**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

Румянцева Анастасия Владимировна

(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

Румянцева Анастасия Владимировна

(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 27**

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

(подпись)  (Ф.И.О.)

Задание получил «9» декабря 2024 г. \_\_\_\_\_\_\_\_\_\_\_\_ Румянцева А.В.(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

Румянцева Анастасия Владимировна

(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 заказов. |  |
| 17.12.24 | Создание приложения. Основные формы приложения. |  |
| 17.12.24 | Разработка библиотеки классов. |  |
| 18.12.24 | Подготовка отчетов и выгрузка документов для печати. |  |
| 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. |  |

Оглавление

ВВЕДЕНИЕ

Учебная практика проводилась в ФГАОУ ВО СПбПУ, Институт среднего профессионального образования, в период с 09.12.2024 по 21.12.2024. Этот этап обучения играет ключевую роль в формировании высококвалифицированных специалистов, поскольку он обеспечивает возможность применения теоретических знаний в практической деятельности и знакомит студентов с реальными профессиональными условиями.

Учебная практика направлена на развитие профессиональных навыков и компетенций, необходимых для успешной работы в области информационных технологий. В рамках практики студенты изучают современные методы и инструменты разработки программного обеспечения, что способствует их профессиональному становлению и подготовке к требованиям рынка труда.

Основной целью данной практики было закрепление теоретических знаний и приобретение практических умений в области разработки и тестирования программных модулей. Для достижения этой цели выполняются задачи, связанные с созданием модулей для управления, включая разработку функциональных требований, проектирование пользовательского интерфейса, программирование компонентов системы, их тестирование и документирование.

Практика позволила студентам решать задачи, приближенные к реальным условиям, что способствовало развитию их профессиональных качеств. Результаты практики подтвердили высокий уровень подготовки студентов и их готовность к решению задач в сфере разработки программных систем.

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 «Гостиница»

### **Описание предметной области**

Основной целью данного модуля является автоматизация процессов управления и учета клиентов в гостинице. Модуль позволяет эффективно управлять бронированием номеров, регистрацией постояльцев, предоставлением дополнительных услуг и расчетами.

Разработка программного модуля для управления гостиницей включает создание программного продукта, который позволяет отелям и гостиницам систематизировать учет клиентов, оптимизировать занятость номеров и повышать уровень обслуживания.

Задача разработки состоит в предоставлении удобного инструмента для управления бронированием, регистрации и обслуживания клиентов, что помогает сократить время обработки заявок и повысить удовлетворенность гостей.

Основные функции и возможности модуля включают:

1. Бронирование номеров:
   * Бронирование номеров с указанием дат заезда и выезда, категории номера, и дополнительных услуг.
2. Управление услугами:
   * Учет дополнительных услуг (питание, SPA).
3. Безопасность и удобство:
   * Аутентификация пользователей с разграничением прав доступа.

Техническое задание

1. Общие сведения

1.1. Наименование проекта: Разработка программного модуля для учета и управления бронированием в гостинице.  
1.2. Заказчик: Гостиница «Mr.BeastiX».  
1.3. Исполнитель: Компания «Дурови feat. Nkeeei».

2. Функциональные требования

2.1. Управление бронированием:

* Добавление бронирований в базу данных с указанием следующих параметров:
  + Номер бронирования;
  + Дата и время создания;
  + Дата заезда и выезда;
  + Тип номера (стандартный, двухэтажный, люкс);
  + ФИО клиента;
  + Контактный телефон;
  + Статус бронирования (забронировано, подтверждено).

2.2. Редактирование бронирований:

* Изменение даты заезда или выезда;
* Изменение типа номера;

2.3. Управление номерами:

* Добавление и редактирование номеров (тип номера, стоимость, статус);

3. Нефункциональные требования

3.1. Кроссплатформенность:

* Поддержка работы на ОС семейства Windows.

3.2. Безопасность:

* Аутентификация пользователей (логин и пароль);
* Разграничение прав доступа в зависимости от роли (администратор, клиент, горничная).
* Доступ производится через отдельную форму, путем ввода логина и пароля(зашифрован), который хранится в БД.

3.3. Удобство использования:

* Возможность быстрого бронирования и управления номерами;

3.4. Производительность:

* Минимальное потребление ресурсов устройства

4. Требования к реализации

4.1. Язык программирования: C# **10.0 и выше** (совместимость с .NET 6 и выше).

4.2. Фреймворк: .NET 8.0 Windows Forms.

4.3. СУБД: MS SQL Server 2019 и выше.

5. Требования к документации

5.1. Техническое задание на разработку программного модуля.  
5.2. Руководство пользователя:

* Введение, где указывают область применения ПО, краткое описывают его возможности, требуемый уровень знаний пользователя и список документов, которые необходимо изучить помимо настоящего руководства;
* Назначение и условия применения, где описывают виды деятельности и функции, которые автоматизированы и условия, при соблюдении которых автоматизация используется;
* Подготовка к работе, где описывают комплектность дистрибутива, порядок установки и загрузки программы, а также способ проверки ее работоспособности;
* Описание операций, представляет собой основной раздел, где описывают функции программы, процессы работы с данными, выполнение конкретных задач пользователя;
* Аварийные ситуации, где описывают действия в нештатных ситуациях – сбоях в программе, ошибок в данных и т.д.;
* Рекомендации по освоению, где приводят методические рекомендации по изучению программы и примеры использования.

5.3. Руководство системного администратора:

* Общие сведения о программе
* Структура программы
* Настройка программы
* Проверка программы
* Дополнительные возможности
* Сообщения системному программисту
* Приложения

6. Руководство по стилю

6.1. Шрифт: Times New Roman, размер 14.  
6.2. Цветовая схема: использование фиолетовых и белых цветов.

6.3. Логотип: обязательное использование логотипа гостиницы на основных окнах (Приложение А).

