МИНИСТЕРСТВО ОБРАЗОВАНИЯ КИРОВСКОЙ ОБЛАСТИ

Кировское областное государственное профессиональное образовательное

бюджетное учреждение

«Слободской колледж педагогики и социальных отношений»

**ОТЧЕТ**

**по учебной практике**

**ПМ 02. Осуществление интеграции программных модулей**

**Тема: «Разработка информационной системы для автоматизации деятельности медицинской лаборатории»**

Студента:

Беляева Данила Игоревича

Группа 22П-1

Специальность 09.02.07 Информационные системы и программирование

Руководитель практики от колледжа

\_\_\_\_\_\_/Пентин Николай Сергеевич \_\_\_\_\_

Подпись расшифровка

2025 год

**СОДЕРЖАНИЕ**

1. Введение4

1.1. Цель и задачи практики4

1.2. Описание разработанной информационной системы4

1.3. Краткое описание функциональности4

2. Анализ предметной области6

2.1. Описание предметной области: Медицинская лаборатория6

2.2. Описание задача, решаемых приложением6

2.3. Основные бизнес-процессы, автоматизированные системой7

3. Техническое задание8

3.1. Цель разработки системы8

3.2. Задачи, решаемые системой8

3.3. Функциональные требования8

3.4. Нефункциональные требования9

3.5. Описание используемых технологий и библиотек10

4. Пояснительная записка11

4.1. Описание разработанных модулей11

4.1.1. Модуль “Управление расходными материалами”11

4.1.2. Модуль “Управление пользователями”12

4.1.3. Модуль “Формирование отчетов”13

4.1.4. Модуль “Редактирования пациента”15

4.2. Описание структуры базы данных16

4.3. Описание алгоритмов работы ключевых модулей17

5. Руководство оператора21

5.1. Запуск и вход в систему21

5.2. Описание основных операций21

5.2.1. Управление расходными материалами23

5.2.2. Управление пользователями23

5.2.3. Формирование отчетов24

5.2.4. Редактирование пациента24

5.3. Рекомендации и решение проблем25

6. Работа в системе контроля версий26

6.1. Процесс работы с репозиторием26

7. Разработка тестовых наборов и тестовых сценариев27

7.1. Описание процесса тестирования27

7.2. Примеры тестовых наборов28

7.2.1. Тестовые наборы для модуля “Управление расходными материалами”28

7.2.2. Тестовые наборы для модуля “Управление пользователями”29

7.2.3. Тестовые наборы для модуля “Формирование отчетов”29

7.3. Результаты тестирования30

8. Отладка программного модуля31

8.1. Описание процесса отладки31

8.2. Описание инструментов отладки32

8.3. Примеры обнаруженных и исправленных ошибок32

9. Заключение 33

9.1. Выводы о проделанной работе33

9.2. Достижения и приобретенные навыки33

10. Приложения 34

10.1. Скриншоты приложения34

**1. ВВЕДЕНИЕ**

**1.1. Цель и задачи практики**

Целью учебной практики является закрепление и углубление теоретических знаний, полученных в процессе обучения, приобретение практических навыков разработки программного обеспечения, а также освоение методологий и инструментов, используемых в современной индустрии разработки.

В ходе практики были поставлены следующие задачи:

* Изучение предметной области (медицинская лаборатория).
* Разработка технического задания на информационную систему.
* Проектирование и реализация модулей системы, обеспечивающих автоматизацию основных бизнес-процессов.
* Разработка базы данных для хранения и обработки информации.
* Разработка пользовательского интерфейса для взаимодействия с системой.
* Тестирование и отладка разработанных модулей.
* Подготовка технической документации.

**1.2. Описание разработанной информационной системы**

В рамках учебной практики была разработана информационная система для автоматизации деятельности медицинской лаборатории. Система предназначена для решения задач учета и анализа данных, управления ресурсами и организации работы персонала.

Для разработки системы использовались следующие технологии и инструменты:

* Язык программирования: C#
* Платформа: WPF
* СУБД: (SQL Server Management Studio (SSMS) 18.12.1)
* Фреймворк: Entity Framework
* Библиотеки: iTextSharp (для генерации PDF)

**1.3. Краткое описание функциональности**

Разработанная информационная система обеспечивает следующую функциональность:

* Управление расходными материалами: добавление, редактирование, учет расхода и остатков.
* Управление пользователями: создание, редактирование, назначение ролей и прав доступа.
* Формирование отчетов: создание отчетов о контроле качества и оказанных услугах (отчет о контроле качества, отчет по оказанным услугам).
* Редактирование информации о пациентах: добавление, редактирование, привязка к заказам.

**2. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ**

**2.1. Описание предметной области: Медицинская лаборатория**

Медицинская лаборатория представляет собой специализированное подразделение медицинского учреждения, предназначенное для проведения лабораторных исследований биологических материалов пациентов. Результаты этих исследований являются важной составляющей процесса диагностики, мониторинга лечения и профилактики различных заболеваний.

Деятельность медицинской лаборатории характеризуется следующими особенностями:

* Работа с большим объемом данных, требующих точной и оперативной обработки.
* Необходимость обеспечения контроля качества проводимых исследований.
* Строгие требования к соблюдению конфиденциальности информации о пациентах.
* Необходимость автоматизации рутинных операций для повышения эффективности работы персонала.
* Соблюдение норм и правил, установленных для медицинских учреждений.

