**Вариант 1 - Мультимедийное приложение (каталог мемов)**

**Цель работы:** Разработать приложение WPF, представляющее собой каталог мемов различной тематики. Создать визуально красивый, удобный и понятный интерфейс.

**Задачи:**

1. Создать TreeView со списком категорий и названий мемов.
2. Создать ComboBox для фильтрации категорий.
3. Добавление мема в каталог, осуществляющееся через диалоговое окно, которое имеет следующие поля ввода: название мема, расположение на диске, категория.
4. В главном окне создать кнопку «Удалить мем». При нажатии на эту кнопку, выделенный мем должен удаляться.
5. Хранение данных о мемах должно быть реализовано с помощью json файла. При этом в главном окне программы должны быть 2 кнопки: «Сохранить» и «Загрузить». При сохранении информация о добавленных 3 мемах должна записаться в файл, при загрузке должен отобразиться список ранее сохраненных мемов.
6. реализовать возможность добавления мема по url.

**Ход работы:**

В начале работы создан класс Meme (рисунок 1), который является родителем для классов Stud, Progr, Bra.

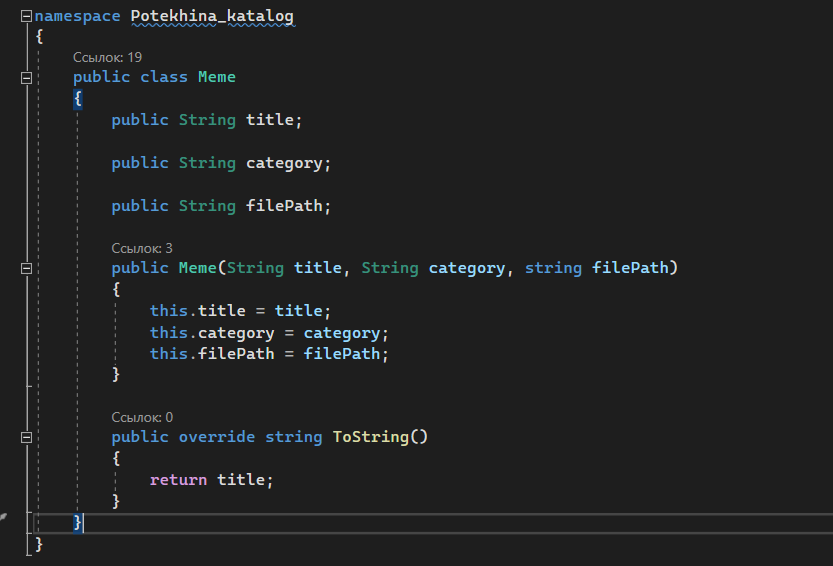


Рисунок 1 – код класса Meme

Дальнейший шаг – разработка интерфейса, который выглядит следующим образом (рисунок 2).

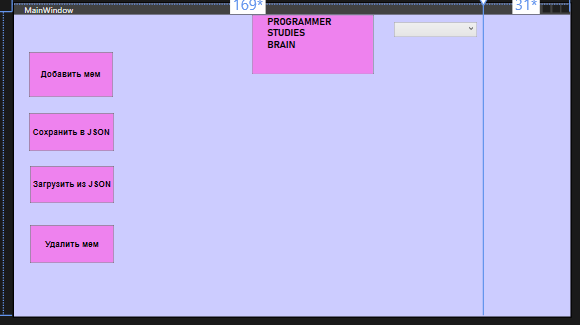


Рисунок 2 – Интерфейс главного окна

Кнопка «Добавить мем» переносит нас в новое окно (ADDMemes), в котором и происходит добавление мема. Код для данной кнопки представлен на рисунке 3.

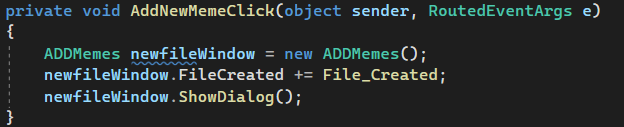


Рисунок 3 – Код для кнопки «Добавить мем»

В данном коде используется обработчик событий File\_Created, который добавляет мем в определенную категорию.

