Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«**Национальный исследовательский университет ИТМО**»

Факультет Программной инженерии и компьютерной техники

Курсовая работа

по дисциплине «Информационные системы»

Этап №1

**Преподаватель:**   
Коновалов Арсений Антонович

**Выполнили:**

Медведев Владислав Александрович

Михайлов Дмитрий Андреевич

**Группа:** Р3306

Санкт-Петербург, 2025

**Содержание.**

Задание…………..……………………………………………….…3

Предметная область………………………………………….…….4-5

Требования………………………………………………………….6

Функциональные требования………………………………..6-12

Нефункциональные требования……………………………13-15

Прецеденты………………………………………………………..15-24

Архитектура будущей системы…………….…………………….24-25

Вывод………………………………………………………………25

**Задание.**

1. Согласовать с преподавателем предметную область, для которой будет разрабатываться информационная система.
2. Составить подробное текстовое описание предметной области.
3. Сформулировать, зачем нужна информационная система для представленной предметной области, какие задачи она позволит решить.
4. Составить функциональные/нефункциональные требования к разрабатываемой информационной системе.
5. Построить модели основных прецедентов (прецеденты согласуются с преподавателем), составить их описание.
6. Предложить архитектуру будущей системы. При составлении архитектуры необходимо учитывать, что все этапы курсовой работы необходимо будет демонстрировать на сервере helios. Согласовать с преподавателем технологии и фреймворки, которые будут использоваться при реализации системы. Для реализации системы можно использовать:  
   a. Frontend: React, Angular, Vue, Next JS, JSF, Spring MVC (Thymeleaf или другой шаблонизатор).  
   b. Backend: основанный на Jakarta EE или Spring MVC  
   ​​​​​​​c. БД: PostgreSQL
7. Составить отчет.
8. **Предметная область.**

Библиотечная система.

* Текстовое описание.

Информационная система библиотечного сервиса предназначена для управления каталогами книг, пользователями, их авторизациями, процессами выдачи и возврата книг в любой подключённой к сервису библиотеке, а также контролем за сроками возврата и начислением штрафов.

* Основные участники.

1. Пользователь: поиск материалов, оформление выдачи/бронирования, продление сроков, оплата штрафов.
2. Библиотекарь: добавление и модификация записей материалов, управление пользователями, формирование выдачи и штрафов.
3. Системный администратор: настройка доступа, резервное копирование, подключение библиотек к сервису.

* Основные процессы.

1. Каталоги и учёт материалов: добавление новых материалов, устранение дубликатов, учет состояния.
2. Выдача и возврат материалов: проверка статуса, регистрации операции, уведомления пользователю, расчет штрафов.
3. Бронирование: формирование брони на экземпляры, уведомления о готовности.
4. Управление пользователями: создание аккаунтов, роли, штрафы, предупреждения.
5. Управление лицензиями на ресурсы: контроль доступа к электронным ресурсам, срок подписки, ограничение числа одновремённых пользователей.
6. Поиск и навигация: поддержка поисковых запросов, фильтры по автору, теме, году, формату.
7. Учёт и отчётность: отчётность по выдаче, возвратам, задолженностям, эффективности работы сотрудников.

* Примеры основных бизнес-процессов.

Изображение выглядит как диаграмма, текст, План, Технический чертеж

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рис. 1: BPMN-диаграмма для визуализации бизнес-процесса “бронирование”.

Изображение выглядит как текст, диаграмма, План, Технический чертеж

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рис. 2: BPMN-диаграмма для визуализации бизнес-процесса “возврат”.