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**

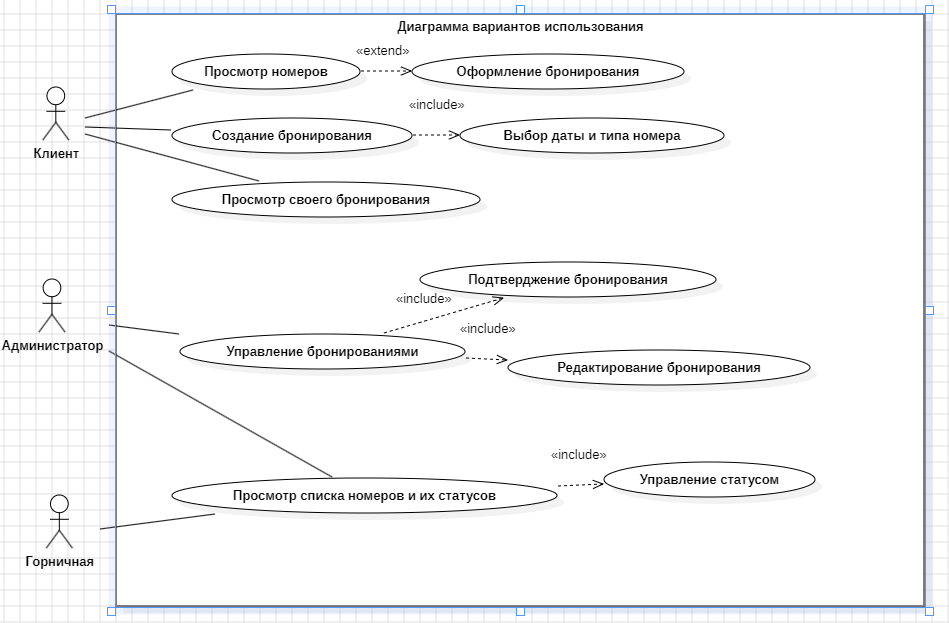


Рисунок 1 - Диаграмма вариантов использования

### **Диаграммы последовательности**

**Сценарий:** Оформление бронирования

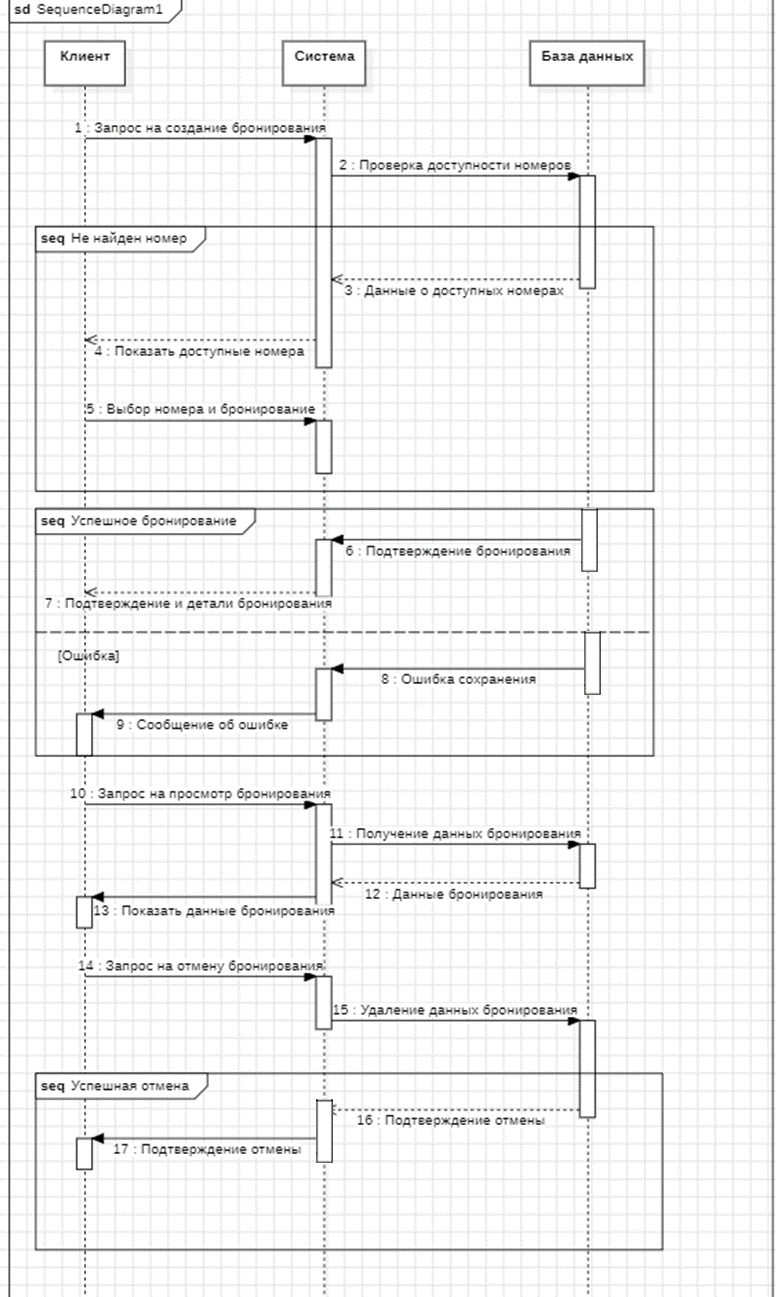


Рисунок 2 - Диаграмма последовательности

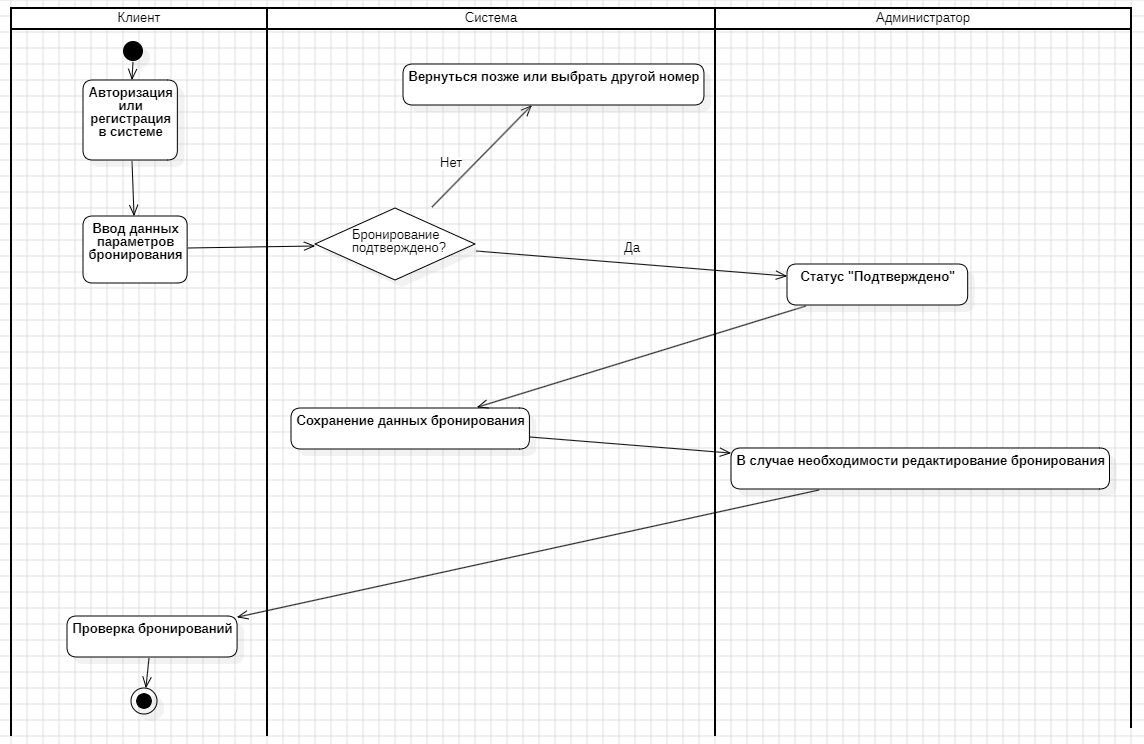


Рисунок 3 – Диаграмма активности

### **ER-диаграмма базы данных**

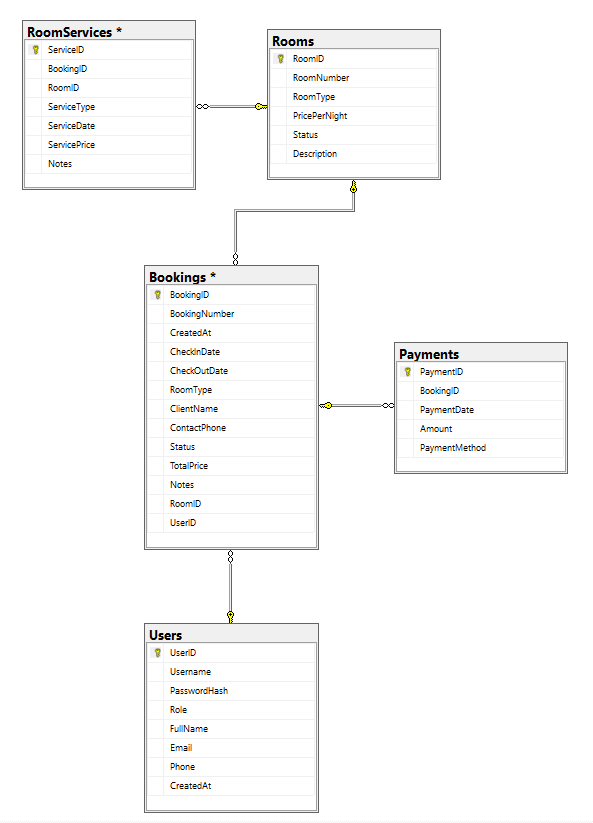


Рисунок 4 – ER-диаграмма базы данных

### **Словарь данных**

Таблица 1 «Users»

|  |  |  |
| --- | --- | --- |
| Имя столбца | Тип данных | Разрешить NULL |
| UserID | int | Нет |
| Username | Nvachar(50) | Нет |
| Password | Nvarchar(255) | Нет |
| Role | Nvarchar(20) | Нет |
| Name | Nvarchar(100) | Да |
| ContactPhone | Nvarchar(15) | Да |

