**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение

высшего образования

«Санкт-Петербургский политехнический университет Петра Великого»

(ФГАОУ ВО «СПбПУ»)

**Институт среднего профессионального образования**

УТВЕРЖДАЮ

Зам. директора по УПР

С.Ю. Назаров\_\_\_\_\_\_\_\_

«\_\_\_\_»\_\_\_\_\_\_\_\_20\_\_\_г.

**ОТЧЕТ**

**по учебной практике УП.02.01**

По профессиональному модулю ПМ.02 «Осуществление интеграции программных модулей»

(код и наименование)

Специальность09.02.07 «Информационные системы и программирование»

(код и наименование специальности)

Студент(ка) 4 курса 42919/4 группы

Смирнов Андрей Александрович

(Фамилия, имя, отчество)

Место прохождения практики: ФГАОУ ВО «СПбПУ» Институт среднего профессионального образования, пр. Энгельса д. 23

(наименование и адрес организации)

Период прохождения практики

с «9» декабря 2024 г. по «21» декабря 2024 г.

Руководитель практики Прокофьев А.А.

(подпись) (расшифровка подписи)

Итоговая оценка по практике\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Санкт-Петербург

2024

**ЗАДАНИЕ**

**на учебную практику** **УП.02.01**

по профессиональному модулю ПМ.02 «Осуществление интеграции программных модулей»

(код и наименование)

Специальность09.02.07 Информационные системы и программирование

(код и наименование специальности)

Студент(ка) 4 курса 42919/4 группы

Смирнов Андрей Александрович

(Фамилия, имя, отчество)

Место прохождения практики: ФГАОУ ВО «СПбПУ» Институт среднего профессионального образования, Энгельса 23

(наименование и адрес организации)

Период прохождения практики

с «9» декабря 2024 г. по «21» декабря 2024 г.

**Виды работ, обязательные для выполнения**

1. Участие в выработке требований к программному обеспечению
2. Стадии проектирования программного обеспечения
3. Разработка модулей программного обеспечения
4. Тестирование программных модулей и их интеграции
5. Разработка программной документации. Стандарты кодирования

**Индивидуальное задание: ВАРИАНТ 13**

Задание выдал «9» декабря 2024 г. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Прокофьев А.А.

(подпись)  (Ф.И.О.)

Задание получил «9» декабря 2024 г. \_\_\_\_\_\_\_\_\_\_\_\_ Смирнов А.А.(подпись)  (Ф.И.О.)

СОГЛАСОВАНО

Председатель ПЦК

\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_\_» \_\_\_\_\_\_\_\_\_ 202\_\_ г.

**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение

высшего образования

«Санкт-Петербургский политехнический университет Петра Великого»

(ФГАОУ ВО «СПбПУ»)

**Институт среднего профессионального образования**

**ДНЕВНИК**

**прохождения учебной практики УП 02.01**

По профессиональному модулю ПМ.02 «Осуществление интеграции программных модулей»

(код и наименование)

Специальность09.02.07 Информационные системы и программирование

(код и наименование специальности)

Студент(ка) 4 курса 42919/4 группы

Смирнов Андрей Александрович

(Фамилия, имя, отчество)

Место прохождения практики ФГАОУ ВО «СПбПУ» Институт среднего профессионального образования, пр. Энгельса д. 23

(наименование и адрес организации)

Период прохождения практики

с «9» декабря 2024 г. по «21» декабря 2024 г.

Руководитель практики Прокофьев А.А.

(подпись) (расшифровка подписи)

Итоговая оценка по практике

Санкт-Петербург

2024

**Содержание дневника**

|  |  |  |
| --- | --- | --- |
| Дата | Виды выполненных работ и заданий по программе практики | Подпись руководителя практики |
| **1** | **2** | **3** |
| 09.12.24 | Анализ предметной области. |  |
| 09.12.24 | Выявление требований к программе. |  |
| 10.12.24 | Разработка технического задания. |  |
| 10.12.24 | UML. Проектирование диаграммы вариантов использования. |  |
| 11.12.24 | UML. Проектирование диаграммы последовательности. |  |
| 11.12.24 | UML. Проектирование диаграммы активности. |  |
| 12.12.24 | Моделирование структуры ПО. |  |
| 12.12.24 | Проектирование инфологической и даталогической модели данных. |  |
| 13.12.24 | Проектирование интерфейса пользователя. Создание Wireframe эскизов. |  |
| 13.12.24 | Разработка дизайна программы в соответствии с руководством по стилю. |  |
| 14.12.24 | Разработка базы данных. |  |
| 14.12.24 | Разработка словаря данных. |  |
| 16.12.24 | Создание приложения. Форма авторизации. |  |
| 16.12.24 | Создание приложения. Форма врачей. |  |
| 17.12.24 | Создание приложения. Основные формы приложения. |  |
| 17.12.24 | Разработка библиотеки классов. |  |
| 18.12.24 | Отладка программных модулей. |  |
| 19.12.24 | Модульное тестирование. |  |
| 19.12.24 | Создание тестовых случаев. |  |
| 20.12.24 | Интеграционное тестирование. |  |
| 20.12.24 | Разработка самодокументирующегося кода. |  |
| 21.12.24 | Инспекция кода на соответствие стандартам кодирования. |  |
| 21.12.24 | Подготовка отчета и размещение результатов в репозитории контроля версий. |  |

**Содержание дневника**

|  |  |  |
| --- | --- | --- |
| Дата | Виды выполненных работ и заданий по программе практики | Подпись руководителя практики |
| **1** | **2** | **3** |
| 09.12.24 | Участие в выработке требований к программному обеспечению. |  |
| 11.12.24 | Стадии проектирования программного обеспечения. |  |
| 13.12.24 | Разработка модулей программного обеспечения. |  |
| 19.12.24 | Тестирование программных модулей и их интеграции. |  |
| 20.12.24 | Разработка программной документации. Стандарты кодирования. |  |

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 9](#_Toc185498796)

[Задание №1 «Разработка технического задания» 10](#_Toc185498797)

[Задание №2 «Разработка алгоритмов и диаграмм» 16](#_Toc185498803)

[Задание №3 «Разработка макетов» 26](#_Toc185498813)

[Задание №4 «Разработка программы» 42](#_Toc185498818)

[Задание №5 «Тестирование» 55](#_Toc185498824)

[Задание №6 «GitHub.Оценка проекта» 70](#_Toc185498828)

[ЗАКЛЮЧЕНИЕ 82](#_Toc185498834)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 83](#_Toc185498835)

ВВЕДЕНИЕ

Учебная практика проходила в ФГАОУ ВО СПбПУ Институт среднего профессионального образования в период с 09.12.24 по 21.12.24. Этот этап является важным компонентом подготовки квалифицированных специалистов, так как позволяет закрепить теоретические знания на практике и получить опыт работы в реальных условиях.

Учебная практика представляет собой вид учебно-вспомогательного процесса, направленного на развитие практических навыков и умений, необходимых для успешной профессиональной деятельности. В ходе практики студенты приобретают опыт работы с современными технологиями и инструментами, что способствует их профессиональному росту и адаптации к требованиям рынка труда.

Целью практики являлось закрепление практических умений и получение дополнительных навыков в разработке программных модулей программного обеспечения для компьютерных систем. Для достижения поставленной цели был выполнен ряд задач, включая разработку спецификаций отдельных компонент, конструирование пользовательского интерфейса, проектирование и разработку модулей программного обеспечения, модульное тестирование, а также документирование программных продуктов.

В ходе практики была работа над реальными задачами, что позволило им приобрести навыки, необходимые для работы в сфере информационных технологий. Результаты практики отражают уровень подготовки студентов и их готовность к профессиональной деятельности.

Задание №1 «Разработка технического задания»

**Цель работы:** развитие навыков разработки технического задания (ТЗ) и понимания процесса взаимодействия между заказчиком и исполнителем.

Поделиться на пары: каждый из вас будет и исполнителем, и заказчиком. Заказчику необходимо разработать задание для исполнителя согласно его варианту, которое будет состоять из описания предметной области со всеми пожеланиями к готовому программному продукту, включая внешний вид (желаемый цвет фона, шрифт и т.д.). После составления предметной области необходимо поменяться готовым заданием друг с другом и приступить к обсуждению, по результатам которого исполнитель, так же каждый из вас, должен будет составить подробное техническое задание со всеми функциями, описанные заказчиком. Во время обсуждения предметная область может дорабатываться. Учтите при составлении предметной области и технического задания следующие требования:

* Минимум 3 роли (например, администратор, сотрудник и клиент);
* Возможность реализовать не менее 5 таблиц в БД;
* Программный продукт должен включать основные функции, такие как: удаление/добавление/изменение всех ключевых таблиц, их корректное отображение, вывод ошибок и предупреждений, авторизация, масштабирование, составление и выгрузка отчетов, админ панель.

Варианты:

* 1. Торговля
  2. Турагентство
  3. Библиотека
  4. Поликлиника
  5. ГАИ
  6. Цветочный магазин
  7. Типография
  8. Авиакомпания
  9. Аптека
  10. Склад
  11. Фабрика
  12. Банк
  13. Прокат автомобилей
  14. Салон красоты
  15. Школа
  16. Почта
  17. Фитнес-клуб
  18. Спортивный клуб (Футбольный клуб/Баскетбольный)
  19. Автосалон
  20. Музей
  21. Ресторан
  22. Поставка товаров
  23. Страховая компания
  24. Компьютерная мастерская
  25. Компьютерный клуб
  26. Ломбард
  27. Гостиница
  28. Химчистка
  29. Театр
  30. Агентство недвижимости

Выполнение задания

Вариант «Поликлиника»

### **Описание предметной области**

Основной целью данного программного продукта является автоматизация процессов учета и обработки данных в поликлинике. Программа предоставляет сотрудникам удобный инструмент для управления информацией о пациентах, врачах, расписании и медицинских историях, обеспечивая прозрачность и эффективность работы.

Разработка базы данных для поликлиники – это процесс создания программного продукта, который позволяет медицинским учреждениям эффективно управлять данными и организовывать работу. Задача разработки состоит в создании удобного и функционального программного обеспечения, которое минимизирует время обработки данных, упрощает взаимодействие между сотрудниками поликлиники и позволяет улучшить качество медицинских услуг. Результат использования программы:

Снижение нагрузки на администраторов за счет автоматизации рутинных задач, повышение точности учета данных пациентов и расписания., оптимизация времени работы врачей и доступность информации.