1. **Функциональные и нефункциональные требования.**
   1. **Функциональные требования.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Требование** | **Приоритет** | **Трудоемкость** | **Стабильность** | **Дополнительная информация** |
| FR 1.1 | Возможность полнотекстового поиска | Must have | 30 часов | Высокая | Система должна предоставить возможность полнотекстового поиска по названию, автору, ISBN, теме, серии, году издания и ключевым словам. |
| FR 1.2 | Фильтрация результатов | Must have | 16 часов | Высокая | Система должна предоставить возможность фильтрации результатов по жанру, языку, типу носителя, доступности (в выдаче/на руках). |
| FR 1.3 | Поддержка сортировки результатов | Must have | 16 часов | Высокая | Система должна поддерживать сортировку результатов по релевантности, дате добавления, алфавиту, году издания. |
| FR 1.4 | Визуализация обложек материалов | Should have | 30 часов | Высокая | Система должна визуализировать обложки материалов и отображать минимальную информацию о каждом элементе. |
| FR 1.5 | Поддержка сохранённых и недавно выполненных запросов | Should have | 16 часов | Средняя | Система должна поддерживать сохранённые и недавно выполненные запросы, создание персональных подборок. |
| FR 2.1 | Регистрация нового пользователя | Must have | 30 часов | Высокая | Система должна предоставлять возможность регистрации нового пользователя с проверкой уникальности идентификатора и валидацией данных. |
| FR 2.2 | Аутентификация пользователей | Must have | 30 часов | Высокая | Система должна предоставлять возможность аутентификации пользователей с поддержкой пароля, двухфакторной идентификации и временной блокировкой после нескольких неудачных попыток. |
| FR 2.3 | Ролевое управление доступом | Must have | 20 часов | Высокая | Система должна реализовывать ролевое управление доступом: роли Пользователь, Библиотекарь, Системный администратор. |
| FR 2.4 | Управление пользовательскими профилями | Must have | 26 часов | Высокая | Система должна реализовывать управление пользовательскими профилями: редактирование данных, смена пароля, просмотр истории действий. |
| FR 2.5 | Управление установленными лимитами | Must have | 20 часов | Высокая | Система должна реализовывать управление установленными лимитами выдачи по каждому пользователю (кол-во книг, срок выдачи, штрафы). |
| FR 3.1 | Оформление выдачи материалов | Must have | 16 часов | Высокая | Система должна реализовывать оформление выдачи материалов с выбором материала и привязкой к пользователю. |
| FR 3.2 | Возможность бронирования материалов | Must have | 20 часов | Высокая | Система должна предоставлять возможность бронирования материалов, которые в данный момент недоступны, с уведомлением об освобождении. |
| FR 3.3 | Продление срока выдачи | Should have | 20 часов | Высокая | Система должна предоставлять возможность продления срока выдачи через интерфейс пользователя с учётом правил библиотеки (макс. срок, очереди). |
| FR 3.4 | Отмена выдачи или бронирования | Must have | 16 часов | Высокая | Система должна предоставлять возможность отмены выдачи или бронирования по запросу пользователя. |
| FR 3.5 | Уведомления о предстоящем сроке возврата и просрочке | Must have | 10 часов | Высокая | Система должна уведомлять о предстоящем сроке возврата и просрочке через электронную почту. |
| FR 3.6 | Поддержка штрафных санкций за просрочку | Must have | 16 часов | Высокая | Система должна реализовывать поддержку штрафных санкций за просрочку с автоматическим расчётом и списанием средств. |
| FR 4.1 | Управление библиотекарем | Must have | 20 часов | Высокая | Система должна реализовывать добавление, редактирование и удаление материалов (книги, журналы, аудиовизуальные материалы, электронные копии). |
| FR 4.2 | Управление библиотекарем | Must have | 16 часов | Высокая | Система должна реализовывать управление связями материалов с авторами, жанрами, издательствами, копиями и экземплярами. |
| FR 4.3 | Управление библиотекарем | Must have | 16 часов | Высокая | Система должна реализовывать управление экземплярами материалов: инвентаризация, статус (в доступности, выдан, зарезервирован, в ремонте), локация на полке. |
| FR 4.4 | Управление библиотекарем | Must have | 20 часов | Высокая | Система должна реализовывать формирование выдач и возвратов с учётом статусов экземпляров и ограничений пользователя. |
