Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Российский экономический университет имени Г.В. Плеханова»

Московский приборостроительный техникум

**Курсовой проект**

ПМ 11 Разработка, администрирование и защита баз данных

МДК 11.01. «Технология разработки и защиты баз данных»

Специальность 09.02.07 «Информационные системы и программирование»

Квалификация: Программист

Тема: «Разработка информационной системы "Производство микроэлектроники" (Реализация подсистем: бухгалтерия, производственный отдел, склад, отдел кадров, учет посетителей, отдел поддержки)»

**Пояснительная записка**

Листов: 36

Руководитель\_\_\_\_\_\_\_\_\_\_\_\_\_/ Н.С. Митасов

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Консультант \_\_\_\_\_\_\_\_\_\_\_\_\_ / А.А. Шимбирёв

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2021 г.

Исполнитель\_\_\_\_\_\_\_\_\_\_\_\_\_/ А.А. Дрюпин

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

2021

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Российский экономический университет имени Г.В. Плеханова»

Московский приборостроительный техникум

«Утверждаю»

Заместитель директора по учебной работе

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д.А. Клопов

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Задание

на выполнение курсового проекта (курсовой работы)

Дрюпину Андрею Александровичу

(фамилия, имя, отчество студента — полностью)

студенту группы П50-1-18 специальности 09.02.07 «Информационные системы и программирование» по МДК 11.01 «Технология разработки и защиты баз данных»

1. Исходные данные к проекту (работе):
   1. Тема: «Разработка информационной системы "Производство микроэлектроники" (Реализация подсистем: бухгалтерия, производственный отдел, склад, отдел кадров, учет посетителей, отдел поддержки)».
   2. Состав курсового проекта:
      1. Задание КП
      2. Пояснительная записка
      3. Программа (исходные данные) на электронном носителе
      4. Презентация и инсталляционный пакет программы на электронном носителе
   3. Содержание пояснительной записки:

ВВЕДЕНИЕ

1. ОБЩАЯ ЧАСТЬ
   1. Цель разработки
   2. Средства разработки
2. СПЕЦИАЛЬНАЯ ЧАСТЬ
   1. Постановка задачи
      1. Входные данные
      2. Выходные данные
      3. Подробные требования к проекту
   2. Внешняя спецификация
      1. Описание задачи
      2. Входные и выходные данные
      3. Методы
      4. Тесты
      5. Контроль целостности данных
   3. Проектирование
      1. Схема архитектуры приложения
      2. Логическая схема данных
      3. Физическая схема данных
      4. Структурная схема
      5. Функциональная схема
      6. Диаграмма классов
      7. Схема тестирования
      8. Схема пользовательского интерфейса
   4. Результат работы программы
3. ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ
   1. Инструментальные средства
   2. Отладка программы
   3. Защитное программирование
   4. Характеристики программы

ЗАКЛЮЧЕНИЕ

СПИСОК ИСПОЛЬЗУЕМЫХ МАТЕРИАЛОВ

ПРИЛОЖЕНИЕ А. Текст программы

ПРИЛОЖЕНИЕ Б. Сценарий и результаты тестовых испытаний

ПРИЛОЖЕНИЕ В. Руководство пользователя

ПРИЛОЖЕНИЕ Г. Скрипт базы данных

1. Содержание задания по проекту (работе) ⁠— перечень вопросов, подлежащих разработке

|  | Разрабатываемый вопрос | Объем от всего задания, % | Срок выполнения |
| --- | --- | --- | --- |
| А | Описательная часть проекта (введение, общее описание и т. д.) |  | 01.11.2021 |
| 1. | Введение |  | 01.11.2021 |
| 2. | Цель разработки |  | 01.11.2021 |
| 3. | Средства разработки |  | 01.11.2021 |
| Б | Анализ задачи и её постановка |  | 22.11.2021 |
| 1. | Определение требований к программе |  | 15.11.2021 |
| 2. | Спецификация программы (описание задачи, описание входных и выходных данных, методы) |  | 15.11.2021 |
| 3. | Тесты, контроль целостности данных |  | 22.11.2021 |
| В | Проектирование и реализация |  | 29.11.2021 |
| 1. | Схемы проекта (схема архитектуры, логическая схема данных, физическая схема данных, функциональная и структурная схемы, диаграмма классов, схема тестирования, схема пользовательского интерфейса) |  | 15.11.2021 |
| 2. | Реализация в инструментальной среде |  | 29.11.2021 |
| Г | Технологическая часть проекта |  | 06.12.2021 |
| 1. | Инструментальные средства разработки |  | 06.12.2021 |
| 2. | Отладка программа |  | 06.12.2021 |
| 3. | Защитное программирование |  | 06.12.2021 |
| 4. | Характеристика программы |  | 06.12.2021 |
| Д | Программная документация |  | 13.12.2021 |
| 1. | Приложение Б. Текст программы |  | 13.12.2021 |
| 2. | Приложение В. Сценарий и результаты тестовых испытаний |  | 13.12.2021 |
| 3. | Приложение Г. Руководство пользователя |  | 13.12.2021 |
| 4. | Приложение Д. Скрипт базы данных |  | 13.12.2021 |
| Е | Экспериментальная часть проекта |  | 20.12.2021 |
| 1. | Программа на машинном носителе.  Информация на носителе разбита на разделы: эксплуатационный пакет, тексты программы, документация. |  | 20.12.2021 |

Руководитель курсового проекта (работы) Митасов Никита Сергеевич, преподаватель

«25» октября 2021 года \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Н.С. Митасов/

Дата выдачи курсового задания «25» октября 2021 года

Срок сдачи законченного проекта (работы) «20» декабря 2021 года

Задание принял к исполнению

«25» октября 2021 года \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / А.А. Дрюпин/

## ВВЕДЕНИЕ

Производство микроэлектроники – важнейшая область промышленности в данный момент из-за повсеместного развития технологий. Готовая продукция заводов микроэлектроники используется в самых разных областях, где есть техника: медицина, производство военной техники, производство бытовой техники, машиностроение.

Таким образом, тема производства микроэлектроники является одной из самых актуальных на данный момент.

Данная курсовая работа направленна на изучение структуры работы и оптимизацию производственного процесса на производствах микроэлектроники при помощи разработки информационной системы производства микроэлектроники.

1. ОБЩАЯ ЧАСТЬ
   1. Цель разработки

Оптимизация трудовых ресурсов сотрудников производства микроэлектроники, структурирование информации, необходимой при производстве.

* 1. Средства разработки

Программные средства, использованные для разработки информационной системы, перечислены в таблице 1.

Таблица 1 – Программные средства

|  |  |  |  |
| --- | --- | --- | --- |
| № | Тип средства | Название средства | Назначение |
| 1 | 2 | 3 | 4 |
| 1 | Текстовый редактор | Microsoft Word 2016 (16.0.4591.1000) | Разработка документации, формирование отчетных документов |
| 2 | Инструментальное средство разработки программных решений | Visual Studio 2019 Community 16.8.3 | Разработка настольного приложения и Web API |
| 3 | Система управления базами данных | MSSQL Server 17 (17.9.1) | Создание и управление базой данных |
| 4 | Средство проектирования | ERWin Data Modeler 7.2.0.1644 | Разработка схем для проектирования приложения |
| 5 | Табличный редактор | Microsoft Excel 2016 (16.0.4266.1001) | Экспорт и импорт данных приложения |

В качестве средств вычислительной техники использовался ноутбук. Его характеристики представлены в таблице 2.

Таблица 2 – Технические средства

| № | Тип оборудования | Наименование оборудования |
| --- | --- | --- |
| 1 | 2 | 3 |
|  | Ноутбук HP Envy x360 13-ar0010ur | |
| 1 | Размер экрана: | 13.3" |
| 2 | Разрешение экрана: | 1920x1080 |
| 3 | Линейка процессора: | AMD Ryzen 5 3500 U |
| 4 | Количество ядер процессора: | 4 |
| 5 | Оперативная память: | 8 ГБ |
| 6 | Тип видеокарты: | встроенная |
| 7 | Видеокарта: | AMD Radeon Vega 8 |
| 8 | Конфигурация накопителей: | SSD |
| 9 | Общий объем всех накопителей: | 128 ГБ |
| 10 | Операционная система | Windows 10 Home |

В качестве средств периферийной техники использовались устройства, приведенные в таблице 3.

