**Инструкция администратора**

**Порядок развертывания ПО**

1. **Установка необходимых компонентов:**
   * Убедитесь, что на вашем компьютере установлены следующие компоненты:
     + .NET Framework (версия 4.7.2 или выше)
     + SQLite (установите SQLite библиотеку для .NET через NuGet)
     + Visual Studio (версия 2017 или выше)
2. **Создание базы данных SQLite:**
   * Создайте файл базы данных orders.db в удобном месте.
   * Создайте таблицы базы данных, выполнив следующий SQL скрипт:

CREATE TABLE OrderHeader (

order\_id INTEGER PRIMARY KEY AUTOINCREMENT,

workshop TEXT NOT NULL,

start\_date TEXT NOT NULL,

end\_date TEXT NOT NULL,

status TEXT NOT NULL

);

1. **Настройка проекта в Visual Studio:**
   * Откройте проект в Visual Studio.
   * Убедитесь, что все ссылки на библиотеки настроены правильно (особенно ссылки на SQLite).
   * Настройте строку подключения к базе данных в файле OrdersForm.cs:

string connectionString = "Data Source=путь\_к\_вашей\_базе\_данных/orders.db;Version=3;";

1. **Сборка и запуск проекта:**
   * Скомпилируйте проект и убедитесь, что сборка прошла успешно.
   * Запустите приложение, чтобы проверить его работоспособность.

**Инструкция пользователя**

**Работа с ПО**

1. **Запуск приложения:**
   * Дважды щелкните на исполняемый файл, чтобы запустить приложение.
2. **Основные функции:**
   * **Добавление заказа:**
     + Введите данные в соответствующие поля (Цех, Дата начала, Дата окончания, Статус).
     + Нажмите кнопку "Добавить заказ".
   * **Редактирование заказа:**
     + Выберите заказ из списка.
     + Отредактируйте данные в соответствующих полях.
     + Нажмите кнопку "Редактировать заказ".
   * **Удаление заказа:**
     + Выберите заказ из списка.
     + Нажмите кнопку "Удалить заказ".
   * **Поиск заказа:**
     + Введите данные для поиска в поле "Поиск по цеху".
     + Нажмите кнопку "Поиск".
3. **Навигация:**
   * Используйте таблицу для выбора и просмотра существующих заказов.
   * Все изменения автоматически обновляются в таблице после выполнения соответствующих действий.

**Инструкция разработчика**

**Стек технологий и архитектура ПО**

1. **Технологии:**
   * Язык программирования: C#
   * Платформа: .NET Framework
   * Среда разработки: Visual Studio
   * База данных: SQLite
   * Пользовательский интерфейс: Windows Forms
2. **Архитектура ПО:**
   * **Frontend:** Windows Forms для создания графического интерфейса пользователя.
   * **Backend:** Логика работы с данными и взаимодействие с базой данных реализованы на C#.
   * **База данных:** SQLite используется для хранения данных о заказах. Таблица OrderHeader хранит основную информацию о заказах.
3. **Основные классы и их функционал:**
   * OrdersForm: основной класс формы, который содержит логику работы с заказами, включая загрузку, добавление, редактирование и удаление заказов.
4. **Структура проекта:**
   * OrdersForm.cs: основной файл формы, содержащий логику работы с заказами.
   * Program.cs: точка входа в приложение.
   * orders.db: файл базы данных SQLite (размещается в папке с проектом или в указанном месте).

**Описание структуры базы данных**

**1. Таблица OrderHeader**

**Описание:**

Таблица OrderHeader содержит информацию о заказах.

**Поля:**

* + order\_id: Уникальный идентификатор заказа (первичный ключ, автоматически увеличиваемый).
  + workshop: Название цеха, где выполняется заказ (текст, не может быть пустым).
  + start\_date: Дата начала заказа (дата, не может быть пустой).
  + end\_date: Дата окончания заказа (дата, может быть пустой).
  + status: Статус заказа (текст, не может быть пустым).

**2. Таблица OrderItem**

**Описание:**

Таблица OrderItem содержит информацию о деталях заказа.

**Поля:**

* + item\_id: Уникальный идентификатор элемента заказа (первичный ключ, автоматически увеличиваемый).
  + order\_id: Идентификатор заказа, к которому относится элемент (внешний ключ, ссылающийся на OrderHeader.order\_id, не может быть пустым).
  + steel\_grade\_id: Идентификатор марки стали (внешний ключ, ссылающийся на SteelGrade.steel\_grade\_id).
  + diameter: Диаметр элемента (вещественное число).
  + wall\_thickness: Толщина стенки элемента (вещественное число).
  + volume: Объем элемента (вещественное число).
  + unit: Единица измерения (текст).
  + status: Статус элемента заказа (текст, не может быть пустым).

**3. Таблица SteelGrade**

**Описание:**

Таблица SteelGrade содержит информацию о марках стали.

**Поля:**

* + steel\_grade\_id: Уникальный идентификатор марки стали (первичный ключ, автоматически увеличиваемый).
  + grade\_name: Название марки стали (текст, не может быть пустым).

**Связи между таблицами**

* + Таблица OrderItem ссылается на таблицу OrderHeader через поле order\_id, образуя связь "многие к одному" (много элементов заказа могут относиться к одному заказу).
  + Таблица OrderItem ссылается на таблицу SteelGrade через поле steel\_grade\_id, образуя связь "многие к одному" (много элементов заказа могут использовать одну марку стали).