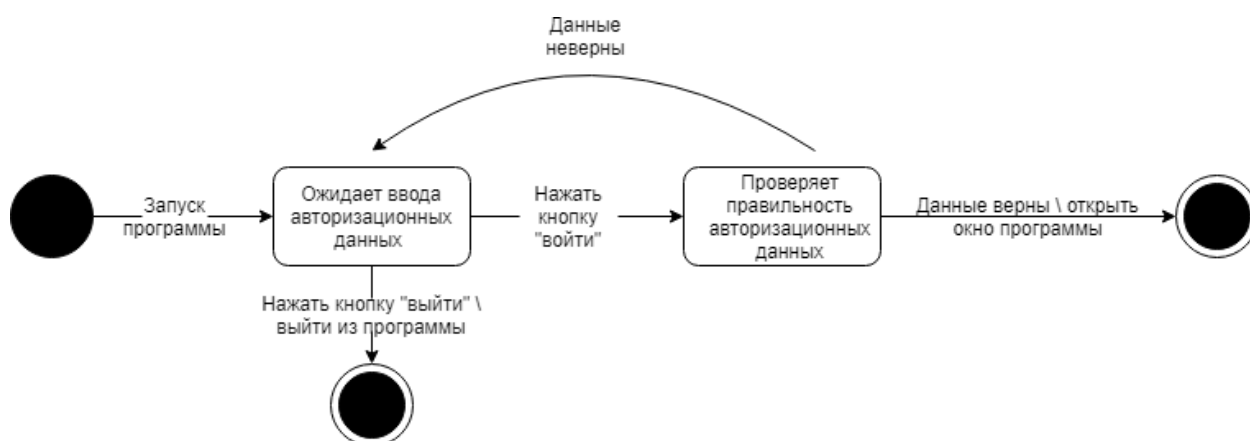


Рисунок 5.2 – Диаграмма состояний процесса авторизации в системе

5.3 Диаграмма классов

Диаграмма классов представляет отражает все типы в системе, а также связи между ними.

Диаграмма классов моделей представлена на рисунке Б.3.1. Она содержит классы Act, Reader, User, Librarian, Book, которые реализуют интерфейсы Table и Entity, а также перечисление RoleEnum, которое реализует абстрактный класс Enum.

На рисунке Б.3.2 представлена диаграмма, на которой перечислены интерфейсы ActRepository, UserRepository, ReaderRepository,

LibrarianRepository, BookRepository, которые в свою очередь наследуют интерфейс CrudRepository.

На рисунке Б.3.3 представлена диаграмма классов UserServiceImpl, LibrarianServiceImpl, ReaderServiceImpl, которые в свою очередь реализуют интерфейсы UserService, LibrarianService, ReaderService соответственно. Также на диаграмме представлены классы GeneralUserDto, UserDto, LibrarianDto, ReaderDto, UserConverter.

На рисунке Б.3.4 представлена диаграмма классов ActServiceImpl, который реализует интерфейс ActService, ActDto, ActConverter.

На рисунке Б.3.6 представлена диаграмма классов BookServiceImpl, который реализует интерфейс BookService, BookDto, BookConverter.

На рисунке Б.3.7 представлена диаграмма классов контроллеров: ActController, UserController, BookController, RegistrationController, которые используют аннотации RequestMapping, Controller.

5.4 Диаграмма компонентов

Диаграмма компонентов отражает компоненты программного обеспечения, а также их связь. Главным компонентом является Server.jar, который использует в своей работы остальные пакеты.

5.5 Диаграмма развертывания

Диаграмма развёртывания показывает расположение физических элементов системы, которые существуют на этапе выполнения. Содержит графические изображения устройств, процессов и связей между ними. Позволяет определить архитектуру системы. Диаграмма содержит 2 устройства – компьютер клиента и компьютер сервера, которые связаны между собой протоколом HTTP.

В данном проекте система представляет собой веб-приложение. Также диаграмма содержит хранилище, которые содержит все данные - базу данных PostgreSQL, которая связана с сервером протоколом JDBC.

5.6 Диаграмма последовательности процесса добавления акта

Диаграмма последовательности описывает поведение системы, отражает временные особенности приема и передачи сообщений.

Диаграмма последовательности процесса добавления акта представлена на рисунке Б.6. Исполнителем является читатель. Процесс добавления акта начинается с запуска приложения, после которого ему необходимо авторизоваться. Пользователь вводит логин и пароль, которые отправляются на сервер. Сервер делает запрос в базу данных и проверяет, существует ли работник с такими учетными данными. В случае успеха, сервер предоставляет читателю доступ к списку доступной литературы, где необходимо выбрать нужную книгу. После выбора пользователя, они отправляются на сервер, который делает запрос в базу данных на добавление акта, и в случае успеха сервер уведомляет пользователя об успешном добавлении, выслав подтверждение добавления.

6 ОПИСАНИЕ АЛГОРИТМОВ, РЕАЛИЗУЮЩИХ БИЗНЕС-ЛОГИКУ СЕРВЕРНОЙ ЧАСТИ ПРОЕКТИРУЕМОЙ СИСТЕМЫ УЧЕТА КНИГ В БИБЛИОТЕКЕ

6.1 Алгоритм работы библиотеки

Блок-схема алгоритма работы библиотеки представлена на рисунке А.1.1. Алгоритм начинается с входа в систему. Если логин и пароль неправильны, пользователь получает сообщение об этом и происходит возврат на страницу авторизации. В случае успешной авторизации пользователь получает доступ к системе – возможности работы с книгами и другое.

6.2 Алгоритм просмотра списка книг

Блок-схема алгоритма просмотра списка книг представлена на рисунке А.1.2. Он, как и алгоритм работы магазина, начинает работу со входа в систему. Если логин и пароль неправильны, пользователь получает сообщение об этом и происходит возврат на страницу авторизации. В случае успешной авторизации пользователь получает доступ к списку книг.

