**PHASE 1 PROJECT: VIRTUAL KEY FOR YOUR REPOSITORIES**

**Documentation**

1. **Project Info**

**Project Name:** Virtual Key

**Developer Name:** Srilekha Vinjamara

1. **Sprints/Tasks**

Since the timeline of the project is for 3 weeks (15 working days), the sprint cycle has been reduced from default 2 weeks to 1 week per sprint.

**Sprint 1:**

Planning and outline of the program, with basic menu driven code for functionalities.

**Tasks done:**

Basic operations done, as follows:

* Retrieving the file names in an ascending order
* Business-level operations:
  + Option to add a user specified file to the application
  + Option to delete a user specified file from the application
  + Option to search a user specified file from the application
  + Navigation option to close the current execution context and return to the main context
* Option to close the application

**Sprint 2:**

Detailed functionalities implemented.

**Tasks done:**

Operations implemented as follows:

* Code to display the welcome screen. It should display:
  + Application name and the developer details
  + The details of the user interface such as options displaying the user interaction information
  + Features to accept the user input to select one of the options listed
* The first option should return the current file names in ascending order. The root directory can be either empty or contain few files or folders in it
* The second option should return the details of the user interface such as options displaying the following:
  + Add a file to the existing directory list
    - You can ignore the case sensitivity of the file names
  + Delete a user specified file from the existing directory list
    - You can add the case sensitivity on the file name to ensure that the right file is deleted from the directory list
    - Return a message if FNF (File not found)
  + Search a user specified file from the main directory
    - You can add the case sensitivity on the file name to retrieve the correct file
    - Display the result upon successful operation
    - Display the result upon unsuccessful operation
  + Option to navigate back to the main context
* There should be a third option to close the application

**Sprint 3:**

Testing functionalities, optimization and code enhancement done.

**Tasks done:**

Techniques used:

* Searching using Linear Search
* Sorting using Collections [Files with caps sorted before filenames in lowercase, due to ASCII values for lowercase characters (97-122) being greater than uppercase (65-90)]
* File Handling
* Exception/Error Handling
* Testing application for various scenarios by running the code, making code bug free

1. **Algorithms used**

**Searching:** Linear Search

**Sorting:** Collections.sort() which uses Merge Sort internally

1. **Core Concepts used**

Searching using Linear Search

Sorting using Collections (Merge Sort used internally)