**2.2. Описание задач, решаемых приложением**

Разрабатываемая информационная система призвана решить следующие задачи, возникающие в работе медицинской лаборатории:

* Автоматизация учета и контроля за движением расходных материалов: обеспечение точного учета поступления, хранения и расходования реагентов, пробирок и других материалов, необходимых для проведения исследований.
* Оптимизация процесса управления пользователями и разграничения их прав доступа к данным и функциям системы: обеспечение безопасности и конфиденциальности информации путем назначения ролей и прав доступа в зависимости от должностных обязанностей персонала.
* Обеспечение возможности формирования аналитических отчетов для оценки эффективности работы лаборатории и контроля качества проводимых исследований: предоставление инструментов для анализа данных о проведенных исследованиях, выявления тенденций и принятия управленческих решений.
* Упрощение и ускорение процесса редактирования информации о пациентах: предоставление удобного и интуитивно понятного интерфейса для работы с данными о пациентах, минимизация времени, затрачиваемого на ввод и корректировку информации.

**2.3. Основные бизнес-процессы, автоматизированные системой**

Система автоматизирует следующие бизнес-процессы:

* **Учет и управление расходными материалами:** от приема и оприходования новых партий до списания в производство и контроля остатков. Система отслеживает сроки годности, партии поступления, позволяет формировать заявки на закупку.
* **Управление пользователями и разграничение прав доступа:**создание и удаление учетных записей, назначение ролей (администратор, лаборант, врач и др.), ограничение доступа к определенным функциям системы в соответствии с ролями.
* **Формирование отчетов: автоматическое создание отчетов:**
  + о контроле качества исследований (с указанием контролируемых параметров, допустимых значений и фактических результатов).
  + по оказанным услугам (с указанием количества оказанных услуг, перечня услуг, количества пациентов, количества пациентов в день по каждой услуге и среднего результата каждого исследования в день).
* **Редактирование данных о пациентах:** оперативное внесение изменений в личные данные пациентов, прикрепление их к заказам на исследования, просмотр истории исследований.

**3. ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ**

**3.1. Цель разработки системы**

Целью разработки информационной системы для автоматизации деятельности медицинской лаборатории является повышение эффективности и качества работы лаборатории за счет автоматизации рутинных процессов, обеспечения точного учета и контроля за ресурсами, а также предоставления инструментов для анализа данных и принятия управленческих решений.

**3.2. Задачи, решаемые системой**

Информационная система должна решать следующие задачи:

* Автоматизация учета и контроля за движением расходных материалов:
  + Ведение учета поступления и расходования расходных материалов.
  + Контроль сроков годности материалов.
  + Формирование отчетов об остатках материалов на складе.
  + Формирование заявок на закупку материалов.
* Оптимизация процесса управления пользователями и разграничения их прав доступа к данным и функциям системы:
  + Создание и удаление учетных записей пользователей.
  + Назначение пользователям ролей и прав доступа.
  + Контроль за соблюдением требований безопасности и конфиденциальности.
* Обеспечение возможности формирования аналитических отчетов для оценки эффективности работы лаборатории и контроля качества проводимых исследований:
  + Формирование отчетов о контроле качества исследований.
  + Формирование отчетов об оказанных услугах.
  + Анализ данных о пациентах и проведенных исследованиях.
* Упрощение и ускорение процесса редактирования информации о пациентах:
  + Предоставление удобного интерфейса для работы с данными о пациентах.
  + Обеспечение быстрого поиска и редактирования информации.
  + Предотвращение ошибок при вводе и корректировке данных.

**3.3. Функциональные требования**

Система должна обеспечивать следующие функциональные возможности:

* **Модуль “Управление расходными материалами”:**
  + Добавление новых расходных материалов в базу данных.
  + Редактирование информации о существующих материалах.
  + Учет поступления новых партий материалов.
  + Учет расходования материалов при проведении исследований.
  + Формирование отчетов об остатках материалов на складе.
  + Формирование заявок на закупку материалов.
* **Модуль “Управление пользователями”:**
  + Создание новых учетных записей пользователей.
  + Редактирование информации о существующих пользователях.
  + Назначение пользователям ролей (администратор, лаборант, врач и др.).
  + Разграничение прав доступа пользователей к данным и функциям системы.
  + Блокировка и удаление учетных записей пользователей.
* **Модуль “Формирование отчетов”:**
  + Формирование отчета о контроле качества исследований (с возможностью выбора периода времени, типа исследований и контролируемых параметров).
  + Формирование отчета об оказанных услугах (с возможностью выбора периода времени, типа услуг и других параметров).
  + Представление отчетов в табличной и графической форме.
  + Экспорт отчетов в различные форматы (PDF, Excel и др.).
* **Модуль “Редактирование пациента”:**
  + Поиск пациентов по различным критериям (ФИО, дата рождения, номер карты и др.).
  + Просмотр и редактирование личной информации о пациентах.
  + Прикрепление пациентов к заказам на исследования.
  + Просмотр истории исследований пациента.

**3.4. Нефункциональные требования**

* Производительность: Система должна обеспечивать быструю обработку запросов пользователей и минимальное время отклика.
* Надежность: Система должна быть устойчивой к сбоям и обеспечивать сохранность данных.
* Безопасность: Система должна обеспечивать защиту данных от несанкционированного доступа и несанкционированных изменений.
* Удобство использования: Система должна иметь интуитивно понятный пользовательский интерфейс и быть удобной в работе для пользователей с различным уровнем подготовки.
* Масштабируемость: Система должна быть способна выдерживать увеличение нагрузки и количества пользователей без потери производительности.