7 РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Для запуска приложения изначально необходимо запустить серверную часть. Для этого необходимо в командную строку ввести команду `java -jar`, после чего указать путь к `server.jar` файлу. В случае успешного запуска в консоли отобразится следующее:

```
Tomcat started on port(s): 8080 (http) with context path ''
Started Application in 13.753 seconds (JVM running for 16.066)
```

Рисунок 6.2 – Сообщение о запуске сервера

После того, как сервер запущен, необходимо в адресной строке браузера ввести <http://localhost:8080/>, после чего откроется главная страница системы учета продаж компьютерной техники, которая имеет следующий вид:



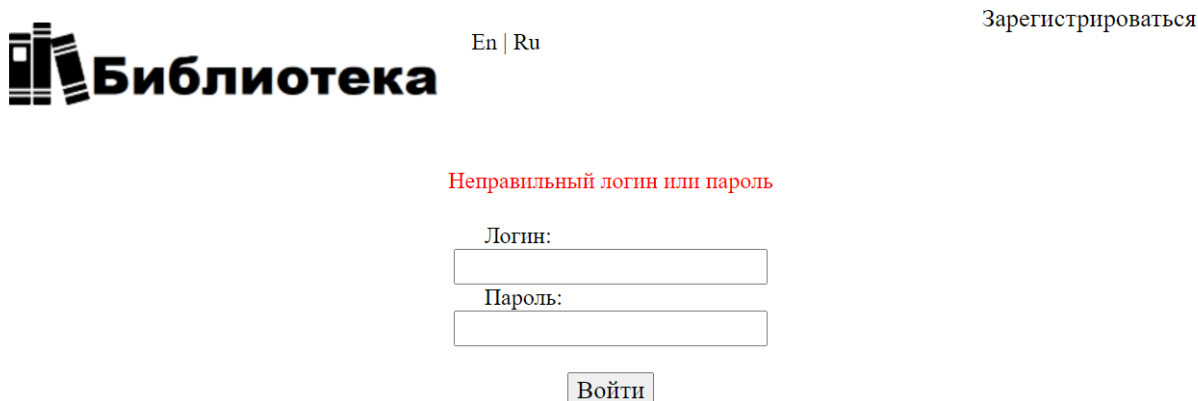
Рисунок 6.3 – Главная страница системы учета

Как видно из рисунка 6.3, неавторизованному пользователю доступны следующие возможности: авторизация, регистрация и смена локали.

Если пользователь ещё не имеет аккаунта, то при нажатии на кнопку «регистрация» он попадает на окно регистрации, где необходимо заполнить форму и создать аккаунт.

В случае успешной регистрации происходит перенаправление на страницу авторизации, где пользователь, используя только что созданный аккаунт, может войти в систему. Для того, чтобы войти в систему, пользователю необходимо в поле логин ввести свой логин, в поле пароль – пароль.

Если введенные пользователем логин и пароль неправильны, отображается сообщение об ошибке и пользователю предлагается ввести учетные данные заново.



The screenshot shows the top of a website with a logo on the left consisting of three books and the word "Библиотека". In the center, there is a language selector "En | Ru". On the right, there is a link "Зарегистрироваться". Below these elements, a red error message "Неправильный логин или пароль" is displayed. Underneath the message are two input fields: the first is labeled "Логин:" and the second is labeled "Пароль:". Below the password field is a button labeled "Войти".

Рисунок 6.7 – Ошибка авторизации

В случае успешной авторизации, пользователь попадает на страницу доступных функций, которая немного отличается в зависимости от роли. Также, после авторизации на каждой странице имеется возможность выхода из системы и возможность смены локали. На рисунке 6.8 показана страница списка книг для покупателя. Доступные покупателю доступна функция «взять книгу» и поиск по каталогу.

Книги

Начните вводить здесь для поиска...

Id	Название	Автор	Год издания	Издательство	Жанр	
3	gfdg	afda	1234	dsafa	fafa	✓

Рисунок 6.8 – Страница доступных функций читателя в списке книг

На рисунке 6.9 показан список читателей для администратора.

Читатели

Начните вводить здесь для поиска...







Id	Имя	Фамилия	Email	Логин	
7	Сергей	Петров	reader@gmail.com	reader	  
9	Марьна	Marozava	Maryna_Marozava@epam.com	maryna	  

Рисунок 6.9 – Страница доступных функций в списке книг для администратора

На рисунке 6.10 показан список доступных функций администратора для работы с библиотекарями.

Библиотекари

[Добавить нового библиотекаря](#)

Начните вводить здесь для поиска...





Id	Имя	Фамилия	Email	Телефон	Логин		
8	Иван	Иванов	ivan@gmail.com	+375444827912	ivan		
5	Петр	Петров	petr@mail.ru	+375123456789	petr		

Рисунок 6.10 – Список доступных функций администратора

В случае, если пользователь имеет роль библиотекаря или администратора, он имеет возможность редактирования информации о книге, для этого необходимо нажать на символ редактирования книги, после которого произойдет перенаправление на страницу изменения книги. Страница редактирования информации о продукте имеет следующий вид:

Редактирование книги

Название:	<input type="text" value="Война и Мир"/>
Автор:	<input type="text" value="Лев Толстой"/>
Год издания:	<input type="text" value="1920"/>
Издательство:	<input type="text" value="Веста"/>
Жанр:	<input type="text" value="Роман"/>
Количество:	<input type="text" value="55"/>
<input type="button" value="Редактировать"/>	

Рисунок 6.11 – Страница редактирования книги

Все пользователи имеют доступ к просмотру заказов. Администратор и библиотекарь имеют возможность закрыть заказ. Страница списка заказов изображена на рисунке 6.12.



Текущие заказы

Id	Название	Читатель	Дата начала	Дата окончания	
21	gfdg	Сергей Петров	2021-12-31	2021-12-01	

Рисунок 6.12 – Страница списка заказов

Вне зависимости от того, какая роль – администратор или библиотекарь, имеется возможность сохранения акта в текстовый файл. С сервера будет загружен текстовый файл, содержащий всю информацию об акте.

Администратор имеет возможность добавить нового библиотекаря. Страница добавления библиотекаря имеет следующий вид:



Добавить нового библиотекаря

Имя	<input type="text"/>
Фамилия	<input type="text"/>
Логин:	<input type="text"/>
Пароль:	<input type="text"/>
Телефон	<input type="text"/>
Email	<input type="text"/>
<input type="button" value="Добавить"/>	

Рисунок 6.13 – Страница добавления библиотекаря

8 РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ РАЗРАБОТАННОЙ СИСТЕМЫ

По итогам разработки системы было проведено мануальное тестирование.

Ручное тестирование – это прямое взаимодействие QA-инженера и приложения. В его процессе можно получить обратную связь о продукте и дизайне пользовательского интерфейса.

Кроме того, тестировщик сможет составить свой отзыв и рекомендации по улучшению ПО, а проведенное тестирование будет свидетельством сравнения ожидаемого и реального результатов.