| FR 4.5 | Управление библиотекарем | Must have | 20 часов | Высокая | Система должна реализовывать управление штрафами для пользователей: начисление, списание, корректировки. |
| FR 5.1 | Управление материалами и каталогом | Should have | 10 часов | Высокая | Система должна реализовывать поддержку версионирования записей материалов (история изменений). |
| FR 5.2 | Массовое импортирование и экспортирование | Must have | 16 часов | Высокая | Система должна реализовывать массовое импортирование и экспортирование данных материалов (CSV/XML/JSON). |
| FR 5.3 | Автоматическое обновление метаданных | Must have | 16 часов | Высокая | Система должна реализовывать автоматическое обновление метаданных по внешним источникам (ISBNdb, OpenLibrary и т.д.) с подтверждением изменений. |
| FR 5.4 | Учет копий материалов | Must have | 20 часов | Высокая | Система должна реализовывать учет копий материалов с уникальными идентификаторами и привязкой к экземплярам. |
| FR 5.5 | Управление связями материалов | Should have | 10 часов | Средняя | Система должна реализовывать управление связями материалов: серии, стихи, переводчики (для мультимедийного контента), резервы на редкость. |
| FR 6.1 | Контроль выдач и сроков возврата | Must have | 20 часов | Высокая | Система должна содержать реализацию системы уведомления о просрочках (автоматические напоминания, уведомления о штрафах). |
| FR 6.2 | Контроль выдач и сроков возврата | Must have | 10 часов | Высокая | Система должна реализовывать расчёт штрафов по заданной политике (интервал, ставка, льготы) и автоматическое списание или отображение для оплаты. |
| FR 6.3 | Контроль выдач и сроков возврата | Should have | 16 часов | Высокая | Система должна осуществлять поддержку нескольких профилей штрафов для разных категорий пользователей (студенты, преподаватели, граждане и т.д.). |
| FR 6.4 | Контроль выдач и сроков возврата | Must have | 16 часов | Высокая | Система должна реализовывать отражение истории выдач с привязкой к штрафам и возвратам. |
| FR 7.1 | Оплата штрафов | Must have | 20 часов | Высокая | Система должна осуществлять поддержку нескольких способов оплаты (онлайн-платежи, банковские переводы, офлайн-оплата через кассу). |
| FR 7.2 | Оплата штрафов | Must have | 20 часов | Высокая | Система должна реализовывать интеграцию с платежными шлюзами и учёт статуса оплаты в системе. |
| FR 7.3 | Оплата штрафов | Should have | 10 часов | Высокая | Система должна осуществлять генерацию квитанций и проверка статуса оплаты для выдачи материалов после оплаты. |
| FR 8.1 | Статистика и отчёты | Must have | 16 часов | Высокая | Система должна реализовывать формирование отчетов по выдачам и возвратам за период (кол-во выданных материалов, среднее время выдачи, просрочки). |
| FR 8.2 | Статистика и отчёты | Should have | 20 часов | Высокая | Система должна реализовывать формирование отчётов по активности пользователей (частота поисков, бронирований, штрафов). |
| FR 8.3 | Статистика и отчёты | Must have | 10 часов | Высокая | Система должна формировать отчёты по состоянию коллекций (количество доступных копий, количество на ремонте, устаревшие издания). |
| FR 8.4 | Статистика и отчёты | Must have | 16 часов | Высокая | Система должна экспортировать аналитические данные в форматы CSV/Excel/PDF. |
| FR 9.1 | Безопасность и аудит | Must have | 16 часов | Высокая | Система должна реализовывать ведение журнала аудита действий пользователей и библиотекарей. |
| FR 9.2 | Безопасность и аудит | Must have | 20 часов | Высокая | Система должна реализовывать резервирование и шифрование конфиденциальных данных и безопасное хранение паролей. |
| FR 9.3 | Безопасность и аудит | Must have | 20 часов | Высокая | Система должна реализовывать управление ролями и доступом к функциям по принципу минимальных прав. |
| FR 9.4 | Безопасность и аудит | Must have | 10 часов | Высокая | Система должна реализовывать регулярное резервное копирование данных с планами восстановления. |
| FR 10.1 | Управление лицензиями и электронными копиями | Should have | 16 часов | Высокая | Система должна реализовывать учёт лицензий на электронные издания и сроков действия. |
| FR 10.2 | Управление лицензиями и электронными копиями | Should have | 16 часов | Высокая | Система должна реализовывать контроль доступа к электронным ресурсам по учетным записям пользователей. |
| FR 10.3 | Управление лицензиями и электронными копиями | Should have | 20 часов | Высокая | Система должна реализовывать мониторинг использования электронных материалов и ограничений. |

