**Report**

1. **Introduction**

Laptop Simulator is an application that simulates a real laptop. It covers the basic functions of a laptop. The purpose of developing this Laptop Simulator is to learn how some functions of laptop are actually work. The functions of the Laptop Simulator includes the on/off button, home page, settings, notepad, install software, uninstall software, open software and music player.

For the A – is display screen. B – Is the on/off button, which can start and close the screen. Besides that, C – is the home page button, when we click into others software, if we need to go back to main page, we can just click the home page button. Next is D – which is the settings button. This button can change the wallpaper of the screen and check the description for the system information. E – Is the Notepad that enables users to record notes in the database as well as displaying the note on the screen of Laptop Simulator. After that, the F and G that is Install Software button and Uninstall Software button. Install Software interface will show the description of each application and user can choose to install their preferred software in the list. Uninstall Software interface enable user to uninstall the installed software in the Laptop Simulator. Furthermore, H – Is the Open Software button that the function can open the software which user has installed. Lastly is I – This is the music player button, which can let user play their favourite music and add their favourite music to the playlist.

As a conclusion for our laptop simulator design, is to let user experience how laptop application actually function.

1. **UML class diagram and designs for the GUIs**



1. **A description of how you designed and developed the final code**

We used object oriented programming method to develop this application. Each function in the application is develop separated by using classes. Each of the class inherits the JFrame class in order to develop the GUI of the application. The GUI components that we used included JButton, JLabel, JTextArea, JPanel and so on. This application also consists of action listeners, which always listen to the action performed by the user, such as click on the button.

Each class that has an interface will have a constructor that initialize all of the GUI components needed. We used different layout managers, such as BorderLayout and FlowLayout, in order to develop different layouts for our GUI components. We put some styling on our GUI components by changing its font family, font colour, font size and width in order to user friendly interface.

We also linked our application to a database by using DBHandler class with some imported external libraries given by lecturer. In the DBHandler class, there are methods that perform different SQL queries, such as SELECT from table and DELETE from table. Each queries are placed inside a try block as the execute query function may throw an exception when there is any error. When there is error while executing the query, the method will return a false and an error message will pop out to alert the user.

This application also included some additional classes in order to enhance the user experience. For example, we imported classes that needed in order to show current time in the application, such as “SimpleDateFormat”, “Calendar” and “Date”. We also included open source MP3 library in our project so that our application is able to play, pause and stop music files. We imported the classes needed for file chooser in order to allow user to upload images for wallpaper and MP3 files for music player.

