Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерных технологий

Кафедра инженерной психологии и эргономики

Дисциплина: управление информационными проектами

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

на тему:

Разработка информационной системы «Автоматизация учета лекарств для сети аптек»

Студент: гр. 910903 Логвин В. В.

Руководитель: Борисик М. М.

Минск 2022

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерного проектирования

УТВЕРЖДАЮ

Заведующий кафедрой ИПиЭ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ T.В. Казак

«\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2021 года

ЗАДАНИЕ

по курсовому проектированию

Студенту(–ке) Логвину Владислав Валерьевичу*–––––––––––*

1. Тема проекта Разработка информационной системы «Автоматизация учета лекарств для сети аптек»

2. Срок сдачи студентом законченного проекта––––1 *03.05.2022 –*

3. Исходные данные к проекту *Основные характеристики информационной системы и базы данных (БД): 1) не менее 10 сущностей; 2) 3–я нормальная форма; 3) назначение – в соответствии темой; 4) БД разрабатывается и реализуется с помощью СУБД PostgreSQL; 5) клиентская часть информационной системы разрабатывается на объектно–ориентированном языке программирования Java и реализуется с использованием средств доступа к данным Spring Data и Hibernate; 6) клиентская часть информационной системы должна обеспечивать возможность добавления выборки, поиска и модификации данных в БД (добавление, удаление, обновление); 7) должно быть реализовано два режима работы с БД: режим администратора и пользователя (в режиме администратора доступны все функции манипулирования данными, в режиме пользователя функции манипулирования данными ограничены в соответствии с предметной областью БД); 8) предусмотреть возможность сохранения резервной копии БД по требованию пользователя и автоматически.*

4. Содержание расчетно–пояснительной записки (перечень вопросов, которые подлежат разработке)

*Введение. 1. Постановка задачи. 1.1 Описание предметной области. 1.2* *Цели и задачи на проектирование 2. Проектирование.2.1 Проектирование схемы данных. 2.2* *Обоснование выбора технологии. 2.3 Описание среды реализации. 3. Программная реализация. 3.1 Физическая структура базы данных. 3.2 Структура информационной системы. 3.3 Тестирование системы. 3.4 Описание применения. Заключение. Список использованных источников. Приложение (листинг программы)*

5. Перечень графического материала (с точным обозначением обязательных чертежей и графиков)

*1. Структура информационной системы*

*2. Схема базы данных*

*3. Схема алгоритма работы пользователя/администратора*

*4. Диаграмма вариантов использования*

*5. Диаграмма взаимодействия*

6. Консультант по проекту *M.М. Борисик –––*

7. Дата выдачи задания  *–*

8. Календарный график работы над проектом на весь период проектирования (с обозначением сроков выполнения и трудоемкости отдельных этапов):

*раздел 1: 01.03.2022 (20 %); \_\_\_\_\_––––––––––––––––––––––––––––––––*

*раздел 2: 29.03.2022 (40 %); ––––––\_\_\_\_\_\_\_\_\_––––––––––––––––––––––*

*раздел 3: 26.04.2022 (30 %); \_\_\_–––––––––––––––––––––––––––––––––*

*оформление пояснительной записки и графического материала к 02.05 – 10 %*

РУКОВОДИТЕЛЬ*– М.М. Борисик*

(подпись)

Задание принял к исполнению *–––––––\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_––*

(дата и подпись студента)

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 5](#_Toc100025778)

[1 ПОСТАНОВКА ЗАДАЧИ 7](#_Toc100025779)

[2.1 Описание предметной области 7](#_Toc100025780)

[2.2 Цели и задачи на проектирование 8](#_Toc100025781)

[2 ПРОЕКТИРОВАНИЕ 9](#_Toc100025782)

[2.1 Проектирование схемы данных 9](#_Toc100025783)

[2.2 Обоснование выбора технологий 10](#_Toc100025784)

[2.3 Описание среды реализации 11](#_Toc100025785)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 12](#_Toc100025786)

[ПРИЛОЖЕНИЕ А 13](#_Toc100025787)

[ПРИЛОЖЕНИЕ Б 14](#_Toc100025788)

[ПРИЛОЖЕНИЕ В 15](#_Toc100025789)

[ПРИЛОЖЕНИЕ Г 16](#_Toc100025790)

[ПРИЛОЖЕНИЕ Д 17](#_Toc100025791)

[ПРИЛОЖЕНИЕ Е 18](#_Toc100025792)

# **ВВЕДЕНИЕ**

Скорость развития технологий в наш век рекордно большая для всей истории человечества. Мир движется к удобствам, экономии ресурсов, автоматизации и оптимизации всех процессов. В контексте информационных технологий во многом это удается благодаря разработке эффективных сервисов, которые направлены на облегчение работы людей, занимающихся рутинной и кропотливой деятельностью, которая, благодаря сервисам, теперь может выполняться практически без человеческого участия или контроля.