ЗАКЛЮЧЕНИЕ

В результате выполнения данного курсового проекта был проведен анализ предметной области, а именно работы системы библиотеки. Разработано приложение, основанное на клиент-серверной архитектуре, связь реализована по протоколу HTTP. Сервер реализован с возможностью обработки запросов от нескольких клиентов. Также были выполнены следующие поставленные задачи:

- Создана база данных
- Создано веб-приложение
- Создан простой и удобный пользовательский интерфейс.

Конечное приложение обладает высокой актуальностью в современном мире. Удобный и понятный пользователю интерфейс облегчает работу и создает комфортные рабочие условия.

Разработанное приложение может быть использовано в небольших библиотеках, так как находится на стадии совершенствования, и добавления нового функционала.

В дальнейшем планируется развитие программного продукта, а именно расширение функционала. Например, можно было бы реализовать связь с другими библиотеками, то есть объединить фонды нескольких библиотек в одну систему для более эффективной работы организаций.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Модернизация библиотек [Электронный ресурс]. – Режим доступа : <http://mukcbs.org/modernizaciya-bibliotek-perspektivy-i-vozmozhnosti>.
- [2] Макконнелл, С. Совершенный код. –СПб. : Питер, 2005. –896 с.
- [3] habr.com [Электронный ресурс] – Электронные данные. – Режим доступа: <https://habr.com/ru/post/346396/>.
- [4] belsoft.by [Электронный ресурс] – Электронные данные. – Режим доступа: http://www.belsoft.by/site/ru/news/news-it/rynok_it
- [5] Человек и библиотека [Электронный ресурс]. – Режим доступа : <http://cheloveknauka.com/>.
- [6] Анализ работы библиотек [Электронный ресурс]. – Режим доступа : <https://studfiles.net/preview/5943887/page:2/>.

