МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Тихоокеанский государственный университет»

Кафедра «Программное обеспечение вычислительной техники и автоматизированных систем»

«Разработка и реализация системного приложения “Файловый менеджер с поддержкой архивов”»

Текстовый документ курсовой работы

по дисциплине «Операционные системы и системное ПО»

КР. 180008865.ТД

Выполнил студент Пшеничный Д.О.

Факультет, группа ФКФН, ПО(аб) – 81

Руководитель работы Федосеев А.А.

Виза: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(доработать, к защите и т.д.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Хабаровск – 2021г.

СОДЕРЖАНИЕ

[1 Постановка задачи 3](#_Toc73947434)

[2 Диаграмма классов 4](#_Toc73947435)

[2.1 Описание классов 4](#_Toc73947436)

[Заключение 8](#_Toc73947438)

[Список использованных источников 9](#_Toc73947439)

[Приложение А 10](#_Toc73947440)

# ПОСТАНОВКА ЗАДАЧИ

Требуется реализовать интерактивное приложение “Файловый менеджер с поддержкой архивов”, ­ обладающее следующими свойствами: навигация по директориям, открытие и создание архивов. Для разработки приложения будет использован язык программирования C#, и среды разработки IntelliJ Rider и Microsoft VisualStudio.

# ДИАГРАММА КЛАССОВ

На рисунке 1 рассматривается диаграмма классов.

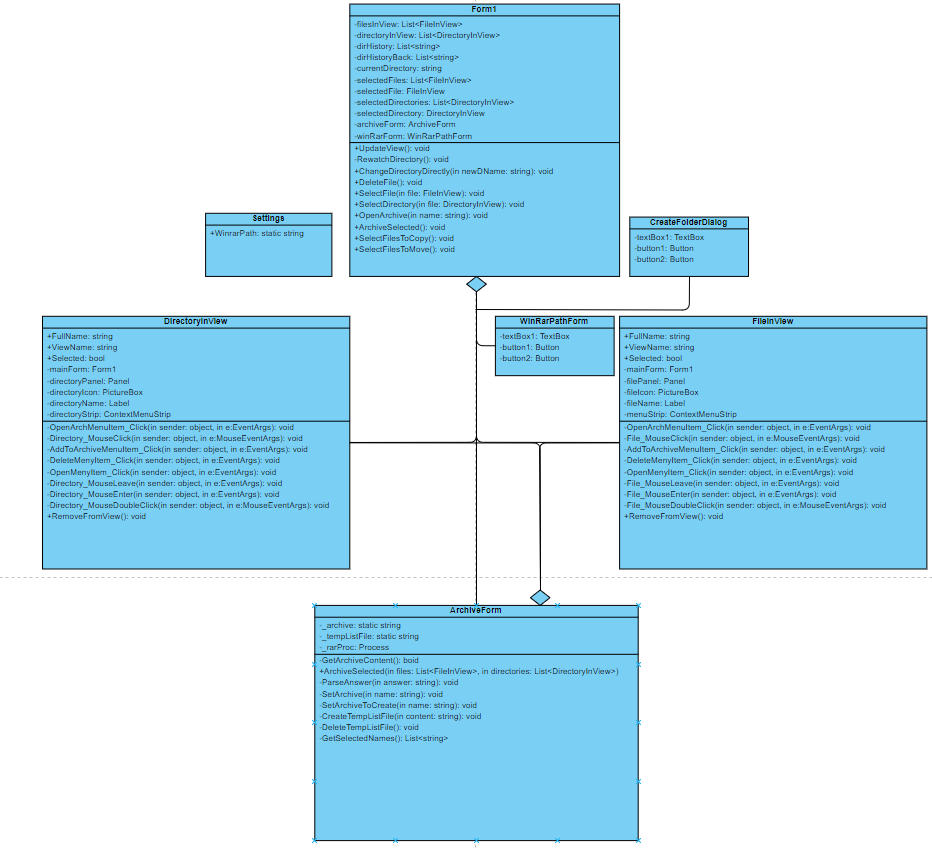


Рисунок 1 - Диаграмма классов

* 1. Описание классов
     1. FileInView – класс, представляющий файл в отображении. Отвечает за управления операциями с файлами в представлении.