Таблица 2 «Bookings»

|  |  |  |
| --- | --- | --- |
| Имя столбца | Тип данных | Разрешить NULL |
| BookingID | int | Нет |
| BookingNumber | Nvachar(20) | Нет |
| CheckInDate | date | Нет |
| CheckOutDate | date | Нет |
| RoomType | Nvarchar(50) | Нет |
| ClientName | Nvarchar(100) | Нет |
| ContactPhone | Nvarchar(15) | Нет |
| Status | Nvarchar(20) | Нет |
| RoomID | int | Да |

Таблица 3 «Rooms»

|  |  |  |
| --- | --- | --- |
| Имя столбца | Тип данных | Разрешить NULL |
| RoomID | Int | Нет |
| RoomNumber | Nvarchar(10) | Нет |
| RoomType | Nvarchar(50) | Нет |
| Status | Nvarchar(20) | Нет |
| PricePerNight | Decimail(10,2) | Нет |

Таблица 4 «RoomServices»

|  |  |  |
| --- | --- | --- |
| Имя столбца | Тип данных | Разрешить NULL |
| ServiceID | Int | Нет |
| ServiceName | Nvarchar(100) | Нет |
| Price | Decimail(10,2) | Нет |
| BookingID | int | Нет |

Таблица 5 «Payments»

|  |  |  |
| --- | --- | --- |
| Имя столбца | Тип данных | Разрешить NULL |
| PaymentID | Int | Нет |
| BookingID | int | Нет |
| Amount | Decimail(10,2) | Нет |
| PaymentDate | datetime | Нет |
| PaymentMethod | Nvarchar(50) | Нет |

Таблица 6 «Roles»

|  |  |  |
| --- | --- | --- |
| Имя столбца | Тип данных | Разрешить NULL |
| RoleID | int | Нет |
| Role | varchar(20) | Нет |



Рисунок 5 – Диаграмма метода создания бронирования (основной функционал)

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**

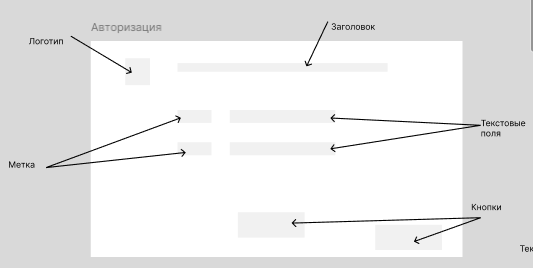


Рисунок 6 – Wareframe-макет окна авторизации

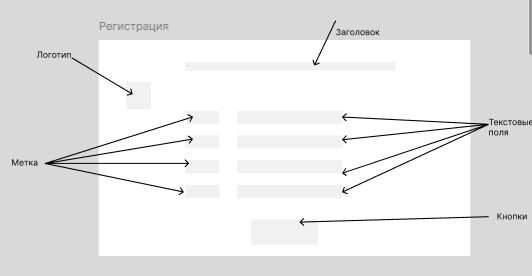


Рисунок 7 – Wareframe-макет окна регистрации

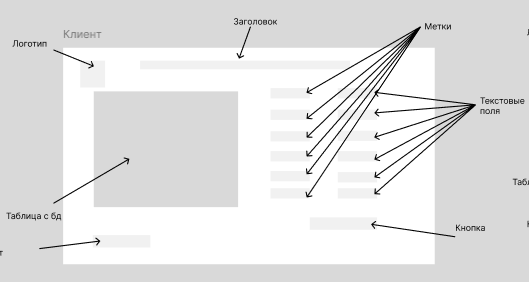
****

Рисунок 8 – Warframe-макет главного окна клиента

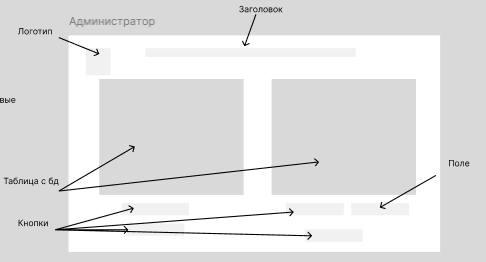


Рисунок 9 – Warframe-макет главного окна администратора

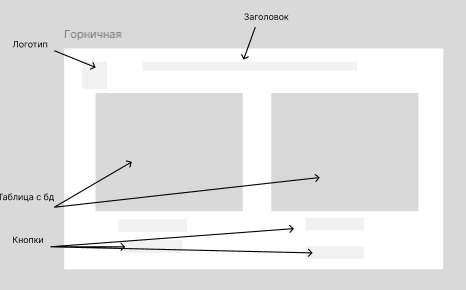
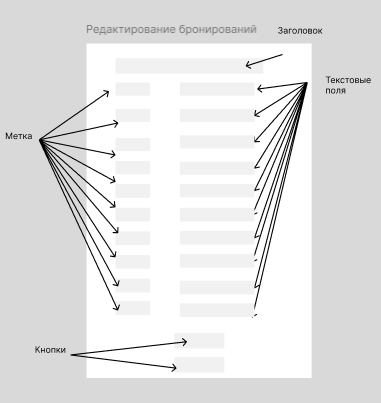