Подобные сервисы способствуют более эффективному распределению кадров: компаниям требуется меньше работников, чтобы обеспечивать деятельность предприятия, и, соответственно, появляется большая денежная выгода, благодаря которой можно также обеспечить более высокие зарплаты сотрудникам, повышая таким образом их мотивацию и продуктивность. Кроме того, если функции, выполняемые сервисами, занимали определенный процент обязанностей работника, то вместе с автоматизацией этих функций данный сотрудник получает возможность перенаправить свое время и усилия на более важные аспекты своей работы.

Автоматизация производства – это процесс в развитии машинного производства, при котором функции управления и контроля, ранее выполнявшиеся человеком, передаются приборам и автоматическим устройствам. Введение автоматизации на производстве позволяет значительно повысить производительность труда, обеспечить стабильное качество выпускаемой продукции, сократить долю рабочих, занятых в различных сферах производства [1].

До внедрения средств автоматизации замещение физического труда происходило посредством механизации основных и вспомогательных операций производственного процесса. Интеллектуальный труд долгое время оставался не механизированным (ручным). В настоящее время операции физического и интеллектуального труда, поддающиеся формализации, становятся объектом механизации и автоматизации [1].

Внедрение программы для автоматизации учета лекарств для сети аптек является технологией абсолютной необходимости. Обусловлено это многими факторами. Во–первых, подобный сервис обеспечивает разумное распределение ресурсов и кадров аптеки, при этом повышая эффективность и оперативность работы. Во–вторых, информация часто нуждается в своевременной коррекции, реализовать которую без помощи программного сервиса в разы дольше и сложнее. В–третьих, экономия времени в современном мире является приоритетной, особенно когда речь идет о таких процессах, как покупка и просмотр информации о различной продукции. Данная программная система позволит централизованно и структурированно хранить, и обрабатывать данные о лекарства, а также удобно осуществлять поиск и сортировку даже в самых объемных базах данных.

# **ПОСТАНОВКА ЗАДАЧИ**

# **Описание предметной области**

Аптека – особая специализированная организация системы здравоохранения, занимающаяся изготовлением, фасовкой, анализом и продажей лекарственных средств. Аптеку традиционно рассматривают как учреждение здравоохранения, а её деятельность формулируют как «оказание фармацевтической помощи населению». Фармацевтическая помощь включает в себя процедуру консультирования врача и пациента с целью определения наиболее эффективного, безопасного и экономически оправданного курса лечения [2].

Аптека осуществляет следующие функции:

* основной задачей аптеки является обеспечение населения, учреждений здравоохранения и других учреждений, предприятий и организаций лекарственными средствами, разрешенными к медицинскому применению в странах СНГ, изделиями медицинского назначения и другими предметами аптечного ассортимента;
* приготовление, контроль и отпуск лекарственных средств населению и лечебно–профилактическим учреждениям по оформленным в соответствии с установленными правилами рецептам и требованиям;
* реализация населению готовых лекарственных средств, разрешенных к медицинскому применению на территории стран СНГ и подлежащих отпуску по рецептам и без рецептов врача, а также изделий медицинского назначения и других предметов аптечного ассортимента;
* обеспечение прикрепленных на мед–обеспечение лечебно–профилактических и других учреждений, предприятий и организаций по оформленным в соответствии с установленными правилами требованиям лекарственными средствами и изделиями медицинского назначения. Прикрепление к аптеке конкретных учреждений, предприятий и организаций осуществляется вышестоящей организацией в порядке подчиненности;
* организация заготовки лекарственного растительного сырья;
* распространение среди населения санитарно–гигиенических знаний и проведение санпросвет работы;
* информация медицинских работников об имеющихся в аптеке и временно отсутствующих лекарственных средствах, поступлении новых лекарственных препаратов и способах их применения [3].

Аптека состоит из нескольких помещений: торговый зал, склад, бухгалтерия, включая кабинет директора. Аптека взаимодействует с поставщиками в лице менеджера, осуществляющими заказы на поставку лекарств, бытовой химии, медицинского оборудования и косметических средств. Каждый препарат имеет свой состав, наименование, артикул и показание к применению. Так же в аптеке есть свой штат сотрудников: фармацевты, бухгалтера, грузчики–водители, менеджеры. Клиент, приходя в аптеку, взаимодействует с фармацевтом, который, в свою очередь, осуществляет продажу. Операции с денежными средствами и отбивку чека осуществляет так же фармацевт. При покупке товара клиент получает чек с указанием товара и его стоимости.

В аптеке ведется документация: кассовая книга (сумма проданного товара, которая отдается бухгалтеру, а затем передается в налоговую для отчетности), приходная накладная, ведется учет доходов и расходов аптеки.

