**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра САПР**

Курсовая РАБОТА

**по дисциплине «Технологии разработки программного обеспечения»**

Тема: Информационно-справочная система программного обеспечения САПР

|  |  |  |
| --- | --- | --- |
| Студент гр. 7302 |  | Литвинов К.Л. |
| Студентка гр. 7302 |  | Петрова Е.М. |
| Студент гр. 7302 |  | Перков Д.. |
| Студент гр. 7302 |  | Стафеев А.А. |
| Преподаватель |  | Новакова Н.Е. |

Санкт-Петербург

2022

**ЗАДАНИЕ**

**на курсовую работу**

|  |  |  |
| --- | --- | --- |
| Студент Литвинов К. Л. | | |
| Группа 6302 | | |
| Тема работы: Информационно-справочная система программного обеспечения САПР | | |
| Исходные данные:  Задание на курсовую работу, язык программирования C#, язык T-SQL, фреймворк Entity Framewor, WPF. | | |
| Содержание пояснительной записки:  «Аннотация», «Содержание», «Введение», «Постановка задачи», «Обоснование выбора основных решений», «Разработка структур данных и основных решений», «Обоснование выбора окончательного решения», «Заключение», «Список использованных источников», «Приложения». | | |
| Дата сдачи курсовой работы: | | |
| Дата защиты курсовой работы: | | |
| Студент |  | Литвинов К. Л. |
| Преподаватель |  | Новакова Н. Е. |

**Аннотация**

В курсовой работе рассматривается командная работа над проектированием и разработкой информационной системы на платформе ASP .NET Framework на языке C# в среде разработки Visual Studio 2019 с использованием в качестве системы управления базами данных Microsoft SQL Server 2019 Developer. Приложение предназначено для управления информацией о продукте на предприятии.

**Summary**

The course work examines teamwork on the design and development of an information system on the ASP .NET Framework platform in C # in the Visual Studio 2019 development environment using Microsoft SQL Server 2019 Developer as a database management system. The application is designed to manage product information in an enterprise.

# 2. обоснование выбора основных решений

## 2.1. Функциональное описание системы

Программа должна обеспечивать следующую функциональность:

* Просмотр информации о программном обеспечение САПР
* Редактирование информации о программном обеспечение САПР
* Создание новой информации о программном обеспечение САПР
* Удаление информации о программном обеспечение САПР

Система должна запускаться на компьютере, на котором установлены:

* .NET 6
* Windows 7 или выше

Технические характеристики компьютера, на котором запускается программа, должны соответствовать минимальным техническим характеристикам для операционной системы Windows 7

**2.2. Формирование предложений по проектируемой системе**

Было принято решение реализовывать информационно-справочная система программного обеспечения САПР в виде настольного приложения по следующим причинам:

1. Возможность использовать приложение без надобности подключения к интернету.
2. Высокая производительность, так как само приложение не требует много ресурсов, и всё содержимое отображается на нативной платформе.

В качестве языка программирования был выбран язык С# с использованием платформы .NET. Данный выбор позволит писать высококачественную объектно-ориентированную структуру без дополнительных нагрузок. Также для этого языка присутствует фреймворк Entity Framework для взаимодействия с базой данных. Он значительно упрощает запросы к БД и интеграцию её в приложение.

В качестве базы данных была выбрана SQL Server 2019. Данный выбор обусловлен тем, что данная БД полностью покрывает все необходимые потребности в разработке системы.

**2.3. Описание архитектуры системы**

При разработке системы использовался паттерн проектирования MVVM. Он является нативным для систем, разработанных при помощи фреймворка WPF. Он предусматривает разделение бизнес-логики приложения, внешнего вида, и моделей, которые используются в системе.

В системе так же разделены уровни самого настольного приложения и процесса взаимодействия с базой данных. Такое разделение позволяет в дальнейшем переиспользовать компоненты системы для другой разработки. К примеру, к смене базы данных или созданию приложения для других операционных систем. Каждый уровень представляет собой отдельный проект, разработанный на языке программирования C#.

