

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«Южно-Уральский государственный университет»
(национальный исследовательский университет)
Высшая школа электроники и компьютерных наук
Кафедра системного программирования

Разработка программы по выявлению больших
файлов-кандидатов на удаление на основе последних
обращений к файлу

Научный руководитель:
доцент кафедры СП, к.ф.-м.н.
И. И. Клебанов

Автор работы:
студент группы КЭ-433
В. А. Ваганов

Актуальность

- Частая установка программ и редкая очистка памяти приводят к накоплению ненужных файлов.
- Избыточное количество файлов замедляет работу компьютера, что особенно критично при работе с большими объемами данных.
- Трудности в определении ненужных файлов затрудняют процесс оптимизации дискового пространства.
- Беспокойство о возможных негативных последствиях удаления для стабильности системы.

Цель и задачи

Цель:

Разработать программу для выявления файлов-кандидатов на удаление

Задачи:

1. Определить критерий «больших» файлов.
2. Спроектировать методы и алгоритмы анализа и оценки, используемые при определении кандидатов на удаление.
3. Спроектировать систему для реализации функциональных требований.
4. Провести тестирования для проверки корректности работы программы.

Анализ аналогов

Были рассмотрены различные аналоги, такие как WinDirStat, TreeSize, DiskSavvy и SpaceSniffer. Большинство этих программ не предоставляют информацию о дате последнего обращения к файлам. Моя программа решает эту проблему, предоставляя доступ к данным о последнем обращении и оценивая файлы на основе этой информации.

Критерий большого файла

Критерии для определения большого файла могут варьироваться в зависимости от контекста и области применения. Несколько общих критериев, которые часто используются для классификации файла как большого:

Размер файла:

1. В цифровом контексте файл обычно считается большим, если его размер превышает несколько гигабайт (GB).
2. В некоторых случаях, особенно при работе с большими данными (Big Data), файлы размером в терабайты (TB) могут не считаться большими файлами.

Контекст использования:

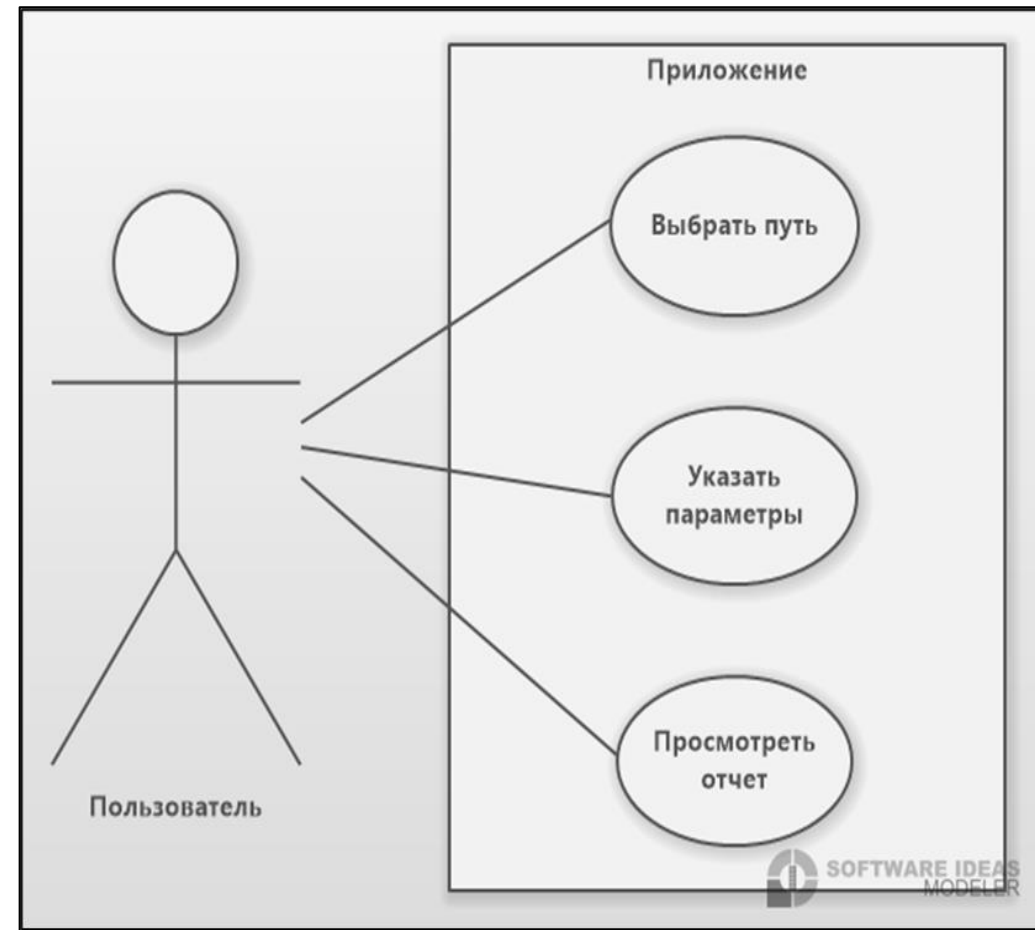
1. В веб-контексте файлы размером более нескольких мегабайт (MB) могут считаться большими, так как это может замедлить загрузку страниц и потребовать больше пропускной способности.
2. В контексте передачи данных через сети файлы, превышающие допустимые лимиты (например, 25 MB для вложений в электронной почте), также считаются большими.

