**Документация по проекту**

**Наименование проекта:** решение кластеризации компании ООО “ИТ-Вижн”

**Выполнили:** Соломонов Артём Максимович, Сусин Иван Дмитриевич

**Специальность:** 09.02.07 Информационные системы и программирование

**Группа:** ИСП-9

**1.1 Проблема**

Завод “К” специализируется на производстве стали различных марок.

Производственный процесс включает несколько этапов:

1. Загрузка сырья: Загрузка руды и других компонентов в доменную печь.
2. Доменная плавка: Получение чугуна.
3. Конвертерный процесс: Переработка чугуна в сталь определенной марки. В конвертер добавляются легирующие элементы для достижения нужных характеристик стали.
4. Разливка: Разливка стали в слитки или заготовки.
5. Прокатка: Формирование из слитков или заготовок листов, прутков, и других видов стального проката.

Завод производит три основных марки стали: A, B и C. Каждая марка требует различного количества сырья и времени на каждом этапе. Завод имеет ограниченные ресурсы:

* Руда: Ограниченное количество доступной руды в день.
* Легирующие элементы: Ограниченное количество каждого легирующего элемента в день (например, никель, хром, марганец).
* Время работы оборудования: Доменная печь, конвертер и прокатный стан имеют ограниченное время работы в день (в часах).

**1.2 Решение**

Разработать консольное приложение на C#, которое поможет оптимизировать производственный план завода “К” для максимизации прибыли. Приложение должно принимать на вход:

* Доступные ресурсы (руда, легирующие элементы, время работы оборудования).
* Требования к производству каждой марки стали (количество сырья, время на этапах, цена за тонну).
* Целевой объем производства каждой марки стали.

Приложение должно выдавать оптимальный производственный план, указывающий, сколько тонн каждой марки стали нужно произвести, чтобы максимизировать прибыль, не превышая доступные ресурсы и удовлетворяя целевым объемам производства.

Дополнительные условия:

* Приложение должно быть написано на C# с использованием .NET Framework или .NET Core.
* Необходимо обеспечить ввод данных из файла.
* Результаты должны быть четко и понятно представлены в консоли.
* Приветствуется использование комментариев в коде для пояснения логики.
* Реализовать проверку на корректность введенных данных (например, проверка на неотрицательность значений).
* Если невозможно достичь целевых объемов производства для всех марок стали, приложение должно сообщать об этом и предлагать альтернативный план, максимально приближенный к целевому.

**1.3 Навигация между экранами**

Приложение состоит из экранов, перечисленных ниже.

* Экран склада: Пользователь вводит количество имеющейся у него руды.
* Экран заказов: Пользователь добавляет заказ с необходимыми ему ресурсами и создаёт план
* Экран выполнения заказа: Пользователь выполняет заказ