Рисунок 10 – Warframe-макет главного окна горничной

  
Рисунок 11 - Warframe-макет окна добавления, редактирования бронирований

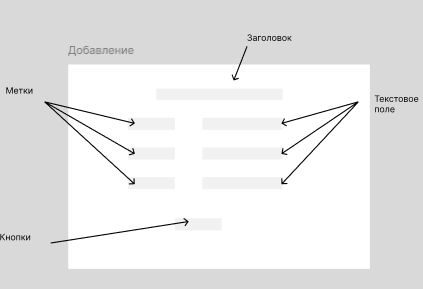


Рисунок 12 - Warframe-макет окна добавления услуги

### **Mockup-макеты**

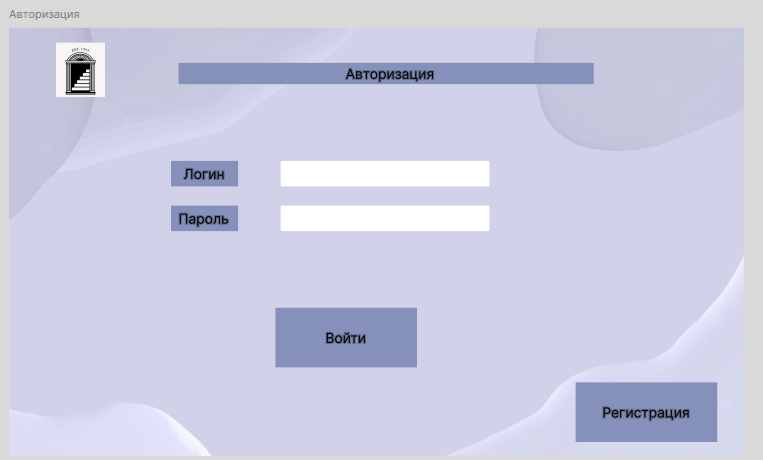


Рисунок 13 – Mockup-макет окна авторизации

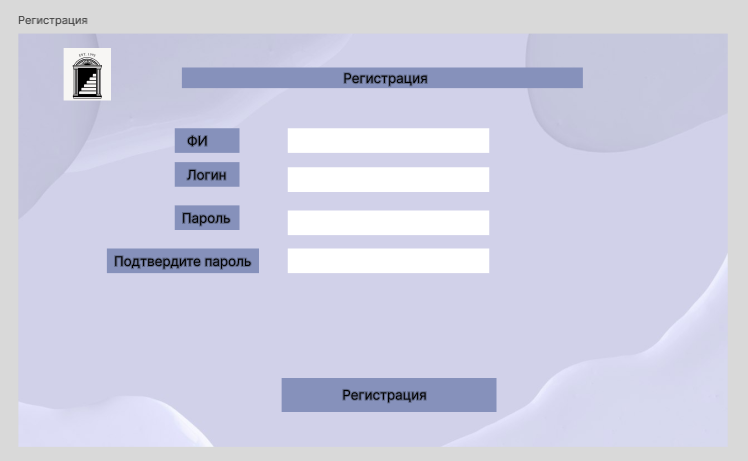


Рисунок 14 – Mockup-макет окна регистрации

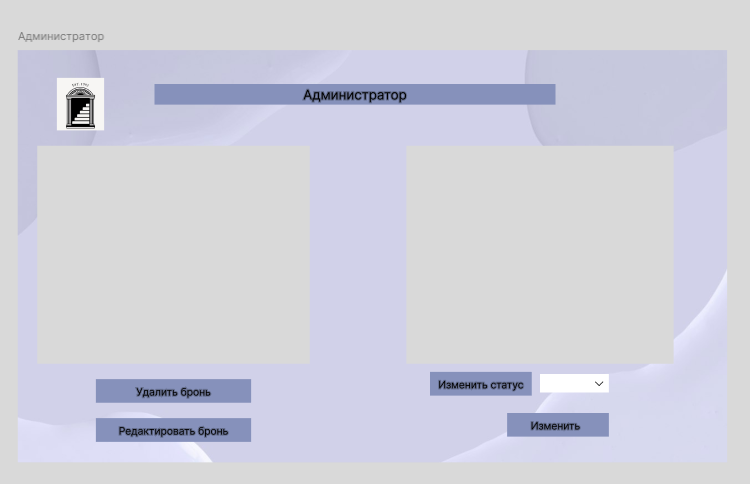


Рисунок 15 – Mockup-макет главного окна администратора

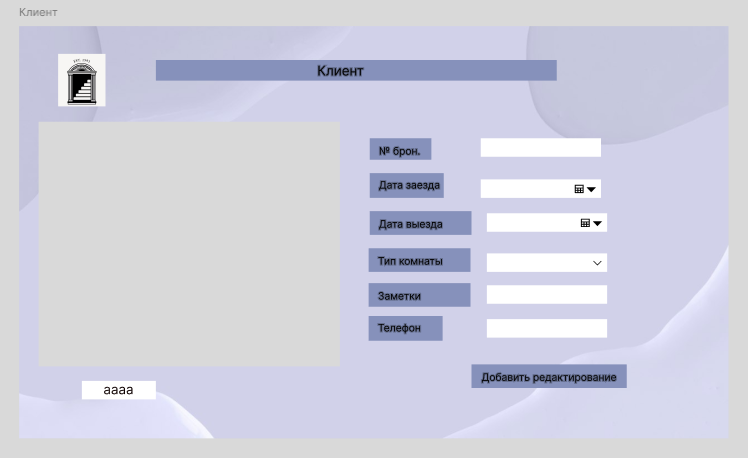


Рисунок 16 – Mockup-макет главного окна клиента

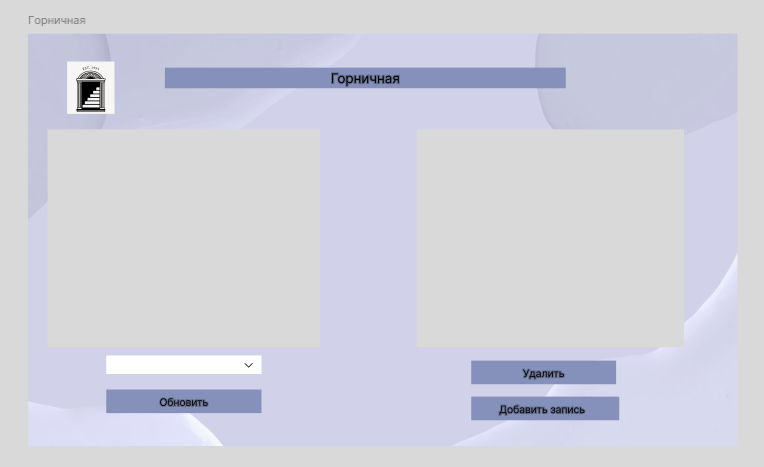


Рисунок 17 – Mockup-макет главного окна горничной

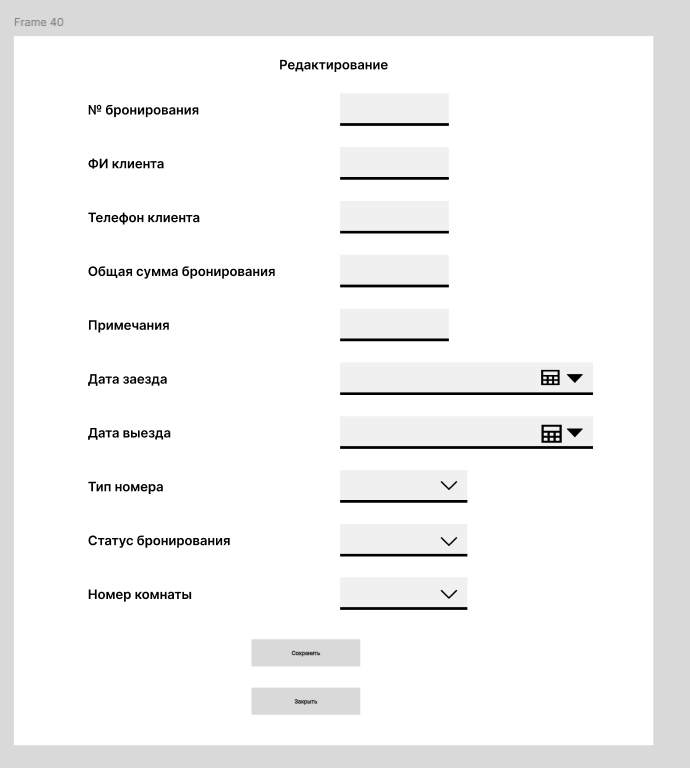


Рисунок 18 – Mockup-макет окна создания бронирований в таблицы

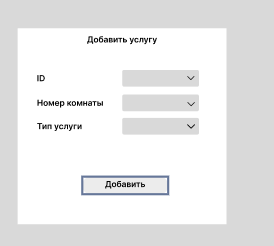