Таблица 3 – Периферийные устройства

| Наименование | Описание |
| --- | --- |
| 1 | 2 |
| Мышь |  |

1. СПЕЦИАЛЬНАЯ ЧАСТЬ
   1. Постановка задачи

Разработка информационной системы "Производство микроэлектроники" (Реализация подсистем: бухгалтерия, производственный отдел, склад, отдел кадров, учет посетителей, отдел поддержки) на платформе WPF.

* 1. Входные и выходные данные

Входные данные:

* Данные для авторизации пользователя (логин, пароль);
* Данные для добавления сотрудника (имя, фамилия, отчество, дата рождения, должность, логин, пароль, зарплата);
* Данные для добавления посетителя (имя, фамилия, отчество, паспорт);
* Данные для добавления поставки (название, количество, дата, контрагент);
* Данные для добавления задачи (название, количество, дата начала, дата дедлайна);
* Данные для отправки сотрудника в отпуск(дата начала и дата окончания);
* Данные для добавления контрагента (название)

Выходные данные:

* Предмет;
* Количество;
* Дата;
* Приход/расход;
* Тип операции.
  + 1. Подробные требования к проекту
* Разработать базу данных для хранения и обработки данных;
* Загрузить базу данных на хостинг;
* Все операции добавления, редактирования, удаления или получения информации из базы данных должны осуществляться с помощью Web API;
* Предусмотреть разграничение функциональных возможностей;
* Пароли пользователей должны быть зашифрованы;
* Изменение баланса организации должно быть осуществлено с помощью триггера добавления операции;
* Должно быть предусмотрено формирование таблиц в Excel.
* Администратор должен иметь возможность выгружать резервные копии данных и восстанавливать их.
* В окнах предусмотреть наличие поиска.

### Описание задачи

После запуска приложения открывается окно авторизации, после прохождения которой, пользователь попадает в окно, соответствующее его должности (hr, вахтер, кладовщик, менеджер поставок, бухгалтер, начальник производства).

В окне сотрудника отдела кадров отображаются сотрудники, есть возможность добавления сотрудника, увольнения сотрудника и отправления его в отпуск.

В окне бухгалтера отображаются поставки и расходы. Помимо этого, бухгалтер может сгенерировать таблицу в Excel.

В окне менеджера поставок отображаются данные о поставках, есть возможность создать новую поставку, создать контрагента, создать новый товар.

В окне начальника производства отображаются данные о текущих задачах по производству, есть возможность создавать новые задачи и менять статус уже существующих.

В окне кладовщика отображаются продукты, имеющиеся на складе в результате поставок или производства, кладовщик имеет возможность изменить способ хранения продукта.

В окне вахтера отображаются данные о посетителях, вахтер может менять статус посетителя на уже ушедшего.



Рисунок 1 - Диаграмма прецедентов.

### Входные и выходные данные

### В таблице 4 представлены входные данные, вводимые пользователем при взаимодействии с информационной системой.

Таблица 4 – Входные данные

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Имя | Тип | Размер в байтах | Диапазон изменений | Точность представления | Форма ввода |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | Логин | Текст | 8 | Латинские символы и цифры | [A-Za-zА-Яа-я0-9] | Ввод с клавиатуры |
| 2 | Пароль | Текст | 8 | Латинские символы и цифры | [A-Za-zА-Яа-я0-9] | Ввод с клавиатуры |
| 3 | Паспорт | Целое число | 8 | Цифры | [0-9] | Ввод с клавиатуры |
| 4 | Название | Текст | 8 | Кириллические символы и цифры | [A-Za-zА-Яа-я0-9] | Ввод с клавиатуры |
| 5 | Зарплата | Целое число | 8 | Цифры и знак «-» | [0-9] и знак «-» | Ввод с клавиатуры |
| 6 | Дата окончания | Дата | 8 | Цифры и точка | [0-9] и знак «.» | Ввод с клавиатуры или выбор в DatePicker |
| 7 | Дата начала | Дата | 8 | Цифры и точка | [0-9] и знак «.» | Ввод с клавиатуры или выбор в DatePicker |
| 8 | Фамилия | Текст | 8 | Кириллические символы | [A-Za-zА-Яа-я] | Ввод с клавиатуры |
| 9 | Дата | Дата | 8 | Цифры и точка | [0-9] и знак «.» | Ввод с клавиатуры или выбор в DatePicker |
| 10 | Количество | Целое число | 8 | Цифры | [0-9] | Ввод с клавиатуры |
| 11 | Имя | Текст | 8 | Кириллические символы | [A-Za-zА-Яа-я] | Ввод с клавиатуры |
| 12 | Отчество | Текст | 8 | Кириллические символы | [A-Za-zА-Яа-я] | Ввод с клавиатуры |
| 13 | Дата рождения | Дата | 8 | Цифры и точка | [0-9] и знак «.» | Ввод с клавиатуры или выбор в DatePicker |
| 14 | Должность | Текст | 8 | Кириллические символы | [A-Za-zА-Яа-я] | Выбор в ComboBox |

Таблица 5 – Выходные данные

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Имя | Тип | Размер в байтах | Диапазон изменений | Точность представления | Форма вывода |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | Предмет | Текст | 300 | Латинские и кириллические символы, специальные символы и цифры | [A-Za-zА-Яа-я0-9] | Элемент таблицы |
| 2 | Количество | Текст | 300 | Латинские и кириллические символы, специальные символы и цифры | [A-Za-zА-Яа-я0-9.] | Элемент таблицы |
| 3 | Тип операции | Текст | 300 | Латинские и кириллические символы, специальные символы и цифры | [A-Za-zА-Яа-я0-9.] | Элемент таблицы |
| 4 | Приход/расход | Текст | 300 | Латинские и кириллические символы, специальные символы и цифры | [A-Za-zА-Яа-я0-9] | Элемент таблицы |

### Методы

* Объектно-ориентированное программирование – работа с классами и методами, выделение связей между объектами, инкапсуляция;
* Структурное программирование – разбиение программы на взаимодействующие между собой компоненты;
* Применение паттерна MVC, позволяющего отделить логику приложения от визуальной части.

### Тесты

Сценарий и результаты тестовых испытаний представлены в приложении Б.

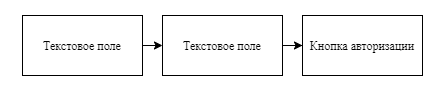


Рисунок 2 – Схема тестирования авторизации



Рисунок 3 – Схема тестирования добавления пользователя

### Контроль целостности данных

### Была проведена работа для обеспечения целостности данных. Во все поля была добавлена защита от некорректно-вводимой информации, а также обработаны все исключения приложения.

### В базе данных отсутствуют транзитивные зависимости, была проведена работа по приведению базы данных к 3 нормальной форме.