Основные функции и возможности модуля включают:

1. Регистрация: процесс создания учётной записи клиента, который включает в себя ввод регистрационных данных (ФИО, номер телефона, дата рождения, номер водительского удостоверения, тип коробки передач, дата получения прав, логин, пароль). Каждому зарегистрированному клиенту присваивается свой идентификационный номер.
2. Фильтрация каталога автомобилей: удобный инструмент, позволяющий найти нужный автомобиль по выбранным параметрам (марка, тип автомобиля, тип коробки передач, класс автомобиля).
3. Прокат автомобиля: клиенты могут отправить заявку на прокат автомобиля. Для этого нужно выбрать в списке желаемый автомобиль (тип коробки передач и стаж вождения должны соответствовать требуемому для данного автомобиля), срок аренды и отправить заявку. Таблица с классами автомобилей и требуемым стажем для них расположены рядом со списком автомобилей (Эконом – любой стаж, комфорт – от двух лет, бизнес – от пяти лет).
4. Редактирование списка автомобилей: администратор и сотрудник могут добавлять, изменять или удалять данные по автомобилям в списке.
5. Смена статуса заявки на прокат: сотрудник может менять статус заявки (Ожидает обработки, заявка принята, заявка отклонена, завершено).
6. Работа с таблицами: администратор может редактировать некоторые таблицы.
7. Внешний вид: Фон формы голубого цвета, все таблицы и кнопки белого цвета. Шрифт Verdana размером 12 пт, края кнопок скруглённые. В качестве логотипа использовать логотип автопарка.

### **Техническое задание**

1. **Общие сведения**
   1. **Наименование проекта**

Разработка программного модуля для Поликлиники.

* 1. **Заказчик**

Поликлиника №34

* 1. **Исполнитель:**

ИП «Смирнов»

1. **Функциональные требования**
   1. **Регистрация клиента**

Возможность добавления пациентов в базу данных с указанием следующих параметров

* Номер;
* ФИО;
* Дата рождения;
* Телефон;
* Пол;
* Адрес;.
  1. **Возможность добавления врачей в базу данных с указанием следующих параметров:**
* Номер;
* ФИО;
* Дата рождения;
* Телефон;
* Специализация;
* Дата приема;
* Дата увольнения **;**
  1. **Возможность добавления истории болезни в базу данных с указанием следующих параметров:**
* Номер;
* Номер пациента;
* Номер врача;
* Дата явки;
* Диагноз;
* Рекомендации;
  1. **Возможность создания расписания в базе данных с указанием следующих параметров:**
* Номер;
* Номер врача
* Дата
* Время начала приема
* Время конца приема
* Кабинет
  1. **Врачи могут редактировать историю болезни и записывать пациента на прием:**
* Администраторы могут записывать на прием и отменять запись, вносить данные о новых пациентах, создавать расписание врачей.
* Системный администратор может выдавать, изменять и удалять доступ к ИС

1. **Нефункциональные требования**
   1. **Кроссплатформенность**

* Поддержка работы на ОС Windows 10 и выше.
  1. **Безопасность**
* Логин и пароль для доступа к приложению;
* Доступ к данным должен быть ограничен в зависимости от роли.
  1. **Удобство использования**
* Уведомления и подсказки.
  1. **Производительность**
* Минимальное время отклика на запросы пользователя (до пяти секунд).

1. **Требования к реализации**

* Язык программирования: C# (Windows Forms .NET Framework 4.8).
* СУБД: Microsoft SQL Server 19.

1. **Требования к документации**

* Техническое задание на разработку программного модуля.
* Руководство системному программисту.

1. **Руководство по стилю**

* Шрифт Times New Roman, размер 12 пт, цвет шрифта чёрный.
* Цвет фона формы #7FB2FF.
* Текстовые поля должны быть белого цвета.
* Должен использоваться логотип в виде красного креста

Задание №2 «Разработка алгоритмов и диаграмм»

**Цель работы**: разработка и реализация системы проектирования и моделирования, включающей создание алгоритмов в виде блок-схем и UML-диаграмм, а также проектирование базы данных, соответствующей требованиям третьей нормальной формы (3НФ) с обеспечением ссылочной целостности. Работа направлена на развитие навыков анализа предметной области, проектирования архитектуры системы, разработки алгоритмов и баз данных, а также на практическое применение методологий моделирования и проектирования.

По составленному техническому заданию необходимо разработать алгоритмы в виде блок-схем и UML-диаграммы:

* диаграммы вариантов использования;
* диаграммы последовательности;
* диаграммы активности.

Все диаграммы и блок-схемы должны подробно отображать разные процессы.

Также необходимо разработать ER-диаграмму для БД + словарь данных. Обязательно 3 нормальная форма с обеспечением ссылочной целостности. При разработке диаграммы обратите внимание на согласованную осмысленную схему именования, создайте необходимые первичные и внешние ключи, определите ограничения внешних ключей, отражающие характер предметной области.

Создайте все необходимые сущности, определите отношения, создайте ограничения на связи между сущностями (при наличии всех связей), приведите базу данных к 3НФ (при наличии всех сущностей и связей).

Создайте базу данных, используя предпочтительную платформу, на сервере баз данных. Создайте таблицы основных сущностей, атрибуты, отношения и необходимые ограничения.

Выполните названия таблиц и полей в едином стиле. Заполните таблицы данными.

Выполнение задания

### **Диаграммы вариантов использования**



Рисунок – Диаграмма вариантов использования

### **Диаграммы последовательности**

**Сценарий:** Запись пациента



Рисунок 2 – Диаграмма последовательности

### **Диаграммы активности**

**Сценарий**: Добавление сотрудника



Рисунок – Диаграмма активности

### **ER-диаграмма базы данных**



Рисунок – ER-диаграмма базы данных

### **Словарь данных**

Таблица – Словарь данных.

|  |  |  |  |
| --- | --- | --- | --- |
| Врачи | | | |
| Ключ | Наименование | Тип данных | Обязательность |
| PK | Номер | INT | Да |
|  | ФИО | NVARCHAR(256) | Да |
|  | Дата\_рождения | DATE | Да |
|  | Телефон | NVARCHAR(12) | Да |
|  | Дата\_приема | DATE | Да |
|  | Дата\_увольнения | DATE | Нет |
|  |  |  |  |
| Пациениты | | | |
| Ключ | Наименование | Тип данных | Обязательность |
| PK | Номер | BIGINT | Да |
|  | ФИО | NVARCHAR(256) | Да |
|  | Дата\_рождения | DATE | Да |
|  | Телефон | NVARCHAR(12) | Да |
|  | Пол | BIT | Да |
|  | Адрес | NVARCHAR(256) | Да |

Продолжение таблицы 1.

|  |  |  |  |
| --- | --- | --- | --- |
| Истории\_болезней | | | |
| Ключ | Наименование | Тип данных | Обязательность |
| PK | Номер | INT | Да |
| FK | Номер\_врача | INT | Да |
| FK | Номер\_пациента | BIGINT | Да |
|  | Дата\_приема | DATE | Да |
| FK | Диагноз | BIGINT | Да |
|  | Рекомендации | NVARCHAR(256) | Да |
|  |  |  |  |
| Пароли | | | |
| Ключ | Наименование | Тип данных | Обязательность |
| PK | Номер | BIGINT | Да |
| FK | Номер\_врача | NVARCHAR(256) | Да |
|  | Логин | NVARCHAR(256) | Да |
|  | Пароль | NVARCHAR(256) | Да |
|  | Действующий | BIT | Да |
|  |  |  |  |
| Расписание | | | |
| Ключ | Наименование | Тип данных | Обязательность |
| PK | Номер | BIGINT | Да |
| FK | Номер\_врача | INT | Да |
|  | Дата | DATE | Да |
|  | Время\_начала | TIME(7) | Да |
|  | Время\_конца | TIME(7) | Да |
|  | Кабинет | NVARCHAR(20) | Да |
|  |  |  |  |
| Диагнозы | | | |
| Ключ | Наименование | Тип данных | Обязательность |
| PK | Номер | BIGINT | Да |
|  | Название | NVARCHAR(256) | Да |
|  | Описание | NVARCHAR(MAX) | Нет |
|  |  |  |  |
| Специализации | | | |
| Ключ | Наименование | Тип данных | Обязательность |
| PK | Номер | BIGINT | Да |
|  | Название | NVARCHAR(256) | Да |
|  |  |  |  |
| Врачи\_специализации | | | |
| Ключ | Наименование | Тип данных | Обязательность |
| FK | Номер | BIGINT | Да |
| FK | Название | NVARCHAR(256) | Да |

Окончание таблицы 1.

|  |  |  |  |
| --- | --- | --- | --- |
| Врачи\_специализации | | | |
| Ключ | Наименование | Тип данных | Обязательность |
| PK | Номер | BIGINT | Да |
| FK | Номер\_врача | BIGINT | Да |
| FK | Номер\_пациента | BIGINT | Да |
|  | Дата | DATE | Да |
|  | Время | TIME | Да |
|  | Примечание | NVARCHAR(256) | Нет |

