|  |  |
| --- | --- |
| ­­­  **К Г Э У** | МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«КАЗАНСКИЙ ГОСУДАРСТВЕННЫЙ ЭНЕРГЕТИЧЕСКИЙ УНИВЕРСИТЕТ»**  (ФГБОУ ВО «КГЭУ») |

**Техническое задание**

**Дисциплина: Проектный практикум**

|  |  |
| --- | --- |
| **Выполнил:** | Гайфутдинов И.И. |
| **Группа:** | ТРП-1-20 |
| **Проверил:** | Алексеев И.П. |
|  |  |
|  |  |

Казань – 2024

**Оглавление**

[1. Наименование и область применения 3](#_Toc154401682)

[2. Основание для разработки 3](#_Toc154401683)

[3. Назначение разработки 3](#_Toc154401684)

[4. Технические требования к программе или программному изделию 3](#_Toc154401685)

[Требования к функциональным характеристикам: 3](#_Toc154401686)

[Требовании к надежности: 7](#_Toc154401687)

[Требования к составу и параметрам технических средств: 7](#_Toc154401688)

[Стадии и этапы разработки: 8](#_Toc154401689)

[5. Средства разработки 8](#_Toc154401690)

1. **Наименование и область применения**

Программное обеспечение – АРМ администратора автосалона.

Продукт предназначен для учета и хранения информации, необходимой в ходе работы в должности администратора в автосалоне.

1. **Основание для разработки**

Основанием для разработки служит задание от преподавателя дисциплины «Проектный практикум по управлению разработкой и разработке программного обеспечения» Алексеева И.П.

1. **Назначение разработки**

Система создана с целью обеспечения эффективной работы администраторов автосалонов. Это достигается через разработку десктопного приложения, предназначенного для автоматизации ключевых сфер деятельности. В числе этих сфер - запись клиентов на просмотр автомобилей, управление данными об автотранспорте в наличии, а также систематизация информации о клиентах.

Проект "АРМ администратора автосалона" рассчитан на создание благоприятной платформы, направленной на повышение эффективности и точности предоставляемых услуг. Он учитывает важность обеспечения конфиденциальности и безопасности хранения данных.

1. **Технические требования к программе или программному изделию**

## **Требования к функциональным характеристикам:**

2.1. Учет и хранение данных об автомобилях

Эта подсистема разработана для эффективного управления жизненным циклом данных об автомобилях и их технических параметрах. Внутри нее реализованы следующие функции:

1. Создание карточки автомобиля

• Предоставление администратору возможности мгновенно создавать новые карточки.

• Возможность ввода основной информации с простотой и удобством.

• Возможность детального ввода технических характеристик, предоставляя полное представление об автомобиле.

• Возможность добавления фото автомобиля.

2. Управление данными об автомобилях

• Предоставление гибких инструментов для изменения основных данных, связанных с автомобилями.

• Возможность корректировки технических характеристик напрямую из карточек автомобилей, обеспечивая максимальную оперативность и точность данных.

• Возможность смены фото автомобиля.

3. Удаление карточек автомобилей

• Предоставление инструментов для удаления карточек автомобилей с полным удалением, связанных данных с ним.

Эта подсистема не только упрощает процессы управления информацией об автомобилях, но и делает его более интуитивным и простым для администраторов.

Учет и хранение данных о клиентах

Эта подсистема разработана для эффективного управления данными клиентов салона. Внутри нее реализованы следующие функции:

1. Создание карточки клиента

• Предоставление администратору возможности создавать новые карточки.

• Простой и удобный интерфейс для ввода основной информации, включая ФИО, дату рождения, пол.

2. Управление данными об клиентах

• Гибкие инструменты для изменения ключевых данных клиентов, позволяя оперативно реагировать на изменения и запросы.

3. Удаление карточек клиентов

• Предоставление инструментов для удаления карточек клиентов с полным удалением, связанных данных с ним.

Эта подсистема не только упрощает процессы управления информацией о клиентах, но и стремится сделать этот процесс более интуитивным и удобным для администраторов, чтобы они могли максимально эффективно взаимодействовать с клиентами.

Учет и хранение данных о заявках на просмотр

Эта подсистема разработана для управления данными о заявках на просмотр автомобилей от клиентов. Внутри нее реализованы следующие функции:

1. Создание документа о просмотре одного из автомобилей салона

• Предоставление администратору возможности создавать новые документы.

• Возможность выбора в документе из списка автомобилей салона.

• Возможность выбора в документе из списка клиентов салона.

• Указание даты приема.