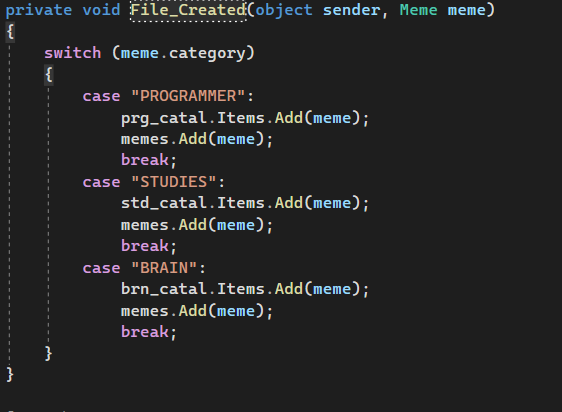


Рисунок 4 - обработчик событий File\_Created

Далее реализована возможность сохранения данных о мемах с помощью json файла. Подключен пакет «Newtonsoft.Json».

В коде для кнопки «save\_Click», объекты «memes» преобразуются в JSON с помощью метода «JsonConvert.SerializeObject» и сохраняются в файл с помощью метода «File.WriteAllText» (рисунок 5).

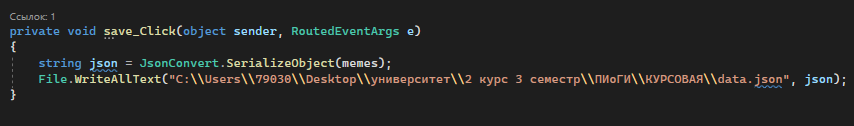


Рисунок 5 – код для сохранения файла

В коде для кнопки «load\_Click», содержимое JSON файла считывается с помощью метода «File.ReadAllText». Затем происходит десериализация JSON обратно в список объектов «Meme» с использованием метода «JsonConvert.DeserializeObject» (рисунок 6).

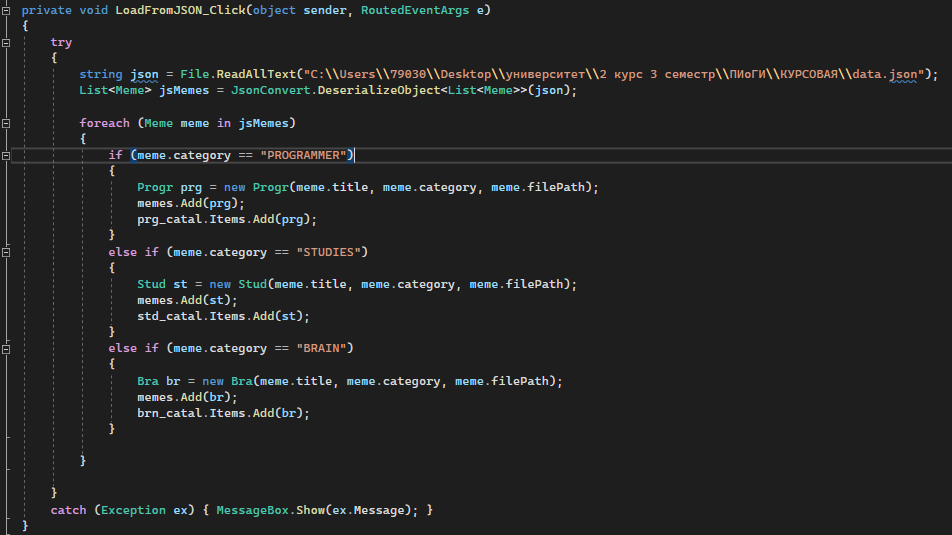


Рисунок 6 – код для загрузки файла

Реализация кнопки «Удалить мем» с помощью метода Remove представлена на рисунке 7.

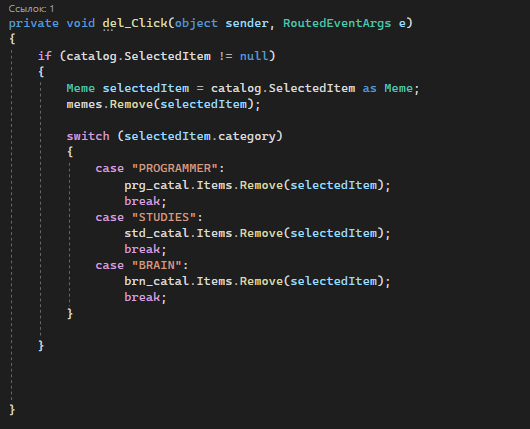


Рисунок 7 – реализация удаления мемов

Интерфейс окна «ADDMemes» представлен на рисунке 8.

