

Система управления продукцией (обои)

Документация к проекту

Разработано в рамках демонстрационного экзамена по специальности
09.02.07 (Информационные системы и программирование)

2025 год

Содержание

1	Описание проекта	2
1.1	Основные функции:	2
2	Требования и установка	2
2.1	Требования:	2
2.2	Установка (для сборки из исходного кода):	2
2.3	Запуск исполняемого файла:	3
2.4	Сборка из исходного кода (опционально):	3
3	Использование	4
3.1	Главное окно	4
3.2	Форма добавления/редактирования	4
3.3	Окно материалов	4
4	Структура проекта	5
5	Описание базы данных	5
6	Метод расчета количества материала	5
7	Тестирование	6
8	Авторы	6

1 Описание проекта

Проект представляет собой приложение для управления продукцией компании, производящей обои. Приложение позволяет просматривать, добавлять и редактировать данные о продукции, а также рассчитывать необходимое количество материалов для производства. Разработано в рамках демонстрационного экзамена по специальности 09.02.07 (Информационные системы и программирование).

1.1 Основные функции:

- Просмотр списка продукции с расчетом стоимости на основе используемых материалов.
- Добавление и редактирование данных о продукции.
- Просмотр списка материалов, необходимых для выбранного продукта.
- Расчет количества материала, необходимого для производства заданного количества продукции.

2 Требования и установка

2.1 Требования:

- **Операционная система:** Windows (или любая ОС с установленным Python для сборки).
- **Python:** Версия 3.8 или выше (только для сборки).
- **Библиотеки Python (только для сборки):**
 - `tkinter` (встроенная в Python).
 - `sqlite3` (встроенная в Python).
 - `Pillow` (для работы с изображениями).
- **SQLite:** Утилита `sqlite3.exe` для работы с базой данных.
- Графические файлы: `res/logo.png`, `res/icon.ico`.

2.2 Установка (для сборки из исходного кода):

1. Убедитесь, что Python установлен:

```
python --version
```

2. Установите библиотеку Pillow:

```
pip install Pillow
```

3. Скачайте проект из репозитория (например, через `git clone`).
4. Убедитесь, что в папке проекта находятся:
 - Исходные файлы: `main.py`, `material_calc.py`.
 - Скрипты БД: `schema.sql`, `schema2.sql`, `data.sql`, `data_import.sql`.
 - Графические файлы: `res/logo.png`, `res/icon.ico`.
 - Исполняемый файл: `sqlite3.exe`.
5. Создайте базу данных:

```
sqlite3 wallpaper_company.db < schema.sql  
sqlite3 wallpaper_company.db < schema2.sql  
sqlite3 wallpaper_company.db < data.sql  
sqlite3 wallpaper_company.db < data_import.sql
```

2.3 Запуск исполняемого файла:

1. Скачайте скомпилированный файл `main.exe` из репозитория.
2. Поместите его в папку с файлом `wallpaper_company.db`.
3. Дважды щелкните на `main.exe` для запуска приложения.

2.4 Сборка из исходного кода (опционально):

1. Установите PyInstaller:

```
pip install pyinstaller
```

2. Перейдите в папку проекта:

```
cd path/to/your/project
```

3. Выполните команду для компиляции:

```
pyinstaller --onefile --windowed --icon=res/icon.ico --add-data "res;res" --add-data "wallpaper_company.db;." main.py
```

Для Linux/Мас замените `;` на `:` в `-add-data`.

4. Найдите скомпилированный файл в папке `dist/main.exe`.
5. Запустите `main.exe`.

3 Использование

3.1 Главное окно

- **Список продукции:** Отображает все продукты с типом, наименованием, артикулом, минимальной стоимостью, шириной рулона и рассчитанной стоимостью.
- **Кнопка "Добавить продукт":** Открывает форму для добавления нового продукта.

3.2 Форма добавления/редактирования

- **Поля:** Артикул, тип продукта (выпадающий список), наименование, минимальная стоимость, ширина рулона.
- **Валидация:** Поля не могут быть пустыми, стоимость и ширина должны быть неотрицательными.
- **Кнопка "Сохранить":** Сохраняет изменения и обновляет список.
- **Кнопка "Назад":** Закрывает форму без сохранения.

3.3 Окно материалов

- **Таблица материалов:** Показывает материалы, используемые продуктом (наименование, требуемое количество).
- **Выбор материала:** Кликните на строку в таблице, чтобы выбрать материал для расчета.
- **Форма расчета:**
 - **Количество продукции:** Сколько единиц продукции нужно произвести.
 - **Параметр 1, Параметр 2:** Например, длина и ширина (в метрах).
 - **Количество на складе:** Автоматически заполняется из базы данных для выбранного материала.
- **Кнопка "Рассчитать":** Вычисляет необходимое количество материала с учетом процента брака и текущего запаса.
- **Кнопка "Назад":** Закрывает окно.

4 Структура проекта

- `main.py`: Основной файл приложения, содержит интерфейс и логику работы.
- `material_calc.py`: Модуль для расчета количества материала (функция `calculate_material`).
- `wallpaper_company.db`: База данных SQLite.
- `schema.sql`, `schema2.sql`: Скрипты создания структуры базы данных.
- `data.sql`, `data_import.sql`: Скрипты для импорта данных из CSV.
- `res/logo.png`, `res/icon.ico`: Графические файлы для логотипа и иконки приложения.
- `main.exe`: Скомпилированный исполняемый файл (доступен после сборки).

5 Описание базы данных

База данных (`wallpaper_company.db`) содержит следующие таблицы:

- `product_type`: Типы продукции (например, "Обои "Флизелиновые обои") с коэффициентом для расчета.
- `material_type`: Типы материалов (например, "Бумага "Краска") с процентом брака.
- `material`: Материалы (наименование, цена, запас на складе).
- `product`: Продукты (артикул, тип, наименование, минимальная стоимость, ширина рулона).
- `product_material`: Связь продуктов и материалов (требуемое количество материала для продукта).

Схема соответствует 3-й нормальной форме с первичными и внешними ключами.

6 Метод расчета количества материала

Функция `calculate_material_quantity` (в `material_calc.py`):

- **Входные параметры:**
 - `product_type_id`: Тип продукции (для получения коэффициента).
 - `material_id`: Идентификатор материала (для получения процента брака и запаса).

- `product_quantity`: Количество продукции.
- `param1, param2`: Параметры (например, длина и ширина).
- `stock_quantity`: Количество материала на складе.

- **Логика:**

1. Базовое количество: `param1 * param2 * coefficient`.
2. Учет брака: `base_quantity * (1 + defect_rate)`.
3. Общее количество: `adjusted_quantity * product_quantity`.
4. Требуемое количество: `ceil(total_quantity - stock_quantity)`.

- **Результат:** Возвращает целое число (необходимое количество материала) или -1 при ошибке.

7 Тестирование

- Приложение протестировано на корректность:
 - Добавление/редактирование продукции.
 - Просмотр материалов.
 - Расчет количества материала с разными входными данными.
- Обработаны исключения:
 - Пустые поля.
 - Отрицательные значения.
 - Некорректные числовые данные.

8 Авторы

Разработано в рамках демонстрационного экзамена, 2025 год.