File Handling

Exception/Error Handling

1. **GitHub Repository link**

<https://www.github.com/SRILEKHA-V19>

1. **Unique Selling Points (USPs)**

Easy to use Menu driven program/application written in IntelliJ IDEA

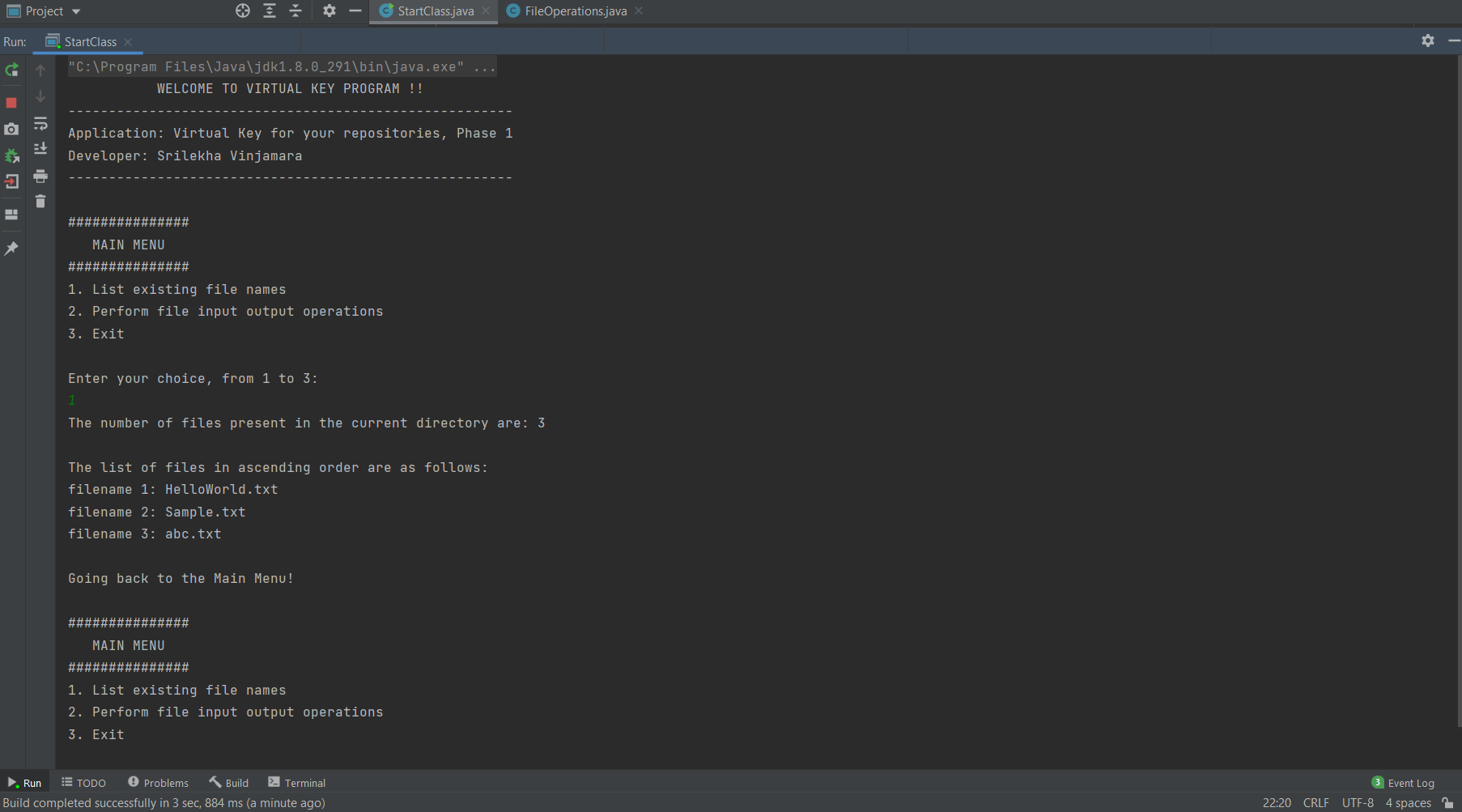
Visual appeal and good understanding of the application flow to the user in the first attempt