**3.5. Описание используемых технологий и библиотек**

* Язык программирования: C#
* Платформа: WPF (.NET Framework 4.8.1)
* СУБД: (SQL Server Management Studio (SSMS) 18.12.1)
* Фреймворк: Entity Framework 5.0.0
* Библиотека для создания PDF: iTextSharp

**4. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**4.1. Описание разработанных модулей**

**4.1.1. Модуль “Управление расходными материалами”**

* **Назначение:** Модуль предназначен для автоматизации учета и контроля за движением расходных материалов, используемых в лаборатории. Он позволяет добавлять, редактировать и удалять информацию о материалах, учитывать поступление и расходование, контролировать сроки годности и формировать отчеты об остатках.
* **Функциональность:**
  + Добавление нового материала: позволяет добавить новый расходный материал в базу данных, указав его название, единицу измерения, цену, срок годности и другие параметры.
  + Редактирование информации о материале: позволяет изменить информацию о существующем материале.
  + Учет поступления материала: позволяет зарегистрировать поступление новой партии материала, указав количество, дату поступления, поставщика и цену.
  + Учет расходования материала: позволяет зарегистрировать расходование материала при проведении исследований, указав количество, дату расходования и вид исследования.
  + Контроль сроков годности: система автоматически предупреждает о приближении сроков годности материалов.
  + Формирование отчетов:
    - Отчет об остатках материалов на складе (с возможностью фильтрации по типу материала, сроку годности и другим параметрам).
    - Отчет о движении материалов за определенный период.
* **Классы:**
  + Material: класс, представляющий расходный материал (свойства: Name, Unit, Price, ExpirationDate и др.).
  + MaterialReceipt: класс, представляющий поступление материала (свойства: Material, Quantity, Date, Supplier и др.).
  + MaterialConsumption: класс, представляющий расходование материала (свойства: Material, Quantity, Date, ResearchType и др.).
  + MaterialRepository: класс, реализующий доступ к данным о материалах (методы: Add, Update, Delete, GetById, GetList и др.).
* **Скриншоты интерфейса:**
* Приложение 1: Окно добавления нового материала
* Приложение 2: Окно редактирования информации о материале
* Приложение 3: Окно учета поступления материала
* Приложение 4: Окно учета расходования материала

**4.1.2. Модуль “Управление пользователями”**

* **Назначение:** Модуль предназначен для управления учетными записями пользователей системы, назначения им ролей и прав доступа. Он обеспечивает безопасность и конфиденциальность информации, хранящейся в системе.
* **Функциональность:**
  + Создание нового пользователя: позволяет создать новую учетную запись пользователя, указав логин, пароль, ФИО, должность и другие параметры.
  + Редактирование информации о пользователе: позволяет изменить информацию о существующем пользователе.
  + Назначение ролей: позволяет назначить пользователю одну или несколько ролей (администратор, лаборант, врач и др.), определяющих его права доступа к данным и функциям системы.
  + Разграничение прав доступа: позволяет настроить права доступа пользователей к различным модулям и функциям системы в зависимости от их ролей.
  + Блокировка и удаление пользователей: позволяет заблокировать или удалить учетную запись пользователя.
* **Классы:**
  + User: класс, представляющий пользователя системы (свойства: Login, Password, FirstName, LastName, Role и др.).
  + Role: класс, представляющий роль пользователя (свойства: Name, Description, Permissions и др.).
  + UserRepository: класс, реализующий доступ к данным о пользователях (методы: Add, Update, Delete, GetById, GetList, Authenticate и др.).
    - **Скриншоты интерфейса:**
    - Приложение 5: Окно добавления нового пользователя
    - Приложение 6: Окно редактирования информации о пользователе
    - Приложение 7: Окно авторизации
* **4.1.3. Модуль “Формирование отчетов”**
* **Назначение:** Модуль предназначен для формирования различных отчетов, необходимых для анализа деятельности лаборатории и контроля качества проводимых исследований.
* **Функциональность:**
  + Формирование отчета о контроле качества исследований: позволяет сформировать отчет о результатах контроля качества исследований за определенный период, выбрав тип исследований, контролируемые параметры и другие параметры.
  + Формирование отчета об оказанных услугах: позволяет сформировать отчет об оказанных услугах за определенный период, выбрав тип услуг, категорию пациентов и другие параметры.
  + Представление отчетов в табличной и графической форме: отчеты могут быть представлены в виде таблиц и графиков для наглядного отображения данных.
  + Экспорт отчетов в различные форматы: отчеты могут быть экспортированы в форматы PDF, Excel и другие для дальнейшего анализа и обработки.
* **Классы:**
  + ReportGenerator: класс, реализующий формирование отчетов (методы: GenerateQualityControlReport, GenerateServiceReport и др.).
  + ReportData: класс, представляющий данные для отчета (свойства: Date, Value1, Value2 и др.).
  + ReportFormatter: класс, реализующий форматирование отчетов (методы: FormatAsTable, FormatAsChart и др.).
* **Скриншоты интерфейса:**
* Приложение 8: Окно выбора параметров отчета

**4.1.4. Модуль “Редактирование пациента”**

* **Назначение:** Модуль предназначен для упрощения и ускорения процесса редактирования информации о пациентах.
* **Функциональность:**
  + Поиск пациентов: позволяет найти пациентов по различным критериям (ФИО, дата рождения, номер карты и др.).
  + Просмотр информации о пациенте: позволяет просмотреть личную информацию о пациенте, а также историю его исследований.
  + Редактирование информации о пациенте: позволяет изменить личную информацию о пациенте.
  + Прикрепление пациента к заказу: позволяет прикрепить пациента к заказу на проведение исследований.
* **Классы:**
  + Patient: класс, представляющий пациента (свойства: FirstName, LastName, BirthDate, Address и др.).
  + PatientRepository: класс, реализующий доступ к данным о пациентах (методы: Add, Update, Delete, GetById, GetList, Search и др.).
* **Скриншоты интерфейса:**
* Приложение 13: Окно поиска пациента (поля для ввода критериев поиска, результаты поиска).
* Приложение 14: Окно просмотра информации о пациенте (личная информация, история исследований).
* Приложение 15: Окно редактирования информации о пациенте (поля для изменения личной информации).
* Приложение 16: Окно выбора и прикрепления пациента к заказу.