* 1. **Нефункциональные требования.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Требование** | **Приоритет** | **Трудоемкость** | **Стабильность** | **Дополнительная информация** |
| U 1 | Разработка и анализ элементов интерфейса | Must have | 30 часов | Высокая | Система должна иметь элементы интерфейса, соответствующие стандартам ISO. Элементы интерфейса обязаны быть не только понятными, но и максимально информативными для пользователя. Их дизайн и расположение должны обеспечивать лёгкость восприятия и понимание функционала. |
| U 2 | Поддержка всех браузеров | Must have | 20 часов | Высокая | Система должна иметь внешнее оформление, отображающееся корректно во всех популярных браузерах. (Google, FireFox, Yandex, Opera). |
| U 3 | Поддержка мобильных устройств | Must have | 20 часов | Высокая | Система должна иметь внешнее оформление, отображающееся корректно на мобильном устройстве. |
| U 4 | Валидация | Must have | 20 часов | Высокая | Система должна иметь формы, поддерживающие валидацию на стороне клиента и сервера, показывая понятные сообщения об ошибках в реальном времени. |
| U 5 | Интуитивно понятный интерфейс | Must have | 30 часов | Высокая | Система должна иметь интуитивно понятный интерфейс с минимальным числом кликов до выполнения основных действий: не более 3 кликов для большинства операций. |
| R 1 | Надежность | Must have | 30 часов | Высокая | Система должна работать в 99.9% времени в месячном окне (SLA по доступности). |
| R 2 | Надежность | Must have | 26 часов | Высокая | Система должна реализовывать регистрацию и мониторинг событий с алертингом: критические инциденты должны приходить в ответственные лица в течение 5 минут. |
| R 3 | Резервное копирование | Must have | 28 часов | Высокая | Система должна реализовывать резервное копирование данных должно выполняться не менее раз в 24 часа, тестированное восстановление не чаще чем раз в квартал. |
| P 1 | Время отклика | Should have | 30 часов | Высокая | Система должна реализовывать время отклика API под нормальной нагрузкой: среднее не более 200 мс, p95 не более 500 мс. |
| P 2 | Пропускная способность | Must have | 40 часов | Высокая | Система должна поддерживать до N транзакций в секунду без деградации. |
| P 3 | Кэширование запросов | Should have | 30 часов | Высокая | Система должна реализовывать следующее: часто запрашиваемые данные должны кэшироваться на уровне слоя приложения или кэша уровня сервиса приблизительно на 60–120 секунд. |
| S 1 | Совместимость | Must have | 30 часов | Высокая | Система должна быть совместима с UNIX-серверной вычислительной мощностью |

