EPITRELLO –Documentation

- Annanya & Prashanth

Introduction:

Epitrello is a program that implements a system to Manage and Control the tasks for the users. Also to make better decisions about assignments and tasks based on priority and estimated time for each task.

We have made use of H2 Database and its jar file is already present in the Project.

We have made use of Array List as our Data Structure since it is dynamic and can be incremented or decremented when required. Since, the users, list and tasks can be created and deleted as required, we prefer to use this Data Structure.

Steps Before Running the Project

1. Choose your desirable Database, if it is other than H2 please insert the respective Database JAR file under the lib folder of the Project.
2. In the eclipse , Goto Properties → Java Build Path, Select classpath and add your desired JAR present under lib.
3. Run your desired Database and run the Create Query present in the Project path "sql→init.sql".
4. Close your Database connection.
5. Insert your Database JDBC URL , ID and password in the fields db.url, db.id and db.passwrd respectively in the configuration file present in conf.properties files under Epitrello.

6. Now, you can run the Main.java class present under "src→ fr.epita.epitrello.launcher→Main.java
7. The output will not be generated in the console, you can find the output in the path mentioned under "output.file" in the configuration file.
8. You can find the logs in the path " temp→application.log"

.

Testing:

We have performed some basic operations and have tested the same with the java file under test→fr.epita.epitrello.service.test→Tester.java file and the test output is generated in the test.txt

.

Output of Main with comments:

Success // output of adduser for Thomas

Success // output of adduser for AmirAli

Success // output of adduser for AmirAli

Success // output of addList for Code

Success // output of addList for Description

Success // output of addList for Misc

Success // output of addTask for "Do Everything"

Success // output of editTask for "Do Everything"

Success // output of assignTask of "Do Everything" to Rabih

Do Everything// output of printTask for "Do Everything"

Write the whole code

Priority: 10

Estimated Time: 12

Assigned To: Rabih

Success // output of addTask for "Destroy code formatting"

Success // output of assignTask of " Destroy code formatting " to Thomas

Success // output of addTask for "Write Description"

Success // output of assignTask of "Write Description" to AmirAli

Success // output of addTask for "Upload Assignment"

Success // output of completeTask for "Do Everything"

Rabih // output of printUsersByPerformance

AmirAli

Thomas

Thomas// output of printUsersByWorkload

AmirAli

Rabih

1 | Upload Assignment | Unassigned | 1h

// output of printUnassignedTasksByPriority

Success// output of deleteTask for "Upload Assignment"

1 | Write Description | AmirAli | 3h

2 | Destroy code formatting | Thomas | 1h

// output of printAllUnfinishedTasksByPriority

Success // output of addTask for "Have fun"

Success // output of moveTask for "Have fun" to “Code”

Have fun// output of printTask for "Have fun"

Just do it

Priority: 2

3

Estimated Time: 10

Unassigned

List Code // output of printList for "Code"

10 | Do Everything | Rabih | 12h

2 | Destroy code formatting | Thomas | 1h

2 | Have fun | Unassigned | 10h

list Code// output of printAllList

10 | Do Everything | Rabih | 12h

2 | Destroy code formatting | Thomas | 1h

2 | Have fun | Unassigned | 10h

list Description

1 | Write Description | AmirAli | 3h

list Misc

//output of printUserTasks for AmirAli

1 | Write Description | AmirAli | 3h

//output of printUnassignedTasksByPriority

2 | Have fun | Unassigned | 10h

//output of printAllUnfinishedTasksByPriority

1 | Write Description | AmirAli | 3h

2 | Destroy code formatting | Thomas | 1h

2 | Have fun | Unassigned | 10h