### **Диаграмма базы данных, созданная в Microsoft SQL Server Management Studio 19**

Изображение выглядит как текст, диаграмма, План, Технический чертеж

Автоматически созданное описание

Рисунок – Диаграмма базы данных

### **Запросы SQL**

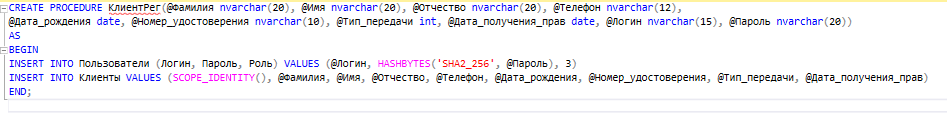


Рисунок – Запрос SQL для создания процедуры регистрации клиента

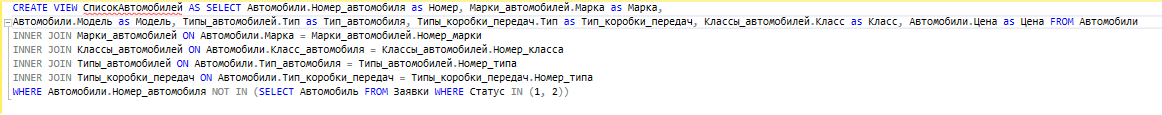


Рисунок – Запрос SQL для создания представления со списком автомобилей

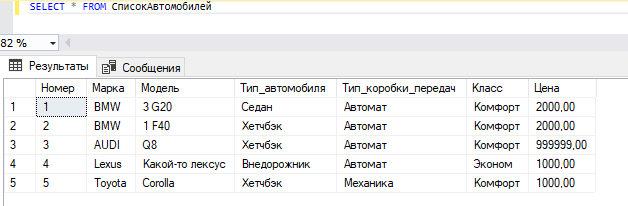


Рисунок – Запрос SQL для вывода результата из представления

### **Алгоритмы программы**



Рисунок – Алгоритм авторизации



Рисунок 9 – Алгоритм авторизации

Задание №3 «Разработка макетов»

**Цель работы:** изучение и применение различных видов макетов (Wireframe, Mockup, Prototype) для проектирования интерфейса программного продукта, соответствующего требованиям заказчика. Работа направлена на развитие навыков визуализации и прототипирования пользовательских интерфейсов, а также на создание удобной и интуитивно понятной карты навигации приложения. Дополнительно целью является освоение анимации в прототипах для повышения интерактивности и удобства использования будущего продукта.

Изучить информацию о разных видах макетов. Ссылка на ресурс: <https://blog.sibirix.ru/mockup-wireframe-prototype-sketch/>

Необходимо разработать макеты интерфейса будущего программного продукта, опираясь на требования заказчика. Вы должны представить макеты Wireframe (черно-белые) и Mockup (цветные). Работу выполнять в графическом редакторе, например Figma. За скрины форм с Visual Studio максимум 3 балла.

Разработать карту навигации приложения.

Дополнительное задание: разработать прототипы (макеты с анимацией).

Выполнение задания

### **Wireframe-макеты**



Рисунок – Wireframe-макет окна авторизации

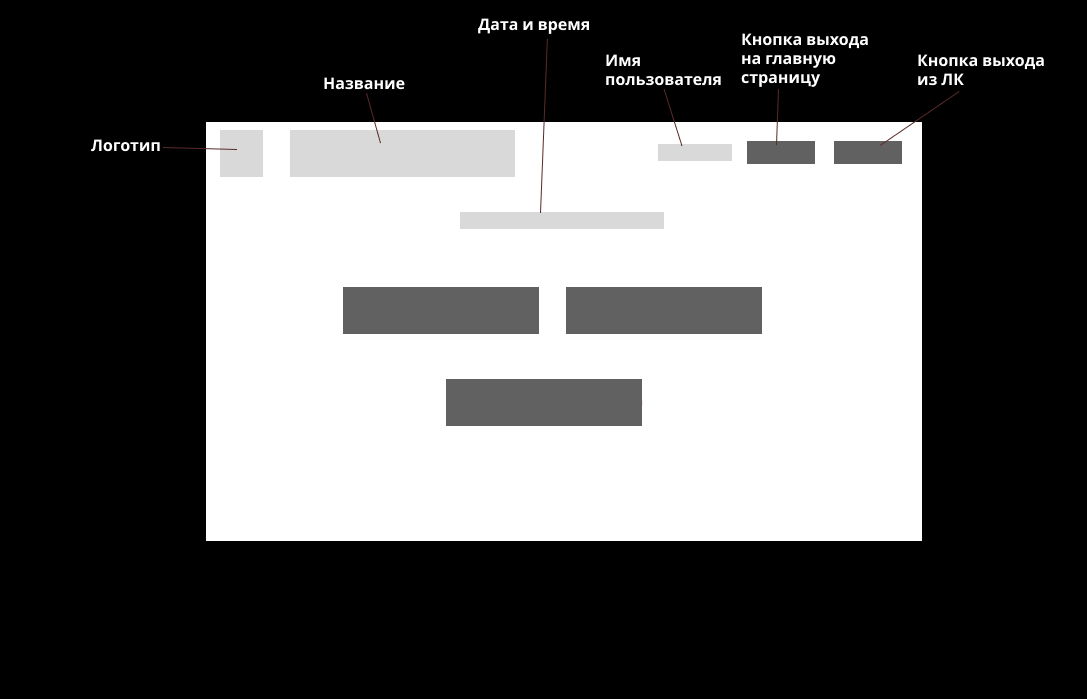


Рисунок – Wireframe-макет главного окна врача

Рисунок – Wireframe-макет окна просмотра списка пациентов

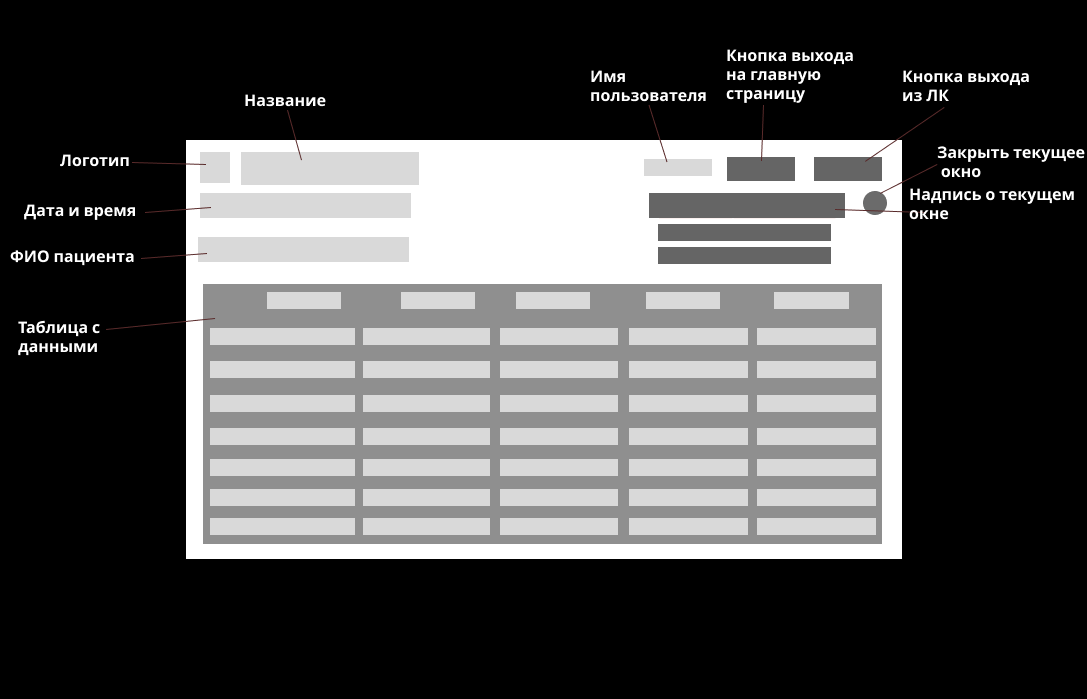


Рисунок – Wireframe-макет окна с историей болезни пациента

Рисунок – Wireframe-макет окна с записями врача на сегодня



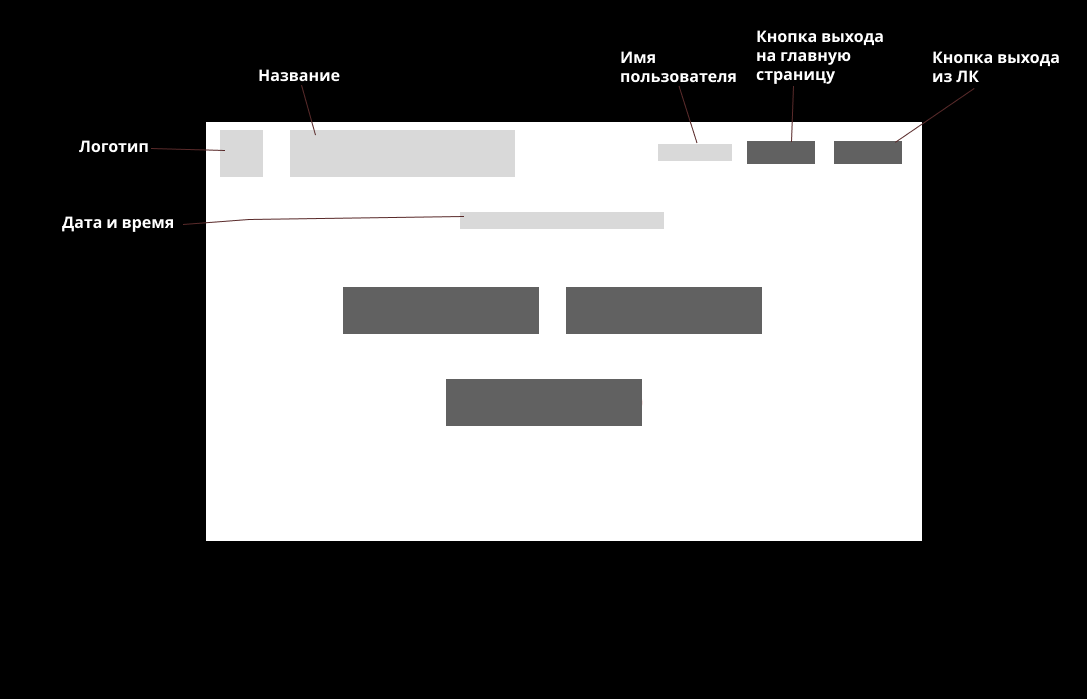
Рисунок – Wireframe-макет окна график работы