1. **Прецеденты.**

|  |  |
| --- | --- |
| **Имя** | Регистрация пользователя |
| **ID** | 1 |
| **Описание** | Система должна предоставлять возможность регистрации нового пользователя с проверкой уникальности идентификатора и валидацией данных. |
| **Актёры** | Неавторизованный пользователь |
| **Предусловие** | Пользователь находится на главной странице сайта, будучи неавторизованным. |
| **Основной поток** | Система предлагает зарегистрироваться. |
| **Постусловие** | Пользователь регистрируется и получает свой читательский билет с уникальным идентификатором. |
| **Альтернативный поток** | Пользователь вводит неверные данные, после чего система уведомляет его об этом. |

|  |  |
| --- | --- |
| **Имя** | Авторизация пользователя |
| **ID** | 2 |
| **Описание** | Система должна предоставлять возможность входа зарегистрированного пользователя с проверкой уникальности идентификатора и валидацией данных. |
| **Актёры** | Неавторизованный пользователь |
| **Предусловие** | Пользователь находится на главной странице сайта, будучи неавторизованным. |
| **Основной поток** | Система предлагает войти в систему. |
| **Постусловие** | Пользователь входит в свой аккаунт. |
| **Альтернативный поток** | Пользователь вводит неверные данные, после чего система уведомляет его об этом. |