Рисунок 19 - Mockup-макет окна создания услуг в таблицы

### **Карта навигации приложения**



Рисунок 20 – Карта навигации клиента

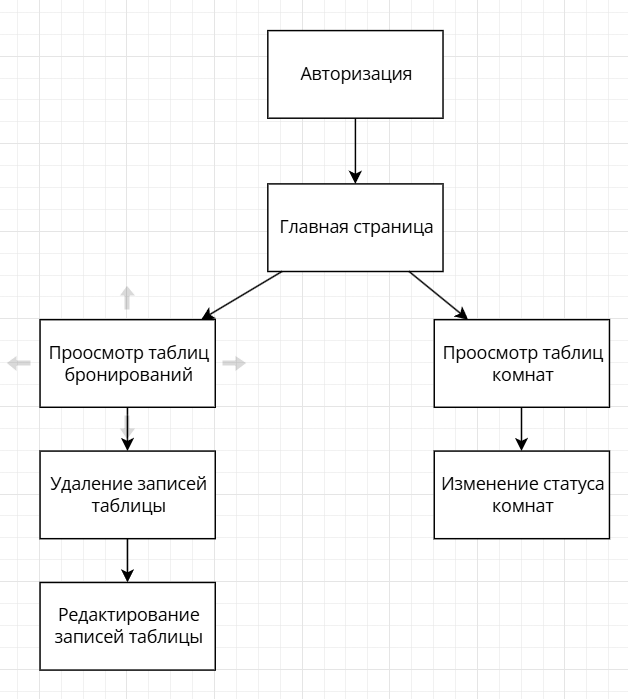


Рисунок 21 – Карта навигации администратора

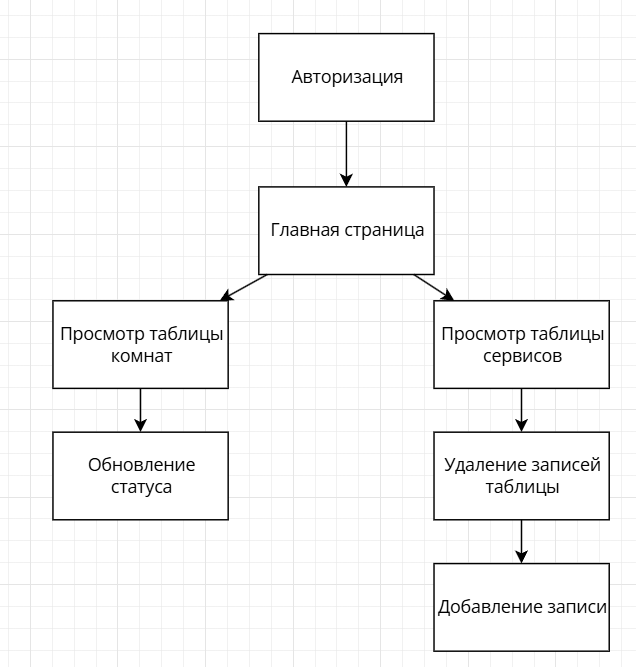


Рисунок 22 – Карта навигации горничной

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 были реализованы следующие окна приложения:

### **Главное окно, авторизация и регистрация**

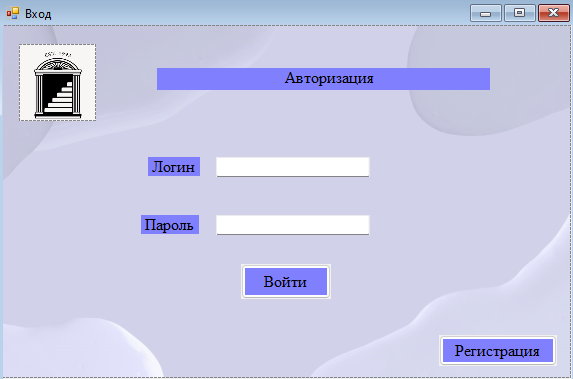


Рисунок 23 – Окно авторизации

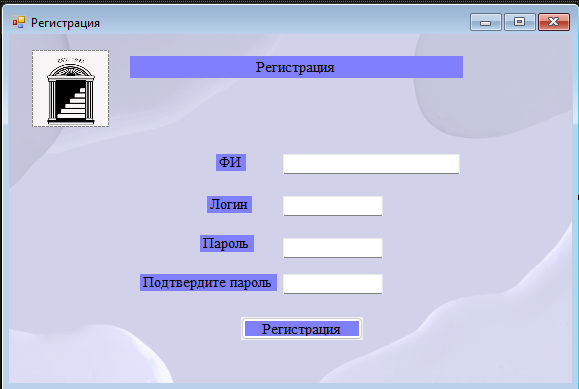


Рисунок 24 – Окно регистрации

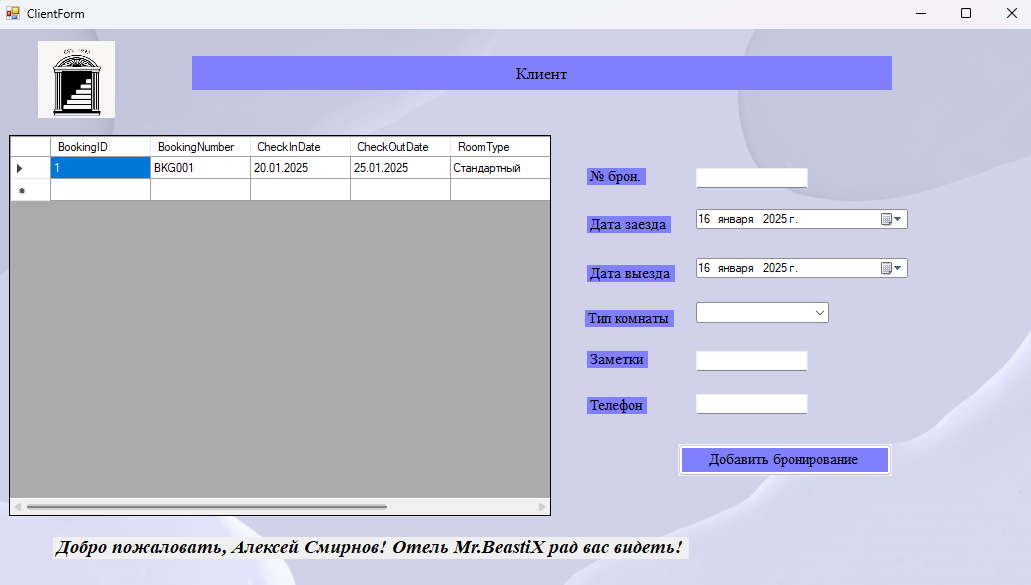


Рисунок 25 – Главное окно клиента с именем «Алексей Смирнов»

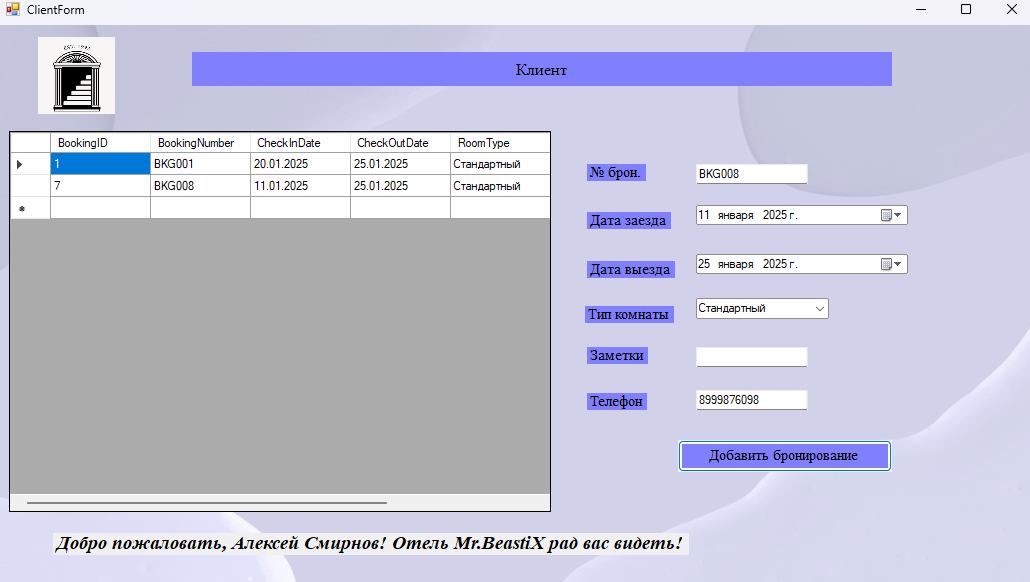


Рисунок 26 – Добавление нового бронирования для «Алексей Смирнов»