**2.4. Принятие основных решений по безопасности и отказоустойчивости системы**

Обеспечение безопасности системы состоит в применении следующих решений:

1. Сокрытие исходного программа от всех, кто не является разработчиком на проекте
2. Использование локальной базы данных для каждого при разработке
3. Сокрытие ключей доступа к базе данных

Отказоустойчивость системы обеспечивается следующими мероприятиями:

1. Использование резервных источников питания для работоспособности сервера.

2. Переключение на дублирующий сервер в случае возникновения неисправностей на основном.

3. Своевременной резервное копирование БД.

4. Своевременное резервное копирование документации с основного сервера.

**3. Разработка структур данных и основных решений**

**3.1. Разработка структур данных основных компонентов системы**

**3.1.1. Описание классов**

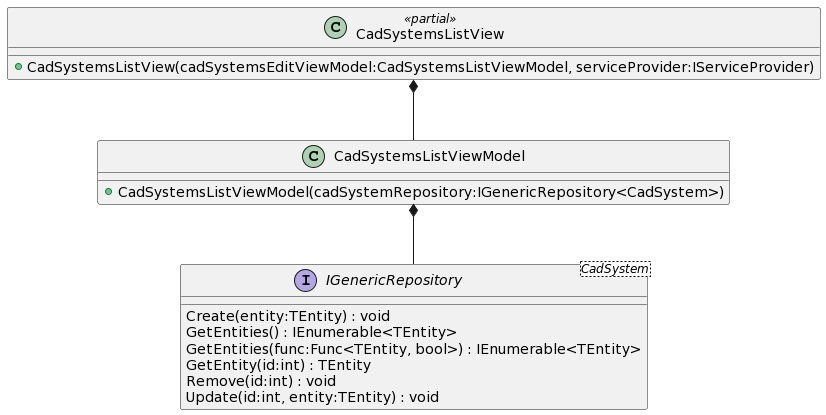
В проекте используется фреймворк Entity Framework. Благодаря нему можно организовать доступ редактирования моделей базы данных. Так как операции в большинстве случаев одинаковые, то было решено создать типовой класс для взаимодействия с базой данных.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 9 – диаграмма классов сервиса авторизации

Так как для форм WPF используется типовой паттерн MVVM, то и связь классов для каждой формы будет примерно одинаковой. Рассмотрим на примере CadSystemsList.



View имеет связь через композицию с ViewModel, а она в свою очередь имеет такую же связь с репозиторием для взаимодействия с данными базы данных.

**3.1.2. Описание структур данных в БД**

На рисунке 12 представлена ER-диаграмма, которая описывает связи между выделенными ранее сущностями.