|  |  |
| --- | --- |
| **Имя** | Аутентификации пользователя |
| **ID** | 3 |
| **Описание** | Система должна предоставлять возможность аутентификации пользователей с поддержкой пароля, двухфакторной идентификации и временной блокировкой после нескольких неудачных попыток. |
| **Актёры** | Неавторизованный пользователь |
| **Предусловие** | Пользователь находится на главной странице сайта, будучи неавторизованным. |
| **Основной поток** | Система предлагает пройти процесс аутентификации. |
| **Постусловие** | Пользователь проходит аутентификацию и входит в аккаунт. |
| **Альтернативный поток** | После нескольких неудачных попыток система выдает временную блокировку. |

|  |  |
| --- | --- |
| **Имя** | Поиск материалов |
| **ID** | 4 |
| **Описание** | Система должна предоставить возможность полнотекстового поиска по названию, автору, ISBN, теме, серии, году издания и ключевым словам. |
| **Актёры** | Авторизованный пользователь |
| **Предусловие** | Пользователь находится на главной странице сайта, будучи авторизованным. |
| **Основной поток** | 1. Пользователь вводит критерии поиска. 2. Система возвращает список соответствующих материалов. 3. Пользователь выбирает материал для просмотра подробной информации. |
| **Постусловие** | Отображён список результатов и выбранный материал открыт для просмотра. |
| **Альтернативный поток** | Нет результатов — отображено сообщение об отсутствии совпадений. |

|  |  |
| --- | --- |
| **Имя** | Фильтрация результатов |
| **ID** | 5 |
| **Описание** | Система должна предоставить возможность фильтрации результатов по жанру, языку, типу носителя, доступности (в выдаче/на руках). |
| **Актёры** | Авторизованный пользователь |
| **Предусловие** | Пользователь находится на главной странице сайта, будучи авторизованным. |
| **Основной поток** | 1. Пользователь вводит критерии фильтрации. 2. Система возвращает список соответствующих материалов. 3. Пользователь выбирает материал для просмотра подробной информации. |
| **Постусловие** | Отображён список отфильтрованных результатов. |
| **Альтернативный поток** | Нет результатов — отображен пустой список. |

|  |  |
| --- | --- |
| **Имя** | Сортировка результатов |
| **ID** | 6 |
| **Описание** | Система должна поддерживать сортировку результатов по релевантности, дате добавления, алфавиту, году издания. |
| **Актёры** | Авторизованный пользователь |
| **Предусловие** | Пользователь находится на главной странице сайта, будучи авторизованным. |
| **Основной поток** | 1. Пользователь вводит критерии сортировки. 2. Система возвращает список соответствующих материалов. 3. Пользователь выбирает материал для просмотра подробной информации. |
| **Постусловие** | Отображён список отсортированных результатов. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Визуализация обложек материалов |
| **ID** | 7 |
| **Описание** | Система должна визуализировать обложки материалов и отображать минимальную информацию о каждом элементе. |
| **Актёры** | Авторизованный пользователь |
| **Предусловие** | Пользователь находится на главной странице сайта, будучи авторизованным. |
| **Основной поток** | 1. Пользователь просматривает каталог. 2. Система возвращает список материалов с визуализированными обложками и краткой информацию сбоку. 3. Пользователь выбирает материал для просмотра подробной информации. |
| **Постусловие** | Отображён список, где у каждого элемента есть обложка и краткое описание. |
| **Альтернативный поток** | - |
| **Имя** | Поддержка сохранённых и недавно выполненных запросов |
| **ID** | 8 |
| **Описание** | Система должна поддерживать сохранённые и недавно выполненные запросы, создание персональных подборок. |
| **Актёры** | Авторизованный пользователь |
| **Предусловие** | Пользователь находится на главной странице сайта, будучи авторизованным. |
| **Основной поток** | 1. Пользователь нажимает на поисковую строку. 2. Система возвращает список недавних запросов. |
| **Постусловие** | Пользователь получает список недавних запросов. |
| **Альтернативный поток** | Нет результатов – не было запросов. |