### Перечень всех возможных аномалий отображен в приложении Б. Сценарий тестовых испытаний.

* 1. Проектирование
     1. Логическая модель данных

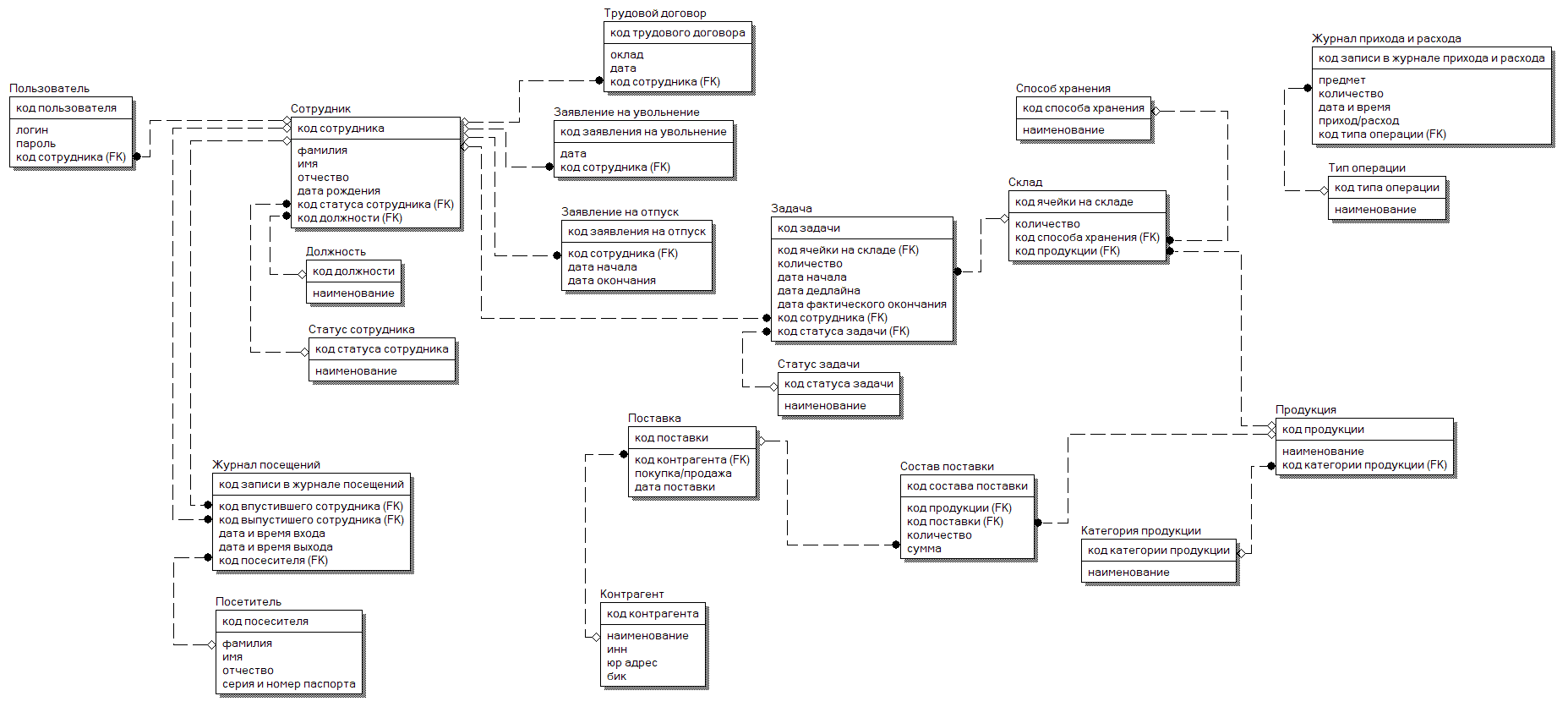


Рисунок 4 – Логическая модель данных

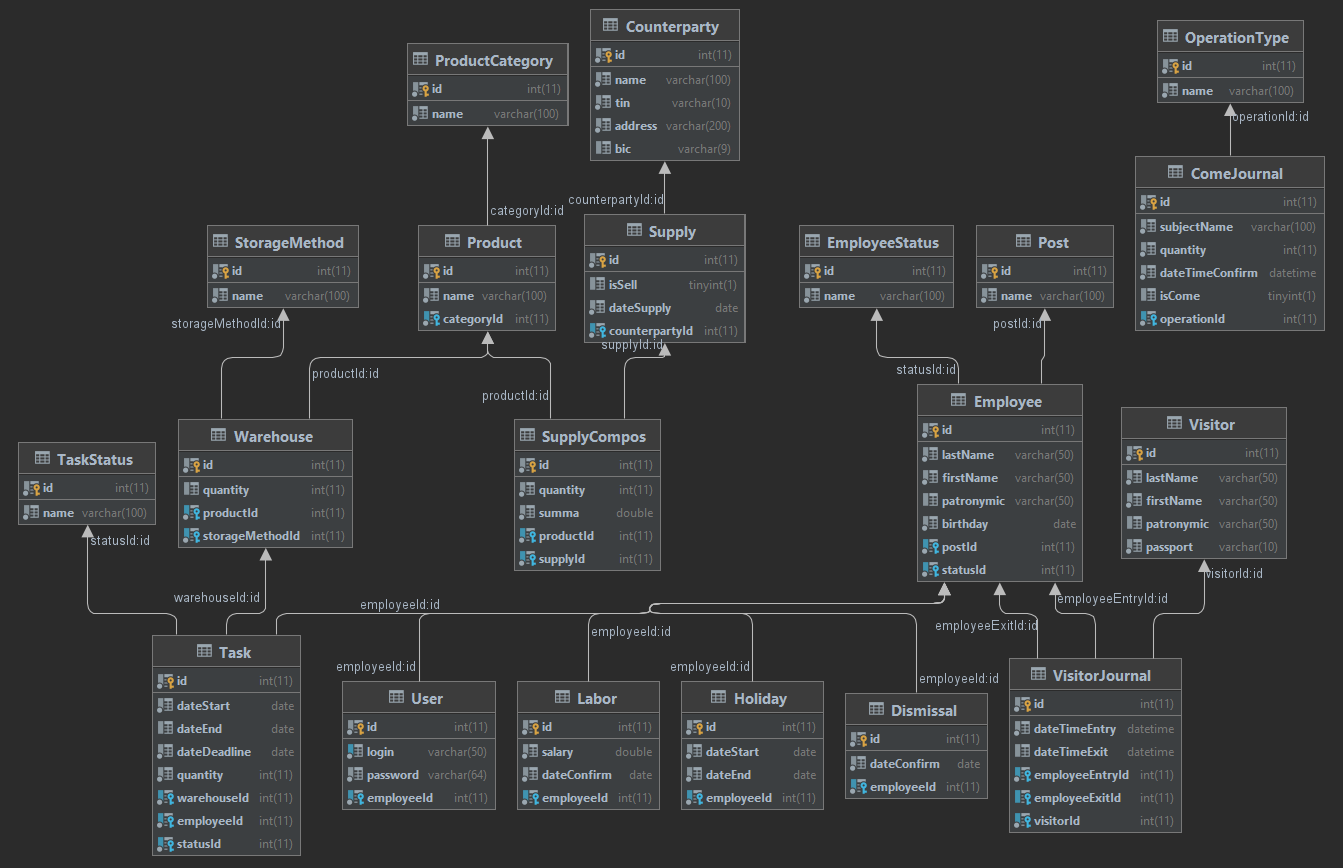


Рисунок 5 – Физическая модель данных

### Схема архитектуры программы

Схема архитектуры информационной системы представлена на рисунке 6.

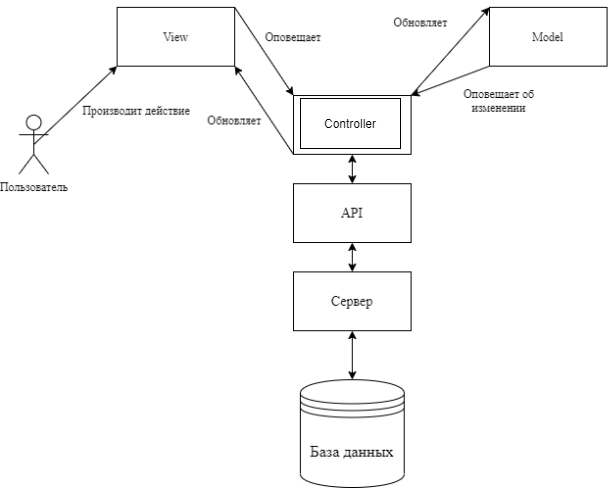


Рисунок 6 – Схема архитектуры информационной системы

### Структурная схема программы

Структурная схема информационной системы представлена на рисунке 7.

Структурная схема API представлена на рисунке 8.

Описание модулей информационной системы представлено в таблице 6.

Описание модулей API представлено в таблице 7.