1. **Suitable screen shots of the program in operation**

Figure 1 Laptop Simulator starting up with start-up animation



Figure 2 Home page of application

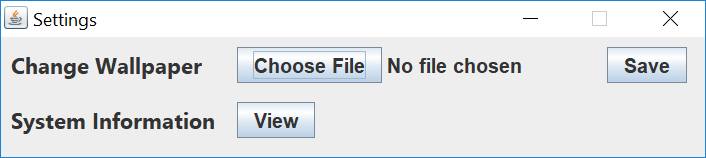


Figure 3 Settings interface

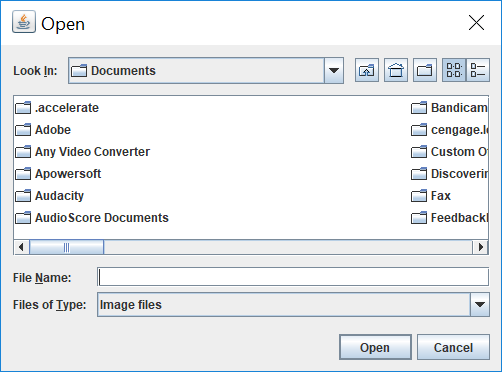


Figure 4 File Chooser interface

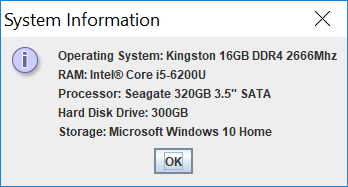


Figure 5 System Information interface

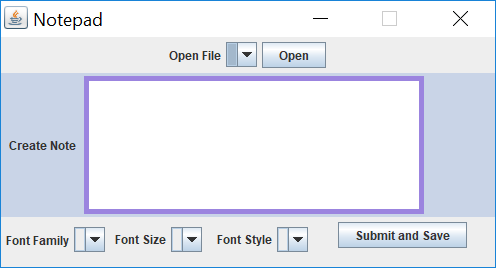


Figure 6 Notepad interface

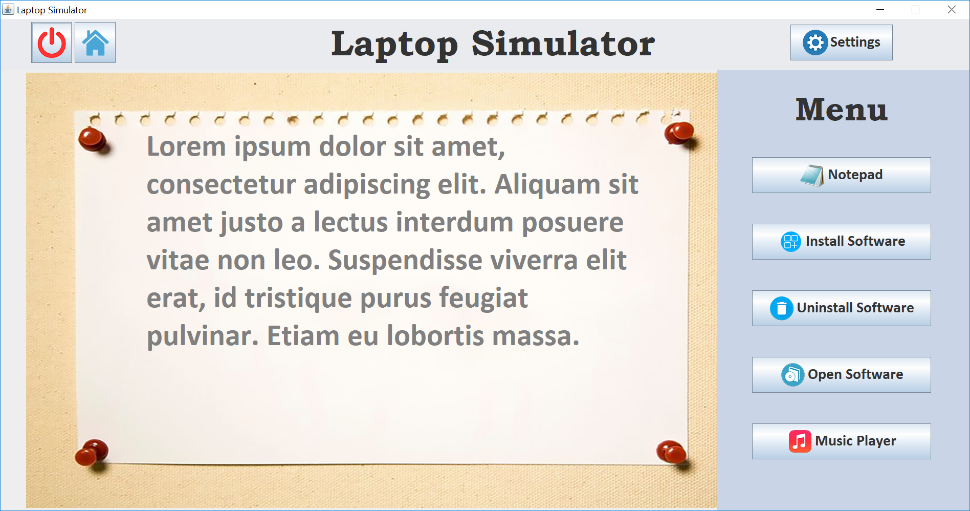
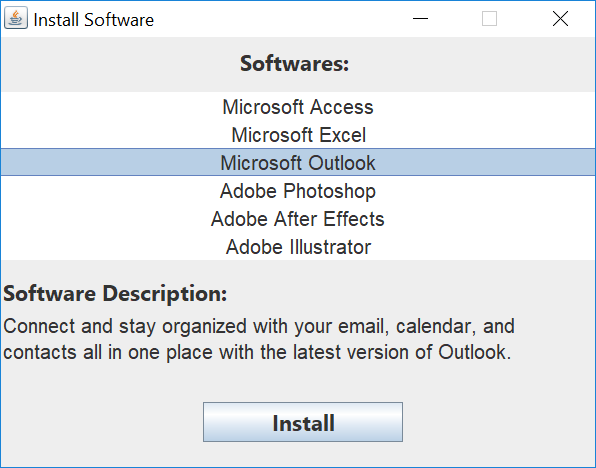
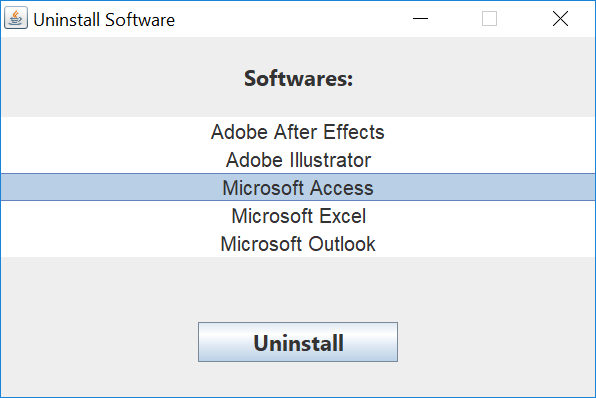


Figure 9 Uninstall Software interface

Figure 8 Install Software interface

Figure 7 Notepad’s text displayed on the screen

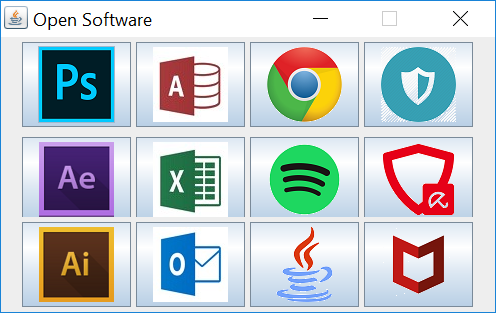
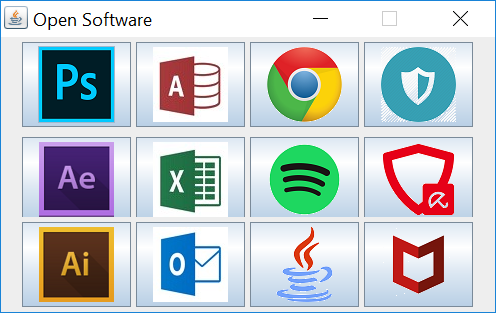
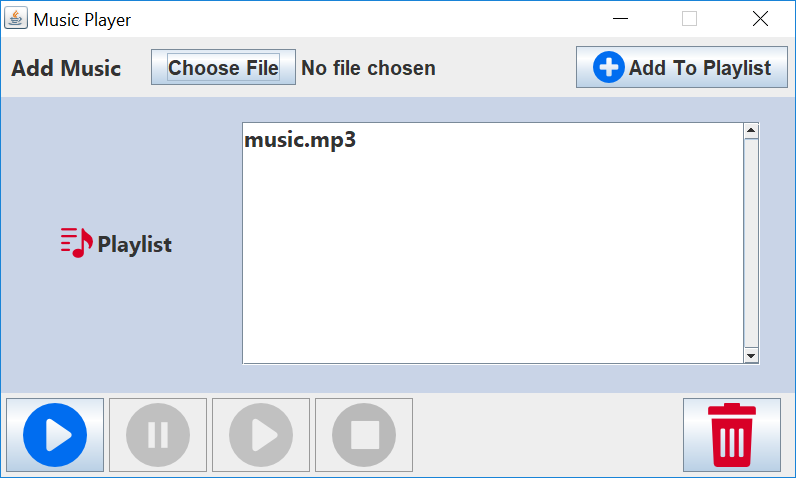


Figure 12 Music Player interface

Figure 11 Software opened appeared on screen

Figure 10 Open Software interface



Figure 13 Laptop Simulator shutting down with shut down animation

1. **Details of any faults and failures, strengths of your system**

The faults of this application is it has a slow response and execution speed on running a class. This may be caused by importation of various built-in classes and external libraries as well as redundancy of if-else statements. This application also not able to store the wallpaper location chose by the user once it change to another wallpaper as the database table always store only one record. For the notepad, users are not able to save the notes they written with specified name and the font styles for the note are not changeable. For install software function, users can only install the software listed on the list and adding of other software are not supported by the application. For the music player, there is no “Next” and “Back” button that enable users to play next song or previous song in the playlist.

The strength of the application is it can display the current time on the screen and the greetings (“Good Afternoon”) will be changed according to the time. For example, when the time is 10AM, the greeting will be “Good Morning”. Besides, the wallpaper changed by the user will be preserved even the application is closed and restarted again as the wallpaper location will be store in the database.