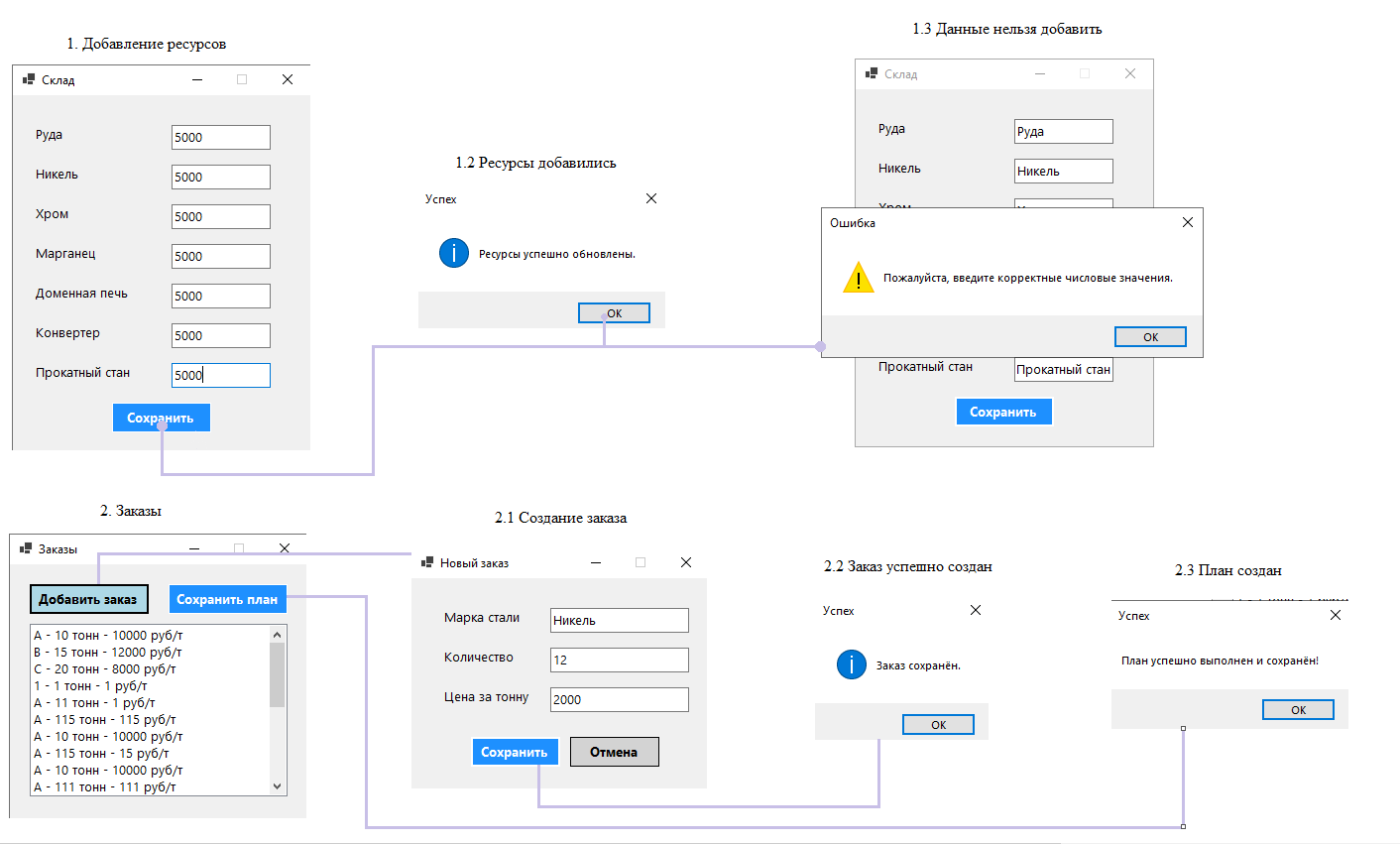
****

Рисунок 1.1 – Переходы между экранами

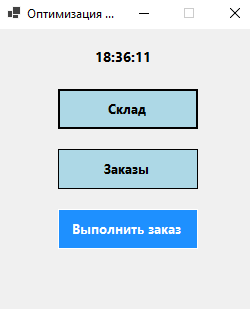


Рисунок 1.2 – Главное меню

**1.4 Техническая документация по проекту:**

**1.4.1 Класс MainForm**

Наследуется от System.Windows.Forms.Form. Представляет главное окно приложения

*Поля доступа private:*

|  |  |
| --- | --- |
| timer | Указываем полный путь |
| components | Контейнер для управления компонентами. |
| btnWarehouse | Кнопка ‘Склад’ |
| btnOrders | Кнопка ‘Заказы |
| btnExecute | Кнопка ‘Выполнить заказ |
| labelTime | Метка для отображении времени |

*Методы доступа private:*

|  |  |
| --- | --- |
| InitializeClock | Настраивает и запускает таймер для отображения времени |
| UpdateTime | Обработчик события Tick таймера; обновляет текст метки с текущим временем |
| OpenWarehouseForm | Открывает форму WarehouseForm |
| OpenOrdersForm | Открывает форму OrdersForm |
| ExecuteOrder | Создаёт экземпляр OrderExecution и вызывает его метод Execute() |
| InitializeComponent | Создаёт и настраивает элементы интерфейса формы |
| Dispose | Переопределённый метод для освобождения ресурсов. |

**1.4.2 Класс WarehouseForm**

Наследуется от System.Windows.Forms.Form.