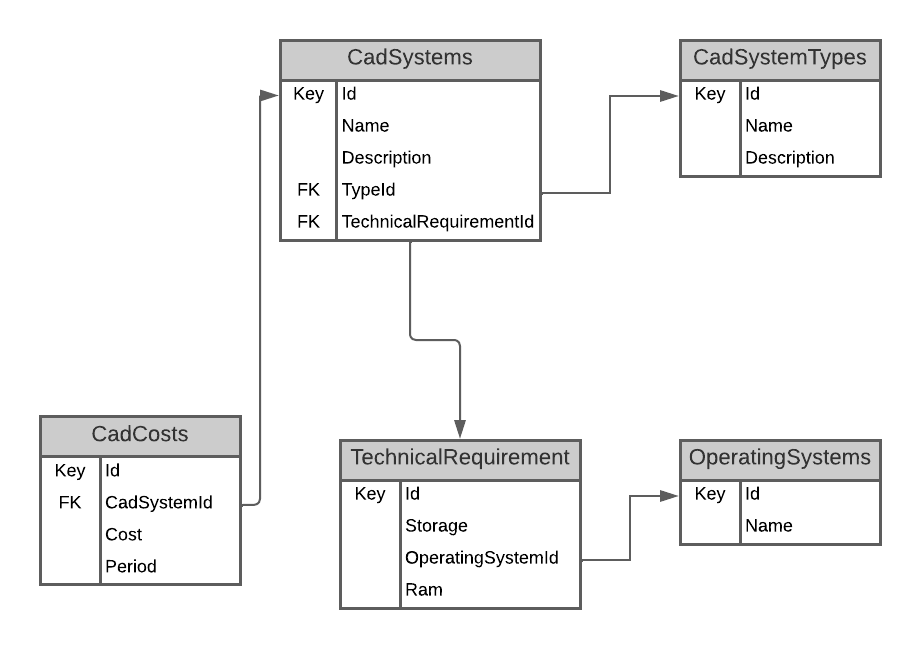


Рисунок 12 – ER-диаграмма базы данных

Далее в таблицах представлено описание структур таблиц БД.

Таблица 3. Структура CadSystems

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Описание структуры таблицы БД** | | | Наименование таблицы БД:  **Таблица Информации о системах САПР** | | | | Имя таблицы:  CadSystems | | | |
| Дата разработки:  25.05.2022 | | |
| Порядковый номер таблицы:  **1** | | |  | | | |
|  |  |  | |  | |  |  | |  |  |
|  | № п/п | Наименование поля | | Спецификация данных | | | | | |  |
|  | Имя поля | Тип данных | | Ключ | Ограничения целостности | |  |
|  | 1 | Идентификатор | | Id | int | | P |  | |  |
|  | 2 | Имя | | Name | nvarchar(10) | |  | not null | |  |
|  | 3 | Описание | | Description | nvarchar(100) | |  | not null | |  |
|  | 4 | Тип системы | | TypeId | int | | F | not null | |  |
|  | 5 | Технические требования системы | | TechnicalRequirementId | int | | F | not null | |  |
|  |  |  | |  |  | |  |  | |  |

Таблица 4. Структура CadSystemTypes

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Описание структуры таблицы БД** | | | Наименование таблицы БД:  **Таблица типы систем САПР** | | | | Имя таблицы:  CadSystemTypes | | | |
| Дата разработки:  25.05.2022 | | |
| Порядковый номер таблицы:  **2** | | |  | | | |
|  |  |  | |  | |  |  | |  |  |
|  | № п/п | Наименование поля | | Спецификация данных | | | | | |  |
|  | Имя поля | Тип данных | | Ключ | Ограничения целостности | |  |
|  | 1 | Идентификатор | | Id | int | | P |  | |  |
|  | 2 | Имя | | Name | nvarchar(10) | |  | not null | |  |
|  | 3 | Описание | | Description | nvarchar(100) | |  | not null | |  |
|  |  |  | |  |  | |  |  | |  |

Таблица 4. Структура TechnicalRequirement

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Описание структуры таблицы БД** | | | Наименование таблицы БД:  **Таблица технические требования систем САПР** | | | | Имя таблицы:  TechnicalRequirement | | | |
| Дата разработки:  25.05.2022 | | |
| Порядковый номер таблицы:  **3** | | |  | | | |
|  |  |  | |  | |  |  | |  |  |
|  | № п/п | Наименование поля | | Спецификация данных | | | | | |  |
|  | Имя поля | Тип данных | | Ключ | Ограничения целостности | |  |
|  | 1 | Идентификатор | | Id | Int | | P |  | |  |
|  | 2 | Размер постоянной памяти | | Storage | Int | |  | not null | |  |
|  | 3 | Операционная система | | OperatingSystemId | int | | F | not null | |  |
|  | 4 | Размер оперативной памяти | | Ram | int | |  | not null | |  |
|  |  |  | |  |  | |  |  | |  |