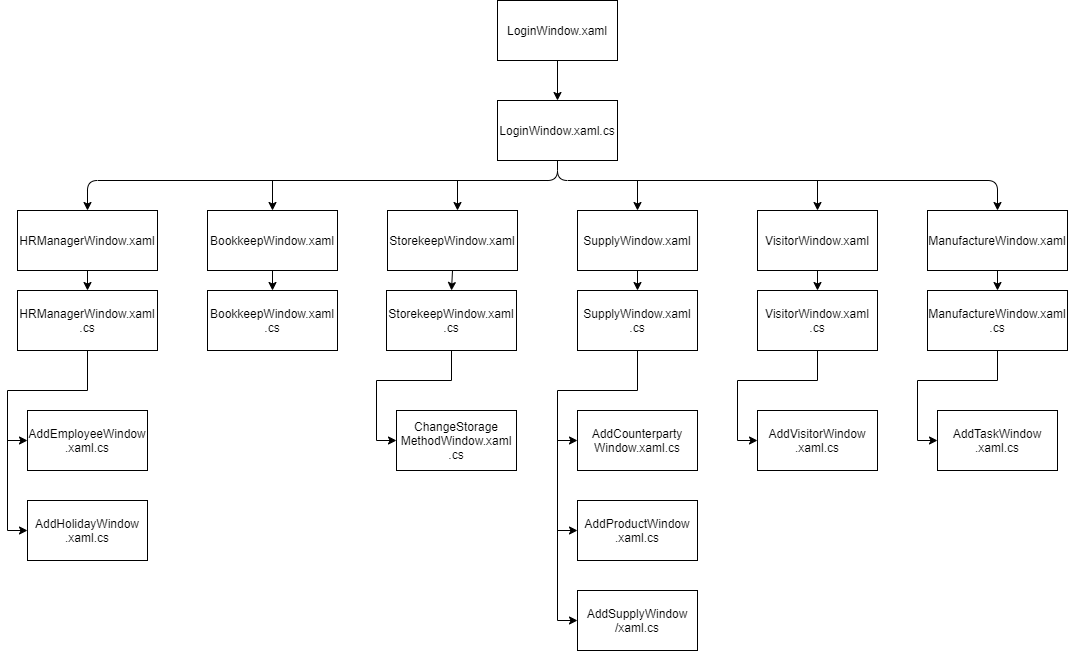
Рисунок 7 – Структурная схема информационной системы

Таблица 6 – Описание модулей информационной системы

|  |  |  |
| --- | --- | --- |
| № | Название модуля | Описание модуля |
|  | LoginWindow.xaml | Отображение графики |
|  | HRManagerWindow.xaml | Отображение графики |
|  | BookkeeperWindow.xaml | Отображение графики |
|  | StorekeeperWindow.xaml | Отображение графики |
|  | SupplyWindow.xaml | Отображение графики |
|  | VisitorWindow.xaml | Отображение графики |
|  | ManufactureWindow.xaml | Отображение графики |
|  | AddEmployeeWindow.xaml | Отображение графики |
|  | AddHolidayWindow.xaml | Отображение графики |
|  | ChangeStorageMethodWindow.xaml | Отображение графики |
|  | AddCounterpartyWindow.xaml | Отображение графики |
|  | AddProductWindow.xaml | Отображение графики |
|  | AddSupplyWindow.xaml | Отображение графики |
|  | AddVisitorWindow.xaml | Отображение графики |
|  | AddTaskWindow.xaml | Отображение графики |
|  | LoginWindow.xaml.cs | Окно авторизации |
|  | HRManagerWindow.xaml.cs | Окно отдела кадров |
|  | BookkeeperWindow.xaml.cs | Окно бухгалтерии |
|  | StorekeeperWindow.xaml.cs | Окно склада |
|  | SupplyWindow.xaml.cs | Окно отдела поставок |
|  | VisitorWindow.xaml.cs | Окно вахты |
|  | ManufactureWindow.xaml.cs | Окно производства |
|  | AddEmployeeWindow.xaml.cs | Окно добавления сотрудника |
|  | AddHolidayWindow.xaml.cs | Окно добавления отпуска |
|  | ChangeStorageMethodWindow.xaml.cs | Окно изменения способа хранения |
|  | AddCounterpartyWindow.xaml.cs | Окно добавления контрагента |
|  | AddProductWindow.xaml.cs | Окно добавления товара |
|  | AddSupplyWindow.xaml.cs | Окно добавления поставки |
|  | AddVisitorWindow.xaml.cs | Окно добавления посетителя |
|  | AddTaskWindow.xaml.cs | Окно добавления задачи |

Рисунок 8 – Структурная схема API

Таблица 7 – Описание модулей API

|  |  |  |
| --- | --- | --- |
| № | Название модуля | Описание модуля |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Функциональная схема

Функциональная схема программы представлена на рисунке 9.

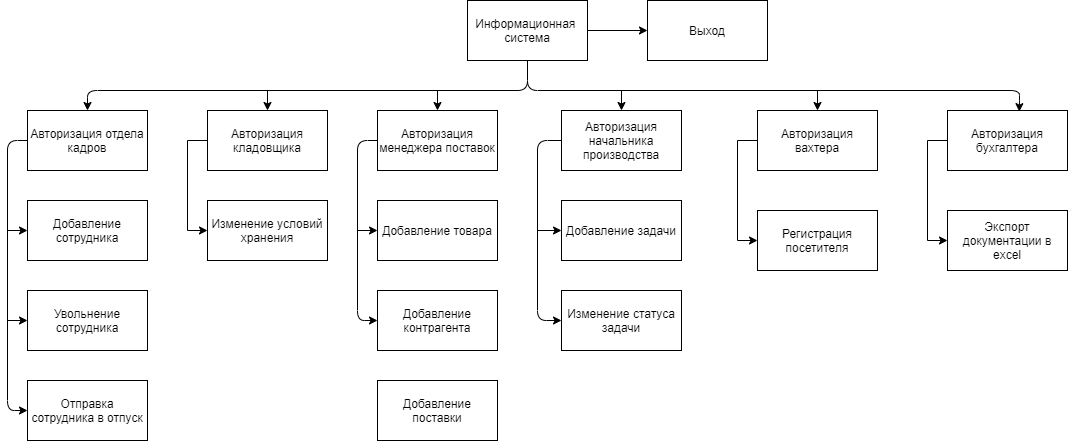


Рисунок 9 – Функциональная схема информационной системы

### Диаграмма классов

Диаграмма классов информационной системы представлена на рисунке 10 и 11.

Описание методов информационной системы представлено в таблице 8.

Диаграмма классов API представлена на рисунке 10 и 11.

Описание методов API представлено в таблице 9.

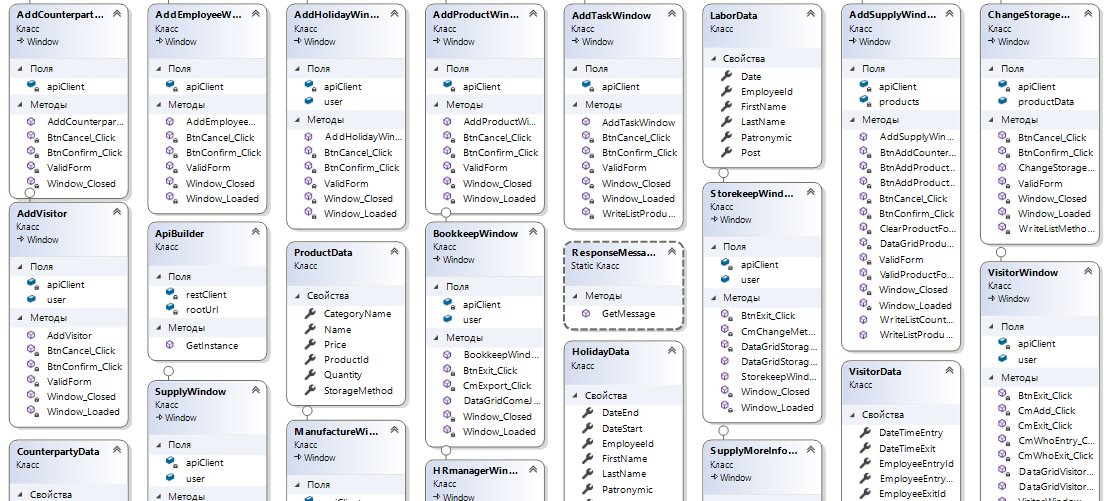


Рисунок 10 - Диаграмма классов информационной системы (1)