*Поля доступа private:*

|  |  |
| --- | --- |
| txtIron | Поле для отображения/ввода количества руды ("Ruda") |
| txtNickel | Поле для никеля |
| txtChrome | Поле для хрома |
| txtManganese | Поле для марганца |
| txtBlastTime | Время работы доменной печи |
| txtConverterTime | Время работы конвертера |
| txtRollingTime | Время работы прокатного стана |
| btnSave | Кнопка сохранения данных |

*Методы доступа public:*

|  |  |
| --- | --- |
| WarehouseForm | Конструктор формы — вызывает InitializeComponent() и загружает данные о ресурсах |

*Методы доступа private:*

|  |  |
| --- | --- |
| LoadResources | Загружает данные из DatabaseManagerи устанавливает их в соответствующие текстовые поля |
| SaveResources | Сохраняет значения из текстовых полей в базу данных через DatabaseManager. Вызывает сообщение об успешном сохранении или ошибке |
| WarehouseForm\_Load | Событийный метод загрузки формы. В данный момент пустой |
| InitializeComponent | Создаёт элементы интерфейса формы |

**1.4.3 Класс SteelOrder**

*Поля доступа public:*

|  |  |
| --- | --- |
| SteelType | Тип стали (например, "нержавеющая", "углеродистая") |
| Quantity | Количество стали в тоннах |
| OrePerTon | Норма расхода руды на 1 тонну стали |
| NickelPerTon | Норма расхода никеля на 1 тонну |
| ChromiumPerTon | Норма расхода хрома на 1 тонну |
| ManganesePerTon | Норма расхода марганца на 1 тонну |
| BlastFurnaceTimePerTon | Время работы доменной печи на 1 тонну |
| ConverterTimePerTon | Время работы конвертера на 1 тонну |
| RollingMillTimePerTon | Время работы прокатного стана на 1 тонну |
| ProfitPerTon | Прибыль с одной тонны стали |
| TotalProfit | Общая прибыль |
| RequiredOre | Требуемое количество руды |
| RequiredNickel | Требуемое количество никеля |
| RequiredChromium | Требуемое количество хрома |
| RequiredManganese | Требуемое количество марганца |
| RequiredBlastFurnaceTime | Требуемое время для доменной печи |
| RequiredConverterTime | Требуемое время для конвертера |
| RequiredRollingMillTime | Требуемое время для прокатного стана |
| ResourceRequirements | Словарь всех требуемых ресурсов с их количеством |

**1.4.4 Класс OrdersForm**

Наследуется от System.Windows.Forms.Form.

*Поля доступа private:*

|  |  |
| --- | --- |
| components | Контейнер компонентов для освобождения ресурсов. |
| btnSavePlan | Кнопка "Сохранить план" |
| lstOrders | Список заказов |
| components | Контейнер компонентов для управления ресурсами. |
| lstOrders | Список заказов (вывод информации). |
| btnAddOrder | Кнопка "Добавить заказ". |
| btnSavePlan | Кнопка "Сохранить план". |

*Методы доступа public:*

|  |  |
| --- | --- |
| OrdersForm | Конструктор формы — вызываетInitializeComponent()и инициализирует контейнер |