Рисунок – Wireframe-макет окна главного окна администратора

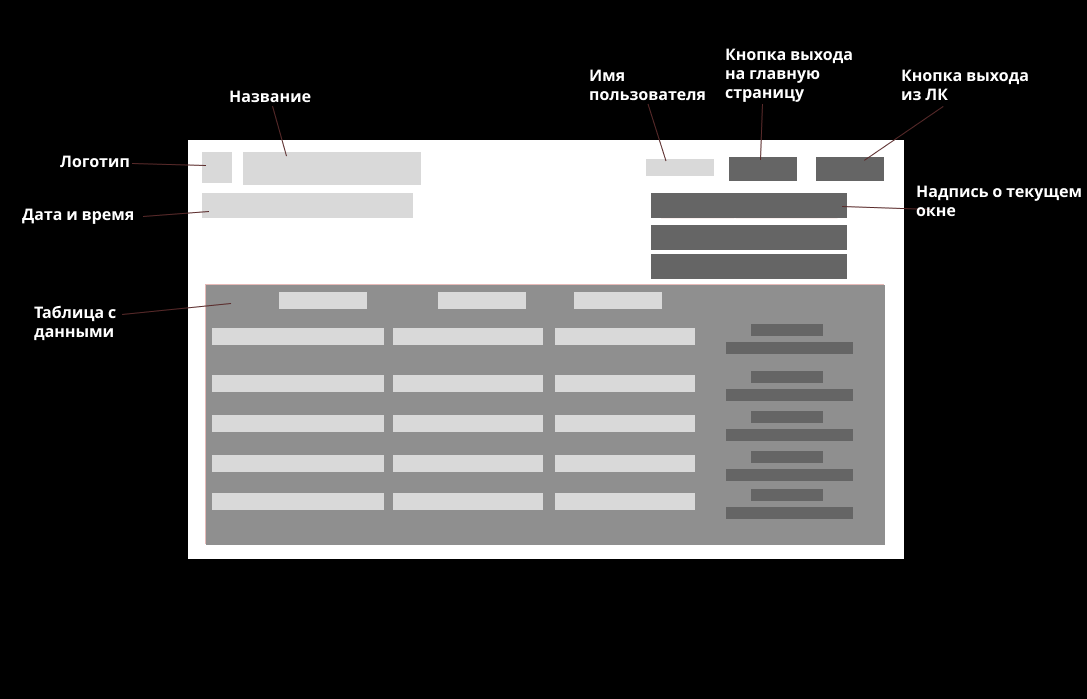


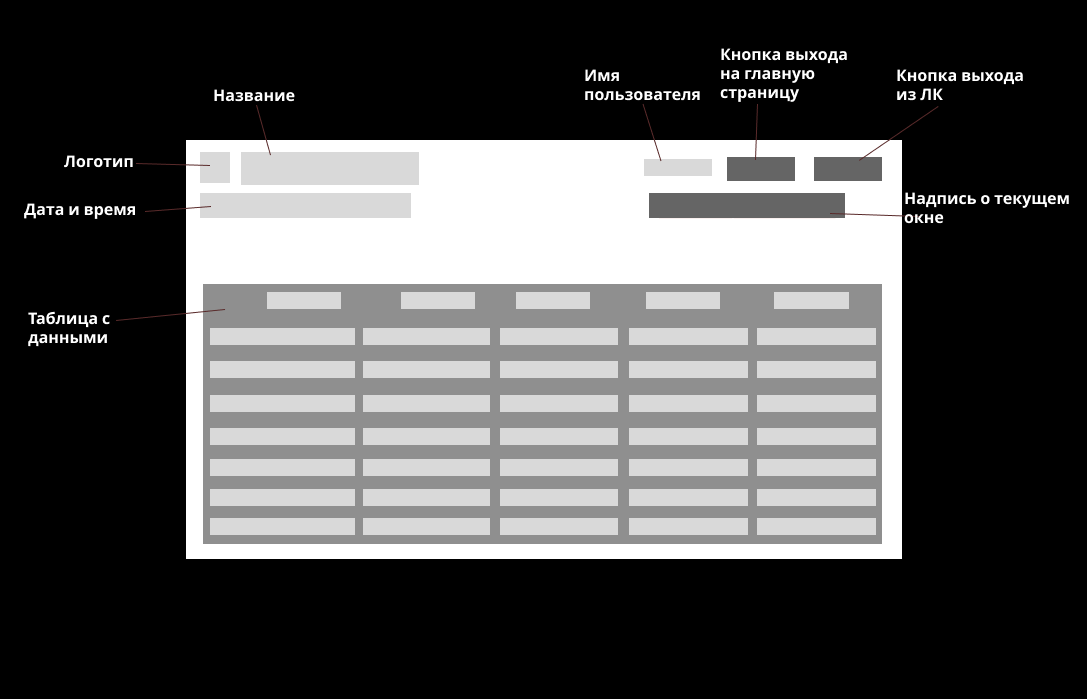
Рисунок – Wireframe-макет списка пациентов