ПРИЛОЖЕНИЕ А
(обязательное)
Блок-схемы алгоритма

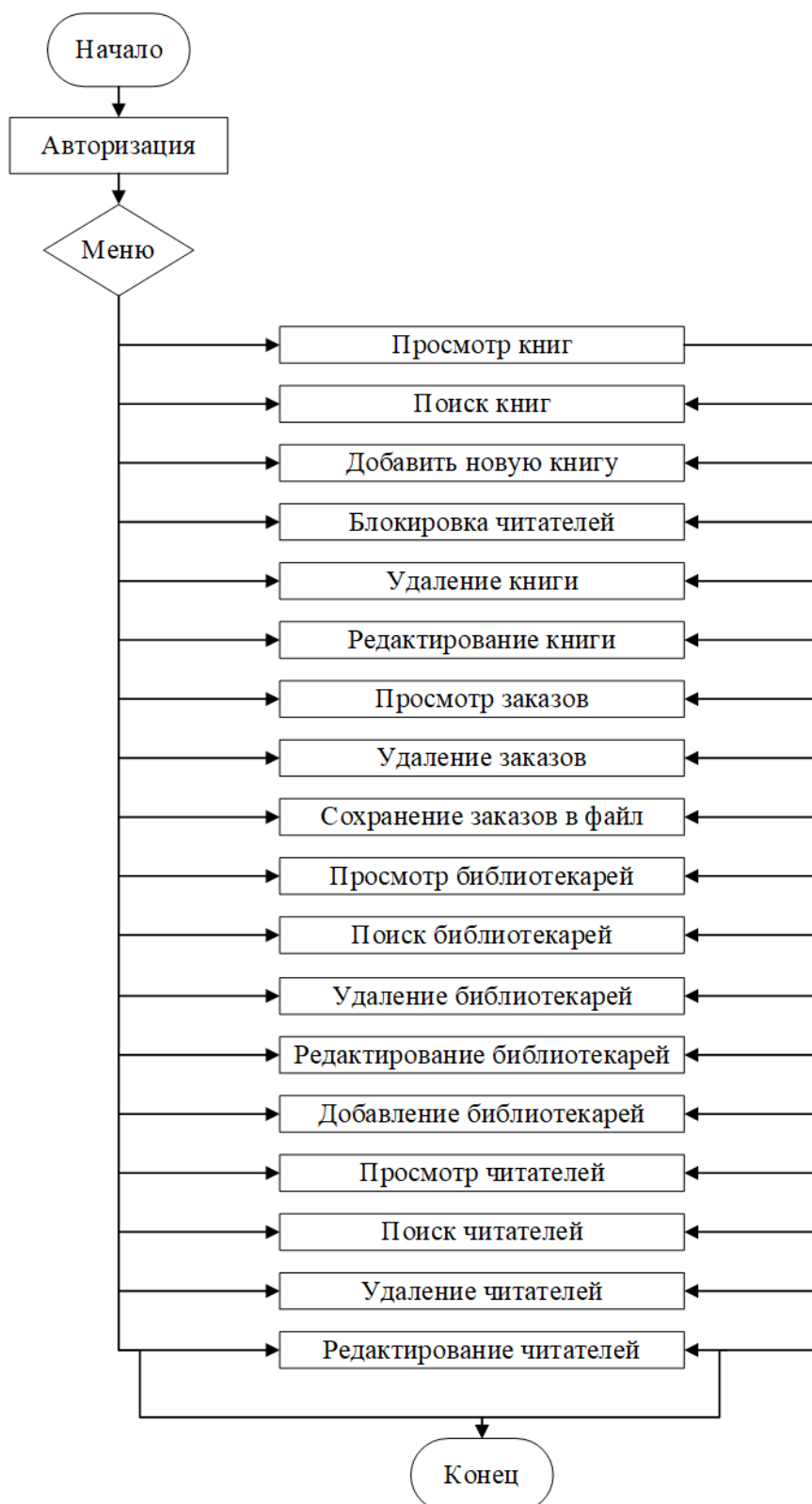


Рисунок А.1.1 – Блок схема алгоритма работы системы

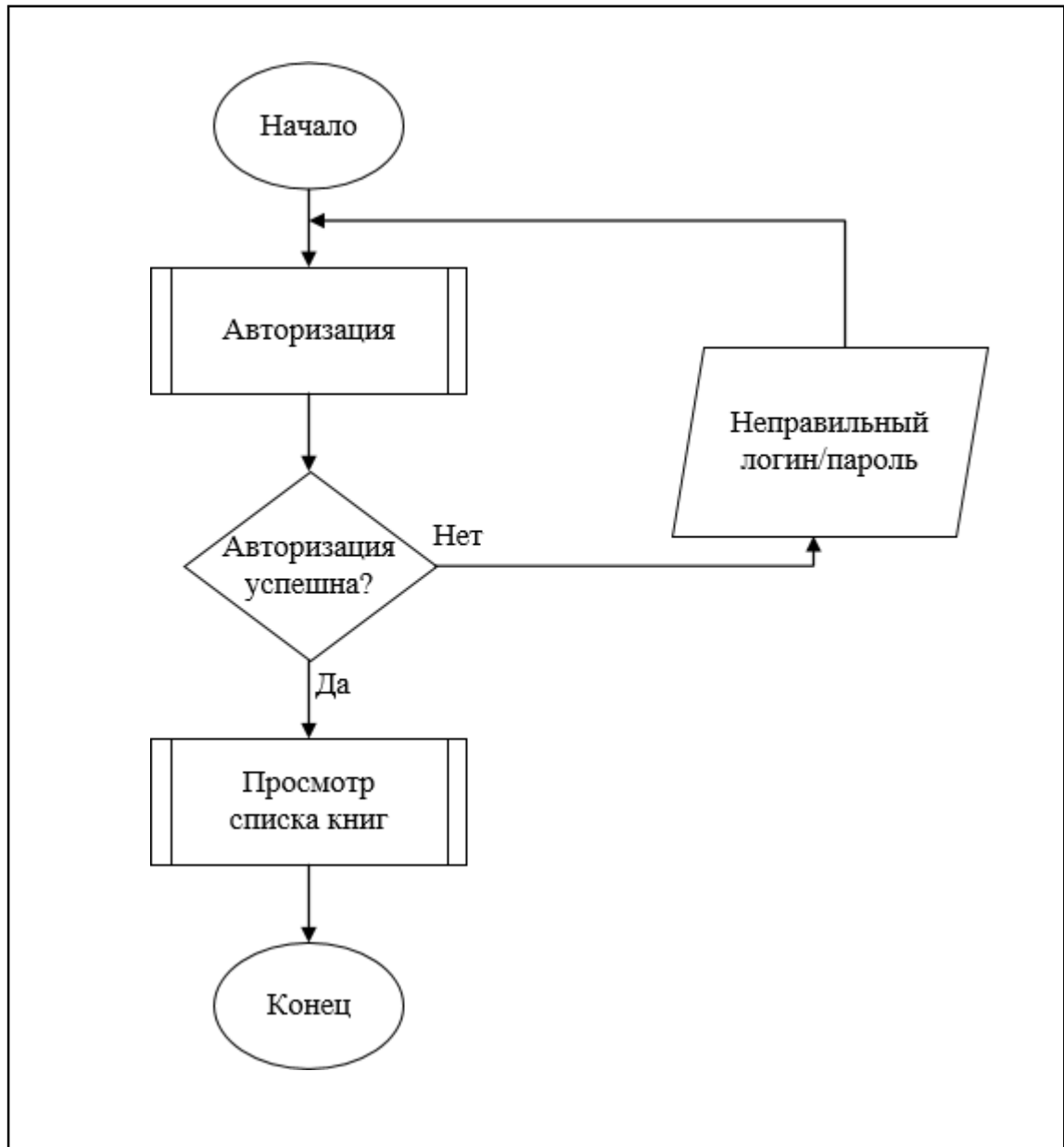


Рисунок А.1.2 – Блок-схема алгоритма просмотра списка книг

ПРИЛОЖЕНИЕ Б **(обязательное)** **Модели представления системы**

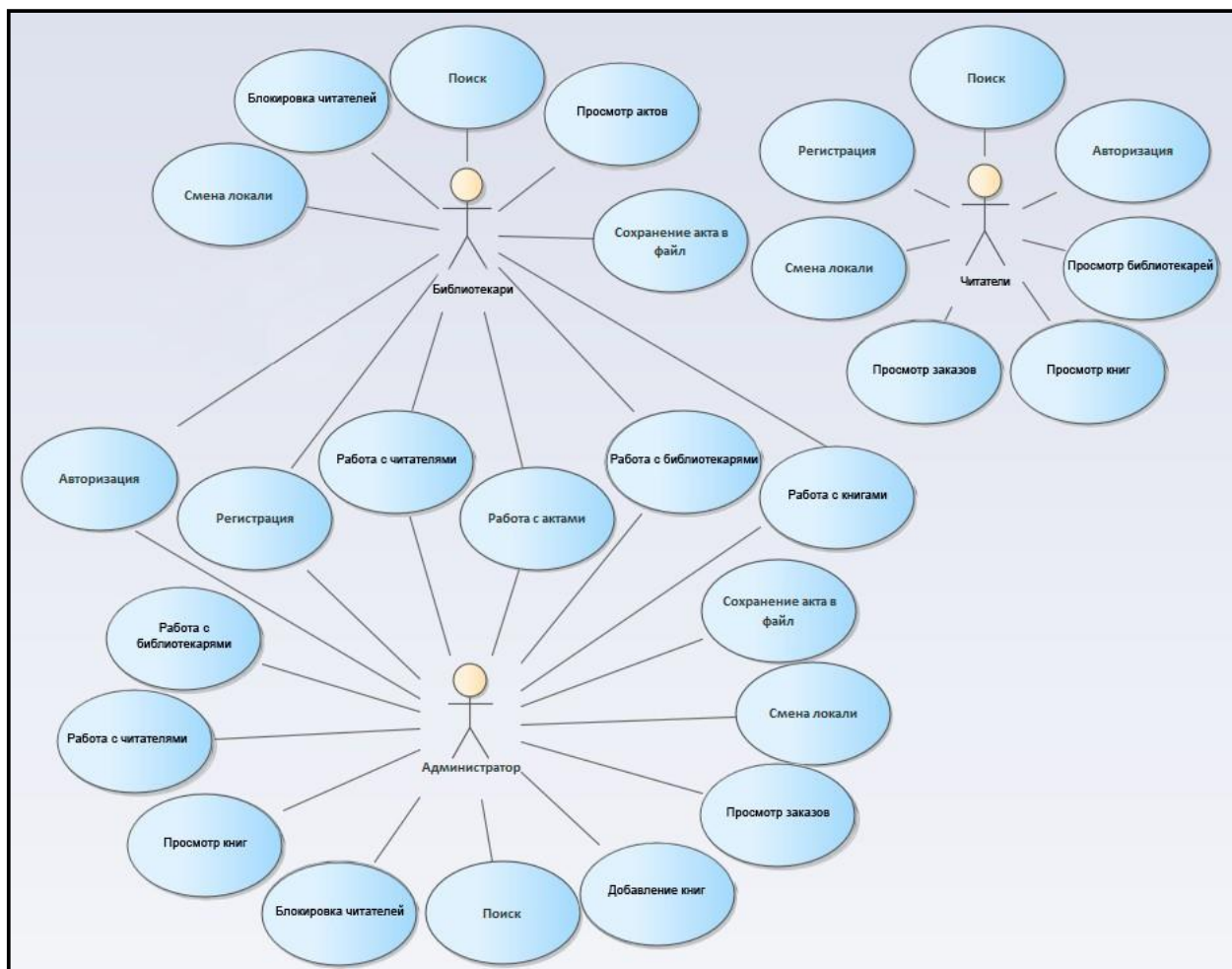


Рисунок Б.1 – Диаграмма вариантов использования (Use case diagram)