Таблица 4. Структура OperatingSystems

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Описание структуры таблицы БД** | | | Наименование таблицы БД:  **Таблица технические требования систем САПР** | | | | Имя таблицы:  **OperatingSystems** | | | |
| Дата разработки:  25.05.2022 | | |
| Порядковый номер таблицы:  **4** | | |  | | | |
|  |  |  | |  | |  |  | |  |  |
|  | № п/п | Наименование поля | | Спецификация данных | | | | | |  |
|  | Имя поля | Тип данных | | Ключ | Ограничения целостности | |  |
|  | 1 | Идентификатор | | Id | Int | | P |  | |  |
|  | 2 | Имя | | Name | nvarchar(10) | |  | not null | |  |
|  |  |  | |  |  | |  |  | |  |

Таблица 4. Структура CadCosts

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Описание структуры таблицы БД** | | | Наименование таблицы БД:  **Таблица технические требования систем САПР** | | | | Имя таблицы:  **CadCosts** | | | |
| Дата разработки:  25.05.2022 | | |
| Порядковый номер таблицы:  **55** | | |  | | | |
|  |  |  | |  | |  |  | |  |  |
|  | № п/п | Наименование поля | | Спецификация данных | | | | | |  |
|  | Имя поля | Тип данных | | Ключ | Ограничения целостности | |  |
|  | 1 | Идентификатор | | Id | int | | P |  | |  |
|  | 2 | Имя | | CadSystemId | int | | F |  | |  |
|  | 3 | Стоимость | | Cost | double | |  | not null | |  |
|  | 4 | Период оплаты | | Period | nvarchar(100) | |  | not null | |  |
|  |  |  | |  |  | |  |  | |  |

**3.2. Описание решений по реализации бизнес-логики системы**

Для описания бизнес-процессов, реализованных в разрабатываемой системе, были выбраны два типа UML-диаграмм: диаграмма состояний и диаграмма прецедентов.

На рисунке 13 представлена диаграмма состояний для сущности Документ в системе. Данная диаграмма позволяет описать бизнес-процесс работы пользователей с файлами.