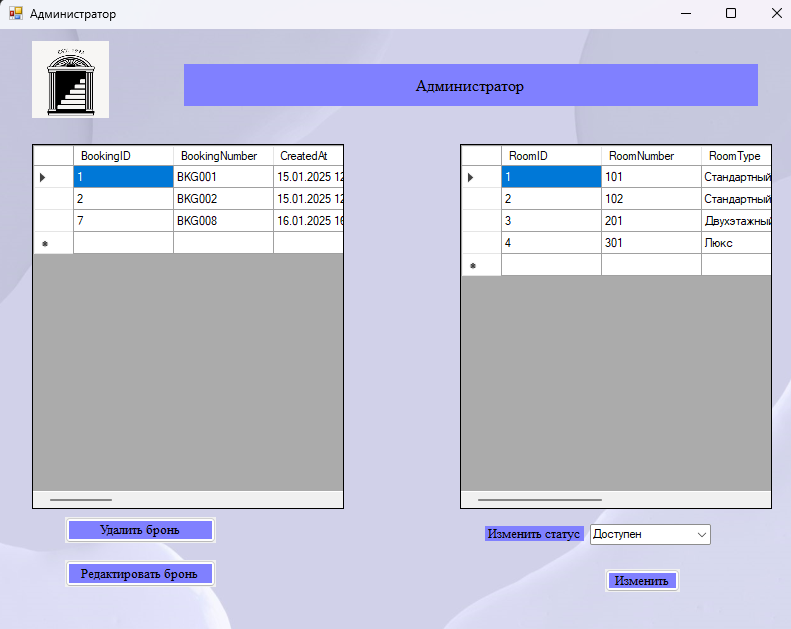


Рисунок 27 – Главное окно администратора

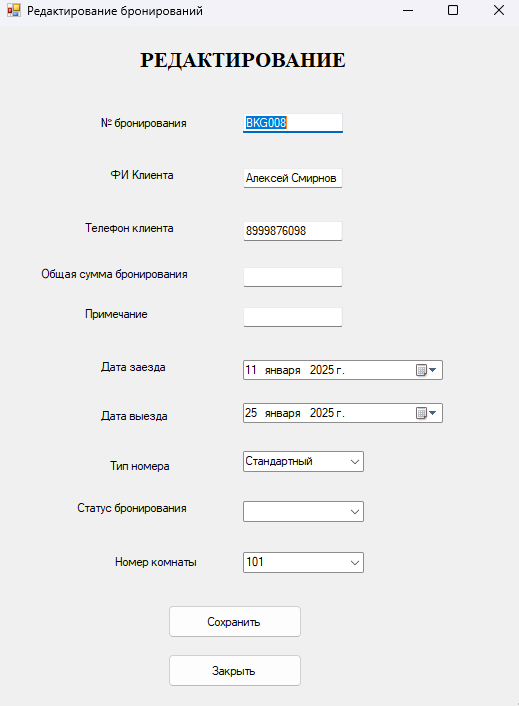


Рисунок 28 – Окно редактирования бронирования

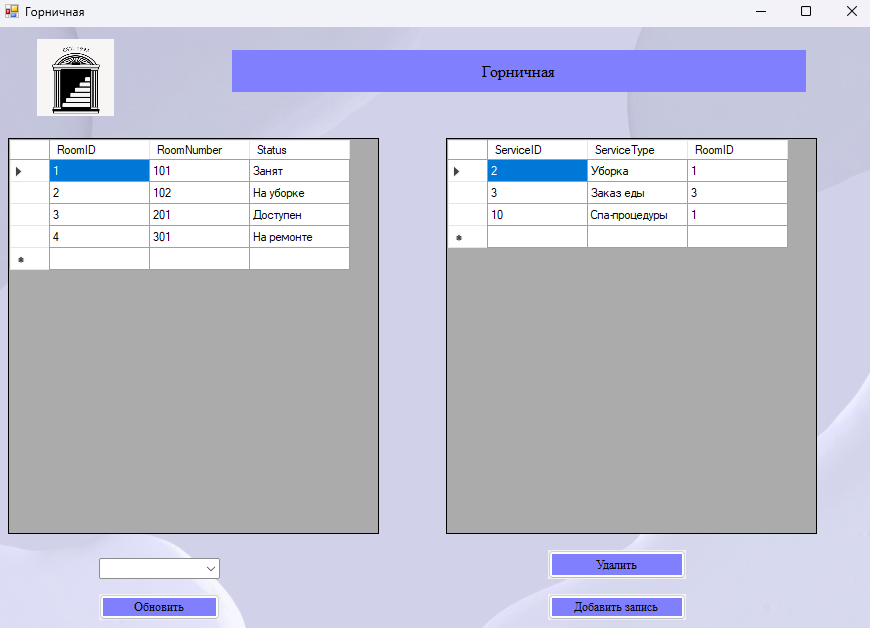


Рисунок 29 – Главное окно горничной

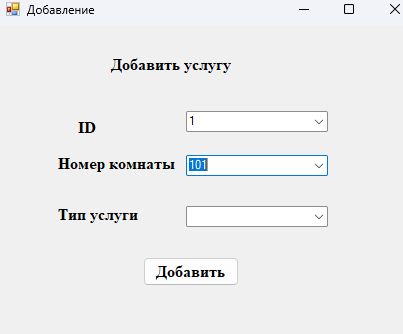


Рисунок 30 – Окно добавления услуги

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**

|  |  |
| --- | --- |
| **Название проекта** | Mr.BeastiX |
| **Рабочая версия** | 1.0 |
| **Имя тестирующего** | Румянцева Анастасия Владимировна |
| **Дата(ы) теста** | 15.01.2025 |

|  |  |
| --- | --- |
| **Тестовый пример #** | TC\_UI\_1 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | Регистрация нового пользователя с пустыми полями |
| **Краткое изложение теста** | Проверка поведения системы при попытке регистрации пользователя с незаполненными полями. |
| **Этапы теста** | 1. Открыть окно регистрации. 2. Оставить все поля пустыми. 3. Нажать кнопку "Зарегистрировать". |
| **Тестовые данные** | * ФИ: "" * Логин: "" * Пароль: "" * Подтверждение пароля: "" |
| **Ожидаемый результат** | Появление сообщения: "Все поля должны быть заполнены." |
| **Фактический результат** | Сообщение: "Все поля должны быть заполнены." |
| **Статус** | Зачет (Рисунок 31) |
| **Предварительное условие** | Открыто окно регистрации. |
| **Постусловие** | Данные в базе данных не изменились. |
| **Примечания/комментарии** | - |

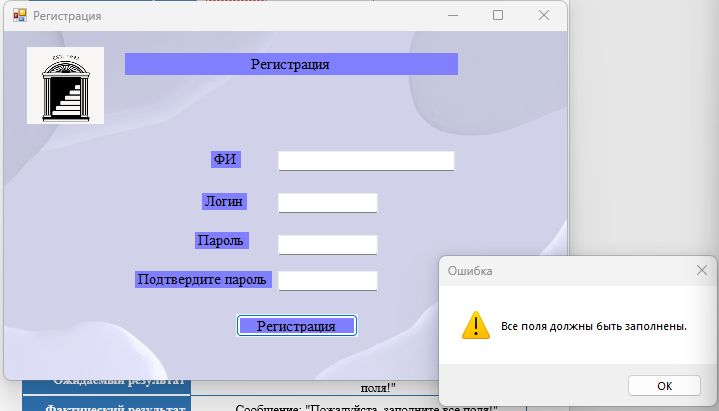


Рисунок 31 – Успешный вывод ошибки

|  |  |
| --- | --- |
| **Тестовый пример #** | TC\_UI\_2 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | Создание нового бронирования с корректными данными |
| **Краткое изложение теста** | Проверка корректного создания нового бронирования с валидными данными. |
| **Этапы теста** | 1. Открыть главное окно клиента. 2. Ввести номер бронирования (например “BKG007”) 3. Выбрать из списка дат дату заезда и выезда 4. Выбрать из выпадающего списка тип комнаты (например “Стандартный”) 5. Ввести номер телефона в поле 6. Нажать кнопку "Добавить бронирование". |
| **Тестовые данные** | * № брон.: BKG007 * Дата заезда: 18 января 2025 * Дата выезда: 19 января 2025 * Тип комнаты: Стандартный * Заметки: - * Телефон: 8909876543 |
| **Ожидаемый результат** | Появление сообщения: "Бронирование сохранено!"  Появление нового бронирования в таблице бронирований |
| **Фактический результат** | Сообщение: "Бронирование сохранено!"  Появление нового бронирования в таблице бронирований. |
| **Статус** | Зачет (Рисунок 32) |
| **Предварительное условие** | Открыто главное окно клиента. |
| **Постусловие** | В базе данных появилось новый заказ с указанными данными. |
| **Примечания/комментарии** | - |