Рисунок – Wireframe-макет окна с расписанием врачей

Изображение выглядит как снимок экрана, Прямоугольник, текст, дизайн

Автоматически созданное описание

Рисунок – Wireframe-макет окна администратора с историей болезни пациента

Изображение выглядит как снимок экрана, текст, Прямоугольник, дизайн

Автоматически созданное описание

Рисунок – Wireframe-макет окна администратора записи пациентов ко врачам

Изображение выглядит как снимок экрана, текст, диаграмма, дизайн

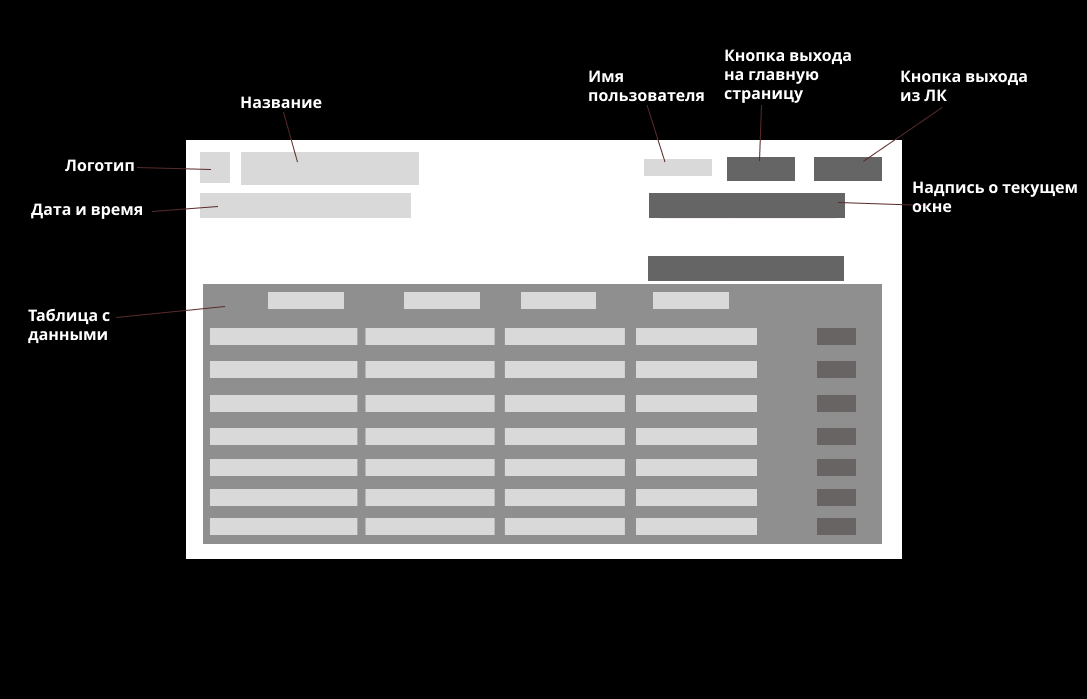
Автоматически созданное описание

Рисунок – Wireframe-макет главного окна системного администратора

Рисунок – Wireframe-макет окна с пользователями

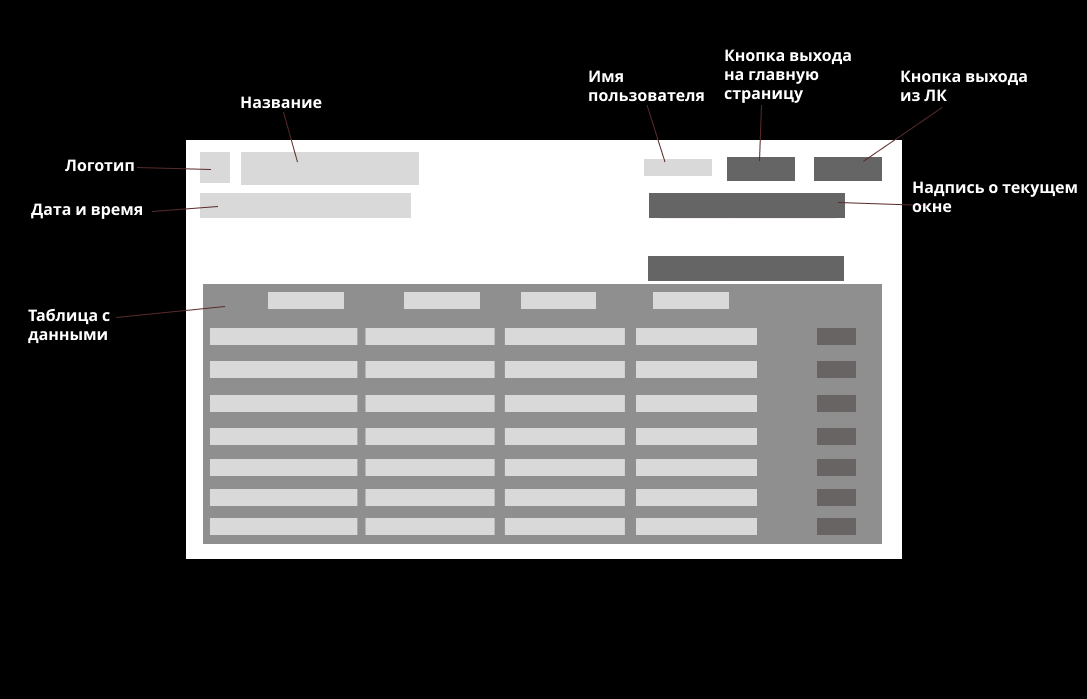


Рисунок – Wireframe-макет окна просмотра паролей

Изображение выглядит как снимок экрана, текст, диаграмма, дизайн

Автоматически созданное описание

Рисунок 24 – Wireframe-макет окна создания новой записи

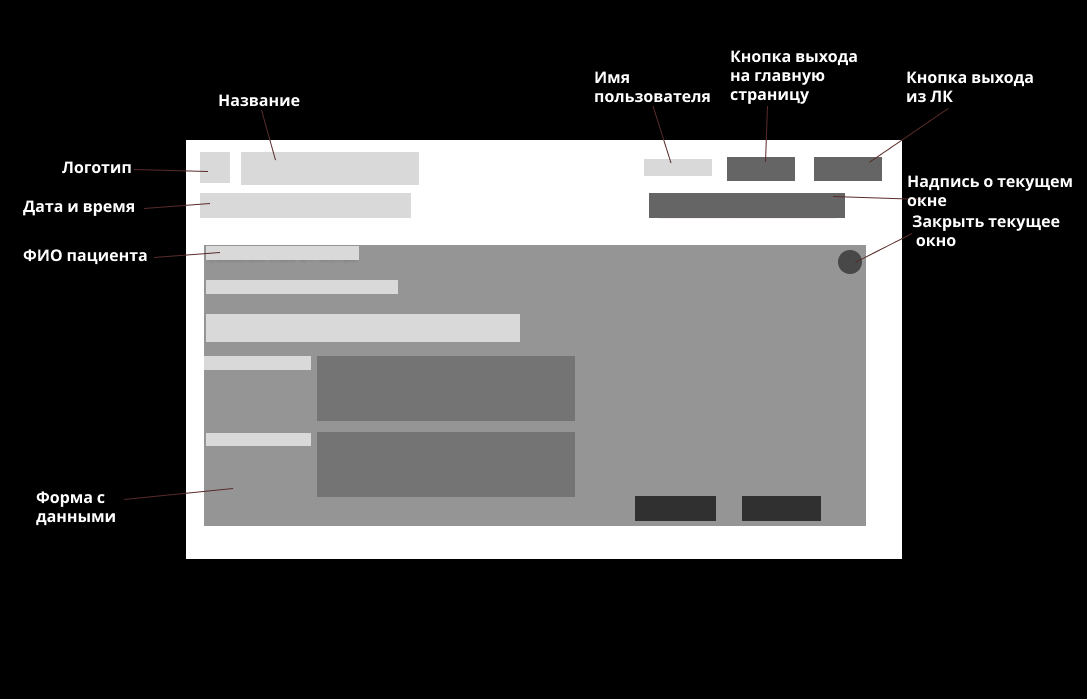


Рисунок 25 – Wireframe-макет окна создания записи с комбобоксами

### **Mockup-макеты**

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание

Рисунок 26 – Mockup-макет окна авторизации

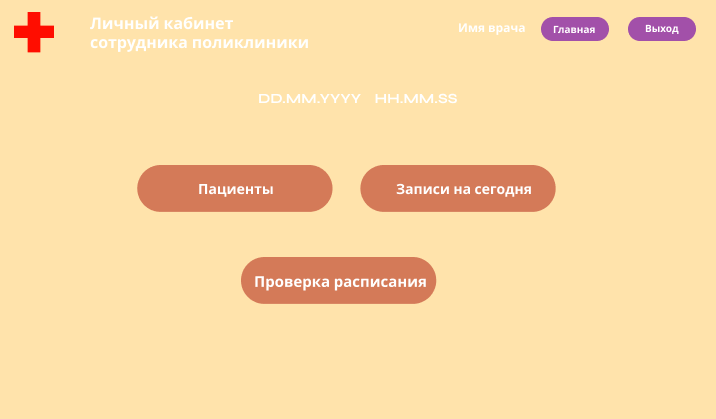


Рисунок 27 – Mockup-макет главного окна врача

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 28 – Mockup-макет окна просмотра списка пациентов

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

Автоматически созданное описание

Рисунок 29 – Mockup-макет окна с историей болезни пациента

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 30 – Mockup-макет окна с записями врача на сегодня

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 31 – Mockup-макет окна график работы

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание

Рисунок 32 – Mockup-макет окна главного окна администратора

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 33 – Mockup-макет списка пациентов

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рисунок 34 – Mockup-макет окна с расписанием врачей

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание

Рисунок 35 – Mockup-макет окна администратора с историей болезни пациента

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание

Рисунок 36 – Mockup-макет окна администратора записи пациентов ко врачам

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

Автоматически созданное описание

Рисунок 37 – Mockup-макет главного окна системного администратора



Рисунок 38 – Mockup-макет окна новой записи на прием

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

Автоматически созданное описание

Рисунок 39 – Mockup-макет окна новой записи в истории болезней

### **Карта навигации приложения**



Рисунок – Карта навигации

Задание №4 «Разработка программы»

**Цель работы:** разработка программного продукта, соответствующего составленным алгоритмам, диаграммам, макетам и техническому заданию. Работа направлена на создание последовательного и удобного пользовательского интерфейса, обеспечение единообразия внешнего вида компонентов системы, а также реализацию обработки исключительных ситуаций для повышения стабильности и удобства использования приложения. Целью также является соблюдение стандартов кодирования, написание чистого и понятного кода, а также проведение отладки и тестирования программного обеспечения для обеспечения его корректной работы.

Необходимо разработать программный продукт, опираясь на составленные алгоритмы, диаграммы, макеты и техническое задание.

  Все компоненты системы должны иметь единый согласованный внешний вид, соответствующий макетам, а также следующим требованиям:

- последовательный пользовательский интерфейс, позволяющий перемещаться между существующими окнами в приложении (в том числе обратно, например, с помощью кнопки «Назад»);

- соответствующий заголовок на каждом окне приложения.

Выполнить исходный код модуля в соответствии гайдлайну: идентификаторы должны соответствовать соглашению об именовании, например (CodeConvention), стилю CamelCase (для C# и Java), snake\_case (для Python) и <https://its.1c.ru/db/v8std#browse:13:-1:31> (для 1C).

Допустимо использование не более одной команды в строке.

Необходимо использовать комментарии для пояснения неочевидных фрагментов кода. Запрещено комментирование кода. Хороший код воспринимается как обычный текст. Не используйте комментарии для пояснения очевидных действий. Комментарии должны присутствовать только в местах, которые требуют дополнительного пояснения.

Реализовать программные обработки исключительных ситуаций в приложении. Уведомляйте пользователя о совершаемых им ошибках или о запрещенных в рамках задания действиях, запрашивайте подтверждение перед удалением, предупреждайте о неотвратимых операциях, информируйте об отсутствии результатов поиска и т.п. Окна сообщений соответствующих типов (например, ошибка, предупреждение, информация) должны отображаться с соответствующим заголовком и пиктограммой. Текст сообщения должен быть полезным и информативным, содержать полную информацию о совершенных ошибках пользователя и порядок действий для их исправления. Также можно использовать визуальные подсказки для пользователя при вводе данных.

Выполнить отладку модуля.

Выполнить отладку программного обеспечения с использованием инструментальных средств. Сохранить и представить результаты в скриншотах.

Выполнение задания

На основе Mockup-макетов (Рисунок 24–40) были реализованы следующие окна приложения:

### **Главное окно и авторизация**

Изображение выглядит как текст, снимок экрана, программное обеспечение, Операционная система

Автоматически созданное описание

Рисунок – окно авторизации

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок –главное окно врача

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание

Рисунок –окно с пациентами

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание

Рисунок – окно просмотра истории болезней

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

Автоматически созданное описание

Рисунок – окно создания новой записи к доктору

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

Автоматически созданное описание

Рисунок – окно создания новой записи в историю болезней

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание

Рисунок – окно записанных пациентов на сегодня

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок – окно графика работы

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

Автоматически созданное описание

Рисунок – окно списка пациентов для главного врача

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

Автоматически созданное описание

Рисунок – окно истории записей пациента для главного врача

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок – окно с графиком работ для главного врача

Изображение выглядит как текст, снимок экрана, программное обеспечение, Операционная система

Автоматически созданное описание

Рисунок – главное окно системного администратора

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок – окно со списком парролей

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

Автоматически созданное описание

Рисунок – окно изменения паролей

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

Автоматически созданное описание

Рисунок – окно создания паролей

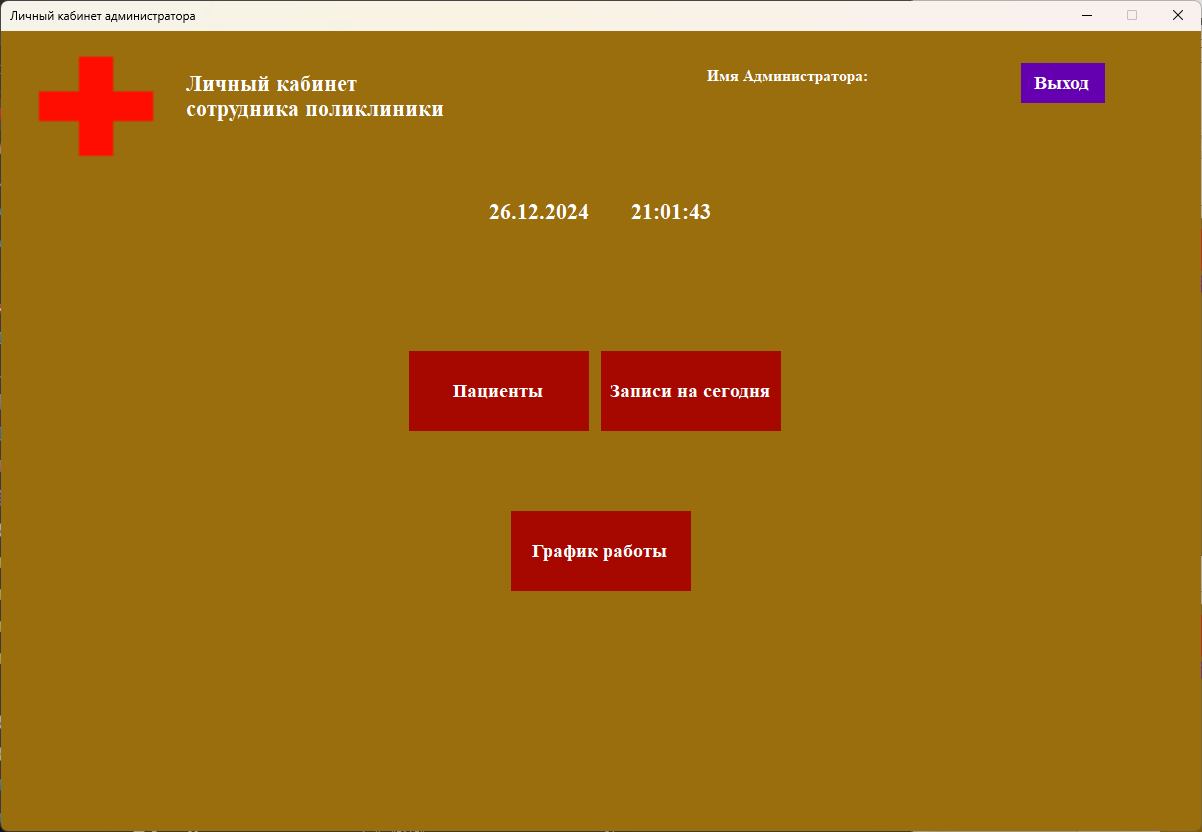


Рисунок – главное окно администратора

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

Автоматически созданное описание

Рисунок – окно списка пациентов для администратора

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

Автоматически созданное описание

Рисунок 58 – окно списка записей пациентов на сегодня для администратора

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 59 – окно графика работы для администратора