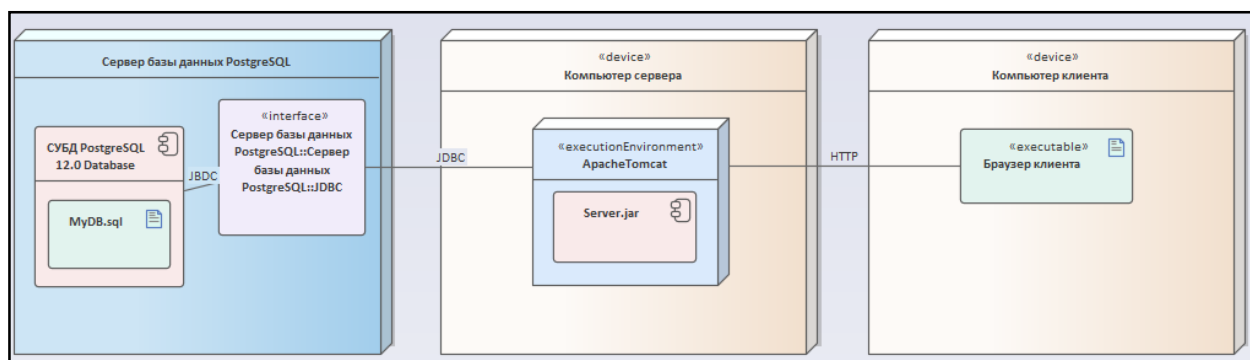


Рисунок Б.2 – Диаграмма развёртывания (Deployment diagram)

Продолжение приложения Б



Рисунок Б.3.1 – Диаграмма классов (Class diagram)

Продолжение приложения Б

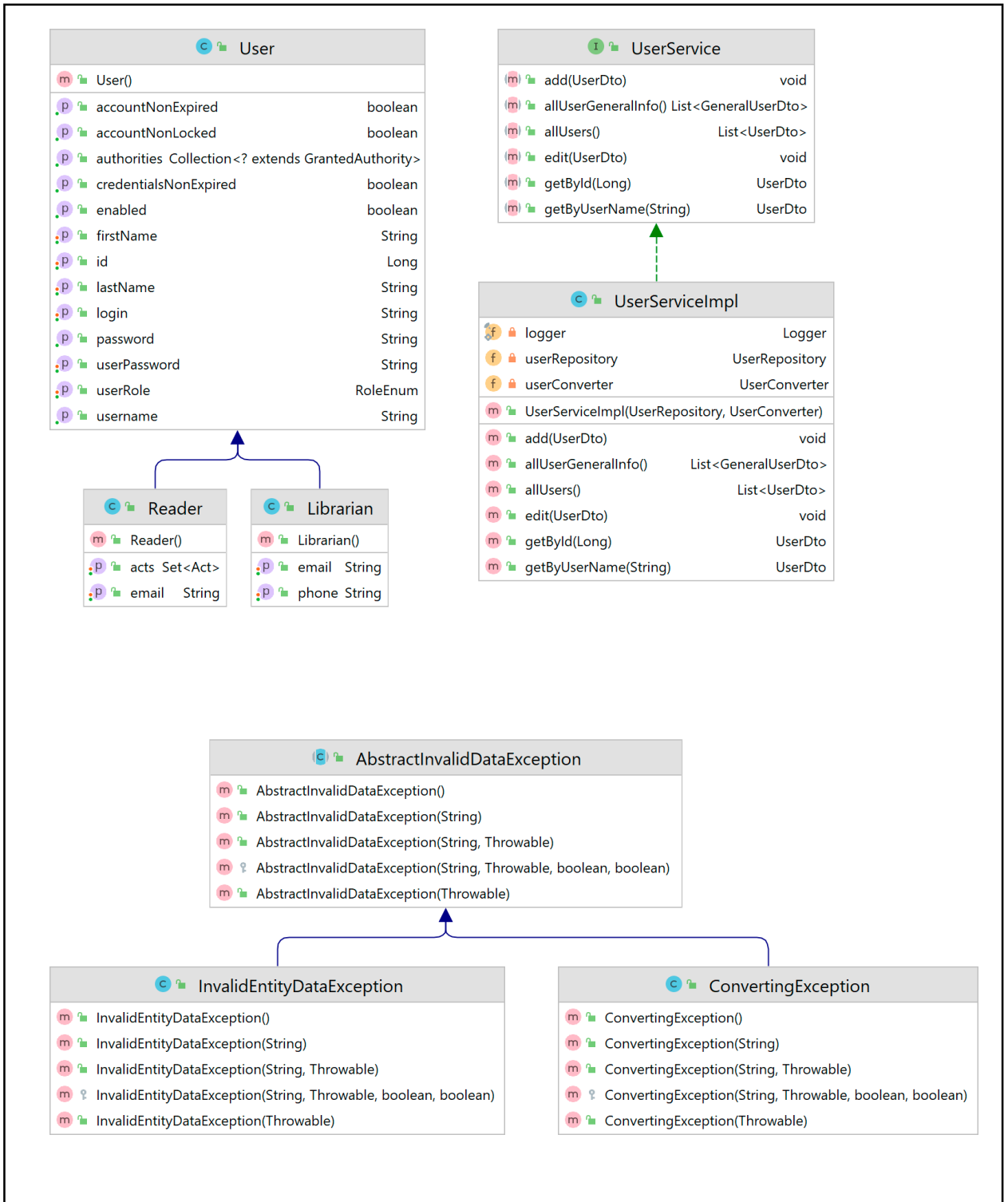


Рисунок Б.3.2 – Диаграмма классов (Class diagram)

