LOCKEDME.COM

SPECIFICATION DOCUMENT

JAVIER LUQUE

POST GRADE FULL STACK DEVELOPMENT
IMPLEMENT OOPS USING JAVA WITH DATA STRUCTURES
COHORT 1
2022

Table of Contents

Problem Background

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.

Developer Details

Javier Luque will be the developer of this demo representing LockedMe.com; he is currently a junior Java developer with little experience but eager to learn and to show what he has learned up to this point.

He will be also the project manager and will be in charge of the planning, monitoring and execution of this demo.

Project Details

This project will be divided into two springs (details on the sprint execution document). Estimation were based in hours per day. Third week is planned for changes requested by the user after the demo has been displayed.

All the documentation for this project could be found in the following Github repo:

jluqueco/**LockedMe**

This project was built under two Java classes:

LockedMe – This class contains all the business logic as well as the algorithms for the functionality that the customer is requesting.

LockedMeMain – this class deals with the front end of the application. I t also instanciates and call the methods needed from LockedMe class.

Sprints

As stated before, two sprints were planned for this project, each sprint were planned and executed by the developer team. Also, all Scrum activities were executed for this project as documented on the Github repo.

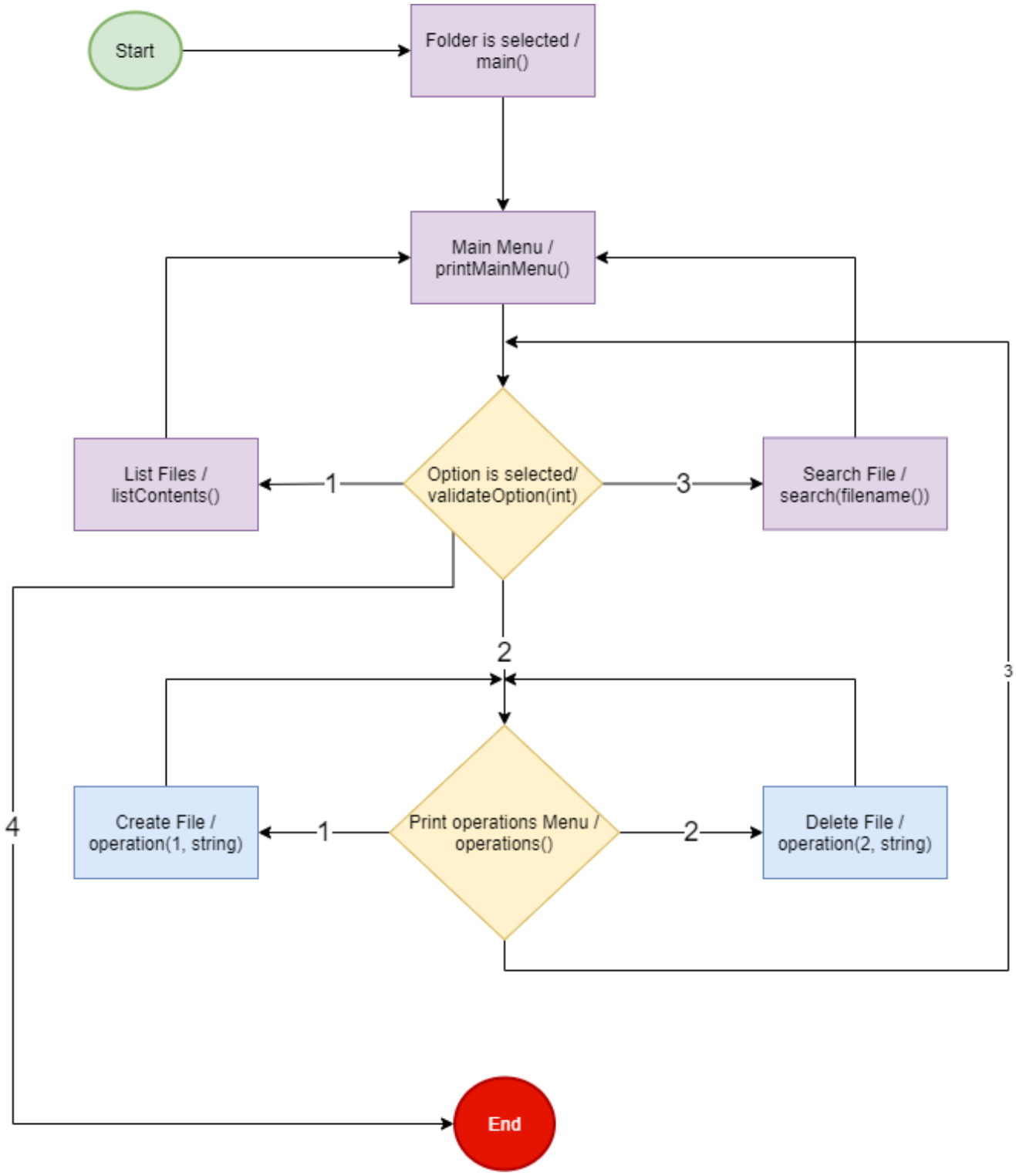
SPRINT [NUMBER] BACKLOG													
ID	User Story	Tasks	Owner	Status	Estimated effort	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
SP001	As an user I need to be able to list The files stored in a folder.	Code "list files": Select folder, define a structure, sort and display	jluque	Completed	20	8						
		Create empty menus	jluque	Completed			8					
		Demo, show list files functionalities	jluque	Completed				2	2			
	As an user I need to be able to add a file to the existent directory.	Code Insert file: define folder, check duplicated exception, store	jluque	Completed	20			6	3			
		Prepare and execute demo	jluque	Completed					3	8		
SP002	As an user I need to be able to delete a file in the existent directory.	Code Delete file: define folder, watch checked exceptions.	jluque	Completed	10	8						
		other exceptions	jluque	Completed			2					
	As an user I need to be able to search for a file in the existent directory.	Code Search: define folder, check letter case	jluque	Completed	22		6	2				
		Prepare a complete demo	jluque	Completed				4	4			
		Complete any changes required from Demo	jluque	Completed					4	2		
					72	16	16	14	16	10	0	0

