# Virtual Key for Your Repositories Specification Document

# Table of Contents


***Introduction*** ..... ***- 2 -***

***Problem Statement*** ..... ***- 2 -***

***Software Development Life Cycle*** ..... ***- 3 -***

***Product Description*** ..... ***- 3 -***

**Main Menu Capabilities** ..... **- 4 -**

**Business Level Operations Sub-Menu Capabilities** ..... **- 4 -**

***Concepts Used*** ..... ***- 4 -***

***FlowChart*** ..... ***- 5 -***

***Conclusion*** ..... ***- 5 -***

***GitHub Link*** ..... ***- 5 -***

## *Introduction*

- This is the project report for the Project "Virtual Key for Your Repositories" under "PG FSD Implement OOPS using JAVA with Data Structures and Beyond" course of Post Graduate Program in Full Stack Web Development from Simplilearn.
- Project submitted by : Krishnaveni Rajan
- Email Id: krishnaveni07.rajan@gmail.com
- Submission Date: 22-October-2021
- Language/Tools Used : Core Java. Project written in Eclipse and committed to GitHub
- SCDL : Agile Scrum

## *Problem Statement*

Company Lockers Pvt. Ltd. hired you as a Full Stack Developer. They aim to digitize their products and chose LockedMe.com as their first project to start with. You're asked to develop a prototype of the application. The prototype of the application will be then presented to the relevant stakeholders for the budget approval. Your manager has set up a meeting where you're asked to present the following in the next 15 working days (3 weeks):

- Specification document - Product's capabilities, appearance, and user interactions
- Number and duration of sprints required
- Setting up Git and GitHub account to store and track your enhancements of the prototype
- Java concepts being used in the project
- Data Structures where sorting and searching techniques are used.
- Generic features and three operations:
    - 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

The goal of the company is to deliver a high-end quality product as early as possible.

The flow and features of the application:

- Plan more than two sprints to complete the application
- Document the flow of the application and prepare a flow chart
- List the core concepts and algorithms being used to complete this application
- 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 in order 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

## *Software Development Life Cycle*

This project uses the Agile Scrum methodology for the Software Development Life Cycle.

- Total no. of Sprints -2
- Duration of the Sprint – 5 days each
- No. of Resources – 1

## *Product Description*

This product is named as “Virtual Key” for the Repository. This product is created as a prototype hence it is built using the command line interaction for the users. Based on the approval of the Prototype the Product will be implemented in User interactable product with front and back ends. Let us see about the capabilities of the product,

- Welcome screen will be displayed to the user with the “Main Menu”
- Main Menu provide 3 different options for the user
    1. Display Files in Ascending Order
    2. Business Level Operation
    3. Close Application

### Main Menu Capabilities

- When user provides '1', then the files that are in the Repo's root directory will be displayed in the Ascending order.
- When user provides '2', the Business Level Operations with it's sub-menu will be displayed as
    1. Add File
    2. Delete File
    3. Search File
    - ♦ Go back to Main menu
- When user provides '3', application will be closed.
- When user provides a different input, proper message will be displayed.
- User will also be given option to continue back from the Main Menu after performing the 1st or 2nd operation.

### Business Level Operations Sub-Menu Capabilities

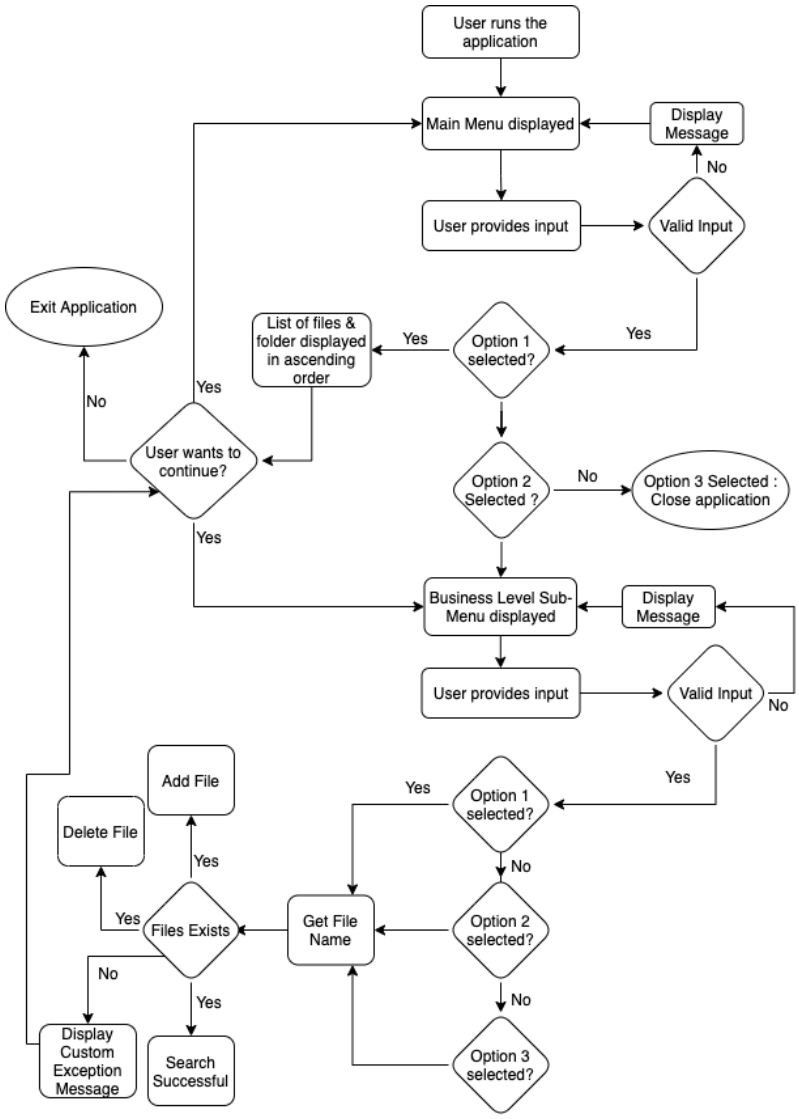
- When user provides '1', user will be asked for the filename. With the name provided the application will create a file if it does not exist already.
    - ♦ If the file with same name is added, proper message will be displayed.
    - ♦ As this option is to 'Add', the file will not be overwritten.
- When user provides '2', user will be asked for the filename. With the name provided the application will delete a file if it exists already
    - ♦ If the file does not exist, proper message will be displayed.
- When user provides '3', user will be asked for the filename. With the name provided the application will search for the file. If it exists, search operation is considered successful.
    - ♦ If the file does not exist, search operation is unsuccessful and proper message will be displayed.
- User will also be given option to continue back from the Sub-Menu after performing the 1st or 2nd or 3rd operation.
- If the user selects not to continue with the Sub-Menu, it will be returned to Main Menu and confirm with user for continuation.

## *Concepts Used*

- File Handling
- Exception Handling : Built-in, Custom
- Constructor
- Array Sort
- Switch case

- Loops : do-while, if-else, for
- Variable & Objects : Methods, File, Scanner, Char, String,

## *FlowChart*




## *Conclusion*

Using core java's different concepts, data type, in-built methods and options, it was feasible to make the Virtual Key for the Repository while the user interaction was maintained through-out the application. It was easy to implement when the tasks are clear, requirement is well understood and implemented with an idea of how to do and which feature to use.

GitHub Link : https://github.com/krajan07/FSDP1Project