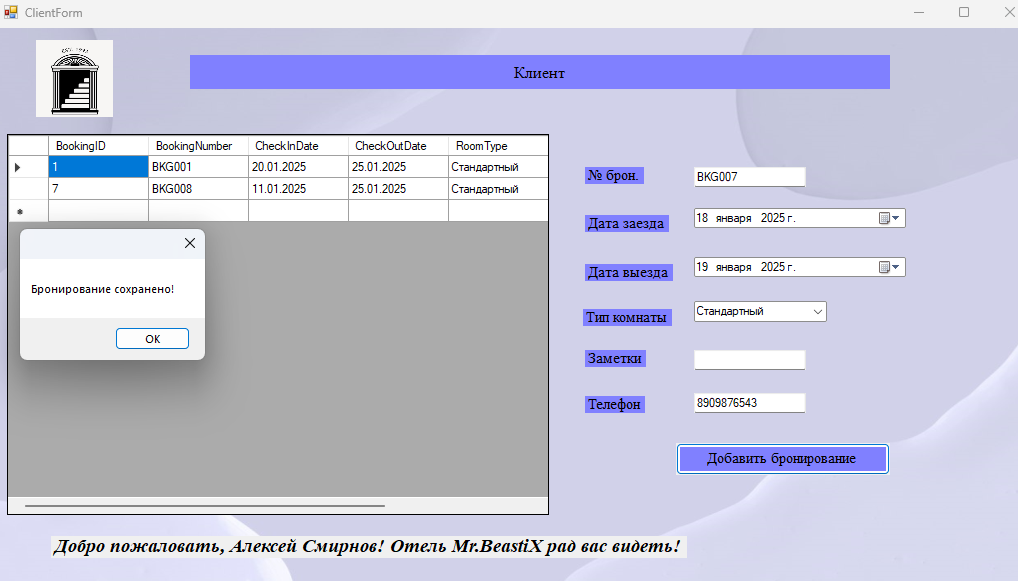


Рисунок 32 – Успешный вывод сообщения

|  |  |
| --- | --- |
| **Тестовый пример #** | TC\_UI\_3 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | Регистрация нового пользователя |
| **Краткое изложение теста** | Проверка корректного создания нового клиента с валидными данными. |
| **Этапы теста** | 1. Открыть окно регистрации пользователя. Данные пользователя:  * ФИ: " Сергей Дубов " * Логин: " kek" * Пароль: "321" * Подтверждение пароля: "321" * Нажать кнопку "Регистрация"  1. Ввести новые валидные данные в поле входа:  * Логин: "kek" * Пароль: "321" * Нажать кнопку "Войти". |
| **Тестовые данные** | * ФИ: " Сергей Дубов " * Логин: " kek" * Пароль: "321" * Подтверждение пароля: "321" |
| **Ожидаемый результат** | 1. Появление сообщения: "Регистрация успешна."  2. Появление сообщения «Добро пожаловать, Сергей Дубов» |
| **Фактический результат** | 1. Появление сообщения: "Регистрация успешна."  2. Появление сообщения «Добро пожаловать, Сергей Дубов» |
| **Статус** | Зачет (Рисунок 33-34) |
| **Предварительное условие** | Открыто окно регистрации и окно входа. |
| **Постусловие** | В базе данных появился новый пользователь с указанными данными. |
| **Примечания/комментарии** | - |

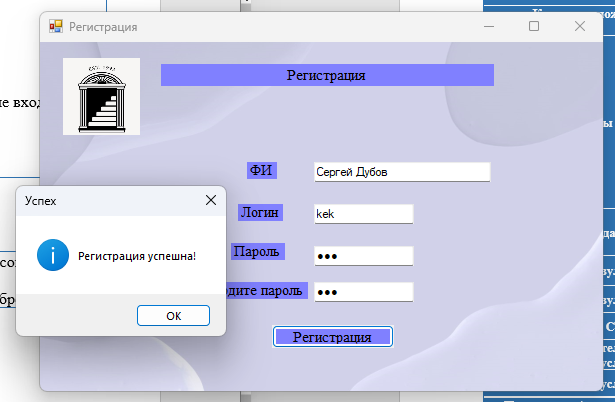


Рисунок 33 – Успешная регистрация

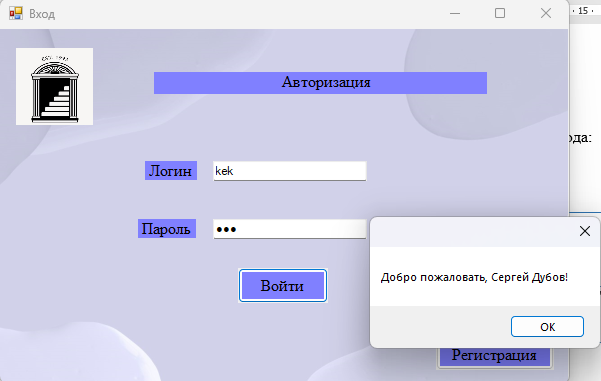


Рисунок 34 – Успешный вход

|  |  |
| --- | --- |
| **Тестовый пример #** | TC\_UI\_4 |
| **Приоритет тестирования** | Низкий |
| **Заголовок/название теста** | Проверка вывода ошибки при нажатии на кнопку «Редактировать». |
| **Краткое изложение теста** | Проверка вывода ошибки при не выбранной строке таблицы и нажатия на кнопку «Редактировать» |
| **Этапы теста** | 1. Открыть главное окно администратора. 2. Не выбирать всю строку в таблице, только ячейку или столбец. 3. Нажать кнопку "Редактировать бронь". |
| **Тестовые данные** | Пользователь с ролью "Администратор". |
| **Ожидаемый результат** | Появление сообщения: «Выберите бронирование для редактирования» |
| **Фактический результат** | Сообщение: "Выберите бронирование для редактирования! |
| **Статус** | Зачет (Рисунок 35) |
| **Предварительное условие** | Открыто главное окно администратора. |
| **Постусловие** | Данные в базе данных не изменились. |
| **Примечания/комментарии** | - |