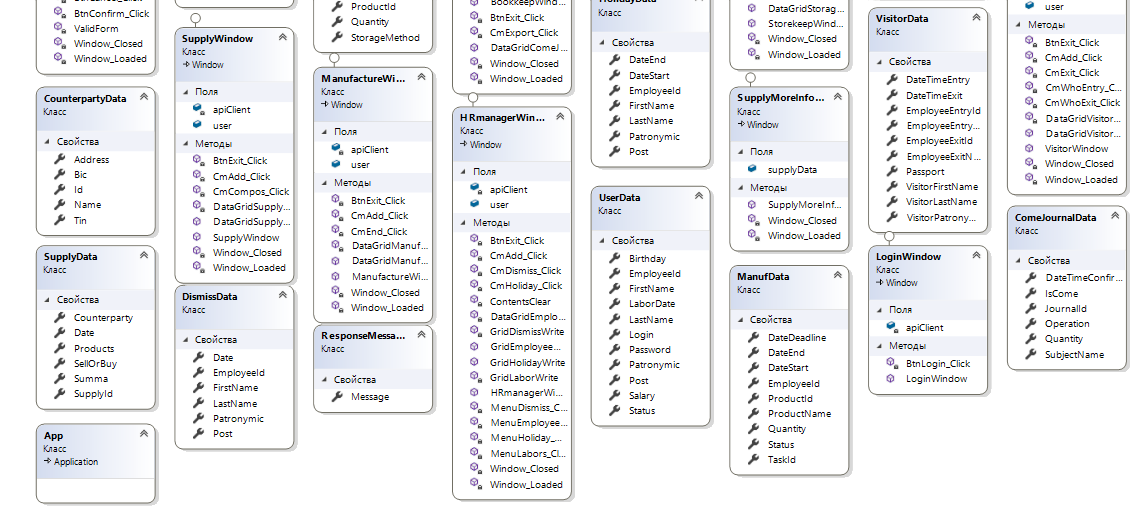


Рисунок 11 - Диаграмма классов информационной системы (2)

Таблица 8 - Описание методов

| № | Название класса | Название метода | Описание |
| --- | --- | --- | --- |
| 1 | Data | ComeJournal |  |
| 2 | Data | Counterparty |  |
| 3 | Data | Dismiss |  |
| 4 | Data | Holiday |  |
| 5 | Data | Labor |  |
| 6 | Data | Manuf |  |
| 7 | Data | Product |  |
| 8 | Data | Supply |  |
| 9 | Data | User |  |
| 10 | Data | Visitor |  |

Таблица 9 – Описание методов API

| № | Название класса | Название метода | Описание |
| --- | --- | --- | --- |
| 1 | Controllers | Get() | Получение всех объектов по запросу |
| 2 | Controllers | Get(int id) | Получение конкретного объекта по его ID |
| 3 | Controllers | Put(int id, object object) | Обновление данных объекта по его ID и новому экземпляру объекта |
| 4 | Controllers | Post(object object) | Добавление нового объекта в БД |
| 5 | Controllers | Delete(int id) | Удаление объекта из БД |
| 6 | CybersportContext | OnConfiguring() | Настройка класса контекста |
| 7 | CybersportContext | OnModelCreating() | Создание моделей и связей между ними |



### Модель данных

Схема данных программы представлена на рисунке 15.

Словарь данных представлен в таблице 10.