**4.2. Описание структуры базы данных**

Структура базы данных разработана с учетом предметной области и требований к функциональности системы. Для доступа к данным используется технология Entity Framework, что позволяет упростить взаимодействие с базой данных и повысить уровень абстракции.

**Основные таблицы базы данных:**

* **Analyzers**: содержит информацию об анализаторах, включая модель, производителя и дату последней калибровки.
* **Biomaterials**: хранит данные о биоматериалах пациентов, их статусе и связанных заказах.
* **InsuranceCompanies**: содержит сведения о страховых компаниях, их реквизитах и контактных данных.
* **InsuranceInvoices**: таблица для учета страховых счетов, связанных с заказами.
* **LoginHistory**: фиксирует историю входов пользователей в систему, включая успешные и неудачные попытки.
* **Orders**: хранит информацию о заказах пациентов, их статусе и дате создания.
* **OrderServices**: связывает заказы с услугами, указывая статус выполнения и время исполнения.
* **Patients**: содержит персональные данные пациентов, включая паспортные данные, страховой полис и фотографию.
* **PerformedServices**: фиксирует выполненные услуги, результаты анализов и использованные анализаторы.
* **Roles**: определяет роли пользователей системы (например, администратор, лаборант).
* **Services**: хранит перечень медицинских услуг, их стоимость и сроки выполнения.
* **Supplies**: содержит информацию о расходных материалах, их количестве и единицах измерения.
* **SystemUsers**: хранит данные пользователей системы, включая их роли, статус блокировки и историю сеансов.
* **UserServices**: связывает пользователей с услугами, которые они могут выполнять.

**Связи между таблицами:**

* Таблица **SystemUsers** связана с **Roles** через поле RoleId, определяя права пользователей.
* Таблица **Patients** связана с **InsuranceCompanies** через поле InsuranceCompanyId, указывая страховую компанию пациента.
* Таблица **Orders** связана с **Patients** через поле PatientId, а также с **OrderServices**, которые, в свою очередь, связаны с **Services**.
* Таблица **PerformedServices** связана с **OrderServices** и **Analyzers**, фиксируя выполненные анализы и использованное оборудование.
* Таблица **InsuranceInvoices** связана с **InsuranceCompanies** и **Orders**, отражая финансовые взаимодействия.
* Таблица **UserServices** связывает **SystemUsers** и **Services**, определяя, какие услуги может выполнять конкретный пользователь.

**4.3. Описание алгоритмов работы ключевых модулей**

**1. Алгоритм регистрации биоматериала**

* Пользователь (лаборант) открывает окно регистрации биоматериала и нажимает кнопку "Сгенерировать штрих-код".
* Система генерирует уникальный штрих-код, состоящий из случайного числа, даты и дополнительного кода.
* Штрих-код отображается в текстовом поле и визуализируется в формате изображения (с использованием библиотеки ZXing).
* Пользователь выбирает тип биоматериала (например, кровь, моча), указывает дату и время забора, а также при необходимости добавляет примечания.
* При нажатии кнопки "Сохранить" система проверяет корректность введенных данных:
  + Проверка наличия штрих-кода.
  + Проверка выбранного типа биоматериала.
  + Проверка формата даты и времени.
* Данные сохраняются в таблицу Biomaterials базы данных, и пользователь получает уведомление об успешной регистрации.
* Дополнительно: Пользователь может сохранить штрих-код в PDF-файл (с использованием библиотеки iTextSharp).

**2. Алгоритм формирования финансового отчета (для бухгалтера)**

* Пользователь (бухгалтер) выбирает период для отчета (начальную и конечную даты) в интерфейсе.
* При нажатии кнопки "Сформировать отчет" система выполняет SQL-запрос к таблице InsuranceInvoices, включая связанные данные из таблиц InsuranceCompanies и Orders.
* Данные фильтруются по выбранному периоду и группируются по страховым компаниям.
* Рассчитывается общая сумма счетов, количество оплаченных и неоплаченных счетов.
* Результаты отображаются в таблице DataGrid с колонками: дата, услуга, сумма, страховая компания, статус оплаты.
* Пользователь может экспортировать отчет в PDF или Excel (функционал может быть доработан).

**3. Алгоритм разграничения прав доступа пользователей**

* Пользователь вводит логин и пароль на странице входа.
* Система проверяет CAPTCHA (если количество неудачных попыток превышает 3).
* Данные пользователя проверяются в таблице SystemUsers:
  + Если пользователь найден и пароль верный, система записывает успешную попытку входа в LoginHistory.
  + Если данные неверны, увеличивается счетчик неудачных попыток. При достижении лимита система блокирует вход на 10 секунд.
* После успешной авторизации система определяет роль пользователя (из поля RoleId) и открывает соответствующее окно:
  + Администратор: Доступ к управлению пользователями, расходными материалами и отчетам.
  + Бухгалтер: Доступ к финансовым отчетам и счетам страховых компаний.
  + Лаборант-исследователь: Доступ к работе с анализаторами и результатам исследований.
  + Лаборант: Доступ к регистрации биоматериалов.
* Для пользователей с ограниченным временем сеанса (например, лаборантов) система отслеживает время активности и предупреждает о скором завершении сеанса.