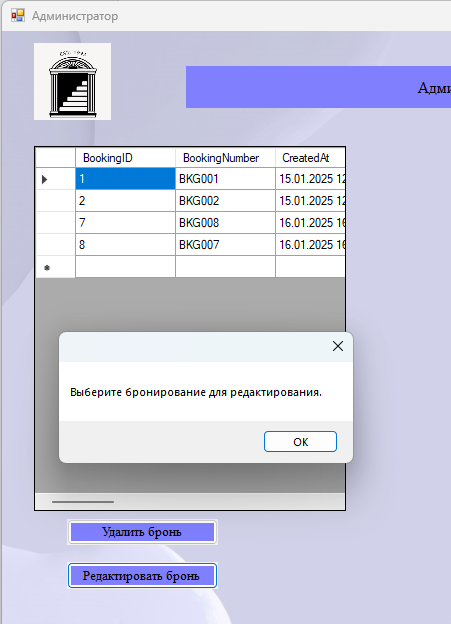


Рисунок 35 – Успешный вывод ошибки

### **Unit-tests**

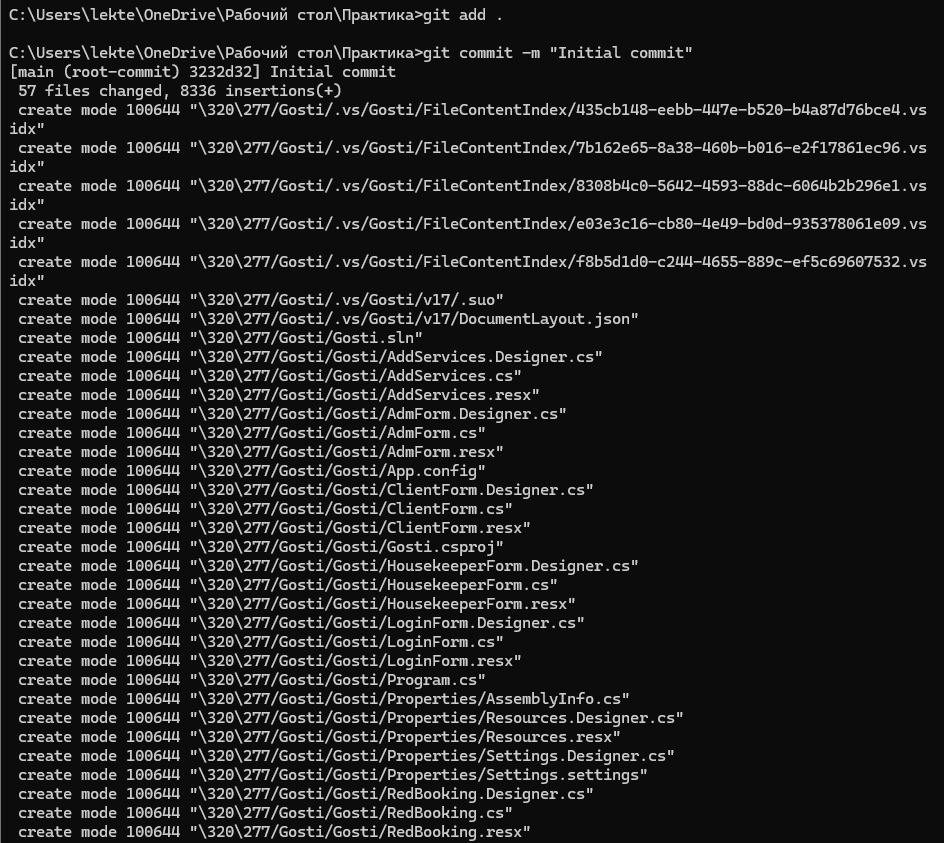
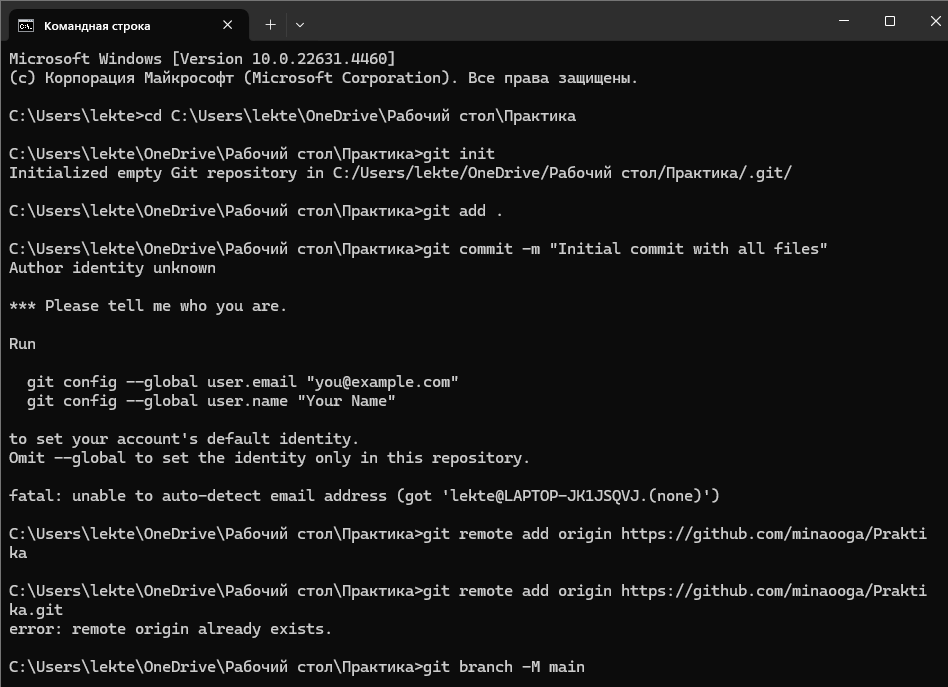
Задание №6 «GitHub.Оценка проекта»

**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.



https://github.com/minaooga/Praktika

**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: 1.5**
  + 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: 3**

- Алгоритмы оформлены в соответствии со стандартами.

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

* **Балл: 3**

- Программный модуль разработан по имеющемуся алгоритму в среде разработки и практически соответствует техническому заданию с незначительными отклонениями.

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

* **Балл: 4**
  + Исходный код модуля соответствует гайдлайну.
  + Название приложения, форм и переменных четко отражают их назначение.

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

* **Балл: 0**
  + 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: 1,5**
  + Выполнена отладка программного обеспечения с использованием инструментальных средств.

**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: 0**
  + Результаты тестов не представлены.
* **Г3Д1: Использование инструментальных средств для тестирования функциональности программы**
* **Балл: 0**
  + Инструментальные средства при выполнении тестирования не использовались

**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: 3**
  + 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.

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

* **Балл: 1,5**
  + Выполнено резервное копирование базы данных.
  + Не выполнено восстановление состояния базы данных на заданную дату.

**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: 0**
  + Выбраны принципы регистрации и система паролей.
  + Созданы и обоснованы группы пользователей.

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

* **Балл: 0**
  + Не разработан документ "Руководство системному программисту" в соответствии со стандартом.

**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.

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

* **Балл: 4**
  + Не предложены варианты модификации программного обеспечения

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

* **Балл: 0**
  + Не 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: 2**
  + Определены некоторые качественные характеристики программного кода.
  + Выявлено несколько фрагментов некачественного кода.

**Итоговый балл: 47/80**

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

**1. Соответствие функциональным требованиям**

* **Оценка: 5(4)/5**
  + Проект полностью реализует все функциональные требования, описанные в ТЗ.
  + Все основные функции реализованы корректно.

**2. Удобство интерфейса**

* **Оценка: 5/5**
  + Интерфейс соответствует требованиям ТЗ: используются нейтральные цвета (белый фон, черный текст, фиолетовый цвет(градиент))
  + Пользовательский интерфейс понятен и удобен для работы.
  + Все элементы управления (кнопки, поля ввода) расположены логично, что упрощает взаимодействие с системой.

**3. Надежность и стабильность**

* **Оценка: 5(3)/5**
  + Система устойчива к ошибкам: обработаны исключительные ситуации.
  + Реализовано резервное копирование базы данных.
  + Авторизация пользователей выполнены корректно, что обеспечивает безопасность данных.

**4. Производительность**

* **Оценка: 4/5**
  + Система работает быстро и эффективно, однако система может зависнуть при получении большого потока информации.

**5. Соответствие техническим требованиям**

* **Оценка: 5/5**
  + Используется язык программирования C# : .NET 8.0 Windows Forms.
  + Интерфейс разработан с использованием Windows Forms.
  + База данных реализована на Microsoft SQL Server 2019.
  + Система работает на Windows 10 и выше, что соответствует требованиям заказчика.

**6. Гибкость и масштабируемость**

* **Оценка: 4/5**
  + Однако для полной масштабируемости рекомендуется доработка архитектуры базы данных и оптимизация запросов.

**7. Качество кода**

* **Оценка: 5(4)/5**
  + Код хорошо структурирован, соответствует стандартам и гайдлайнам.
  + Использованы комментарии и описания функций, что упрощает поддержку и развитие проекта.
  + Реализована обработка ошибок и исключений, что повышает надежность системы.

**8. Тестирование и отладка**

* **Оценка: 4(3)/5**
  + Результаты тестирования оформлены в виде протокола, что соответствует требованиям ТЗ.
  + Однако не было выполнено модульное тестирование.
  + Система продемонстрировала высокую стабильность и корректность работы.

**9. Документация**

* **Оценка: 4(3)/5**
  + Не предоставлена полная документация, включая "Руководство системному программисту" и описание архитектуры системы.
  + 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.
  + Заказчик удовлетворен результатом и считает, что проект полностью оправдал ожидания.

**Итоговая оценка: 40/50**

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

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.

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.

ПРИЛОЖЕНИЕ А

  
Рисунок 41 – Логотип

**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

Румянцева Анастасия Владимировна

(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