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

Автоматически созданное описание

Рисунок 60 – окно добавления записи для графика работы

Задание №5 «Тестирование»

**Цель работы:** разработка и документирование 5 Test-Case, которые проверяют наиболее важный процесс в системе. Эти тест-кейсы должны охватывать различные сценарии и ситуации, чтобы обеспечить полное тестирование функциональности. Выбранный процесс должен быть критически важным для функционирования системы, и тест-кейсы должны быть оформлены в соответствии с предоставленным шаблоном.

Подготовьте 5 Test-Case, проверяющие один наиболее важный на ваш взгляд процесс в системе. Важно, чтобы тестовые данные предусматривали различные ситуации. При оформлении используйте представленный ниже шаблон.

Аннотация теста:

|  |  |
| --- | --- |
| **Название проекта** |  |
| **Рабочая версия** |  |
| **Имя тестирующего** |  |
| **Дата(ы) теста** |  |

Расшифровка тестовых информационных полей:

|  |  |
| --- | --- |
| **Поле** | **Описание** |
| **Название проекта** | Название тестируемого проекта |
| **Рабочая версия** | Версия проекта/программного обеспечения (первый тест считается 1.0). |
| **Имя тестирующего** | Имя того, кто проводил тесты |
| **Дата(ы) теста** | Дата(ы) проведения тестов – это один или несколько дней. Если тесты проводились в более протяженный период времени, нужно отметить отдельную дату для каждого теста. |
| **Тестовый пример #** | Уникальный ID для каждого тестового примера. Следуйте некоторым конвенциям, чтобы указать типы тестов. Например,‘TC\_UI\_1′ означает‘user interface test case #1′ ( ТС\_ПИ\_1: тестовый случай пользовательского интерфейса#1) |
| **Приоритет тестирования**  *(Низкий/Средний/Высокий)* | Насколько важен каждый тест. Приоритет тестирования для бизнес-правил и функциональных тестовых случаев может быть средним или высоким, в то время как незначительные случаи пользовательского интерфейса могут иметь низкий приоритет. |
| **Заголовок/название теста** | Название тестового случая. Например, Подтвердите страницу авторизации с действительным именем пользователя и паролем. |
| **Краткое изложение теста** | Описание того, что должен достичь тест. |
| **Этапы теста** | Перечислите все этапы теста подробно. Запишите этапы теста в том порядке, в котором они должны быть реализованы. Предоставьте как можно больше подробностей и разъяснений. Пронумерованный список – хорошая идея. |
| **Тестовые данные** | Перечислите/опишите все тестовые данные, используемые для данного тестового случая. Так, фактические используемые входные данные можно отслеживать по результатам тестирования. Например, Имя пользователя и пароль для подтверждения входа. |
| **Ожидаемый результат** | Каким должен быть вывод системы после выполнения теста? Подробно опишите ожидаемый результат, включая все сообщения/ошибки, которые должны отображаться на экране. |
| **Фактический результат** | Каким должен быть фактический результат после выполнения теста? Опишите любое релевантное поведение системы после выполнения теста. |
| **Предварительное условие** | Любые предварительные условия, которые должны быть выполнены до выполнения теста. Перечислите все предварительные условия для выполнения этого тестового случая. |
| **Постусловие** | Каким должно быть состояние системы после выполнения теста? |
| **Статус**  *(Зачет/Незачет)* | Если фактический результат не соответствует ожидаемому результату, отметьте тест как неудачный. В ином случае обновление пройдено. |
| **Примечания/комментарии** | Используйте эту область для любых дополнительных заметок/комментариев/вопросов. Эта область предназначена для поддержки вышеуказанных полей (например, если есть некоторые особые условия, которые не могут быть описаны в любом из вышеуказанных полей, или если есть вопросы, связанные с ожидаемыми или фактическими результатами). |

Тестовый пример #1:

|  |  |
| --- | --- |
| **Тестовый пример #** |  |
| **Приоритет тестирования** |  |
| **Заголовок/название теста** |  |
| **Краткое изложение теста** |  |
| **Этапы теста** |  |
| **Тестовые данные** |  |
| **Ожидаемый результат** |  |
| **Фактический результат** |  |
| **Статус** |  |
| **Предварительное условие** |  |
| **Постусловие** |  |
| **Примечания/комментарии** |  |

Модульные тесты

В рамках проекта разработайте тесты, максимально полно покрывающие функционал метода. Ничего страшного, если ваш метод работает не совсем идеально и тесты могут быть не пройдены в связи с этим - в данном модуле это не так важно. Обратите внимание, что имена тестов должны отражать их суть, т.е. вместо TestMethod1() тест следует назвать, например, GetQuantityForProduct\_NonExistentProductType() для тестирования случая передачи несуществующего типа продукции.

Реализуйте 10 unit-тестов. Важно, чтобы тестовые данные предусматривали различные ситуации.

Выполнение задания

### **Tests-case**

|  |  |
| --- | --- |
| **Поле** | **Описание** |
| **Название проекта** | App |
| **Рабочая версия** | 1.0 |
| **Имя тестирующего** | Смирнов Андрей Александрович |
| **Дата(ы) теста** | 19.12.2024 |

|  |  |
| --- | --- |
| **Поле** | **Описание** |
| **Тестовый пример #** | AF\_TC\_1 |
| **Приоритет тестирования**  *(Низкий/Средний/Высокий)* | Высокий |
| **Заголовок/название теста** | Проверка формы авторизации с действительным логином и паролем. |
| **Краткое изложение теста** | Проверка формы авторизации на вывод сообщения о успешной авторизации при вводе действительного логина и пароля. |
| **Этапы теста** | 1. Ввести Логин в текстовое поле для логина (см. Рисунок 61). 2. Ввести Пароль в текстовое поле для пароля (см. Рисунок 62). 3. Нажать на кнопку «Войти». |
| **Тестовые данные** | Логин = «1»  Пароль = «1» |
| **Ожидаемый результат** | Отображается окно «Личный кабинет врача» |
| **Фактический результат** | Отображается окно «Личный кабинет врача» (см. Рисунок 63). |
| **Предварительное условие** | Открыть приложение, увидеть форму авторизации. |
| **Постусловие** | Открывается форма клиента. |
| **Статус**  *(Зачет/Незачет)* | Зачет |
| **Примечания/комментарии** |  |

Изображение выглядит как текст, снимок экрана, программное обеспечение, Операционная система

Автоматически созданное описание

Рисунок 61 – Ввод логина

Изображение выглядит как текст, снимок экрана, программное обеспечение, Операционная система

Автоматически созданное описание

Рисунок 62 – Ввод пароля

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

Автоматически созданное описание

Рисунок 63 – Открытия главного окна

|  |  |
| --- | --- |
| **Поле** | **Описание** |
| **Тестовый пример #** | AF\_NTC\_1 |
| **Приоритет тестирования**  *(Низкий/Средний/Высокий)* | Высокий |
| **Заголовок/название теста** | Проверка формы авторизации с недействительным логином и паролем. |
| **Краткое изложение теста** | Проверка формы авторизации на вывод сообщения о неверных данных при вводе действительного логина и пароля. |
| **Этапы теста** | 1. Ввести Логин в текстовое поле для логина (см. Рисунок 64). 2. Ввести Пароль в текстовое поле для пароля (см. Рисунок 65). 3. Нажать на кнопку «Войти». |
| **Тестовые данные** | Логин = «12345»  Пароль = «ыв1234» |
| **Ожидаемый результат** | Выводится окно с сообщением «Неверный логин или пароль, либо ваша учетная запись неактивна.». |
| **Фактический результат** | Выводится окно с сообщением «Неверный логин или пароль, либо ваша учетная запись неактива.» (см. Рисунок 66). |
| **Предварительное условие** | Открыть приложение, увидеть форму авторизации. |
| **Постусловие** | Пользователь остаётся на форме авторизации. |
| **Статус**  *(Зачет/Незачет)* | Зачет |
| **Примечания/комментарии** |  |

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 64 – Ввод логина

Изображение выглядит как текст, снимок экрана, программное обеспечение, Операционная система

Автоматически созданное описание

Рисунок 65 – Ввод пароля

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание

Рисунок 66 – Вывод сообщения о неверном логине или пароле

|  |  |
| --- | --- |
| **Поле** | **Описание** |
| **Тестовый пример #** | CCF\_TC\_1 |
| **Приоритет тестирования**  *(Низкий/Средний/Высокий)* | Средний |
| **Заголовок/название теста** | Проверка создания нового пациента |
| **Краткое изложение теста** | Проверка создания нового пациента при вводе всех значений отображение этого пациента в таблице пациентов |
| **Этапы теста** | 1. Ввести ФИО в текстовое поле (см. Рисунок 67). 2. Ввести дату рождения в текстовое поле. (см. Рисунок 68). 3. Ввести номер телефона в текстовое поле (см. Рисунок 69). 4. Ввести адрес в текстовое поле (см. Рисунок 70). 5. Нажать кнопку «Сохранить » (см. Рисунок 71). |
| **Тестовые данные** | ФИО – «Мочкина Карина Юрьевна»  Дата рождения «10.001.2001»  Номер телефона – «+79648532141»  Адрес – «г. Москва, ул. Ленина, д. 10» |
| **Ожидаемый результат** | Пациент присутствует в таблице пациенты |
| **Фактический результат** | Пациент присутствует в таблице пациенты (см. Рисунок 72). |
| **Предварительное условие** | Авторизироваться в качестве администратора, увидеть форму с созданием ового пациента |
| **Постусловие** | В таблице пациентов новый пацциент |
| **Статус**  *(Зачет/Незачет)* | Зачет |
| **Примечания/комментарии** |  |