Продолжение приложения Б

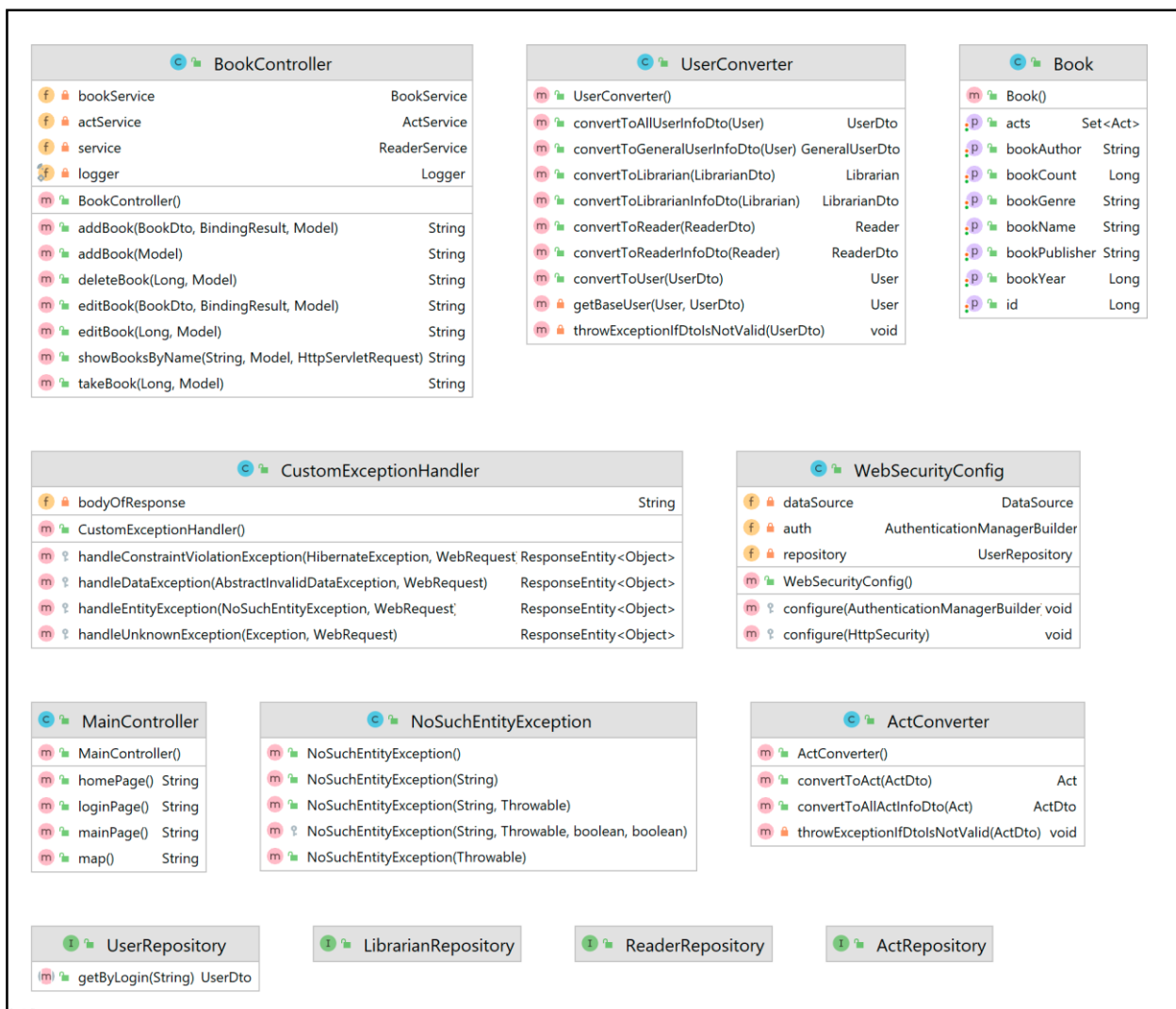


Рисунок Б.3.3 – Диаграмма классов (Class diagram)