*Методы доступа private:*

|  |  |
| --- | --- |
| LoadOrders | Загружает список заказов из OrderManager и выводит их в lstOrders |
| SaveProductionPlan | Основной метод планирования производства: проверяет ресурсы, вычисляет прибыль, генерирует отчет и обновляет БД |
| LoadResourcesFromDatabase | Получает текущие ресурсы со склада из базы данных |
| CheckResources | Проверяет, достаточно ли ресурсов для выполнения всех заказов |
| CalculateTotalProfit | Вычисляет общую прибыль по всем заказам |
| DeductResources | Уменьшает остатки ресурсов в базе после выполнения заказов |
| GenerateReport | Генерирует отчёт и сохраняет его в файл |

**1.4.5 Класс Order**

*Поля доступа public:*

|  |  |
| --- | --- |
| Steel Type | Тип стали (например, "нержавеющая") |
| Quantity | Количество стали в тоннах |
| Ore Per Ton | Расход руды на 1 тонну |
| Nickel Per Ton | Расход никеля на 1 тонну |
| Chromium Per Ton | Расход хрома на 1 тонну |
| Manganese Per Ton | Расход марганца на 1 тонну |
| Blast Furnace Time Per Ton | Время работы доменной печи на 1 тонну |
| Converter Time Per Ton | Время конвертера на 1 тонну |
| Rolling Mill Time Per Ton | Время прокатного стана на 1 тонну |

**1.4.6 Класс OrderManager**

*Методы доступа public:*

|  |  |
| --- | --- |
| SaveOrder | Сохраняет новый заказ в базу данных SQLite |
| GetAllOrders | Получает все заказы из БД и возвращает их в виде списка объектовOrder |

**1.4.7 Класс OrderExecution**

*Методы доступа public:*

|  |  |
| --- | --- |
| Execute | Запускает полный цикл выполнения заказов: загрузка данных, проверка ресурсов, отчёт, списание. |

*Методы доступа private:*

|  |  |
| --- | --- |
| LoadOrdersFromDatabase | Загружает заказы из базы и преобразует их в список объектов SteelOrder |
| LoadResourcesFromDatabase | Получает текущие значения ресурсов (руды, времени станков и т.д.) из БД. |
| CheckResources | Проверяет, хватает ли ресурсов для всех заказов. Возвращает true или false |
| CalculateTotalProfit | Считает общую прибыль по всем заказам. |
| GenerateReport | Генерирует текстовый отчет и сохраняет его на диск. |
| DeductResources | Списывает использованные ресурсы из базы данных. |

**1.4.8 Класс ReportGenerator**

*Методы доступа public:*

|  |  |
| --- | --- |
| GenerateReport | Генерирует текстовый отчет о производственном плане и сохраняет его в файл на диск. Также выводит сообщения об успехе или ошибке. |

**1.4.9 Класс NewOrderForm**

Наследуется от System.Windows.Forms.Form.

*Поля доступа private:*

|  |  |
| --- | --- |
| txtSteelType | Поле ввода марки стали |
| txtQuantity | Поле ввода количества |
| txtProfit | Поле ввода прибыли за тонну |
| btnSave | Кнопка сохранения заказа |
| btnCancel | Кнопка отмены |
| components | Контейнер компонентов для управления ресурсами. |
| txtSteelType | Поле ввода марки стали |
| txtQuantity | Поле ввода количества |
| txtProfit | Поле ввода прибыли за тонну |
| btnSave | Кнопка сохранения заказа |
| btnCancel | Кнопка отмены |

*Методы доступа public*

|  |  |
| --- | --- |
| NewOrderForm | Вызывает метод InitializeComponent()  для инициализации элементов формы. |

*Методы доступа private*

|  |  |
| --- | --- |
| btnSave\_Click | Обрабатывает нажатие на кнопку "Сохранить": проверяет данные и вызываетOrderManager.SaveOrder |
| btnCancel\_Click | Обрабатывает нажатие на кнопку "Отмена": закрывает форму без сохранения. |

**1.5 Ссылка на открытый репозиторий**

<https://github.com/Levikaci/case-1_case-1.1>