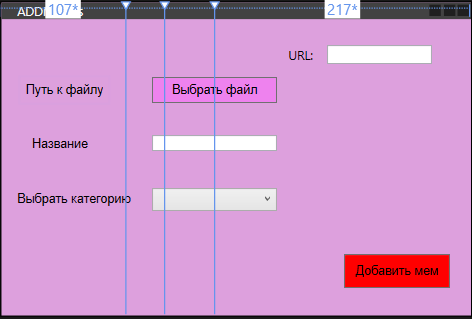


Рисунок 8 – Интерфейс окна «ADDMemes»

Кнопка «Выбрать файл» работает с помощью кода, представленного на рисунке 9.

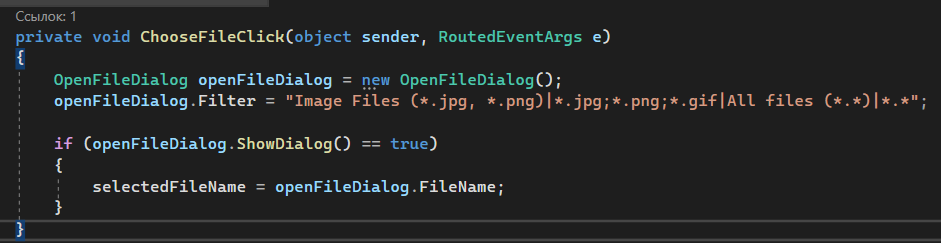


Рисунок 9 – Код для кнопки «Выбрать файл»

Она перенаправляет нас в диалоговое окно на нашем компьютере, откуда мы можем выбрать и загрузить нужную нам картинку.

Чтобы добавить мем, помимо загрузки картинки необходимо ввести ее название в TextBox и выбрать категорию, к которой она относится. При нажатии на кнопку «Добавить мем», он добавляется в TreeView. Часть кода для этой кнопки представлена на рисунке 10.

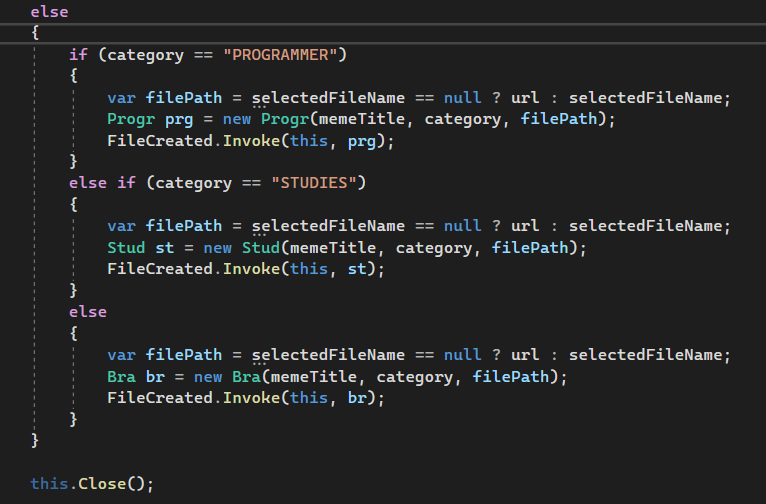


Рисунок 10 – Код для кнопки «Добавить мем

В коде также приведен метод, «updateImagess», который обновляет элемент управления «imagess», который служит для отображения изображений. Следующий метод «TreeViewItem\_Selected» вызывается после выбора элемента в «TreeView». Он вызывает соответствующий метод «updateImagess» для обновления изображения в элементе «imagess (рисунок 11).