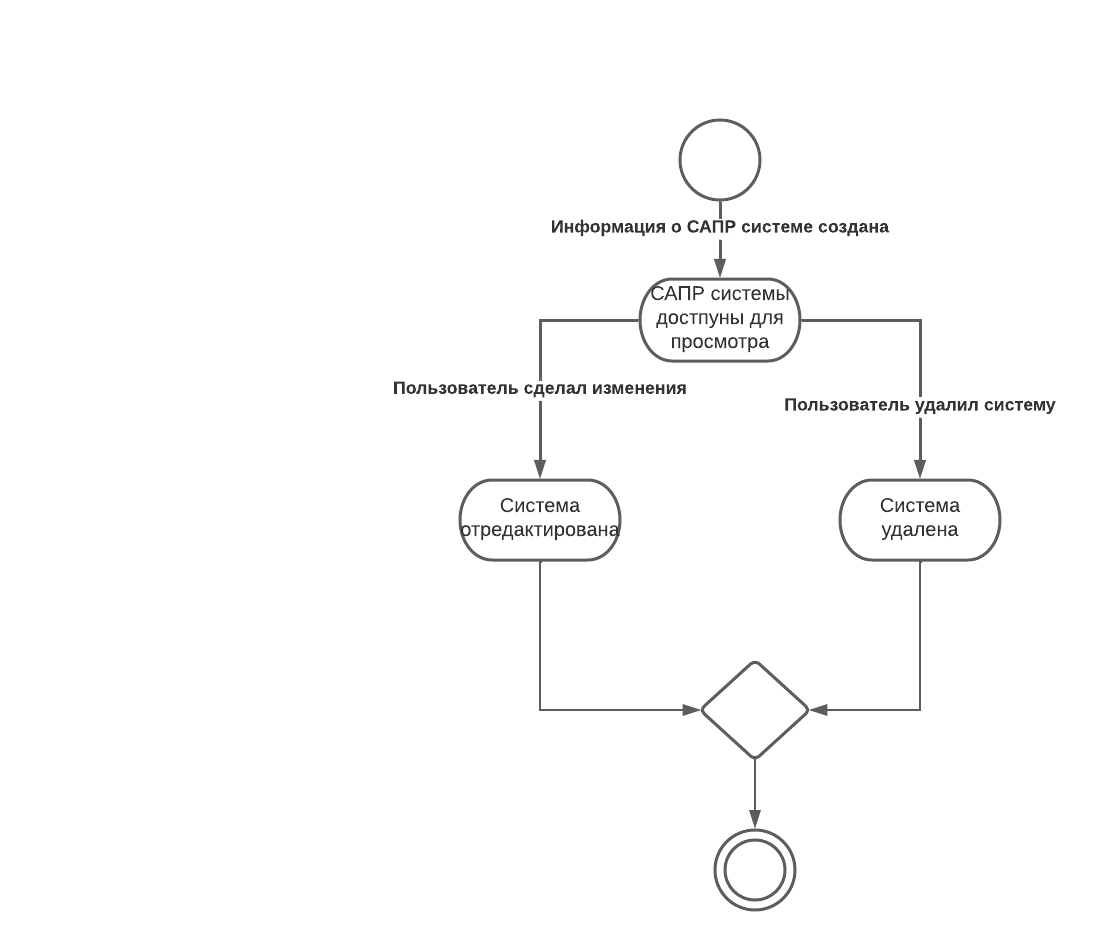


Рисунок 13 – Диаграмма состояний сущности Документ

По диаграмме видно, что у системы может быть три состояния. Редактирование, Просмотр, и удаление.

На рисунке 14 представлена диаграмма прецедентов, которая позволяет описать роли пользователей с точки зрения их возможностей внутри системы.

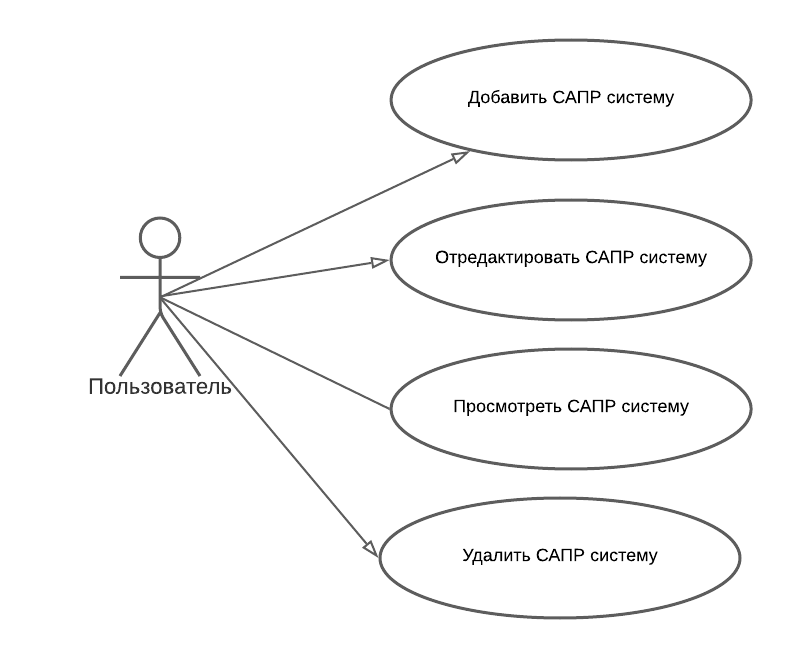


Рисунок 14 – Диаграмма прецедентов

По этой диаграмме видно, что пользователя есть 4 варианта взаимодействия с системой.

Большая часть бизнес-логики реализована в коде с использованием Entity Framework.