Поля:

* string FullName – полное имя файла;
* string ViewName – имя файла в отображении;
* bool Selected – флаг выбора файла;
* Panel filePanel– панель, хранящая визуальные компоненты представления файла;
* Form1 mainForm– главная форма программы;
* PictureBox fileIcon – иконка файла;
* Label fileName – имя файла в отображении;
* ContextMenuStrip menuStrip – контекстное меню, используемое для файла;
* List<FileInView> filesToCopy – список файлов, отмеченных для копирования;
* List<FileInView> filesToMove – список файлов, отмеченных для перемещения;
* List<DirectoryInView> directoriesToCopy – список директорий, отмеченных для копирования;
* List<DirectoryInView> directoriesToMove – список директорий, отмеченных для перемещения.

Методы:

* void RemoveFromView() – удаляет визуальные компоненты представления файла с формы;
  + 1. DirectoryInView – класс, по составу и реализации аналогичный классу FileInView, но работающий для директорий.

2.1.3 Form1 – главный класс, представляющий основное окно приложения

Поля:

* DownloadPool downloadPool – поле, отвечающее за загрузку файла;
* Stage window – контейнер, используемый для вывода главного окна;
* TableView<DownloadThread> table – таблица, выводящая основную информацию о загружаемом файле.

Методы:

* static void main(String[] args) – метод, запускающий приложение;
* void stop() – переопределение метода, отвечающее за действия после закрытия приложения;
* void setTable() – создание таблицы;
* void setButtons() – настройка кнопок;
* void start(Stage stage) – переопределение метода, отвечающий за обработку главного окна при открытие;
* void SelectFilesToCopy() – помечает выбранные файлы и папки как копируемые, помещая их в списки filesToCopy и directoriesToCopy;
* void SelectFilesToMove() – помечает выбранные файлы и папки как перемещаемые, помещая их в список filesToMove и directoriesToMove.

2.1.4 ArchiveForm – главный класс, представляющий основное окно приложения

Поля:

* string \_archive – имя создаваемого/распаковываемого архива;
* Process \_rarProc – процесс winrar, используемый для выполнения операций с архивами;
* string \_tempListFile – имя временного файла-списка.

Методы:

* void GetArchiveContent() – получение содержимого архива, и отображение его на форме;
* void ArchiveSelected(List<FileInView> files, List<DirectoryInView> directories) – метод, выполняющий архивацию полученных файлов(files) и директорий(directories);
* List<string> ParseAnswer(string answer) – метод, разбивающий строку answer с содержимым архива на список строк с именами файлов и папок;
* void SetArchive(string name) – установить имя распаковываемого архива name;
* void SetArchiveToCreate(string name) – установить имя создаваемого архива name;
* List<string> GetSelectedNames() – получить список имён выбранных файлов;
* void CreateTempListFile(string content) – создаёт временный файл-лист с содержимым content;
* void DeleteTempListFile() – удаляет временный файл-лист.

# ЗАКЛЮЧЕНИЕ

Выполнение курсовой работы завершилось созданием программного продукта, который обладает функциональными возможностями, определенными заданием.

В приложении А приведены скриншоты, полученные в результате тестирования программы.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Metanit [Электронный ресурс]: онлайн учебник. – Режим доступа: https://metanit.com/ (дата обращения 12.25.2015)

2. MSDN [Электронный ресурс]: документация. [Microsoft]. – Режим доступа: https://docs.microsoft.com/ (дата обращения 12.25.2015)

3. Бен Албахари и Джозеф Албахари. C# 7.0 in a Nutshell: The Definitive Reference, 28 октября 2017 г.

4. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж.: Паттерны объектно-ориентированного проектирования — Прогресс книга, 2020 г. — 448 с.