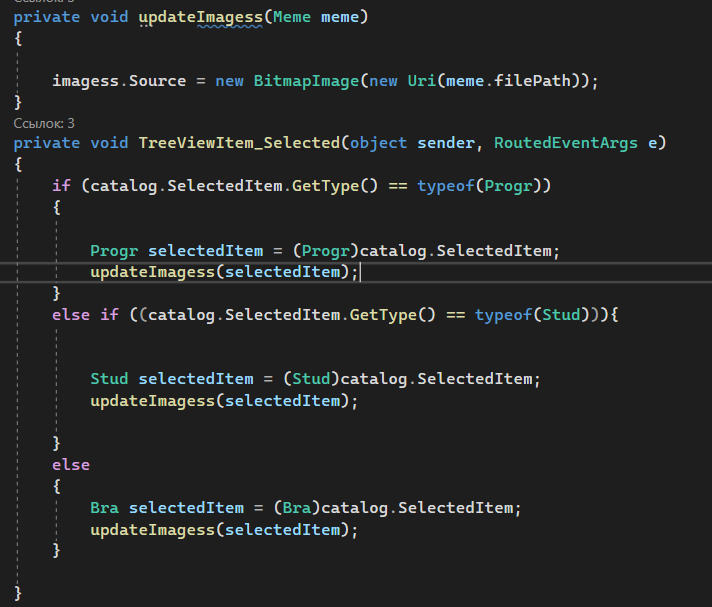


Рисунок 12 – Метод для обновления элемента управления

После завершения написания кода, программа работает следующим образом.

1. При нажатии на кнопку «Добавить мем» открывается диалоговое окно (рисунок 13).

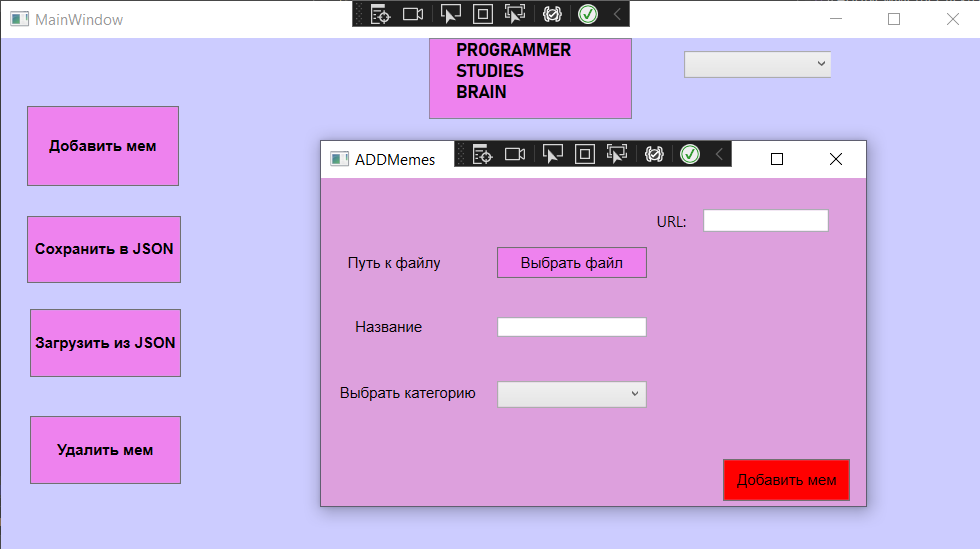


Рисунок 13 – Работа кнопки «Добавить мем»

1. При нажатии на кнопку «Выбрать файл» появляется окно с файлами (рисунок 14).