Продолжение приложения Б

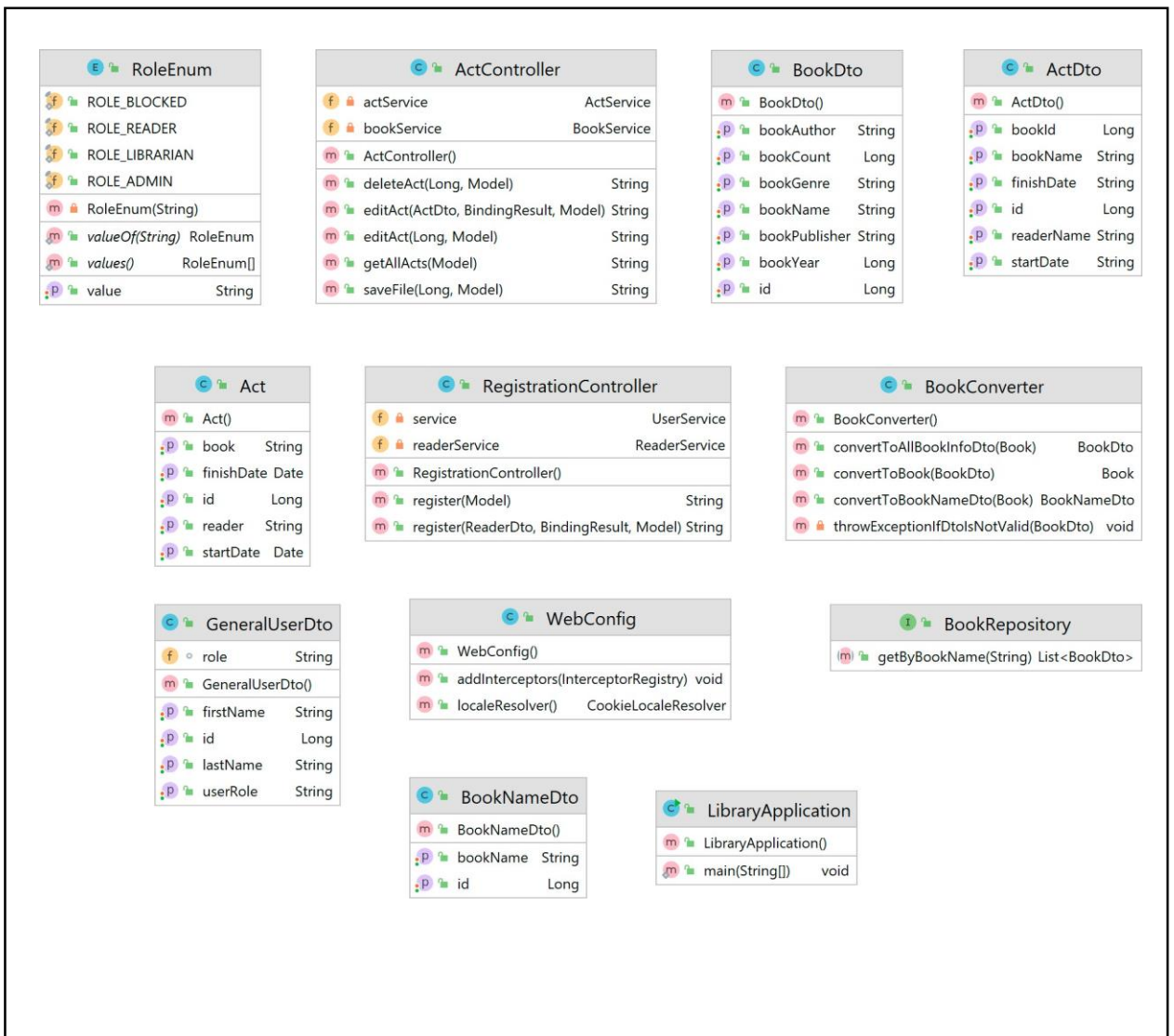


Рисунок Б.3.4 – Диаграмма классов (Class diagram)

Продолжение приложения Б

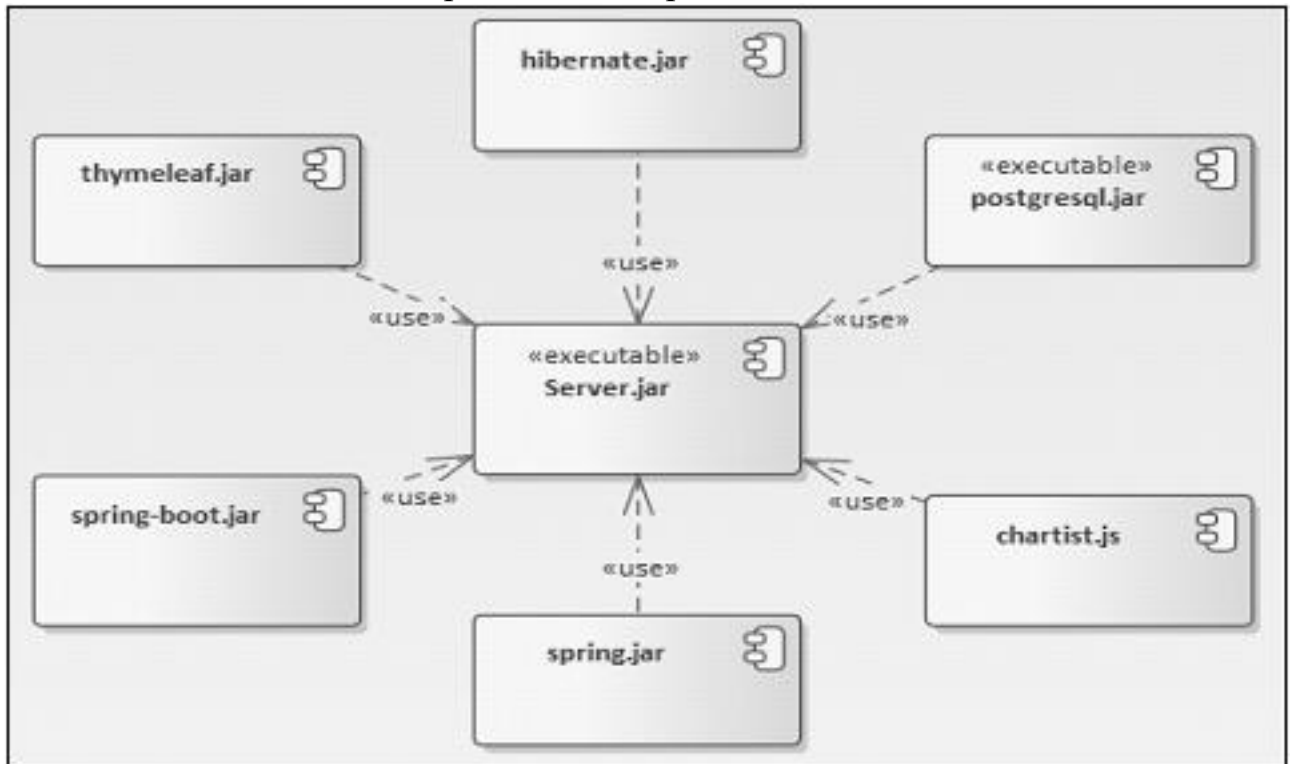


Рисунок Б.4 – Диаграмма компонентов (Component diagram)