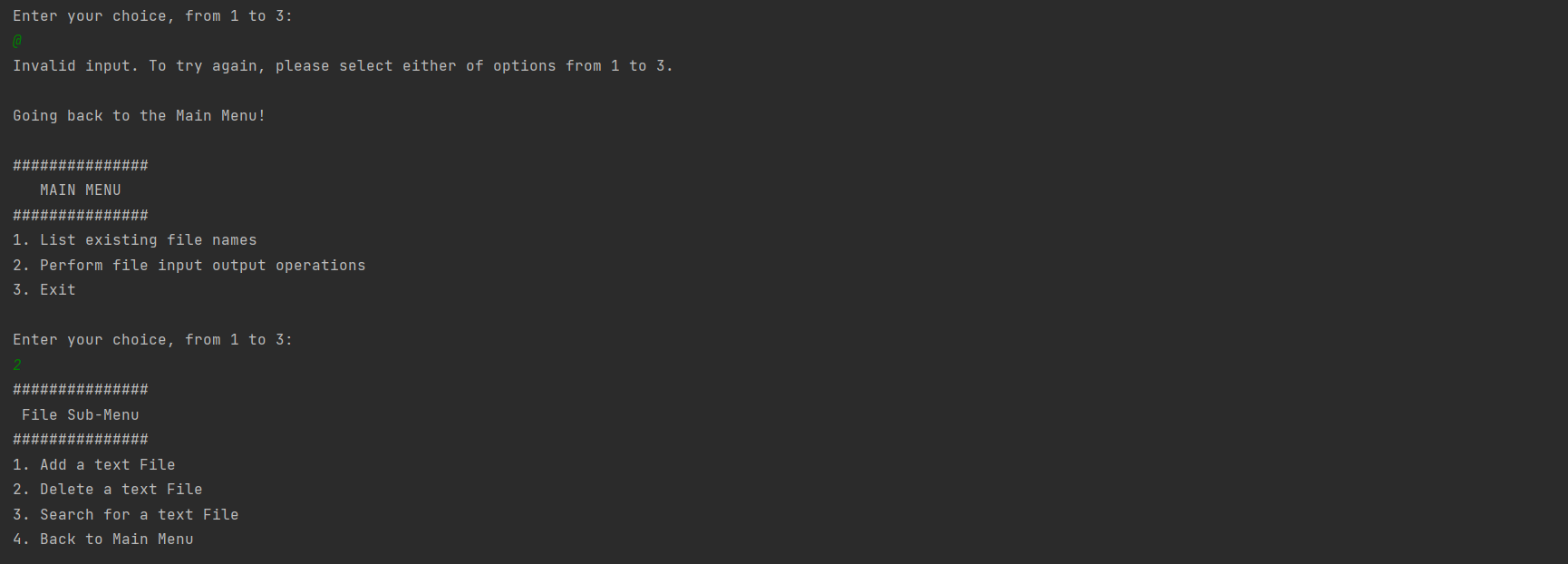
1. **Future Enhancements**

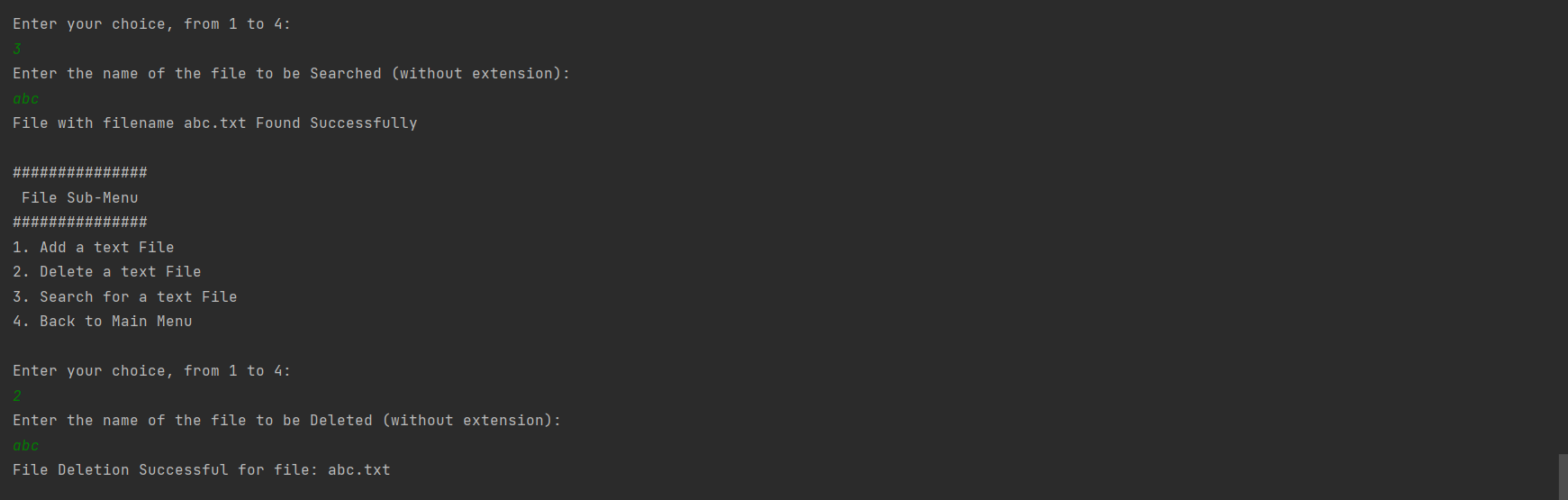
Add more functionality like write into multiple files or append a file

Make it into a basic webapp or add UI instead of using command line to run the application

1. **Screenshots of the application flow**













--- End ---