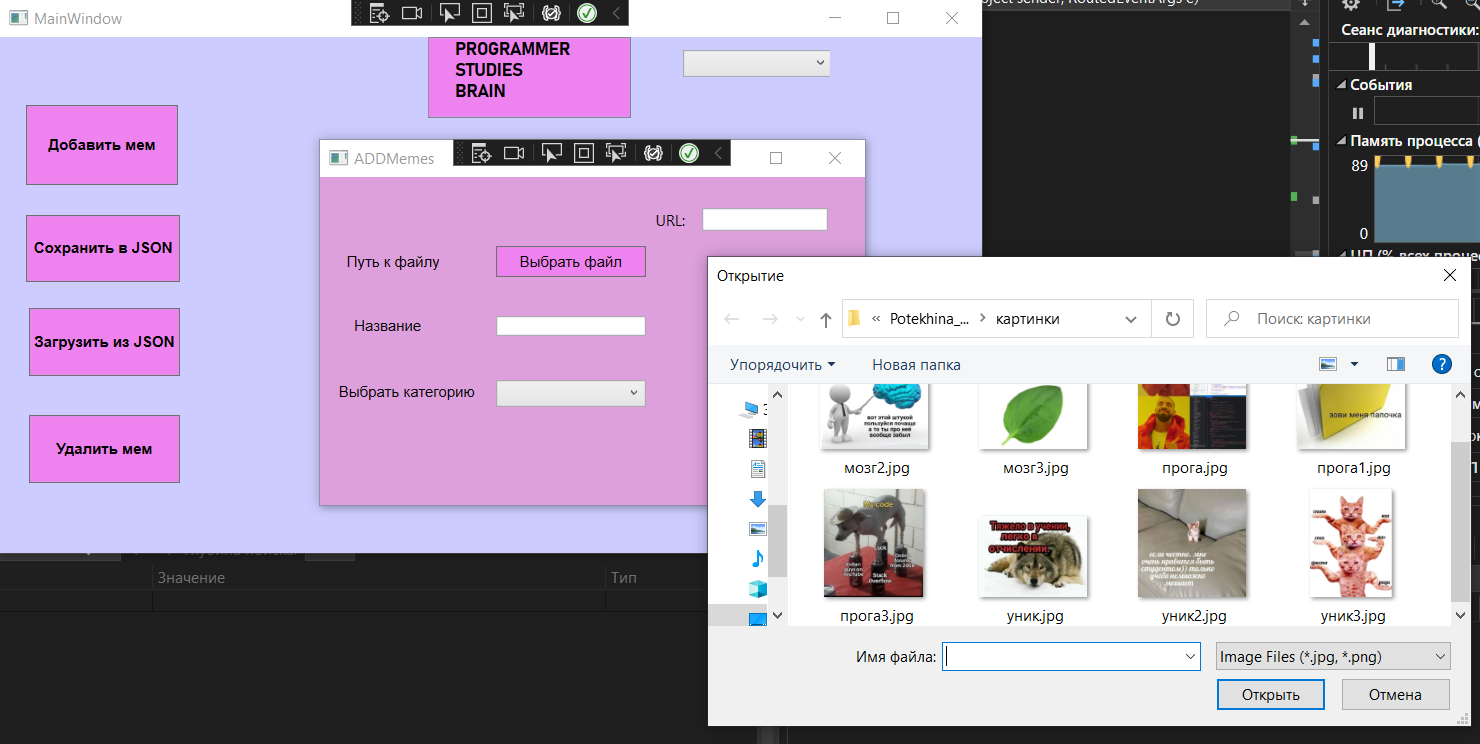


Рисунок 14 – работа кнопки «Выбрать файл»

1. Добавив файл, вводится имя картинки и выбирается подходящая категория (рисунок 15).

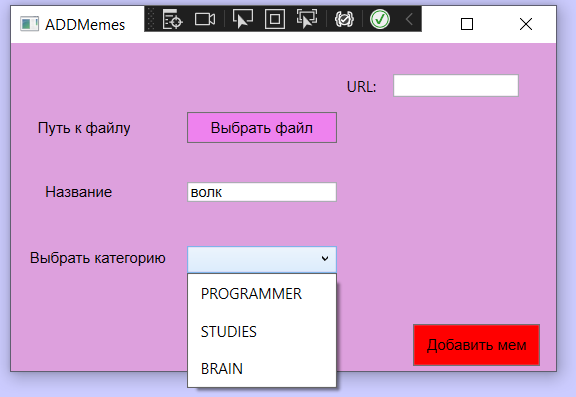


Рисунок 15 – Добавление файла

1. После добавления картинки, название отображается в TreeView. При нажатии на него, отображается выбранное изображение (рисунок 16).