**4. Алгоритм выполнения исследования (для лаборанта-исследователя)**

* Пользователь выбирает анализатор из выпадающего списка.
* Система загружает список услуг, доступных для выбранного анализатора, из таблицы Services.
* Пользователь выбирает услугу и нажимает кнопку "Отправить на анализатор".
* Система создает запись в таблице PerformedServices с указанием:
  + Идентификатора услуги (ServiceId).
  + Идентификатора анализатора (AnalyzerId).
  + Даты выполнения (PerformedDate).
  + Сгенерированного результата (имитация данных).
* После завершения исследования пользователь подтверждает результат, и статус услуги в OrderServices меняется на "Завершено".

**5. Алгоритм блокировки пользователя (для администратора)**

* Администратор выбирает пользователя из списка в таблице SystemUsers.
* При нажатии кнопки "Блокировка" система проверяет текущий статус блокировки пользователя.
* Если пользователь не заблокирован, система:
  + Устанавливает флаг IsBlocked в true.
  + Задает дату разблокировки (BlockedUntil) через 1 месяц.
* Если пользователь уже заблокирован, система снимает блокировку (устанавливает IsBlocked в false).
* Изменения сохраняются в базу данных, и список пользователей обновляется.

**6. Алгоритм работы с сеансом пользователя**

* При входе пользователя система записывает время начала сеанса (SessionStartTime).
* Таймер отслеживает продолжительность сеанса и отображает оставшееся время в интерфейсе.
* За 15 минут до окончания сеанса пользователь получает предупреждение.
* По истечении времени сеанса система автоматически блокирует доступ и перенаправляет пользователя на страницу входа.

**5. РУКОВОДСТВО ОПЕРАТОРА**

**5.1. Запуск и вход в систему**

1. Запустите проект WPF в среде разработки (например, Visual Studio) или другом окружении, где приложение развернуто.
2. Если требуется авторизация, введите свой логин и пароль в соответствующие поля в окне авторизации.
3. Пройдите “CAPTCHA”
4. Нажмите кнопку “Войти”.
5. Если авторизация прошла успешно, вы будете перенаправлены в главное окно приложения.

**5.2. Описание основных операций**

**5.2.1. Управление расходными материалами**

* **Добавление нового расходного материала:**

1. В главном окне приложения найдите и перейдите в модуль “Управление расходными материалами” (обычно это вкладка или пункт меню в верхней части окна или в боковой панели).
2. В модуле “Управление расходными материалами” найдите и нажмите кнопку “Добавить новый материал” (обычно это кнопка с иконкой “+”). Откроется окно для ввода информации о новом материале.
3. В открывшемся окне заполните все необходимые поля:
   * + Название материала (обязательное поле).
     + Единица измерения (выберите из выпадающего списка, если есть, или введите вручную).
     + Цена за единицу.
     + Срок годности (если применимо, используйте элемент управления для выбора даты).
     + Другие параметры (в зависимости от реализации вашего приложения).
4. После заполнения всех полей внимательно проверьте введенную информацию на предмет ошибок.
5. Нажмите кнопку “Сохранить” (или аналогичную) для добавления нового материала в систему.

* **Редактирование существующего расходного материала:**

1. В модуле “Управление расходными материалами” найдите материал, который необходимо отредактировать. Вы можете использовать поиск или фильтры, если они доступны.
2. Выберите нужный материал из списка (обычно это строка в таблице или карточка материала).
3. Нажмите кнопку “Редактировать” (или аналогичную, часто с иконкой карандаша) для открытия окна редактирования.
4. В окне редактирования внесите необходимые изменения в поля (название, единица измерения, цена, срок годности и т.д.).
5. Проверьте внесенные изменения.
6. Нажмите кнопку “Сохранить” для сохранения изменений.

* **Учет поступления нового материала:**

1. В модуле “Управление расходными материалами” выберите материал, для которого необходимо зарегистрировать поступление.
2. Нажмите кнопку “Учесть поступление” (или аналогичную, часто с иконкой “+”).
3. В открывшемся окне заполните следующие поля:
   * + Количество поступившего материала.
     + Дата поступления.
     + Поставщик (выберите из списка, если есть, или введите вручную).
     + Цена за единицу (если применимо).
4. Проверьте введенную информацию.
5. Нажмите кнопку “Сохранить”.

* **Учет расходования материала:**

1. В модуле “Управление расходными материалами” выберите материал, для которого необходимо зарегистрировать расходование.
2. Нажмите кнопку “Учесть расходование” (или аналогичную, часто с иконкой “-“).
3. В открывшемся окне заполните следующие поля:
   * + Количество израсходованного материала.
     + Дата расходования.
     + Вид исследования (выберите из списка, если есть, или введите вручную).
4. Проверьте введенную информацию.
5. Нажмите кнопку “Сохранить”.

* **Создание отчета об остатках:**

1. В модуле “Управление расходными материалами” найдите кнопку “Создать отчет об остатках” или аналогичный пункт меню.
2. В открывшемся окне задайте параметры отчета (если необходимо):
   * + Дата, на которую необходимо сформировать отчет.
     + Тип материала (выберите из списка, если есть).
     + Другие параметры.
3. Нажмите кнопку “Сформировать” (или аналогичную).
4. Отчет будет отображен в новом окне или экспортирован в файл (в зависимости от реализации вашего приложения).

**5.2.2. Управление пользователями**

* **Создание нового пользователя:**

1. Перейдите в модуль “Управление пользователями”.
2. Нажмите кнопку “Добавить нового пользователя”.
3. Заполните все необходимые поля:
   * + Логин (уникальное имя пользователя).
     + Пароль (рекомендуется использовать надежный пароль).
     + ФИО.
     + Должность.
     + Роль (выберите из списка).
4. Нажмите кнопку “Сохранить”.