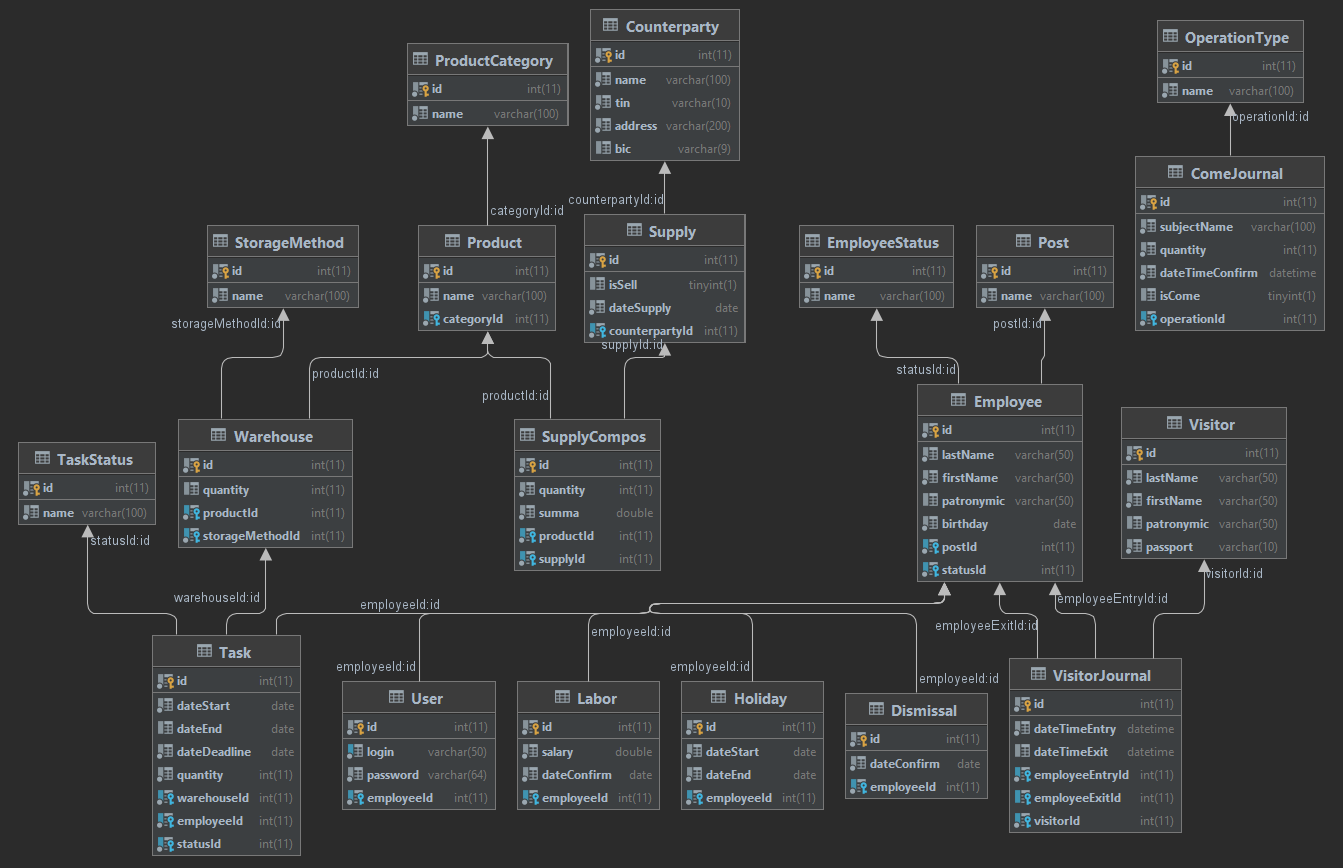


Рисунок 13 – Модель данных

Таблица 10 – Словарь данных

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ключ | Наименование | Тип данных | Ограничения | Null-значение | Описание |
| 1 | 2 | 3 | 4 | 5 | 6 |
| StorageMethod | | | | | |
| PK | id | Int |  | Not null | Первичный ключ |
|  | name | Varchar | A-z, [0-100] |  | Наименование способа хранения |
| ProductCategory | | | | | |
| PK | id | Int |  | Not null | Первичный ключ |
|  | name | Varchar | A-z, [0-100] |  | Наименование категории товара |
| Post | | | | | |
| PK | id | Int |  | Not null | Первичный ключ |
|  | name | Varchar | A-z, [0-100] |  | Наименование |
| OperationTime | | | | | |
| PK | id | Int |  | Not null | Первичный ключ |
|  | name | Varchar | A-z, [0-100] |  | Наименование операции |
| TaskStatus | | | | | |
| PK | id | Int |  | Not null | Первичный ключ |
|  | name | Varchar | A-z, [0-100] |  | Наименование статуса задачи |
| EmployeeStatus | | | | | |
| PK | id | Int |  | Not null | Первичный ключ |
|  | name | Varchar | A-z, [0-100] |  | Наименование статуса сотрудника |
| ComeJournal | | | | | |
| PK | id | int(11) | [0-9]- | Not null | Первичный ключ |
|  | subjectName | varchar(100) | A-z | Not null |  |
|  | quantity | int(11) | [0-9] | Not null |  |
|  | dateTimeConfirm | datetime | [0-9]. | Not null |  |
|  | isCome | tinyint(1) | [0-9] | Null |  |
| FK | operationId | int(11) | [0-9] | Not null | Внешний ключ |
| Counterparty | | | | | |
| PK | id | int(11) | - | Not null | Первичный ключ |
|  | name | varchar(100) | - | Not null |  |
|  | tin | varchar(10) | - | Not null |  |
|  | address | varchar(200) | - | Not null |  |
|  | bic | varchar(9) |  | Null |  |
| Dismissal | | | | | |
| PK | id | int(11) | - | Not null | Первичный ключ |
|  | dateConfirm | date | - | Not null |  |
| FK | employeeId | int(11) | - | Not null | Внешний ключ |
| Employee | | | | | |
| PK | id | int(11) | - | Not null | Первичный ключ |
|  | lastName | varchar(50) | - | Not null |  |
|  | firstName | varchar(50) | - | Not null |  |
|  | patronymic | varchar(50) | - | Null |  |
|  | birthday | date | - | Not null |  |
| FK | postId | int(11) | - | Not null | Внешний ключ |
| FK | statusId | int(11) | - | Not null | Внешний ключ |
| Holiday | | | | | |
| PK | id | int(11) | - | Not null | Первичный ключ |
|  | dateStart | date | - | Not null |  |
|  | dateEnd | date | - | Not null |  |
| FK | employeeId | int(11) |  | Not null | Внешний ключ |
| Labor | | | | | |
| PK | id | int(11) | - | Not null | Первичный ключ |
|  | salary | double | - | Not null |  |
|  | dateConfirm | date | - | Not null |  |
| FK | employeeId | int(11) | - | Not null | Внешний ключ |
| Product | | | | | |
| PK | id | int(11) | - | Not null | Первичный ключ |
|  | name | varchar(100) | - | Not null |  |
| FK | categoryId | int(11) | - | Not null | Внешний ключ |
| SupplyCompos | | | | | |
| PK | id | int(11) | - | Not null | Первичный ключ |
|  | quantity | int(11) | - | Not null |  |
|  | summa | double | - | Not null |  |
| FK | productId | int(11) | - | Not null | Внешний ключ |
| FK | supplyId | int(11) | - | Not null | Внешний ключ |
| Supply | | | | | |
| PK | id | int(11) | - | Not null | Первичный ключ |
|  | isSell | tinyint(1) | - | Null |  |
|  | dateSupply | date | - | Not null |  |
| FK | counterpartyId | int(11) | - | Not null | Внешний ключ |
| Task | | | | | |
| PK | id | int(11) | - | Not null | Первичный ключ |
|  | dateStart | date | - | Not null |  |
|  | dateEnd | date | - | Null |  |
|  | dateDeadline | date | - | Not null |  |
|  | quantity | int(11) | - | Not null |  |
| FK | warehouseId | int(11) | - | Not null | Внешний ключ |
| FK | employeeId | int(11) | - | Not null | Внешний ключ |
| FK | statusId | int(11) | - | Not null | Внешний ключ |
| User | | | | | |
| PK | id | int(11) | - | Not null | Первичный ключ |
|  | login | varchar(50) | - | Not null |  |
|  | password | varchar(64) | - | Not null |  |
| FK | employeeId | int(11) | - | Not null | Внешний ключ |
| Visitor | | | | | |
| PK | id | int(11) | - | Not null | Первичный ключ |
|  | lastName | varchar(50) | - | Not null |  |
|  | firstName | varchar(50) | - | Not null |  |
|  | patronymic | varchar(50) | - | Null |  |
|  | passport | varchar(10) | - | Not null |  |
| VisitorJournal | | | | | |
| PK | id | int(11) | - | Not null | Первичный ключ |
|  | dateTimeEntry | datetime | - | Not null |  |
|  | dateTimeExit | datetime | - | Null |  |
| FK | employeeEntryId | int(11) | - | Not null | Внешний ключ |
| FK | employeeExitId | int(11) | - | Null | Внешний ключ |
| FK | visitorId | int(11) | - | Not null | Внешний ключ |
| Warehouse | | | | | |
| PK | id | int(11) | - | Not null | Первичный ключ |
|  | quantity | int(11) | - | Null |  |
| FK | productId | int(11) | - | Not null | Внешний ключ |
| FK | storageMethodId | int(11) |  | Not null | Внешний ключ |

### Схема пользовательского интерфейса

Схема пользовательского интерфейса информационной системы представлена на рисунке 14. На ней графически продемонстрированы все окна и переходы между ними.

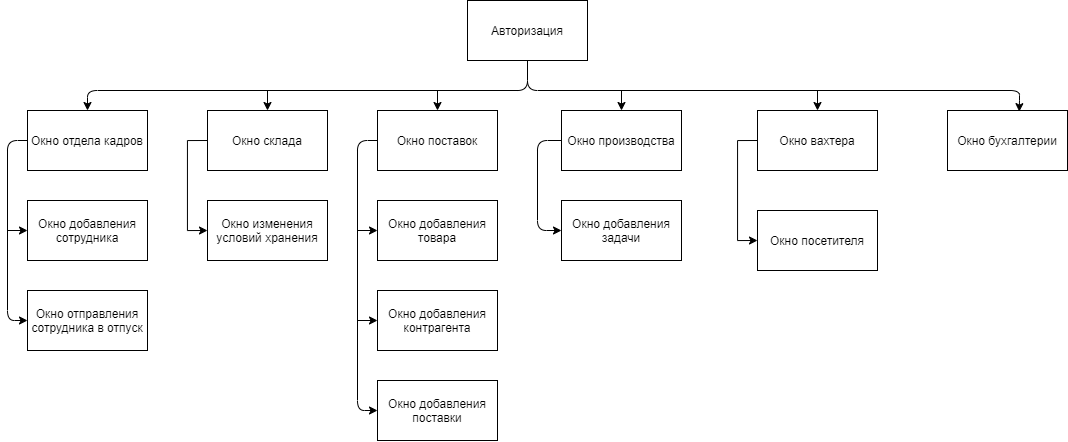


Рисунок 14 – Схема пользовательского интерфейса

* 1. Результаты работы программы

В результате выполнения поставленной задачи, было разработано требуемое программное решение, которое является информационной системой мебельного производства.

Данное решение является клиент-серверным с возможностью распределения функциональных возможностей у каждого пользователя системы.

Информационная система состоит из трех основных частей, а именно:

* Из реляционной базы данных, которая размещается на сервере, рассмотреть состав и структуру которой можно в пункте 2.3.1;
* Из настольного приложения, устанавливаемого на персональный компьютер;
* Из Web-API, размещаемого на сервере. Описание API можно посмотреть в пункте 2.3.3.

Узнать о взаимодействии пользователя с информационной системой можно из Приложения В. Руководство пользователя.

Основные функциональные возможности продемонстрированы на Рисунках 16-20. На Рисунке 15 изображена страница авторизации.