2. Управление документами

• Предоставление гибких инструментов для изменения данных документа, основных полей.

• Фильтрация документов по непоказанным или всем.

3. Удаление документов

• Предоставление инструментов для удаления документов.

Эта подсистема делает простым учет документов о просмотре автомобилей клиентами для администраторов.

Подсистема аутентификации

Подсистема предназначена для идентификации и аутентификации администраторов салона. Без прохождения данного этапа – дальнейшая работа с системой невозможна.

Функции, которые реализуются данной подсистемой:

1. Аутентификация пользователя

• Возможность ввода логина и пароля администратором салона для получения доступа к программе и данным для дальнейшей работы.

2. Управление ролями пользователей

• Возможность наделить выбранного пользователя правами администратора системы, которая позволяет добавлять новых пользователей в систему.

Эта подсистема предназначена для сохранения информационной безопасности в системе и защите конфиденциальных данных.

Шаблон программного интерфейса

Программный интерфейс предназначен для более легкой масштабируемости проекта в будущем при необходимости расширения списка сущностей для учета в системе.

Функции, которые реализуются:

1. Создание/удаление таблицы новой сущности в базе
2. Создание стандартного отображения в виде списка для новой сущности
3. Создание стандартных команд для новой сущности “Добавить”, “Удалить”, “Редактировать”
4. Реализация дополнительных форм создания и редактирования сущности
5. Связь программного интерфейса и стандартных команд с базой данных

Программный интерфейс представляет собой шаблон/скелет с базовыми операциями с возможностью доработки в отдельных модулях.

## **Требовании к надежности:**

Безопасность и конфиденциальность данных: Наш подход к обеспечению безопасности и конфиденциальности данных предусматривает установку мер, направленных на предотвращение несанкционированного доступа или утечки информации. Мы придаем большое значение идентификации и аутентификации пользователей, а также проведению регулярных аудитов доступа для поддержания высокого уровня безопасности.

## **Требования к составу и параметрам технических средств:**

- 64- или 32-разрядный процессор на архитектуре Intel или AMD;

- 2 ГБ оперативной памяти, желательно 4 ГБ;

- 5 ГБ свободного места в памяти компьютера;

- версия операционной системы: не ниже Windows 7, macOS X 10.11, Red Hat Enterprise Linux 6/7.

## **Стадии и этапы разработки:**

1. Проектирование базы данных
2. Реализация авторизации
3. Разработка главной страницы
4. Разработка подсистемы “Автомобили”
5. Разработка подсистемы “Клиенты”
6. Разработка подсистемы “Заявки на просмотр”
7. Разработка подсистемы “Пользователи”
8. Проведение тестирования
9. Написание документации
10. Разработка программного интерфейса
11. Улучшение интерфейса приложения под удобство
12. Проведение тестирования
13. Доработка документации
14. **Средства разработки**

В работе используются язык программирования Python.

В пользу выбора предложенного языка служат:

1. Синтаксис Python напоминает английский язык, что делает код более понятным и легким для написания.
2. Библиотеки для интерфейса, такие как Tkinter, которые позволяют за короткий срок разработать интерфейс.
3. Интеграция с БД: легкая интеграция с реляционными базами данных.
4. Активное сообщество: большое сообщество разработчиков Python значит, что всегда можно найти поддержку, решение проблем и обучающие материалы.

Python — высокоуровневый язык программирования, который стал одним из самых популярных благодаря своей простоте, универсальности и обширной поддержке сообщества.

Tkinter — стандартная библиотека для создания графических интерфейсов (GUI) в языке программирования Python. Вот несколько плюсов Tkinter:

* Встроенность: Tkinter поставляется вместе с Python, поэтому для создания GUI не требуется установка дополнительных библиотек.
* Простота использования: Tkinter довольно прост в освоении, особенно для начинающих. Его синтаксис интуитивно понятен и напоминает структуру других частей Python.
* Портативность: Приложения, созданные с использованием Tkinter, могут быть запущены на различных платформах, таких как Windows, macOS и Linux, без изменений в коде.
* Широкие возможности кастомизации: Tkinter предоставляет различные виджеты (кнопки, текстовые поля, фреймы и др.), которые могут быть легко настроены и адаптированы под нужды разработчика.
* Активное сообщество: Хотя Tkinter является стандартной библиотекой, у него есть активное сообщество разработчиков, готовых предоставить поддержку и решить возможные проблемы.

Tkinter отличный выбором для быстрого создания простых GUI-приложений в Python, особенно если вам нужно что-то легкое и быстрое.

Перечень стандартных программ:

* PyCharm
* Git