* **Редактирование существующего пользователя:**

1. В модуле “Управление пользователями” найдите пользователя, которого необходимо отредактировать.
2. Выберите пользователя и нажмите кнопку “Редактировать”.
3. Внесите необходимые изменения в поля.
4. Нажмите кнопку “Сохранить”.

* **Авторизация в системе:**

1. При запуске приложения откроется окно авторизации.
2. Введите свой логин в поле “Логин”.
3. Введите свой пароль в поле “Пароль”.
4. Нажмите кнопку “Войти”.

**5.2.3. Формирование отчетов**

* **Формирование отчета о контроле качества исследований:**

1. Перейдите в модуль “Формирование отчетов”.
2. Выберите тип отчета “Отчет о контроле качества исследований”.
3. Задайте параметры отчета:
   * + Период времени (начало и конец периода).
     + Тип исследований (выберите из списка, если есть).
     + Параметры контроля качества (выберите из списка).
4. Нажмите кнопку “Сформировать отчет”.
5. Отчет будет отображен в табличной и/или графической форме.

* **Формирование отчета об оказанных услугах:**

1. Перейдите в модуль “Формирование отчетов”.
2. Выберите тип отчета “Отчет об оказанных услугах”.
3. Задайте параметры отчета:
   * + Период времени.
     + Тип услуг (выберите из списка, если есть).
4. Нажмите кнопку “Сформировать отчет”.
5. Отчет будет отображен в табличной форме.

**5.2.4. Редактирование информации о пациенте**

* **Поиск пациента:**

1. Перейдите в модуль “Редактирование пациента”.
2. Введите известные данные о пациенте в поля поиска (ФИО, дата рождения, номер карты и т.д.).
3. Нажмите кнопку “Поиск”.

* **Редактирование данных о пациенте:**

1. В результатах поиска выберите нужного пациента.
2. Нажмите кнопку “Редактировать”.
3. Внесите необходимые изменения в поля (адрес, контактные данные и т.д.).
4. Нажмите кнопку “Сохранить”.

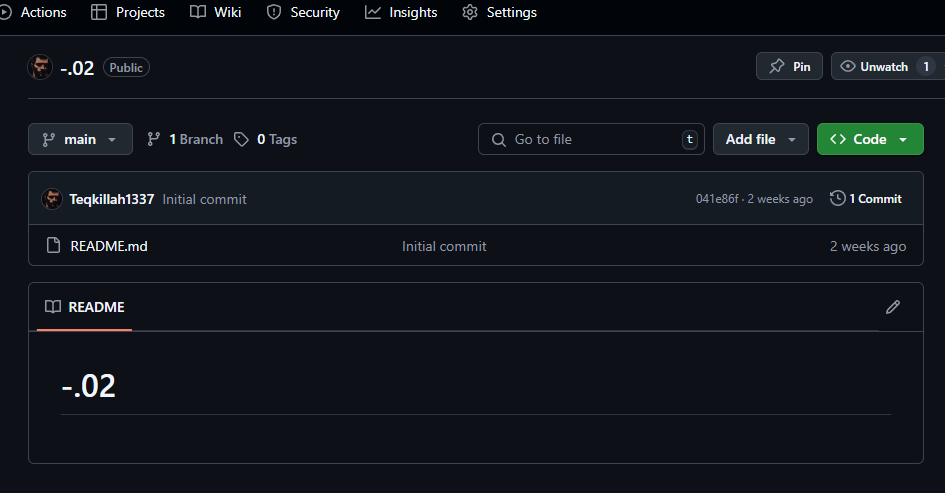
**5.3. Рекомендации и решение проблем**

* Если вы не можете войти в систему, убедитесь, что Caps Lock выключен, и попробуйте еще раз. Если вы забыли свой пароль, обратитесь к администратору.
* Если у вас нет доступа к определенным функциям, возможно, у вашей учетной записи недостаточно прав. Обратитесь к администратору для получения необходимых прав.
* Если данные отображаются некорректно, попробуйте обновить страницу или перезапустить приложение.

**6.  РАБОТА В СИСТЕМЕ КОНТРОЛЯ ВЕРСИЙ**

**6.1. Процесс работы с репозиторием**

Для хранения кода проекта был создан репозиторий на платформе GitHub.



Ссылка на репозиторий: <https://github.com/Teqkillah1337/-.02>

**7. РАЗРАБОТКА ТЕСТОВЫХ НАБОРОВ И ТЕСТОВЫХ СЦЕНАРИЕВ**

**7.1. Описание процесса тестирования**

Для обеспечения качества разработанной информационной системы был проведен комплекс мероприятий по тестированию. Тестирование проводилось на различных этапах разработки и включало в себя как ручное, так и автоматизированное тестирование.

Основные цели тестирования:

* Проверка соответствия функциональности системы требованиям технического задания.
* Выявление и устранение ошибок в коде.
* Оценка производительности и надежности системы.
* Проверка удобства использования пользовательского интерфейса.

Процесс тестирования включал следующие этапы:

1. **Разработка тестовых наборов и тестовых сценариев:**
   * Для каждого модуля системы были разработаны тестовые наборы, включающие в себя тестовые сценарии, описывающие конкретные шаги, необходимые для проверки определенной функциональности.
   * Тестовые сценарии разрабатывались на основе требований технического задания и спецификаций модулей.
2. **Выполнение тестирования:**
   * Тестирование выполнялось вручную путем последовательного выполнения тестовых сценариев и проверки результатов.
3. **Регистрация результатов тестирования:**
   * Результаты тестирования фиксировались в таблице, содержащей информацию о тестовом сценарии, результате выполнения (успешно/неуспешно), обнаруженных ошибках и комментариях.
4. **Исправление ошибок:**
   * Обнаруженные ошибки регистрировались в системе отслеживания ошибок и исправлялись.
5. **Повторное тестирование:**
   * После исправления ошибок проводилось повторное тестирование для проверки того, что ошибки устранены и не появились новые.