Функциональные требования

1. Анализ файлов: анализ размера файлов и определение кандидатов на удаление.
2. Отслеживание активности: функция отслеживания последних обращений к файлам для определения их актуальности.
3. Интерфейс для просмотра результата: интерфейс или отчет показывают кандидатов на удаление с указанием размера и последней активности.
4. Гибкое конфигурирование: возможность настройки критериев для определения "больших" файлов и периода последних обращений.

Варианты использования

Построена диаграмма вариантов использования для модели взаимодействия внешнего актера с приложением. В ходе анализа требований к системе выявлены следующие варианты использования:



Архитектура: главный модуль (Programm)

- **Управляет процессом выполнения программы:**
 - Запускает программу, определяет последовательность задач и контролирует их выполнение.
 - Обрабатывает ошибки и исключения.
- **Взаимодействует с пользователем через консоль:**
 - Принимает ввод от пользователя (путь к папке, размерные и временные пороги).
 - Выводит результаты анализа и оценки файлов.
- **Инициализирует и вызывает методы других модулей:**
 - Создает экземпляры других модулей и вызывает их методы для выполнения задач.
 - Координирует взаимодействие между модулями.

Архитектура: модуль сканирования (FileScanner)

- **Обходит файловую систему:**

- Сканирует все файлы и папки в указанной директории, включая поддиректории.
- Сохраняет информацию о каждом файле (имя, размер, дата последнего доступа).

- **Обеспечивает безопасное извлечение данных:**

- Получает информацию о файлах без изменения их атрибутов.

Архитектура: модуль сортировки (FileSorter)

- **Фильтрует и сортирует файлы:**
 - Выводит файлы-кандидаты на удалении на экран.
 - Сохраняет файлы-кандидаты в файл.
 - Сортирует файлы по баллам.
 - Исключает файлы, размер которых меньше заданного порога.

Архитектура: модуль оценки (Evaluation)

- **Оценивает файлы:**

- Рассчитывает баллы для каждого файла на основе размера и времени последнего доступа.
- Определяет кандидатов на удаление.

- **Сохраняет результаты:**

- Сохраняет результаты оценки для дальнейшего использования.

Функциональное тестирование

№	Название теста	Шаги	Ожидаемый результат	Тест пройден?
1	Сканирование папки	1. Указать путь к корневой папке. 2. Вызвать метод ScanFolder.	Папка просканирована, файлы и подпапки добавлены в структуру данных.	Да
2	Фильтрация файлов по размеру	1. Задать порог размера. 2. Вызвать метод SortAndFilterFiles.	Файлы, превышающие порог, добавлены в список больших файлов.	Да
3	Оценка файлов	1. Задать порог размера и временной порог. 2. Вызвать метод EvaluateFiles.	Файлы оценены по размеру и времени последнего доступа.	Да
4	Вывод отчета	1. Вызвать метод PrintEvaluations.	Список из 10 файлов с наибольшими оценками выведен на экран.	Да
5	Парсинг размерного порога	1. Ввести значение размера (например, 10KB, 100MB). 2. Вызвать метод ParseSizeInput.	Значение корректно распознано и переведено в байты.	Да
6	Обработка некорректного пути	1. Указать несуществующий путь. 2. Вызвать метод ScanFolder.	Отображено сообщение об ошибке пути.	Да
7	Обработка исключений доступа	1. Указать путь к защищенной папке. 2. Вызвать метод ScanFolder.	Отображено сообщение об отсутствии доступа.	Да

Дальнейшая работа

- Создать пользовательский интерфейс
- Добавить возможность удаления файлов в приложении
- Оптимизировать производительность программы

Результаты

1. Проанализирована предметная область.
2. Проведено сравнение существующих аналогов.
3. Разработаны функциональные и нефункциональные требования к программе.
4. Спроектированы и реализованы ключевые компоненты программы, такие как сканер файловой системы, сортировщик файлов и оценщик.
5. Проведено тестирование программы для подтверждения корректности работы системы и ее соответствие заданным требованиям.