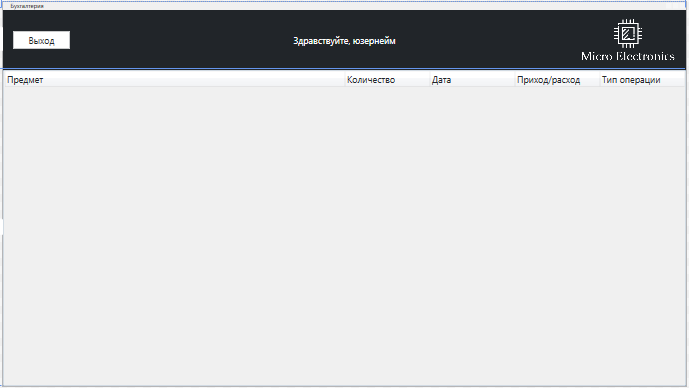
Рисунок 15 - Авторизация

Рисунок 16 – Окно бухгалтера

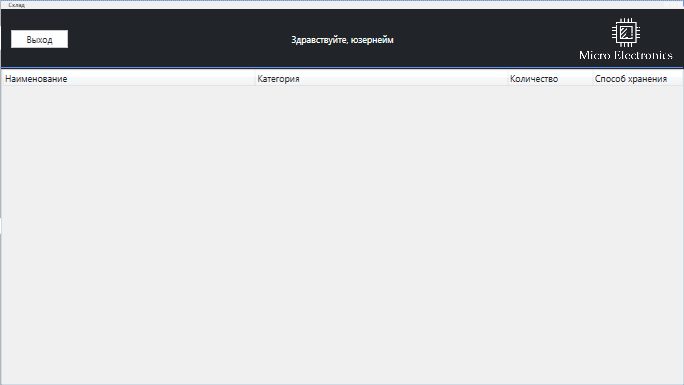


Рисунок 17 – Окно кладовщика

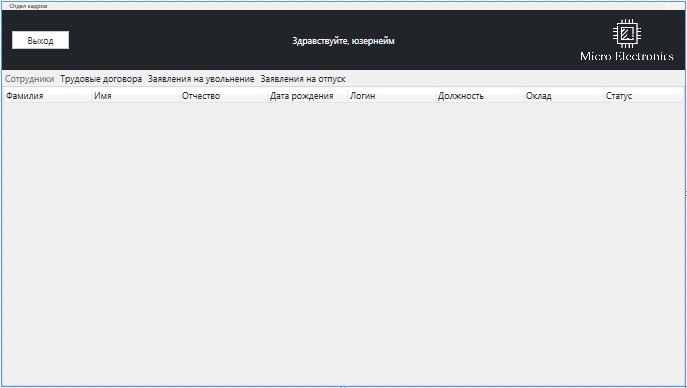


Рисунок 18 – Окно hr

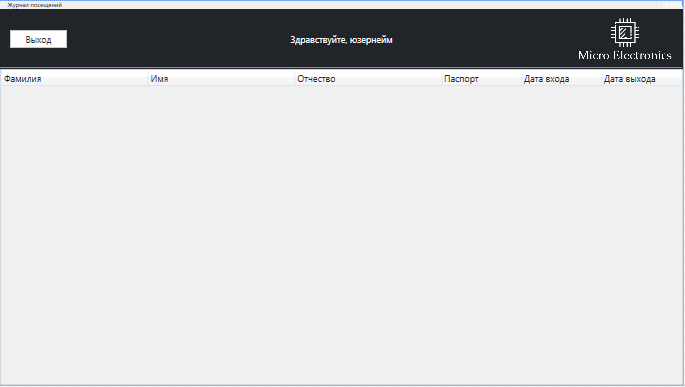


Рисунок 19 – Окно вахтера

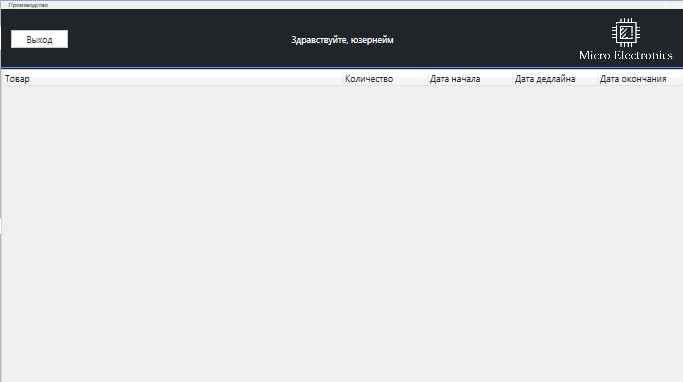


Рисунок 20 – Окно начальника производства

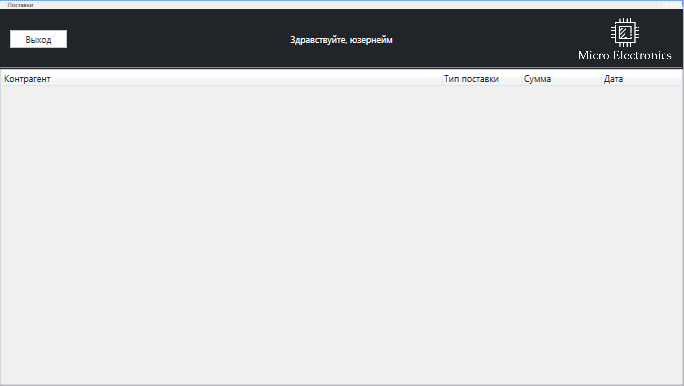


Рисунок – Окно менеджера поставок

1. ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ
   1. Инструментальные средства разработки

Для разработки приложения использовалась среда разработки Microsoft Visual Studio 2019, которая позволяет писать код на языке C# на платформе WPF Преимуществами данной среды разработки является наличие хорошего графического интерфейса, функции автодополнения кода и удобного инструментария отладки.

Для работы с документацией использовался текстовый редактор Microsoft Word 2016, предназначенный для создания, просмотра и редактирования текстовых документов, с локальным применением простейших форм таблично-матричных алгоритмов. Выпускается корпорацией Microsoft в составе пакета Microsoft Office. Его преимуществами являются большая функциональность и удобный интерфейс.

Для работы с базой данных была использована СУБД Microsoft SQL Server. Данное программное решение позволяет отслеживать все изменения, которые происходят при взаимодействии с базой данных. В качестве языка запросов использовался MySQL, обладающий удобным набором инструкций и высокой производительностью.

Для создания схем использовался векторный графический редактор «Draw.io», редактор диаграмм и блок-схем для различных операционных систем.

В качестве операционной системы использовалась операционная система Microsoft Windows 10.

* 1. Отладка программы

Отладка информационной системы осуществлялась с помощью встроенного отладчика среды разработки MS Visual Studio.

При незнании способа или решения исправления ошибки, которая возникала при разработке информационной системы и API, использовалась дополнительная информация, взятая из специализированной литературы, профильных сайтов и официальных мануалов. С открытыми источниками и литературой можно ознакомится в разделе «Список используемых материалов».

Пример ошибок:

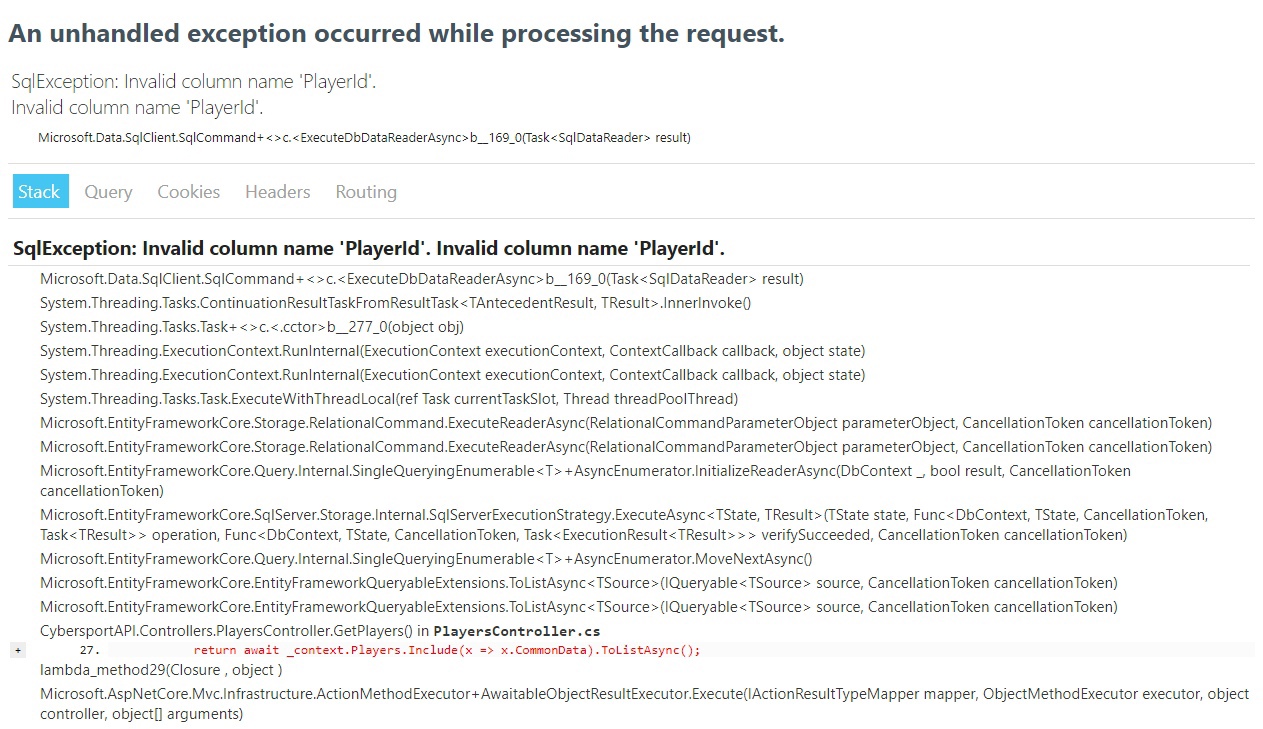


Рисунок 22 – Ошибка связи в моделях (попытка привязять вторую таблицу при связи один-к-одному)