**3.3. Описание решений по организации тестирования системы**

При разработке системы была определена следующий подход к тестированию:

1. Разработчики приложения покрывают код модульными тестами, чтобы снизить вероятность возникновения ошибок при внесении изменений в проект. После внесения изменений в проект необходимо убедиться, что все модульные тесты проходят успешно.

2. Тестировщики составляют тестовый план и занимаются написанием тест-кейсов.

В таблицах 17 – 25 представлены тест-кейсы, разработанные тестировщиками.

Таблица 17. Тест-кейс № 1. Позитивное тестирование авторизации

|  |  |
| --- | --- |
| Номер тест-кейса | 1 |
| Заголовок | Авторизация пользователя через форму для входа |
| Предусловие | Аккаунт пользователя уже был создан ролью Администратор |
| **Шаг** | **Ожидаемый результат** |
| В правом вернем углу нажать на ссылку Login | Открывается страница авторизации с формой для ввода пароля и адреса эл. почты |
| Вводится корректное значение в поле Логин и корректное значение в поле Пароль. Нажать кнопку Войти | Отображается введенный логин и пароль (пароль отображается в виде звездочек). Авторизация проходит успешно. Пользователь перенаправляется на главную страницу в зависимости от его роли |

Таблица 18. Тест-кейс № 2. Негативное тестирование авторизации

|  |  |
| --- | --- |
| Номер тест-кейса | 2 |
| Заголовок | Авторизация пользователя через форму для входа |
| Предусловие | Аккаунт пользователя уже был создан ролью Администратор |
| **Шаг** | **Ожидаемый результат** |
| В правом вернем углу нажать на ссылку Login | Открывается страница авторизации с формой для ввода пароля и адреса эл. почты |
| Вводится некорректное значения в поля Логин и Пароль. Нажать кнопку Войти | Отображается введенный логин и пароль (пароль отображается в виде звездочек). Авторизация не проходит. Пользователь получает уведомление «Неверный логин или пароль» |

Таблица 19. Тест-кейс № 3. Тестирование предоставление прав к проектам

|  |  |
| --- | --- |
| Номер тест-кейса | 3 |
| Заголовок | Предоставление прав к проектам для пользователей |
| Предусловие | Пользователь авторизован под ролью Администратор |
| **Шаг** | **Ожидаемый результат** |
| В правом вернем углу нажать на ссылку Настройка прав доступа | Открывается страница со списком пользователей |
| Нажать на эл. адрес одного из пользователей | Отображается страница с доступами пользователя к проектам |
| Выбрать соответствующие права для каждого из проектов. Нажать кнопку Сохранить изменения | Происходит сохранение измененных прав доступа. Пользователь перенаправляется на страницу Настройка прав доступа |

Таблица 20. Тест-кейс № 4. Позитивное тестирование регистрации нового пользователя

|  |  |
| --- | --- |
| Номер тест-кейса | 4 |
| Заголовок | Регистрация нового пользователя |
| Предусловие | Пользователь авторизован под ролью Администратор |
| **Шаг** | **Ожидаемый результат** |
| В правом вернем углу нажать на ссылку Регистрация нового пользователя | Открывается страница с формой регистрации нового пользователя |
| Заполнить поля Email, Имя, Фамилия, Пароль, Выберите роль. Нажать кнопку Зарегистрировать | Происходит создание пользователя в БД. Пользователя перенаправляет на страницу Настройка прав доступа. |

Таблица 21. Тест-кейс № 5. Негативное тестирование регистрации нового пользователя

|  |  |
| --- | --- |
| Номер тест-кейса | 5 |
| Заголовок | Регистрация нового пользователя |
| Предусловие | Пользователь авторизован под ролью Администратор |
| **Шаг** | **Ожидаемый результат** |
| В правом вернем углу нажать на ссылку Регистрация нового пользователя | Открывается страница с формой регистрации нового пользователя |
| Заполнить поле Email уже существующим существующим значением из БД, Имя, Фамилия, Пароль, Выберите роль. Нажать кнопку Зарегистрировать | Пользователь получает сообщение об ошибке «Регистрация завершилась неуспехом. Пользователь с таким Email уже существует» |
| Заполняются не все поля из формы | Пользователь получает сообщение об ошибке «Заполните это поле» |