# **Цели и задачи на проектирование**

Целью курсовой работы является разработка программного продукта, предназначенного для автоматизации учета лекарств для аптек, а также систематизация и углубление знаний и навыков, выработанных в процессе изучения дисциплины «Управление информационными проектами».

Программный продукт должен быть интуитивным в использовании, понятным и удобным для рядовых пользователей. Все основные действия с базой должны осуществляться путем взаимодействий с интерфейсом приложения, вследствие чего вся сложность реализации будет скрыта от пользователя.

Программа должна осуществлять запросы на чтение, изменение, добавление или удаление данных, хранящихся в БД, в зависимости от соответствующих действий пользователя.

Для достижения поставленных целей необходимо решить следующие задачи:

* проанализировать предметную область;
* разработать структурную схему программы;
* разработать алгоритм работы программного продукта;
* разработать схему базы данных;
* разработать интерфейс пользователя;
* провести тестирование программного продукта.

# **ПРОЕКТИРОВАНИЕ**

# **Проектирование схемы данных**

В качестве концепции взаимодействия была выбрана модель клиент–сервер. Клиент–сервер – вычислительная или сетевая архитектура, в которой задания или сетевая нагрузка распределены между поставщиками услуг, называемыми серверами, и заказчиками услуг, называемыми клиентами. Фактически клиент и сервер – это программное обеспечение [4]. Общение между клиентом и сервером происходит посредством HTTP протокола с использованием GET, POST, PUT, DELETE видами запросов.

Архитектура курсового проекта двухуровневая клиент–серверная. Она основана на использовании только сервера базы – данных (DB–сервера), когда клиентская часть содержит уровень представления данных, а на сервере находится база данных вместе с СУБД и прикладными программами. DB–сервер отличается от файл–сервера тем, что в его оперативной памяти, помимо сетевой операционной системы, функционирует централизованная СУБД, которая обеспечивает совместное использование рабочими станциями базы данных, размещенной во внешней памяти этого DB–сервера [5].

Архитектурный подход – MVC и Redux. Model–View–Controller – схема разделения данных приложения и управляющей логики на три отдельных компонента: модель, представление и контроллер – таким образом, что модификация каждого компонента может осуществляться независимо. Redux представляет собой контейнер для управления состоянием приложения.

В курсовой работе применен REST. REST – архитектурный стиль взаимодействия компонентов распределённого приложения в сети. Другими словами, REST – это набор правил того, как программисту организовать написание кода серверного приложения, чтобы все системы легко обменивались данными и приложение можно было масштабировать.

Для описания функциональности разрабатываемой системы, обозначения рамок проекта и отображения пользовательских требований была разработана диаграмма вариантов использования (см. Приложение Г). В данной диаграмме представлено 4 вида пользователей: гость, владелец аптеки, работник и администратор. Гость – не авторизованный пользователь. Он может авторизоваться как работник либо же зарегистрировать свою аптеку. В случае если гость создаёт аптеку – у него появляются возможности управления внутри своей аптеки. Если же гость авторизовался как пользователь аптеки, то ему становится возможным взаимодействия с аптекой, определённые её владельцем. Человек, зарегистрировавший аптеку, выдаёт права доступа каждому работнику, например, права на добавление и удаление лекарств. Администратор имеет доступ ко всему функционалу системы и может быть выдан только с помощью SQL–запроса.

Алгоритм работы пользователя описан в приложении В.

# **Обоснование выбора технологий**

Для написания курсового проекта для написания серверной части был выбран язык программирования Java с использованием Spring Boot. Java – строго типизированный объектно–ориентированный язык программирования общего назначения. Преимуществами языка являются:

* Использование в корпоративных приложениях, Java способен поддержать строительные блоки системы или различные библиотеки, с их помощью создают необходимые функции;
* Запуск приложений в «песочнице» с устранением распространённых, уязвимых объектов в соответствии с политикой безопасности;
* Независимое положение от платформ;
* Повышенная производительность труда благодаря встроенному механизму, чтобы совместно использовать данные программы на нескольких компьютерах;
* Возможность автоматического управления памятью с одновременным независимым запуском потоков.

Преимуществами Spring Boot являются:

* Быстрая и легкая разработка приложений на основе Spring.
* Автоконфигурация всех компонентов для приложения Spring производственного уровня.
* Готовые встроенные серверы, обеспечивающие ускоренное и более продуктивное развертывание приложений.
* HTTP end–points, позволяющие вводить внутренние функции приложения, такие как показатели, состояние здоровья и другие.

В качестве языка, отвечающего за front–end сайта, были выбраны HTML, CSS, JS с использованием React Framework. Плюсы использования React:

* Виртуальная объектная модель документа;
* Повторное применение компонентов;
* Нисходящий поток данных.