# ПРИЛОЖЕНИЕ А

(обязательное)

Функционирование приложения

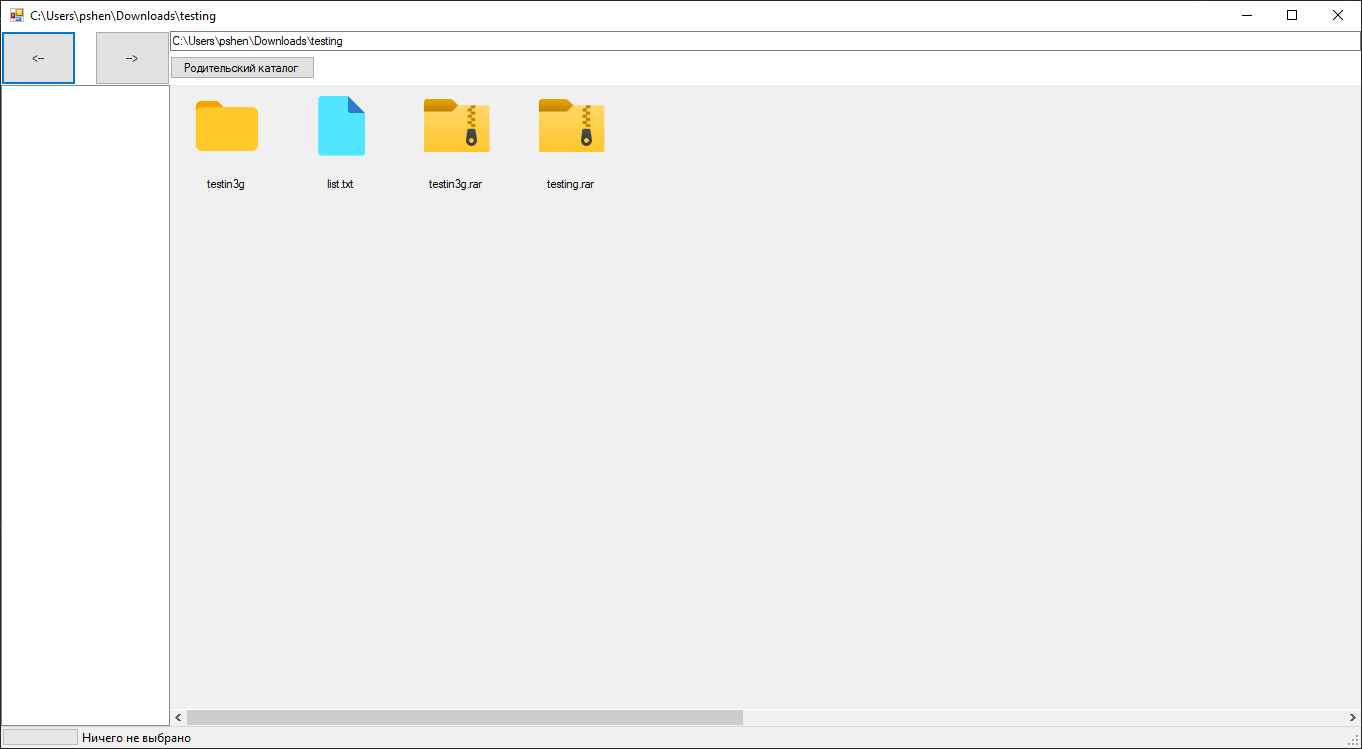


Рисунок А.1 – Основной вид приложения

Продолжение приложения А

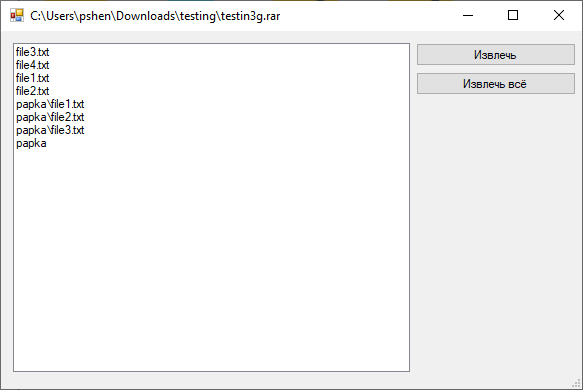


Рисунок А.2 – Окно работы с архивами

Продолжение приложения А

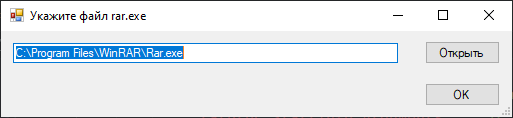


Рисунок А.3 – Окно задания пути к консольной версии WinRar.