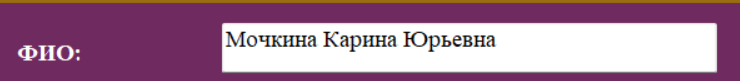


Рисунок 67 – ввод ФИО



Рисунок 68 – ввод даты рождения

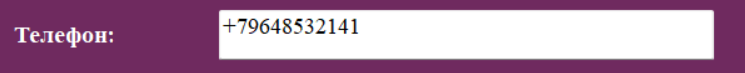


Рисунок 69 – ввод номера телефона

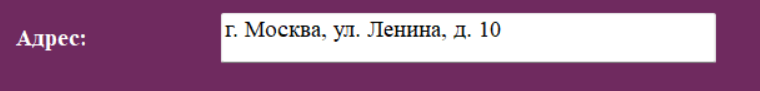


Рисунок 70 – ввод адреса

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание

Рисунок 71 – сохранения записи

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 72 – отображение созданного пациента

|  |  |
| --- | --- |
| **Поле** | **Описание** |
| **Тестовый пример #** | CR\_TC\_1 |
| **Приоритет тестирования**  *(Низкий/Средний/Высокий)* | Высокий |
| **Заголовок/название теста** | Проверка изменения пароля пользователя |
| **Краткое изложение теста** | Проверка изменения пароля пользователя через системного администратора |
| **Этапы теста** | 1. Найти в таблице врача номер 7». 2. Нажать на кнопку «Изменить». 3. Ввести новый пароль. 4. Нажать на кнопку «Сохранить» (Рисунок 73). |
| **Тестовые данные** | Пароль «bhdffHFjd8-» |
| **Ожидаемый результат** | Выводится окно с сообщением «Запись успешно добавлена |
| **Фактический результат** | Выводится окно с сообщением «Запись успешно добавлена.» (см. Рисунок 74). |
| **Предварительное условие** |  |
| **Постусловие** | Заявка на прокат автомобиля записывается в таблицу базы данных с заявками. |
| **Статус**  *(Зачет/Незачет)* | Зачет |
| **Примечания/комментарии** |  |

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 73 – отображение формы врача №7

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

Автоматически созданное описание

Рисунок 74 – изменение пароля

|  |  |
| --- | --- |
| **Поле** | **Описание** |
| **Тестовый пример #** | CR\_NTC\_1 |
| **Приоритет тестирования**  *(Низкий/Средний/Высокий)* | Высокий |
| **Заголовок/название теста** | Проверка запрета на вход деактивированной учетной записи |
| **Краткое изложение теста** | Проверка приложения на запрет входа пользователя, которому была деактивирована учетная запись |
| **Этапы теста** | 1. Деактивировать учетную запись врача №7 через системного администратора (Рисунок 75) 2. Выйти из учетной записи системного администратора 3. Ввести логин и пароль врача 4. Нажать на кнопку «Войти» (см. Рисунок 76). |
| **Тестовые данные** | Отсутствуют |
| **Ожидаемый результат** | Выводится окно с сообщением «Неверный логин или пароль, либо ваша учетная запись неактивна.». |
| **Фактический результат** | Выводится окно с сообщением «Неверный логин или пароль, либо ваша учетная запись неактивна.». (см. Рисунок 76). |
| **Предварительное условие** | Авторизироваться в качестве системного администратора, увидеть фрму с выбранным врачем №7 |
| **Постусловие** | Вход запрещен |
| **Статус**  *(Зачет/Незачет)* | Зачет |
| **Примечания/комментарии** |  |

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

Автоматически созданное описание

Рисунок 75 – деактивация учетной записи

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание

Рисунок 76 – отображение запрета доступа

### **Unit-tests**

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 77 – юнит тесты

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 78 – юнит тесты

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 79 – юнит тесты

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 80 – юнит тесты

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 81 – юнит тесты

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 82 – юнит тесты

Задание №6 «GitHub.Оценка проекта»

**Цель работы:** выполнение задания по выгрузке готового клиентского приложения, приложения-тестировщика (unit-test) и отчета в репозиторий GIT, а также проведение оценки проекта по заданным критериям. В рамках этой работы также необходимо предоставить заказчику оценку готовой программы, сравнив ожидания с реальностью. Это позволит продемонстрировать навыки работы с системой контроля версий GitHub, умение создавать и оформлять проекты, а также проводить анализ и оценку качества разработанного программного обеспечения

В репозиторий GIT выгрузите получившееся клиентское приложение,

приложение-тестировщик unit-test и отчет. В названии отчета укажите ваши номер группы и ФИО.

Провести оценку проекта по таблице (файл «Критерии профиль»). Заказчику также нужно дать свою оценку готовой программе, на сколько ожидания совпали с реальностью.

Выполнение задания

### **GitHub**

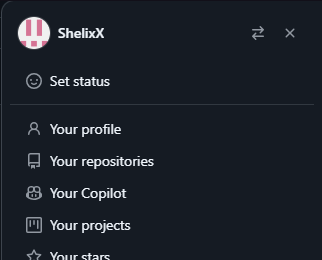


Рисунок – Заходим на аккаунт GitHub

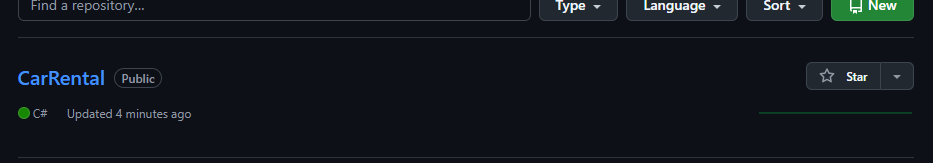


Рисунок – Создаем репозиторий

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок – Инициализируем Git в локальной папке



Рисунок – Добавим все файлы из папки в индекс Git и делаем первый коммит



Рисунок – Подключаем локальный репозиторий к GitHub

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок – Отправляем все в GitHub

### **Ссылка на проект**

https://github.com/amdreiov/clinic

**Оценка проекта по критериям**

**А1Д1: Анализ технического задания, составление краткой спецификации разрабатываемого модуля (входные и выходные данные), формирование алгоритмов в соответствии с техническим заданием**

* **Балл: 3**
  + Техническое задание проанализировано.
  + Составлена краткая спецификация, описаны входные и выходные данные.
  + Алгоритмы разработаны и соответствуют техническому заданию.

**А2Д1: Оформление алгоритмов представлено одним из трех способов и выполнено в соответствии со стандартами (ГОСТ 19.701-90 или ГОСТ 2.105 или ГОСТ 24.301)**

* **Балл: 3**
  + Алгоритмы оформлены в соответствии со стандартами.

**А3Д1: Реализация последовательности алгоритма по этапам (входные данные равны выходным данным), разработка алгоритма с использованием данных, определенных в техническом задании**

* **Балл: 6**
  + Рассмотрены все варианты работы алгоритма.
  + Реализована последовательность алгоритма по этапам.
  + Алгоритм использует все необходимые данные.

**Б1Д1: Программный модуль разрабатывается по имеющемуся алгоритму в среде разработки и полностью соответствует техническому заданию**

* **Балл: 6**
  + Программный модуль разработан по имеющемуся алгоритму.
  + Модуль полностью соответствует техническому заданию.

**Б2Д1: Выполнение исходного кода модуля в соответствии гайдлайну (название приложения, четкость, смысл именования форм и переменных очевидны)**

* **Балл: 2**
  + Исходный код модуля не в полной мере соответствует гайдлайну (название приложения, четкость, смысл именования форм и переменных не везде очевидны)

**В1Д1: Использование отладки и программной обработки исключительных ситуаций, что позволяет исключить фатальные ошибки при работе приложения, а также приложение аварийно не завершает работу**

* **Балл: 4**
  + Выполнена отладка модуля и программная обработка исключительных ситуаций.
  + Фатальные ошибки не возникают, приложение аварийно не завершает работу.

**В2Д1: Выполнение отладки программного обеспечения с использованием инструментальных средств, с пояснением особенностей отладочных классов, сохранение и представление результатов в виде текстового документа**

* **Балл: 3**
  + Выполнена отладка программного обеспечения с использованием инструментальных средств, частично даны пояснения особенностям отладочных классов, результаты представлены в виде текстового документа

**Г1Д1: Оформление результатов тестирования в виде протокола тестирования в соответствии со стандартами**

* **Балл: 3**
  + Результаты тестирования оформлены в виде протокола тестирования, соответствуют стандартам.

**Г2Д1: Определение наборов входных данных, выполнение функционального тестирования модуля по определенному сценарию, проведение тестирования для проверки функциональности программы (хотя бы 1 тест на 1 функцию), представление результатов тестирования**

* **Балл: 1,5**
  + Созданы тесты для проверки функциональности программы не в полном объеме.
  + Наборы входных данных определены верно.
  + Результаты тестирования представлены.

**Г3Д1: Использование инструментальных средств для тестирования функциональности программы**

* **Балл: 3**
  + Выполнено тестирование программного продукта с помощью инструментальных средств.

**Д1Д1: Проведение анализа поставленной задачи и проектирования базы данных (ERD модели) с применением case-средств, создание всех необходимых сущностей, определение отношений, создание ограничений на связи между сущностями (при наличии всех связей), приведение базы данных к 3НФ (при наличии всех сущностей и связей)**

* **Балл: 6**
  + Спроектирована и нормализована база данных в соответствии с поставленной задачей.
  + Уровень нормализации соответствует 3НФ.
  + Таблицы проиндексированы, структура индексов обоснована.