Для работы с базами данных была выбрана СУБД PostgreSQL – свободная объектно–реляционная система управления базами данных. СУБД отличается высокой надёжностью и хорошей производительностью. PostgreSQL поддерживает транзакции (ACID), репликация реализована встроенными механизмами. При этом система расширяемая — можно создавать свои типы данных и индексов, а также расширять поведение при помощи языков программирования.

Серверная часть для взаимодействия с базой данных использует Hibernate – библиотека для языка программирования Java, предназначенная для решения задач объектно–реляционного отображения (ORM), самая популярная реализация спецификации JPA. Библиотека не только решает задачу связи классов Java с таблицами базы данных (и типов данных Java с типами данных SQL), но и также предоставляет средства для автоматической генерации и обновления набора таблиц, построения запросов и обработки полученных данных и может значительно уменьшить время разработки, которое обычно тратится на ручное написание SQL– и JDBC–кода. Hibernate автоматизирует генерацию SQL–запросов и освобождает разработчика от ручной обработки результирующего набора данных и преобразования объектов, максимально облегчая перенос (портирование) приложения на любые базы данных SQL.

# **Описание среды реализации**

Для написания серверной и клиентской частей приложения будет использоваться IntelliJ IDEA – интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java, JavaScript, разработанная компанией JetBrains с использованием Java версии 17.

Для взаимодействия с PostgreSQL будет использоваться pgAdmin 4 – бесплатная программа с графическим интерфейсом для разработки баз данных PostgreSQL.

# **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Автоматизация производства – Википедия [Электронный ресурс]. – Режим доступа – <https://ru.wikipedia.org/wiki/Автоматизация_производства> .
2. Аптека – Википедия [Электронный ресурс]. – Режим доступа – <https://ru.wikipedia.org/wiki/Аптека> .
3. Типовое положение об аптеке [Электронный ресурс]. – Режим доступа – [https://www.vsavm.by/veterinarnyj–nadzor/organizaciya–i–ekonomika–veterinarnoj–farmacii/tipovoe–polozhenie–o–xozraschetnoj–apteke/](https://www.vsavm.by/veterinarnyj-nadzor/organizaciya-i-ekonomika-veterinarnoj-farmacii/tipovoe-polozhenie-o-xozraschetnoj-apteke/) .
4. Клиент–сервер – Википедия [Электронный ресурс]. – Режим доступа – [https://ru.wikipedia.org/wiki/Клиент\_–\_сервер](https://ru.wikipedia.org/wiki/Клиент_—_сервер) .
5. Архитектура “Клиент–сервер” [Электронный ресурс]. – Режим доступа – <http://kgau.ru/istiki/umk/pis/30_1.htm> .

# **ПРИЛОЖЕНИЕ А**

(Обязательное)

Структура информационной системы

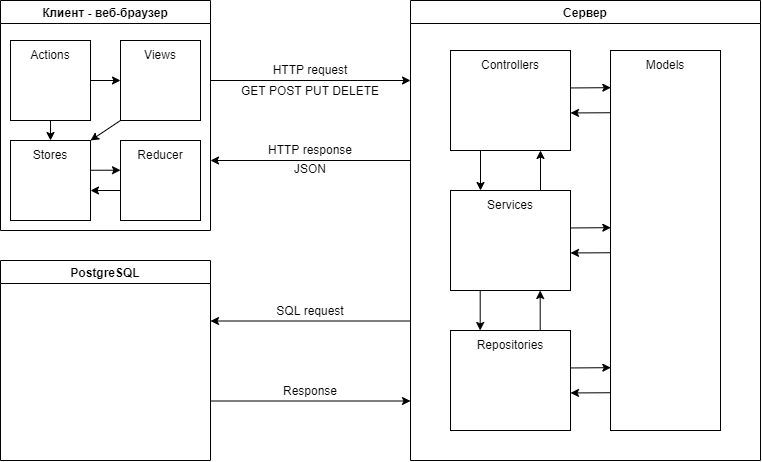


Рисунок А.1 – Структура информационной системы

# **ПРИЛОЖЕНИЕ Б**

(Обязательное)

Схема базы данных

Рисунок Б.1 – Схема базы данных

# **ПРИЛОЖЕНИЕ В**

(Обязательное)

Схема алгоритма работы пользователя

Diagram

Description automatically generated

Рисунок В.1 – Схема алгоритма работы пользователя

# **ПРИЛОЖЕНИЕ Г**

(Обязательное)

Диаграмма вариантов использования

Diagram

Description automatically generated

Рисунок Г.1 – Диаграмма вариантов использования

# **ПРИЛОЖЕНИЕ Д**

(Обязательное)

Диаграмма взаимодействия

Рисунок Д.1 – Структура информационной системы

# **ПРИЛОЖЕНИЕ Е**

(Обязательное)

Листинг кода