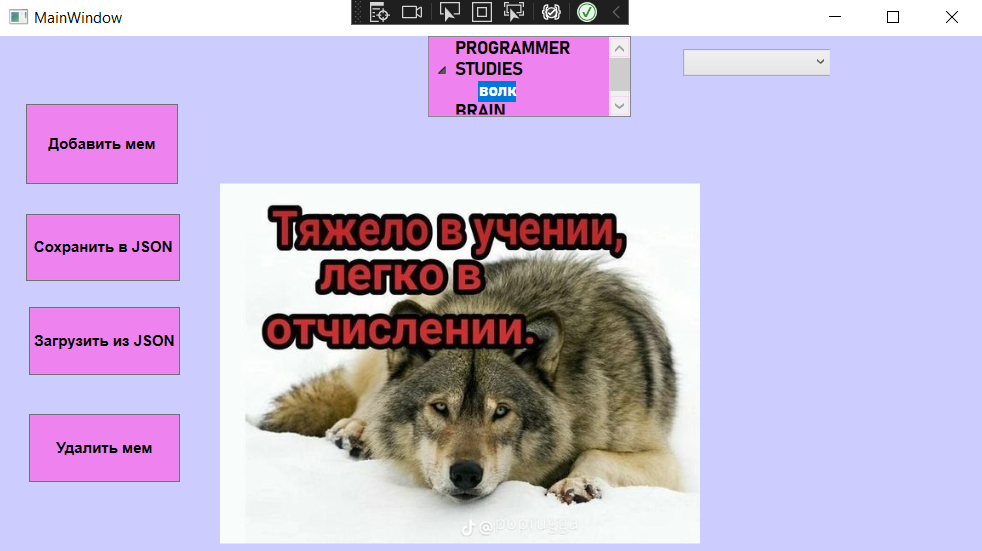


Рисунок 16 – Отображение картинки

1. Так же при выборе в ComboBox нужной категории, в TreeView отображаются названия изображений только этой категории (рисунок 17).

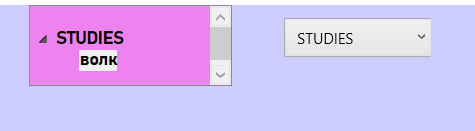
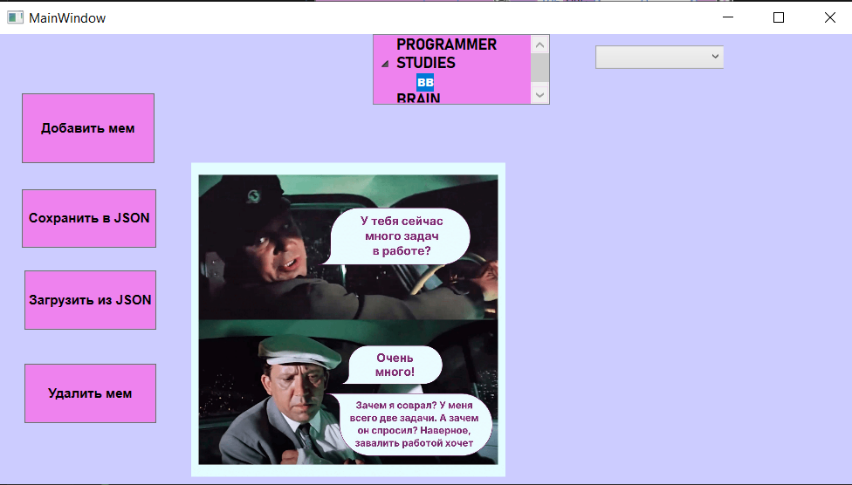
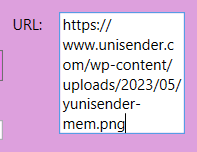


Рисунок 17 – Фильтрация категорий

1. Реализовано добавление мема по url (рисунок 18). В примере взял URL: <https://www.unisender.com/wp-content/uploads/2023/05/yunisender-mem.png>.



а) б)

Рисунок 18 – Реализация URL: а) ввод URL; б) Отображение мема

**Вывод:**

В результате работы разработано мультимедийное приложение, которое предназначено для добавления и поиска мемов различных категорий.

Выполнены все задачи, поставленные в начале работы, такие как: удобный интерфейс; добавление мема в каталог, осуществляющееся через диалоговое окно; хранение данных о мемах реализованное с помощью json файла; реализация добавления мема по url.

В итоге получилось удобное приложение, которое может быть использовано людьми в повседневной жизни.