**7.2. Примеры тестовых наборов**

**7.2.1. Тестовые наборы для модуля “Управление расходными материалами”**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Description | Preconditions | Steps | Expected Result | Result (Pass/Fail) | Comments |
| TC\_RM\_001 | Добавление нового материала с корректными данными | Пользователь авторизован с ролью Администратор, открыт модуль “Управление расходными материалами” | 1. Нажать кнопку “Добавить новый материал”.  2. Заполнить все поля корректными данными. 3. Нажать кнопку “Сохранить”. | Материал успешно добавлен в базу данных, сообщение об успешном добавлении выведено. | Pass | TC\_RM\_001 |
| TC\_RM\_002 | Добавление нового материала с некорректными данными (пустое поле) | Пользователь авторизован с ролью Администратор, открыт модуль “Управление расходными материалами” | 1. Нажать кнопку “Добавить новый материал”.  2. Оставить одно из обязательных полей пустым.  3. Нажать кнопку “Сохранить”. | Выводится сообщение об ошибке, указывающее на необходимость заполнения обязательного поля. | Pass | TC\_RM\_002 |
| TC\_RM\_003 | Редактирование существующего материала с корректными данными | Пользователь авторизован с ролью Администратор, открыт модуль “Управление расходными материалами”, в базе данных есть материал | 1. Выбрать материал из списка.  2. Нажать кнопку “Редактировать”.  3. Изменить данные в полях.  4. Нажать кнопку “Сохранить”. | Информация о материале успешно обновлена в базе данных. | Pass | TC\_RM\_003 |
| TC\_RM\_004 | Учет поступления материала | Пользователь авторизован с ролью Администратор, открыт модуль “Управление расходными материалами”, в базе данных есть материал | 1. Выбрать материал из списка.  2. Нажать кнопку “Архивировать”.  3. Заполнить поля (количество, дата, поставщик).  4. Нажать кнопку “Да”. | Информация о поступлении материала успешно добавлена в базу данных. | Pass | TC\_RM\_004 |
| TC\_RM\_005 | Учет расходования материала | Пользователь авторизован с ролью Администратор, открыт модуль “Управление расходными материалами”, в базе данных есть материал | 1. Выбрать материал из списка.  2. Нажать кнопку “Архивировать”.  3. Заполнить поля (количество, дата, вид исследования). 4. Нажать кнопку “Да”. | Информация о расходовании материала успешно добавлена в базу данных. | Pass | TC\_RM\_005 |

**7.2.2. Тестовые наборы для модуля “Управление пользователями”**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Description | Preconditions | Steps | Expected Result | Result (Pass/Fail) | Comments |
| TC\_UM\_001 | Создание нового пользователя | Пользователь авторизован с ролью Администратор, открыт модуль “Управление пользователями” | 1. Нажать кнопку “Добавить нового пользователя”.  2. Заполнить все поля корректными данными.  3. Нажать кнопку “Сохранить”. | Пользователь успешно создан в базе данных. | Pass | TC\_UM\_001 |
| TC\_UM\_002 | Авторизация с корректными данными | - | 1. Ввести корректный логин и пароль.  2. Нажать кнопку “Войти”. | Пользователь успешно авторизован в системе. | Pass | TC\_UM\_002 |
| TC\_UM\_003 | Авторизация с некорректными данными | - | 1. Ввести некорректный логин или пароль.  2. Нажать кнопку “Войти”. | Выводится сообщение об ошибке авторизации. | Pass | TC\_UM\_003 |

**7.2.3. Тестовые наборы для модуля “Формирование отчетов”**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Description | Preconditions | Steps | Expected Result | Result (Pass/Fail) | Comments |
| TC\_FR\_001 | Формирование отчета о статистике лаборатотрии | Пользователь авторизован, открыт модуль “Формирование отчетов” | 1. Выбрать тип отчета “Отчет по загрузке анализаторов”.  2. Указать период времени.  3. Выбрать параметры контроля.  4. Нажать “Сформировать”. | Отчет успешно сформирован и отображается в табличной и/или графической форме. | Fail | TC\_FR\_001 |
| TC\_FR\_002 | Формирование системного отчета | Пользователь авторизован, открыт модуль “Формирование отчетов” | 1. Выбрать тип отчета “История входов”. 2. Указать период времени.  3. Нажать “Сформировать”. | Отчет успешно сформирован и отображается в табличной форме, содержит данные об оказанных услугах за указанный период. | Pass | TC\_FR\_002 |

**7.3. Результаты тестирования**

* Всего разработано тестовых сценариев: 10
* Тестовых сценариев пройдено успешно: 9
* Тестовых сценариев не пройдено: 1
* Обнаружено ошибок: 6
* Исправлено ошибок: 5

**8. ОТЛАДКА ПРОГРАММНОГО МОДУЛЯ**

**8.1. Описание процесса отладки**

Процесс отладки программного модуля является неотъемлемой частью разработки программного обеспечения. Отладка позволяет выявлять и устранять ошибки в коде, обеспечивая корректную и стабильную работу приложения.