Таблица 22. Тест-кейс № 6. Тестирование скачивания файла последней версии

|  |  |
| --- | --- |
| Номер тест-кейса | 6 |
| Заголовок | Скачивание файла ролью Конструктор |
| Предусловие | Пользователь авторизован под ролью Конструктор |
| **Шаг** | **Ожидаемый результат** |
| В меню выбирается ссылка Мои проекты | Открывается страница с проектами пользователя |
| Выбрать проект. Нажать на ссылку проекта | Пользователь переходит на страницу с документацией по проекту |
| Выбрать файл доступный для скачивания. Нажать кнопку Скачать | Документ скачивается и попадает в загрузки бразузера |

Таблица 23. Тест-кейс № 7. Тестирование перевода файла на редактирование

|  |  |
| --- | --- |
| Номер тест-кейса | 7 |
| Заголовок | Перевод файла на редактирование ролью Конструктор |
| Предусловие | Пользователь авторизован под ролью Конструктор |
| **Шаг** | **Ожидаемый результат** |
| В меню выбирается ссылка Мои проекты | Открывается страница с проектами пользователя |
| Выбрать проект. Нажать на ссылку проекта | Пользователь переходит на страницу с документацией по проекту |
| Выбрать файл доступный для редактирования. Нажать кнопку Редактировать | Документ скачивается и попадает в загрузки браузера. Статус документа на странице изменяется с Доступен для «редактирования» на «На редактировании» |

Таблица 24. Тест-кейс № 8. Тестирование отмены редактирования

|  |  |
| --- | --- |
| Номер тест-кейса | 8 |
| Заголовок | Отмена редактирования ролью Конструктор |
| Предусловие | Пользователь авторизован под ролью Конструктор |
| **Шаг** | **Ожидаемый результат** |
| В меню выбирается ссылка Мои проекты | Открывается страница с проектами пользователя |
| Выбрать проект. Нажать на ссылку проекта | Пользователь переходит на страницу с документацией по проекту |
| Выбрать файл, находящийся на редактировании текущим пользователем. Нажать кнопку «Отменить» | Статус документа на странице изменяется с  «На редактировании» на «Доступен для редактирования» |

Таблица 25. Тест-кейс № 9. Тестирование скачивания предыдущих версий файла

|  |  |
| --- | --- |
| Номер тест-кейса | 9 |
| Заголовок | Скачивание прошлых версий файла ролью Конструктор |
| Предусловие | Пользователь авторизован под ролью Конструктор |
| **Шаг** | **Ожидаемый результат** |
| В меню выбирается ссылка Мои проекты | Открывается страница с проектами пользователя |
| Выбрать проект. Нажать на ссылку проекта | Пользователь переходит на страницу с документацией по проекту |
| Выбрать файл. Нажать на ссылку с названием файла | Пользователь перенаправлен на страницу со всеми версиями текущего файла |
| Выбрать версию файла. Нажать кнопку Скачать | Скачивается версия файла и попадает в папку Загрузки |

Аналогичные тест-кейсы были составлены для ролей Администратор и Проверяющий.

**3.4. Разработка инсталлятора системы и основных решений по автоматизации рутинных задач**

**3.4.1. Разработка инсталлятора системы**

Так как разрабатываемая система является web-приложением, то нет необходимости устанавливать инсталлятор. Для разворачивания приложения на сервере необходимо:

1. Запустить скрипт для создания базы данных.
2. Развернуть приложение в IIS.