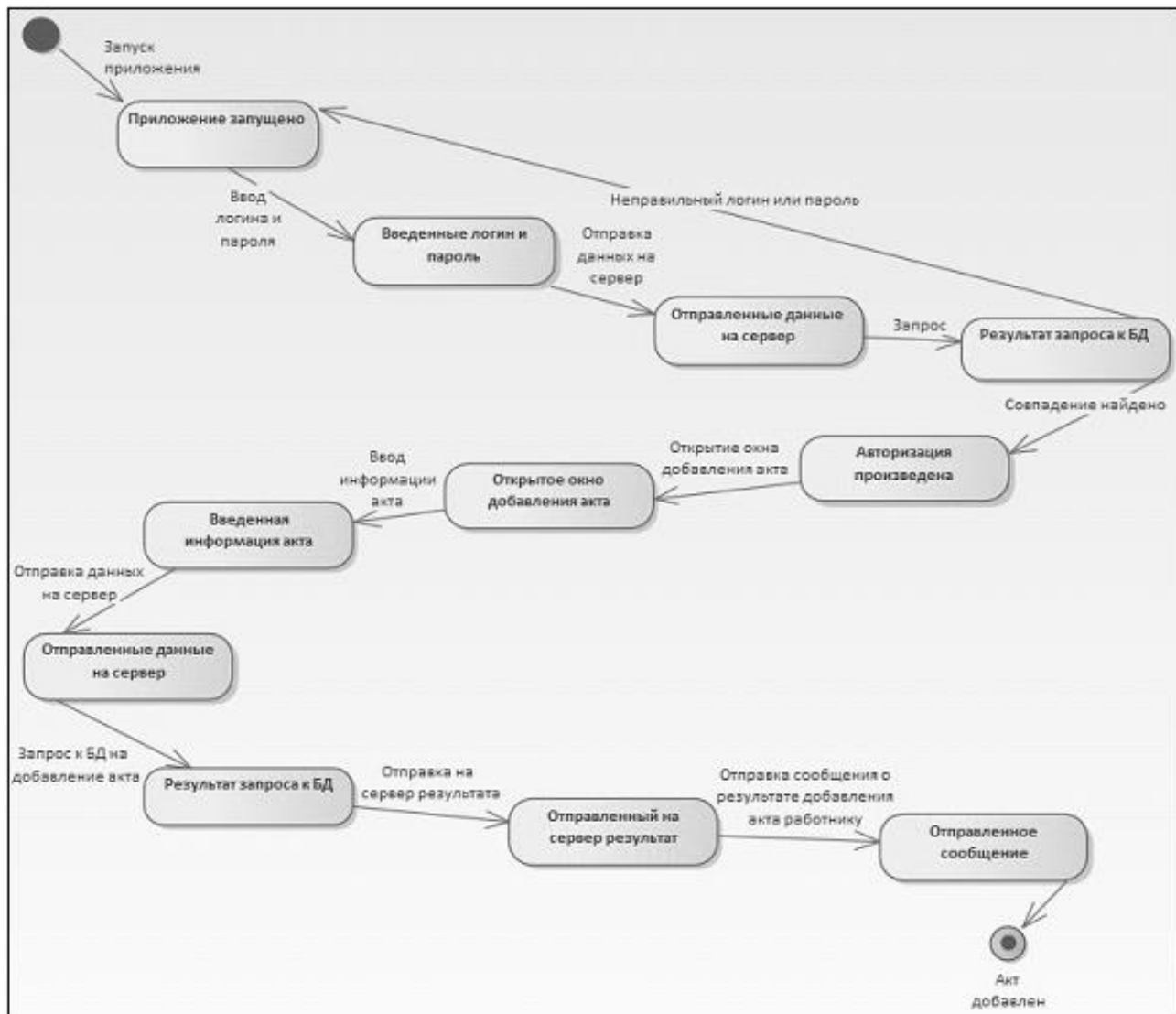


Рисунок Б.5 – Диаграмма состояний процесса добавления заказа (Statement diagram)

Продолжение приложения Б

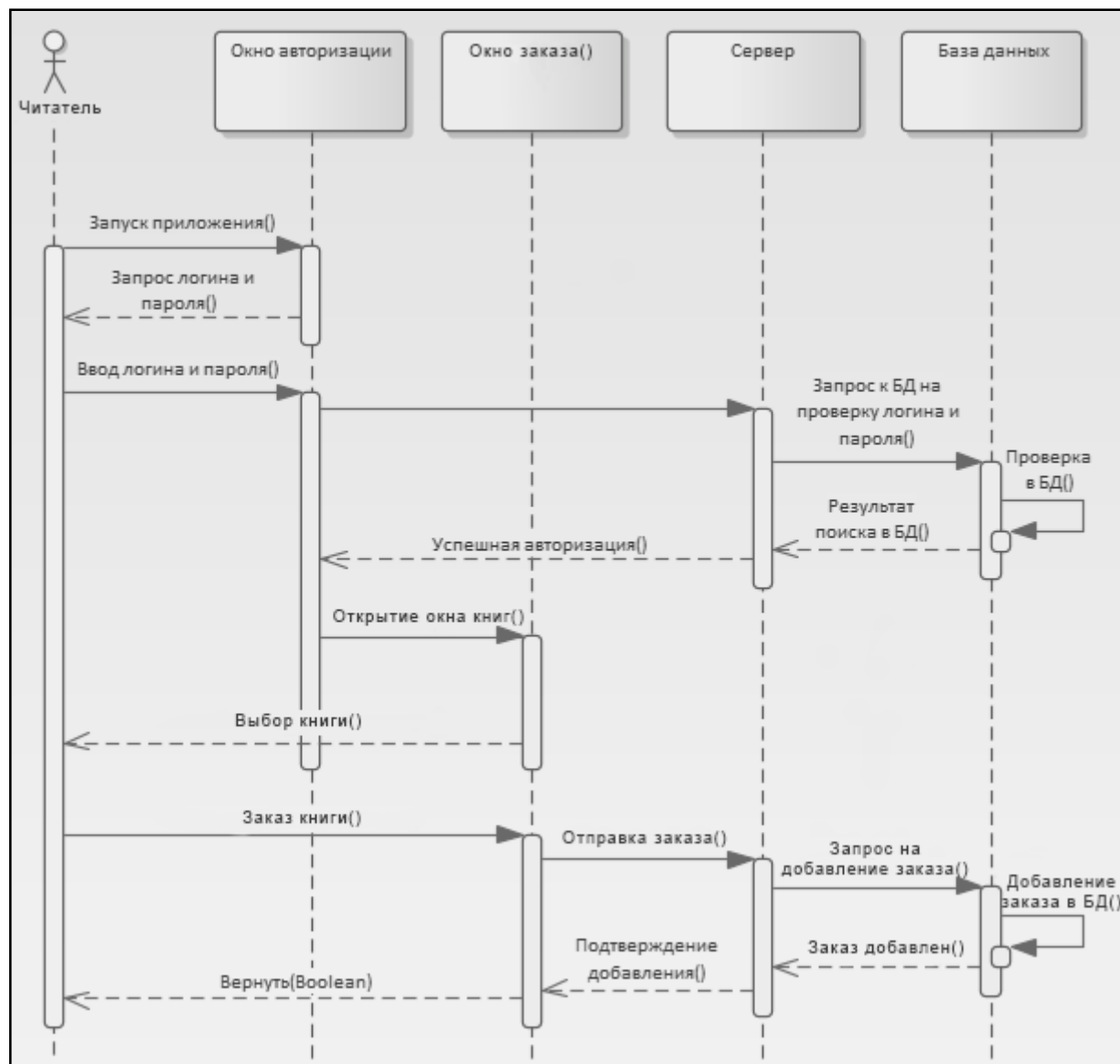


Рисунок Б.6 – Диаграмма последовательности процесса добавления заказа