|  |  |
| --- | --- |
| **Имя** | Оформление выдачи |
| **ID** | 9 |
| **Описание** | Система должна реализовывать оформление выдачи материалов с выбором материала и привязкой к пользователю. |
| **Актёры** | Авторизованный пользователь |
| **Предусловие** | Материал доступен к выдаче; пользователь авторизован |
| **Основной поток** | 1. Пользователь выбирает материал и инициирует выдачу. 2. Система проверяет статус материала и доступность. 3. Система создаёт запись выдачи, связывает с пользователем и устанавливает срок. |
| **Постусловие** | Выдача зарегистрирована; срок выдачи установлен. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Продление срока выдачи |
| **ID** | 10 |
| **Описание** | Система должна предоставлять возможность продления срока выдачи через интерфейс пользователя с учётом правил библиотеки (макс. срок, очереди). |
| **Актёры** | Авторизованный пользователь |
| **Предусловие** | Период продления допустим; материал выдан пользователю. |
| **Основной поток** | 1. Пользователь инициирует продление. 2. Система проверяет наличие ограничений. 3. При возможности продлевает срок и фиксирует новое окончание. |
| **Постусловие** | Срок выдачи обновлён. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Погашение штрафов |
| **ID** | 11 |
| **Описание** | Система должна осуществлять поддержку нескольких способов оплаты (онлайн-платежи, банковские переводы, офлайн-оплата через кассу). |
| **Актёры** | Авторизованный пользователь |
| **Предусловие** | Есть начисленный штраф. |
| **Основной поток** | 1. Пользователь выбирает способ оплаты. 2. Система регистрирует оплату и снимает штраф. |
| **Постусловие** | Штраф погашен; баланс обновлён. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Оформление брони |
| **ID** | 12 |
| **Описание** | Система должна предоставлять возможность бронирования материалов, которые в данный момент недоступны, с уведомлением об освобождении. |
| **Актёры** | Авторизованный пользователь |
| **Предусловие** | Материал доступен для бронирования. |
| **Основной поток** | 1. Пользователь выбирает бронирование. 2. Система создаёт бронь и уведомляет пользователя. |
| **Постусловие** | Бронь создана; пользователь уведомлён. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Отмена брони |
| **ID** | 13 |
| **Описание** | Система должна предоставлять возможность отмены выдачи или бронирования по запросу пользователя. |
| **Актёры** | Авторизованный пользователь |
| **Предусловие** | Есть активная бронь. |
| **Основной поток** | 1. Пользователь инициирует отмену. 2. Система снимает бронь и освобождает материал. |
| **Постусловие** | Бронь удалена. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Управление записями материалов |
| **ID** | 14 |
| **Описание** | Система должна реализовывать добавление, редактирование и удаление материалов (книги, журналы, аудиовизуальные материалы, электронные копии). |
| **Актёры** | Библиотекарь |
| **Предусловие** | Библиотекарь имеет соответствующие полномочия. |
| **Основной поток** | 1. Библиотекарь выбирает действие (добавить/редактировать/удалить). 2. Вводятся данные материала. 3. Система сохраняет изменения. |
| **Постусловие** | Запись материала обновлена. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Управление копиями материалов |
| **ID** | 15 |
| **Описание** | Система должна реализовывать учет копий материалов с уникальными идентификаторами и привязкой к экземплярам. |
| **Актёры** | Библиотекарь |
| **Предусловие** | Материал существует. |
| **Основной поток** | 1. Библиотекарь изменяет число копий. 2. Система пересчитывает доступные копии. |
| **Постусловие** | Доступные копии обновлены. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Формирование выдачи |
| **ID** | 16 |
| **Описание** | Система должна реализовывать формирование выдач и возвратов с учётом статусов экземпляров и ограничений пользователя. |
| **Актёры** | Библиотекарь |
| **Предусловие** | Есть несколько материалов на выдачу для одного пользователя. |
| **Основной поток** | 1. Библиотекарь выбирает материалы и пользователя. 2. Система создаёт записи выдачи. |
| **Постусловие** | Выдачи сформированы. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Формирование штрафов |
| **ID** | 17 |
| **Описание** | Система должна реализовывать управление штрафами для пользователей: начисление, списание, корректировки. |
| **Актёры** | Библиотекарь |
| **Предусловие** | Есть просроченные выдачи. |
| **Основной поток** | 1. Система вычисляет сроки и штрафы по заданным правилам. 2. Библиотекарь может подтвердить и подписать штрафы. |
| **Постусловие** | Штрафы начислены. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Назначение ролей и прав доступа |
| **ID** | 19 |
| **Описание** | Система должна реализовывать управление ролями и доступом к функциям по принципу минимальных прав. |
| **Актёры** | Системный администратор |
| **Предусловие** | Администратор имеет необходимые полномочия. |
| **Основной поток** | 1. Администратор добавляет/изменяет роли. 2. Система применяет настройки. |
| **Постусловие** | Роли и права обновлены. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Резервное копирование данных |
| **ID** | 20 |
| **Описание** | Система должна реализовывать регулярное резервное копирование данных с планами восстановления. |
| **Актёры** | Системный администратор |
| **Предусловие** | Установлены политики резервного копирования. |
| **Основной поток** | 1. Администратор инициирует копирование. 2. Система сохраняет данные в указанный носитель/облако. |
| **Постусловие** | Резервная копия создана. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Восстановление из резервной копии |
| **ID** | 21 |
| **Описание** | Система должна реализовывать регулярное резервное копирование данных с планами восстановления. |
| **Актёры** | Системный администратор |
| **Предусловие** | Наличие резервной копии. |
| **Основной поток** | 1. Администратор выбирает точку восстановления. 2. Система восстанавливает данные. |
| **Постусловие** | Сервис возвращён к рабочему состоянию. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Импорт материалов из внешних источников |
| **ID** | 22 |
| **Описание** | Система должна реализовывать массовое импортирование и экспортирование данных материалов (CSV/XML/JSON). |
| **Актёры** | Библиотекарь |
| **Предусловие** | Формат файла поддерживается. |
| **Основной поток** | 1. Библиотекарь загружает файл. 2. Система валидирует данные и добавляет записи. |
| **Постусловие** | Новые материалы добавлены. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Экспорт каталога |
| **ID** | 23 |
| **Описание** | Система должна реализовывать массовое импортирование и экспортирование данных материалов (CSV/XML/JSON). |
| **Актёры** | Библиотекарь |
| **Предусловие** | Предоставлен фильтр экспорта. |
| **Основной поток** | 1. Библиотекарь выбирает формат и параметры экспорта. 2. Система формирует файл. |
| **Постусловие** | Файл экспортирован. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Поиск и фильтрация по должникам |
| **ID** | 24 |
| **Описание** | Система должна реализовывать управление штрафами для пользователей: начисление, списание, корректировки. |
| **Актёры** | Библиотекарь |
| **Предусловие** | Система содержит данные по выдачам и штрафам. |
| **Основной поток** | 1. Библиотекарь применяет фильтры по статусу просрочки, штрафам. 2. Система отображает список должников. |
| **Постусловие** | Текущий список должников сформирован. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Мониторинг статуса материалов |
| **ID** | 25 |
| **Описание** | Система должна реализовывать управление экземплярами материалов: инвентаризация, статус (в доступности, выдан, зарезервирован, в ремонте), локация на полке. |
| **Актёры** | Системный администратор, Библиотекарь |
| **Предусловие** | База данных содержит статусы. |
| **Основной поток** | 1. Система регулярно обновляет статусы. |
| **Постусловие** | Статусы материалов актуальны. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Уведомления о просрочке |
| **ID** | 26 |
| **Описание** | Система должна содержать реализацию системы уведомления о просрочках (автоматические напоминания, уведомления о штрафах). |
| **Актёры** | Авторизованный пользователь |
| **Предусловие** | Есть просроченные или приближающиеся к сроку выдачи. |
| **Основной поток** | 1. Система отправляет уведомление по выбранному каналу. |
| **Постусловие** | Пользователь информирован. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Статистика использования |
| **ID** | 27 |
| **Описание** | Система должна содержать реализацию системы уведомления о просрочках (автоматические напоминания, уведомления о штрафах). |
| **Актёры** | Библиотекарь, Системный администратор |
| **Предусловие** | Доступны данные за период. |
| **Основной поток** | 1. Запрос статистики по параметрам. 2. Система формирует отчёт. |
| **Постусловие** | Статистический отчёт сформирован. |
| **Альтернативный поток** | - |