Algorithms

For this project, we have used Bubblesort as a sorting algorithm and Binary search as a search algorithm. These algorithms were chosen because of the easy way that these algorithms are implemted. Also try and catch clauses were used to ensure the robustness of the application.

Application Flow

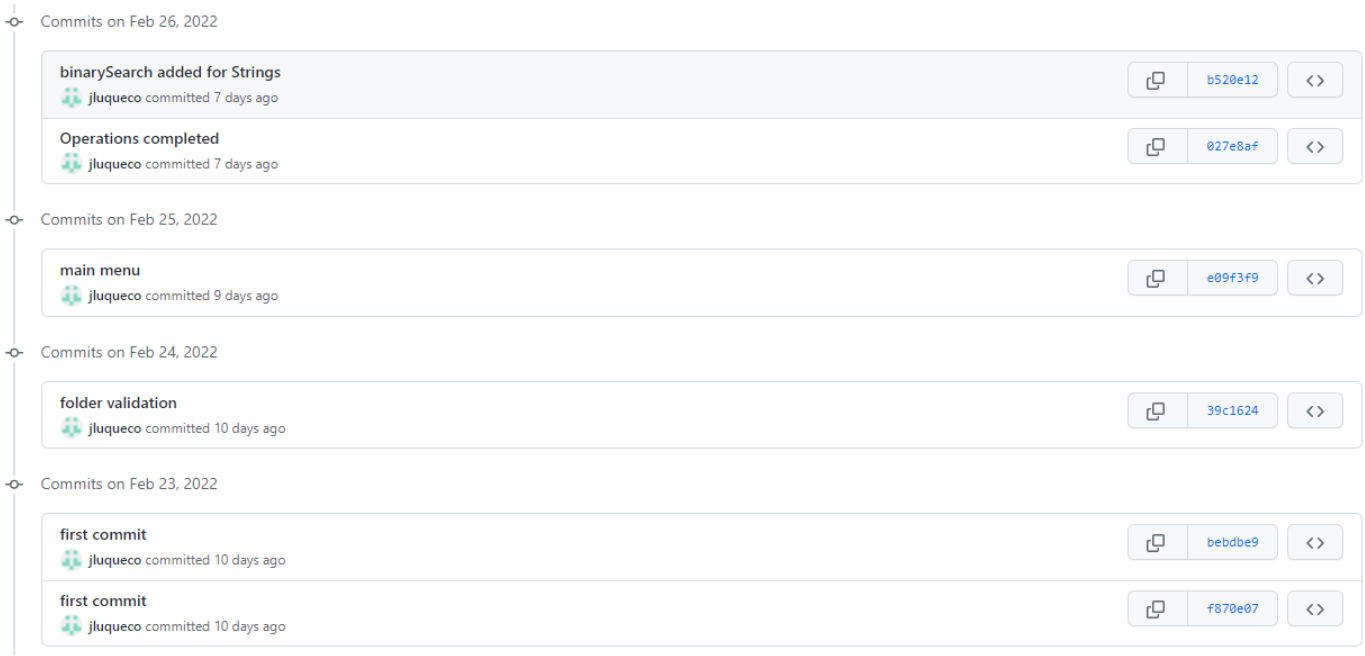
This app displays a menu where the user can choose what to execute. Currently the app is defaulted to a default folder created locally but the application is capable of requesting a folder from the user where he or she wants to execute the desired operations.



Core Concepts

- Casting
- functional programing
- OOPS
- debugging
- sorting data
- searching data
- File Management
- access modifiers
- flow control structures
- Error handling
- Arrays
- List interface
- String handling
- StringBuffer

GitHub Tracking



Conclusion

This demo will be successful due to the fulfillment of all of the customer needs. This application can be enhanced by adding a graphical UI and implementing faster searching and sorting algorithms. The key success of this project is to provide flexibility, reliability and speed to the user.