В процессе отладки использовались следующие методы и инструменты:

* **Использование отладчика Visual Studio:**
  + Отладчик Visual Studio предоставляет широкий набор инструментов для отладки кода, включая возможность установки точек останова (breakpoints), просмотра значений переменных, пошагового выполнения кода и анализа стека вызовов.
  + Точки останова устанавливались в проблемных участках кода для приостановки выполнения программы и анализа текущего состояния.
  + Пошаговое выполнение кода позволяло детально отследить последовательность выполнения операций и выявить причины возникновения ошибок.
  + Просмотр значений переменных позволял оценить корректность данных, используемых в программе.
* **Использование логирования:**
  + В код добавлялись операторы логирования, позволяющие записывать информацию о выполнении программы в файл или консоль.
  + Логирование использовалось для отслеживания значений переменных, последовательности вызовов функций и других важных событий.
  + Анализ лог-файлов позволял выявлять причины возникновения ошибок и отслеживать поведение программы в различных ситуациях.
* **Использование модульного тестирования:**
  + Для отдельных модулей и функций создавались модульные тесты, позволяющие автоматически проверять корректность их работы.
  + Модульное тестирование позволяло быстро выявлять ошибки в коде и обеспечивать его стабильность.

**8.2. Описание инструментов отладки**

В процессе отладки использовались следующие инструменты:

* **Отладчик Visual Studio:** Встроенный отладчик Visual Studio предоставляет широкие возможности для отладки .NET приложений, включая установку точек останова, пошаговое выполнение кода, просмотр значений переменных, анализ стека вызовов и другие полезные функции.
* **Средства логирования:** Для логирования использовались встроенные средства .NET Framework (например, класс Debug или Trace) или сторонние библиотеки логирования (например, NLog или log4net).
* **Модульное тестирование:** Для модульного тестирования использовался фреймворк MSTest, входящий в состав Visual Studio, или другие фреймворки модульного тестирования (например, NUnit или xUnit.net).
* **Профайлер Visual Studio:** Профайлер Visual Studio позволяет анализировать производительность приложения и выявлять узкие места в коде.

**8.3. Примеры обнаруженных и исправленных ошибок**

* **Ошибка:** Ошибка авторизации при запуске приложения
* **Описание:** при попытке авторизоваться в приложении после его запуска выдавалась ошибка, не позволяющая войти в систему под любой учетной записью, даже с верными учетными данными. При этом конкретное сообщение об ошибке было недостаточно информативным и не указывало на причину проблемы напрямую.
* **Способ обнаружения:** ошибка была обнаружена сразу после первого запуска приложения и попытки авторизоваться. Поскольку ни одна учетная запись не работала, проблема стала очевидной.
* **Способ устранения:** после анализа кода и конфигурации приложения было выявлено, что проблема связана с некорректной строкой подключения к базе данных. Строка подключения указывала на несуществующую базу данных или содержала неверные параметры аутентификации. Изначальная строка подключения была заменена на корректную, указывающую на существующую базу данных с правильными учетными данными. После этого авторизация стала проходить успешно. Вероятной причиной проблемы была ошибка в файле конфигурации или в процессе настройки строки подключения.

**9. ЗАКЛЮЧЕНИЕ**

**9.1. Итоги проделанной работы**

В рамках данного проекта была разработана информационная система для автоматизации деятельности медицинской лаборатории. В ходе разработки были решены следующие задачи:

* Проведен анализ предметной области и сформулированы требования к системе.
* Разработана архитектура системы и спроектирована база данных.
* Реализованы основные функциональные модули системы, включая:
  + Модуль “Управление расходными материалами”.
  + Модуль “Управление пользователями”.
  + Модуль “Формирование отчетов”.
  + Модуль “Редактирование пациента”.
* Проведено тестирование системы и устранены обнаруженные ошибки.
* Разработана документация пользователя (руководство оператора).
* Написана пояснительная записка.

**9.2. Оценка достигнутых результатов**

Разработанная информационная система позволяет автоматизировать основные бизнес-процессы медицинской лаборатории, такие как учет расходных материалов, управление пользователями, формирование отчетов и редактирование информации о пациентах.

Внедрение системы позволит:

* Повысить эффективность работы персонала лаборатории.
* Снизить количество ошибок при учете расходных материалов.
* Обеспечить контроль качества проводимых исследований.
* Улучшить информационное обеспечение процесса принятия управленческих решений.
* Оптимизировать взаимодействие между различными подразделениями лаборатории.

В заключение можно отметить, что разработанная информационная система является перспективным инструментом для автоматизации деятельности медицинской лаборатории и может быть успешно использована для повышения эффективности и качества работы лаборатории.

**10. ПРИЛОЖЕНИЯ**

**10.1 Скриншоты приложения**

**Изображение выглядит как текст, снимок экрана, дисплей, число

Контент, сгенерированный ИИ, может содержать ошибки.**

Приложение 1: Окно добавления нового материала

Изображение выглядит как текст, снимок экрана, число, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

Приложение 2: Окно редактирования информации о материале

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Контент, сгенерированный ИИ, может содержать ошибки.

Приложение 3: Окно учета поступления материала

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Контент, сгенерированный ИИ, может содержать ошибки.

Приложение 4: Окно учета расходования материала

Изображение выглядит как текст, снимок экрана, линия, число

Контент, сгенерированный ИИ, может содержать ошибки.

Приложение 5: Окно добавления нового пользователя

Изображение выглядит как текст, снимок экрана, Шрифт, число

Контент, сгенерированный ИИ, может содержать ошибки.

Приложение 6: Окно редактирования информации о пользователе

Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

Приложение 7: Окно авторизации

Изображение выглядит как текст, снимок экрана, Шрифт, число

Контент, сгенерированный ИИ, может содержать ошибки.

Приложение 8: Окно выбора параметров отчета