|  |  |
| --- | --- |
| **Имя** | Интеграция с внешними платежными системами |
| **ID** | 28 |
| **Описание** | Система должна реализовывать интеграцию с платежными шлюзами и учёт статуса оплаты в системе. |
| **Актёры** | Авторизованный пользователь, Библиотекарь |
| **Предусловие** | Подключены платежные провайдеры |
| **Основной поток** | 1. Пользователь выбирает платежного провайдера. 2. Система перенаправляет и получает уведомление об успешной оплате. |
| **Постусловие** | Оплата зарегистрирована. |
| **Альтернативный поток** | - |

1. **Архитектура будущей системы.**

**Архитектура:**

Для реализации информационной системы библиотеки целесообразно использовать многоуровневую (многослойную) архитектуру, включающую несколько взаимосвязанных компонентов. Такое разделение позволит каждому слою системы выполнять строго определённые задачи, обеспечит удобство сопровождения и развитие проекта, а также создаст возможность масштабирования и интеграции новых модулей при необходимости.

**Клиент-серверная архитектура:**

Приложение будет основано на клиент-серверной архитектуре, где frontend и backend отделены друг от друга и взаимодействуют через API. Это позволит легко масштабировать как клиентскую, так и серверную части.

**Фронтенд (Client Side):** Фронтенд будет представлять собой SSR (Server-Side Rendering). Фронтенд взаимодействует с бэкендом через RESTful API.

**Бэкенд (Server Side):** Реализован как монолит c использованием MVC.

**Основные модули бэкенда:**

* **Модуль авторизации и безопасности**: реализация аутентификации и авторизации с помощью JWT-токенов, управление ролями пользователей (читатель, библиотекарь, администратор).
* **Модуль управления книгами**: добавление, редактирование и удаление записей о книгах, хранение информации о доступных экземплярах, предоставление API для поиска и фильтрации книг по названию, автору, жанру или ISBN.
* **Модуль бронирования и выдачи**: обработка запросов на бронирование книг, фиксация выдачи и возврата экземпляров, автоматическая проверка сроков пользования.
* **Модуль уведомлений**: система напоминаний о скором окончании срока бронирования и уведомлений о просрочке через электронную почту. Поддержка асинхронной обработки событий с использованием Kafka.
* **Модуль штрафов**: расчёт штрафов за просрочку возврата, учёт и хранение информации о задолженностях пользователей, отметка об оплате.
* **Модуль управления пользователями**: регистрация и редактирование профилей, хранение истории бронирований и операций, доступ к статистике для библиотекарей и администраторов.

### **База данных:**

Система будет использовать реляционную базу данных PostgreSQL для хранения информации о книгах, экземплярах, пользователях, бронированиях, возвратах и штрафах.

### **Файловое хранилище:**

Для хранения сопутствующих файлов (например, обложек книг, сканов билетов читателей или аватаров пользователей) будет использоваться файловое хранилище, что позволит оптимизировать скорость загрузки и повысить производительность приложения.

**Технологии и фреймворки:**

### **Frontend:** React

**Backend:** Java, Spring Boot, Hibernate, PostgreSQL, MinIO, Kafka

1. **Вывод.**

В результате выполнения первого этапа курсовой работы была выбрана и подробно изучена предметная область библиотечного сервиса. Сформировано детальное текстовое описание предметной области с выделением ключевых сущностей и процессов, определены функциональные и нефункциональные требования к будущей системе. Также была предложена архитектура приложения и согласован набор технологий для реализации. Таким образом, на данном этапе выполнено базовое планирование системы, что создаёт основу для дальнейшего детального анализа и проектирования.