**Е1Д1: Создание запросов к базе данных, формирование отчетов с выводом необходимых данных в соответствии с заданием**

* **Балл: 6**
  + Созданы и корректно работают запросы к базе данных.
  + Сформированные отчеты выводят данные с учетом группировки в полном соответствии с заданием.

**Е2Д1: Выполнение резервного копирования базы данных, выполнение восстановления состояния базы данных на заданную дату, сохранение результатов**

* **Балл: 0**
  + Не выполнено резервное копирование БД; не выполнено восстановление состояния

**Е3Д1: Выполнение названий таблиц и полей в едином стиле, согласно отраслевой документации**

* **Балл: 3**
  + Созданные объекты полностью соответствуют заданию, согласно отраслевой документации.

**Е4Д1: Заполнение базы данных с помощью предложенных средств**

* **Балл: 2**
  + Все таблицы заполнены, данные загружены верно и в правильном формате.

**Ж1Д1: Выбор принципа регистрации и системы паролей, создание групп пользователей**

* **Балл: 1**
  + Выбрана система паролей.
  + Созданы и обоснованы группы пользователей.

**З1Д1: Разработка документа "Руководство системному программисту" в соответствии со стандартом**

* **Балл: 0**
  + Документ "Руководство системному программисту" не разработан.

**З2Д1: Добавление нового пользователя в систему, добавление новой роли, добавление функционала согласно должностным инструкциям пользователя, в соответствии с потребностями заказчика**

* **Балл: 2,5**
  + Добавление нового пользователя в систему.
  + Добавление новой роли отсутствует.
  + Добавление функционала согласно должностным инструкциям пользователя.

**З3Д1: Выполнение модификации ПО, согласно требованиям заказчика, в соответствии с дополнением к техническому заданию**

* **Балл: 4**
  + Дополнительный функционал добавлен полностью.

**З4Д1: Предложение вариантов модификации программного обеспечения, предложения представлены в формате текстового документа**

* **Балл: 3**
  + Предложены варианты модификации программного обеспечения, представлены в формате текстового документа.

**И1Д1: Установка необходимых компонент, в рамках требований заказчика на модификацию программного обеспечения, в соответствии с техническим заданием**

* **Балл: 2**
  + Установка компонент выполнена в полном объеме.

**И2Д1: Настройка компонент программного обеспечения**

* **Балл: 2**
  + Выполнена настройка компонент программного обеспечения.

**К1Д1: Определение качественных характеристик кода: полнота обработки ошибочных данных, наличие тестов для проверки допустимых значений входных данных, наличие средств контроля корректности входных данных, наличие средств восстановления при сбоях оборудования, наличие комментариев в точках входа и выхода в программу, наличие проверки корректности передаваемых данных, наличие описаний основных функций**

* **Балл: 1**
  + Определены качественные характеристики программного кода.
  + Выявлены некоторые фрагменты некачественного кода.

**Итоговый балл: 67/80**

**Оценка заказчика проекта по техническому заданию (ТЗ)**

**1. Соответствие функциональным требованиям**

* **Оценка: 5/5**
  + Проект полностью реализует все функциональные требования, описанные в ТЗ.
  + Все основные функции, такие как регистрация пациентов, ведене истории болезней, разграничение по ролям организованны корректно.
  + Дополнительно реализованы сценарии, которые не были явно указаны в ТЗ, но являются логическим продолжением (например, изменения логина и/или пароля пользователей, изменение данных клиентов).

**2. Удобство интерфейса**

* **Оценка: 5/5**
  + Интерфейс соответствует требованиям ТЗ:
  + Пользовательский интерфейс интуитивно понятен и удобен для работы.
  + Все элементы управления (кнопки, поля ввода, выпадающие списки) расположены логично, что упрощает взаимодействие с системой.

**3. Надежность и стабильность**

* **Оценка: 5/5**
  + Система устойчива к ошибкам: обработаны исключительные ситуации, предотвращены аварийные завершения работы.
  + Аутентификация и авторизация пользователей выполнены корректно, что обеспечивает безопасность данных.

**4. Производительность**

* **Оценка: 4/5**
  + Система работает быстро и эффективно, однако в некоторых сценариях (например, при заполнении таблиц данными) наблюдается незначительная задержка.
  + Для оптимизации производительности рекомендуется доработка алгоритмов заполнения таблиц.

**5. Соответствие техническим требованиям**

* **Оценка: 5/5**
  + Используется язык программирования C#.
  + Интерфейс разработан с использованием Windows Form .NET Framework 4.8.
  + База данных реализована на Microsoft SQL Server 19.
  + Система работает на Windows 10, что соответствует требованиям заказчика.

**6. Гибкость и масштабируемость**

* **Оценка: 4/5**
  + Система легко адаптируется к изменениям, что подтверждается реализацией дополнительных функций, не описанных в ТЗ.
  + Однако для полной масштабируемости рекомендуется доработка архитектуры базы данных и оптимизация запросов.

**7. Качество кода**

* **Оценка: 5/5**
  + Код хорошо структурирован, соответствует стандартам и гайдлайнам.
  + Использование комментарии и описание функций происходят не во всех частях кода
  + Реализована обработка ошибок и исключений, что повышает надежность системы.

**8. Тестирование и отладка**

* **Оценка: 5/5**
  + Проведено комплексное тестирование, включая функциональное, нагрузочное и негативные сценарии.
  + Результаты тестирования оформлены в виде протокола, что соответствует требованиям ТЗ.
  + Система продемонстрировала высокую стабильность и корректность работы.

**9. Документация**

* **Оценка: 4/5**
  + Предоставлена неполная документация, отсутствует "Руководство системному программисту".
  + Предоставленная документация соответствует стандартам и помогает в понимании работы системы.

**10. Соответствие срокам и бюджету**

* **Оценка: 5/5**
  + Проект выполнен в установленные сроки и в рамках бюджета.
  + полностью оправдал ожидания. Заказчик удовлетворен результатом и считает, что проект

**Итоговая оценка: 47/50**

Проект выполнен на высоком уровне, полностью соответствует требованиям технического задания и ожиданиям заказчика. Рекомендуется внедрить систему в эксплуатацию и начать использование в повседневной работе Автопарка «АвтоРент».

ЗАКЛЮЧЕНИЕ

В ходе прохождения учебной практики по профессиональному модулю ПМ.02 «Осуществление интеграции программных модулей» были успешно выполнены все поставленные задачи, что позволило закрепить теоретические знания и приобрести практические навыки в разработке программных модулей, проектировании баз данных, тестировании и документировании программного обеспечения. Практика проводилась в ФГАОУ ВО «СПбПУ» Институт среднего профессионального образования с 9 по 21 декабря 2024 года.

В ходе практики были успешно освоены навыки проектирования, разработки и тестирования программных модулей, что подтверждается выполнением всех поставленных задач. Проект полностью соответствует требованиям технического задания и ожиданиям заказчика. Система демонстрирует высокую стабильность, удобство использования и соответствует современным стандартам разработки программного обеспечения.

Результаты практики подтверждают, что разработанная система готова к внедрению в эксплуатацию и может быть использована для автоматизации деятельности автопарка.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Девятко Н.С. Учебное пособие по МДК 01.02 Прикладное программирование – СПб: ФГАОУ ВО СПбПУ Институт среднего профессионального образования, 2019, 95 с.
2. Павловская Т.А. C#. Программирование на языке высокого уровня: Учебник для вузов. ⎯ СПб.: Питер, 2014. ⎯ 432 с.: ил ⎯ (Серия «Учебник для вузов»).
3. Учебник. Сведения об отладке кода C# с помощью Visual Studio. ⎯ URL: <https://docs.microsoft.com/ru-ru/visualstudio/get-started/csharp/tutorial-debugger?view=vs-2022>.
4. Албахари, Дж. C# 9.0. Карманный справочник. – СПб.: Питер, 2021. – 256 с.
5. Фаронов В.В. SQL. Программирование баз данных для Microsoft SQL Server 2019. – СПб.: Питер, 2020. – 448 с.
6. ГОСТ 19.701-90. ЕСПД. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения.
7. ГОСТ 24.301-80. Информационная технология. Комплекс стандартов на автоматизированные системы. Общие положения.
8. Троелсен Э. C# 9.0 и платформа .NET 5.0. – М.: Вильямс, 2021. – 1408 с.
9. Оглышев А.А. Основы проектирования информационных систем. – М.: Интернет-Университет Информационных Технологий, 2018. – 320 с.

**АТТЕСТАЦИОННЫЙ ЛИСТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ УП.02.01**

по профессиональному модулю

ПМ.02 «Осуществление интеграции программных модулей»

(код и наименование)

Специальность 09.02.07 «Информационные системы и программирование»

(код и наименование специальности)

Студент(ка) 4 курса 42919/4 группы

Смирнов Андрей Александрович

(фамилия, имя, отчество)

Место прохождения практики: ФГАОУ ВО «СПбПУ» Институт среднего профессионального образования, пр. Энгельса д. 23

(наименование и адрес организации)

Период прохождения практики

с «9» декабря 2024 г. по «21» декабря 2024 г.

**Виды и качество выполнения работ**

|  |  |  |
| --- | --- | --- |
| **Виды выполненных работ обучающимся**  **во время практики** | **Объем работ, час.** | **Качество выполнения работ (оценка по пятибальной системе)** |
| **Тема 1.** Участие в выработке требований к программному обеспечению | 9 |  |
| **Тема 2.** Стадии проектирования программного обеспечения | 21 |  |
| **Тема 3.** Разработка модулей программного обеспечения | 24 |  |
| **Тема 4.** Тестирование программных модулей и их интеграции | 9 |  |
| **Тема 5.** Разработка программной документации. Стандарты кодирования | 9 |  |

**Характеристика учебной/профессиональной деятельности обучающегося во время производственной практики (по профилю специальности):**

Общие и профессиональные компетенции, предусмотренные программой практики, освоены/не освоены.

(нужное подчеркнуть)

Итоговая оценка по практике \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель практики Прокофьев А.А. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Ф.И.О.) (подпись)

Дата «21» декабря 2024