**3.4.2. Автоматизация рутинных задач**

Чтобы свести к минимуму риск необратимой потери данных, необходимо создавать резервные копии баз данных для сохранения вносимых изменений на регулярной основе. Хорошо спланированная стратегия резервного копирования и восстановления защищает базы от потери данных, вызванной разными сбоями.

Для создания резервной копии базы данных был написан следующий скрипт:

|  |
| --- |
| DECLARE @pathToBackup nvarchar(max) = N'F:\Backups\BackupPDM\_' + Convert(nvarchar(10), CONVERT (DATE, GETDATE()));  BACKUP DATABASE [PDM]  TO DISK = @pathToBackup  WITH NOFORMAT, NOINIT,  NAME = PDM-Full Database Backup',  SKIP, NOREWIND, NOUNLOAD, STATS = 10 |

Выполнение данного скрипта позволяет создать полную (FULL) резервную копию на диске сервера с именем **BackupPDM\_** + **дата бэкапа**.

Чтобы автоматизировать процесс резервного копирования базы данных, было создано задание (Job) по выполнению резервного копирования согласно установленному расписанию.

В SQL Server Agent был создан Job «BackupPdmDatabase». Job состоит из одного шага – выполнения резервного копирования. Шаг представляет собой выполнение приведенной выше команды T-SQL.

На рисунке 15 представлено расписание, согласно которому будет выполняться задание. Из рисунка видно, что задание будет выполняться каждую неделю по воскресеньям в 2 часа ночи. Предполагается, что в это время нагрузка на сервер является минимальной.

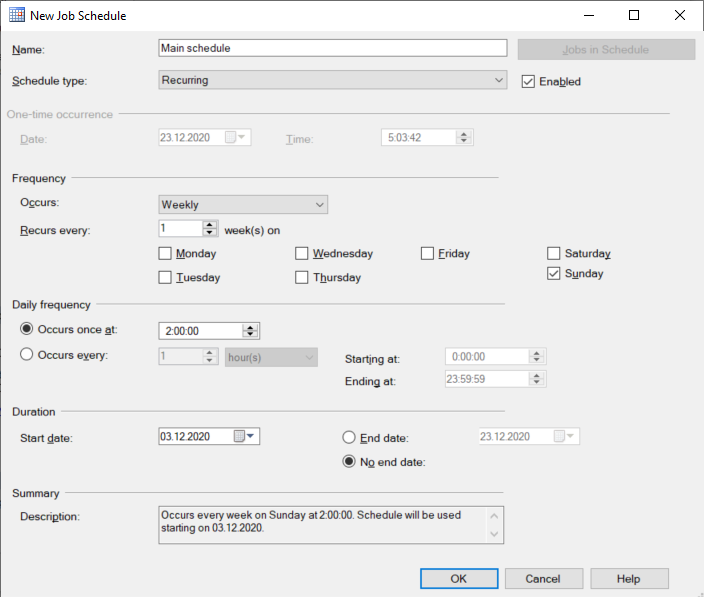


Рисунок 15 – Расписание выполнения задания

Для автоматизации ребилда индексов для таблиц БД было создано задание в SQL Server Agent. Выполнение задания было настроено на 5 часов утра ночи каждое воскресенье. Скрипт представлен ниже:

USE [PDM];

GO

ALTER INDEX ALL ON [dbo].[Documents]

REBUILD WITH (FILLFACTOR = 80, SORT\_IN\_TEMPDB = ON,

STATISTICS\_NORECOMPUTE = ON);

GO

ALTER INDEX ALL ON [dbo].[Users]

REBUILD WITH (FILLFACTOR = 80, SORT\_IN\_TEMPDB = ON,

STATISTICS\_NORECOMPUTE = ON);

GO

ALTER INDEX ALL ON [dbo].[HistoryVersions]

REBUILD WITH (FILLFACTOR = 80, SORT\_IN\_TEMPDB = ON,

STATISTICS\_NORECOMPUTE = ON);

GO