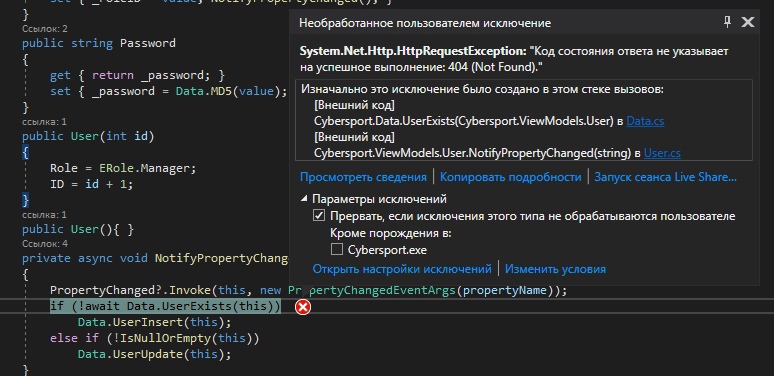


Рисунок 23 – Ошибка поиска данных в API

* 1. Защитное программирование

Особое защитное программирование не применялось при разработке информационной системы.

Для защиты личных данных пользователя его пароль автоматически хешируется с помощью метода MD5. Готовый хеш пароля записывается в базу данных.

При редактировании пароля администратором, пароль также автоматически зашифровывается и новый хеш записывается в БД.



Рисунок 24 – Редактирование пароля



Рисунок 25 – Завершение редактирования и автоматическое хеширование

* 1. Характеристика программы

### В таблице 11 представлены модели информационной системы.

Таблица 11 – Модели информационной системы

| № | Название модуля | Описание модуля | Размер модуля | Кол-во строк |
| --- | --- | --- | --- | --- |
| 1 | ComeJournalData.cs | Модель | 1 кб | 150 |
| 2 | CounterpartyData.cs | Модель | 1 кб | 126 |
| 3 | DismissData.cs | Модель | 1 кб | 44 |
| 4 | HolidayData.cs | Модель | 1 кб | 95 |
| 5 | LaborData.cs | Модель | 1 кб | 121 |
| 6 | ManufData.cs | Модель | 1 кб | 79 |
| 7 | ProductData.cs | Модель | 1 кб | 71 |
| 8 | ResponseMessageData.cs | Модель | 1 кб | 38 |
| 9 | SupplyData.cs | Модель | 1 кб | 65 |
| 10 | UserData.cs | Модель | 1 кб | 119 |
| 11 | VisitorData.cs | Модель | 1 кб | 142 |

### В таблице 12 представлены модели Web-API.

Таблица 12 – Модели Web-API

| № | Название модуля | Описание модуля | Размер модуля | Кол-во строк |
| --- | --- | --- | --- | --- |
| 1 | ComeJournal.cs | Модель | 1 кб | 21 |
| 2 | Counterparty.cs | Модель | 1 кб | 21 |
| 3 | Dismissal.cs | Модель | 10 кб | 283 |
| 4 | Employee.cs | Модель | 1 кб | 18 |
| 5 | EmployeeStatus.cs | Модель | 1 кб | 18 |
| 6 | Holiday.cs | Модель | 1 кб | 17 |
| 7 | Labor.cs | Модель | 1 кб | 21 |
| 8 | MicroElectronsDBContext.cs | Класс контекста данных | 1 кб | 18 |
| 9 | OperationType.cs | Модель | 1 кб | 18 |
| 10 | Post.cs | Модель | 1 кб | 17 |
| 11 | Product.cs | Модель | 1 кб | 18 |
| 12 | ProductCategory.cs | Модель | 1 кб | 19 |
| 13 | StorageMethod.cs | Модель | 1 кб | 18 |
| 14 | Supply.cs | Модель | 1 кб | 18 |
| 15 | SupplyCompo.cs | Модель | 1 кб | 17 |
| 16 | Task.cs | Модель | 1 кб | 18 |
| 17 | TaskStatus.cs | Модель | 1 кб | 19 |
| 18 | User.cs | Модель | 1 кб | 18 |
| 19 | Visitor.cs | Модель | 1 кб | 18 |
| 20 | VisitorJournal.cs | Модель | 1 кб | 17 |
| 21 | Warehouse.cs | Модель | 1 кб | 18 |

### В таблице 13 представлены контроллеры Web-API.

Таблица 13 – Контроллеры Web-API

| № | Название модуля | Описание модуля | Размер модуля | Кол-во строк |
| --- | --- | --- | --- | --- |
| 1 | ComeJournalController.cs | Контроллер | 2 кб | 52 |
| 2 | ManufController.cs | Контроллер | 5 кб | 118 |
| 3 | ProductController.cs | Контроллер | 7 кб | 200 |
| 4 | SupplyController.cs | Контроллер | 6 кб | 156 |
| 5 | UserController.cs | Контроллер | 14 кб | 356 |
| 6 | VisitorController.cs | Контроллер | 6 кб | 132 |

ЗАКЛЮЧЕНИЕ

В изложенном курсовом проекте была изучена актуальная проблема в области автоматизации и оптимизации бизнес-процессов производства микроэлектроники. Основным аспектом в рассматриваемой области является организация всего процесса и разграничение прав доступа.

В ходе решения поставленной задачи была разработана информационная система организации по производству микроэлектроники. Данное решения состоит из трех частей, а именно из реляционной базы данных, настольного приложения и Web-API. База данных была реализована для хранения информации, приложение для удобного представления информации в виде интуитивно понятных функций, а Web-API для связи базы данных и приложения.

При выполнении курсового проекта были закреплены навыки программирования в объектно-ориентированном фреймворке Microsoft Visual Studio на базе платформы WPF, которая реализует архитектурный паттерн MVС (Model-View-Controller). Были закреплены и улучшены знания в области написания программных решений для персональных компьютеров с использованием объектно-ориентированного языка C#, а также языка запросов MySQL для создания базы данных.

После оценки итогов и полученного результата работы программного решения, был сделан вывод о целесообразности использования такого подхода для автоматизации бизнес-процессов производства микроэлектроники. Данное решение позволит сформировать единую систему, в которой взаимодействуют пользователи с разными ролями, возможностями и методами их решения.

СПИСОК ИСПОЛЬЗОВАННЫХ МАТЕРИАЛОВ

1) ГОСТ 19404- 79 ЕСПД. Пояснительная записка. ПЕРЕИЗДАНИЕ Января 2010 г. – С. 1 – 3.

2) ГОСТ 7.80-2000 СИБИД. Библиографическая запись. Заголовок. Общие требования и правила составления. - С. 4 – 5.

3) ГОСТ 19.101-77 ЕСПД. Виды программ и программных документов. С. 1 – 3.

4) ГОСТ 19.103-77 ЕСПД. Обозначение программ и программных документов. С. 1 – 3.

5) ГОСТ 19.105-78 ЕСПД. Общие требования к программным документам. С. 2 – 4.

6) Metanit.com. Паттерн MVC. URL: <https://metanit.com/sharp/wpf/22.1.php>

Дата обращения: 10.11.2021 г.

7) Metanit.com. Команды в MVC. URL: <https://metanit.com/sharp/wpf/22.3.php> Дата обращения: 11.11.2021 г.

8) Эндрю Троелсен, Филипп Джепикс «Язык программирования C# 7 и платформы .NET и .NET Core», 8-е издание, 2018 год – 573с. Дата обращения: 29.11.2021 г.

9) Документация для библиотеки «Material Design», URL: <https://material.io/components/dialogs>