Contents

[GIT HUB repository 2](#_Toc145159209)

[Application Description 2](#_Toc145159210)

[Packages and Classes 2](#_Toc145159211)

[Class info 3](#_Toc145159212)

[Class and Methods 3](#_Toc145159213)

[User Guide 4](#_Toc145159214)

[Display the files of a directory 4](#_Toc145159215)

[Creating a file 7](#_Toc145159216)

[Deleting a File 8](#_Toc145159217)

[Searching a file 10](#_Toc145159218)

# GIT HUB repository

The project has been pushed to the following remote repository:

<https://github.com/stratosg2000/Simplelearn-Phase1-End-Project>

# Application Description

The application is named FileMagic and it includes a welcome message, a main menu and a business submenu. The application provides a) the ability to display the file contents of a user specified directory through main menu and three additional business options in the submenu which include creating a file, deleting a file and searching a file in a user specified directory.

When user is executing the program, he has the ability to perform any set of actions before exiting the program. For example, he can create 2 files of any name in two different directories.

The application guides through the menus to all features and requests from subscriber a) to insert the option for the action he wants to execute, as well as to insert the directory path for displaying the file contents and in business sub menu it additionally requests the filename for which the user desires to either create, delete or search according to his desire.

# Packages and Classes

The application uses the following packages and classes

1. The package **filemagic.com.main** includes **App** class.
2. The package **filemagic.com.util** includes **MainMenu**, **SubMenu** and **Welcome** classes.
3. The package **filemagic.com.service** includes **Service** class

# Class info

The application FileMagic has the following classes:

1. Class **App** includes the main method
2. Class **Welcome** provides the welcome message with the application details.
3. Class **MainMenu** provides the following 3 options of the application.

1: Display the files of a folder in ascending order.

2: Business level operation (for adding, deleting and searching a file).

0: To exit from the app.

1. Class **SubMenu** for presenting the business option of creating, deleting and searching a file

1: To create a file.

2: To delete a file.

3: To search a file.

0: To return to main menu.

1. Class **Service** for implementing the functionality required by the menus, thus it provides the method to display the files of a folder, creating a file, deleting a file and searching a file.

# Class and Methods

Each class provides the below methods:

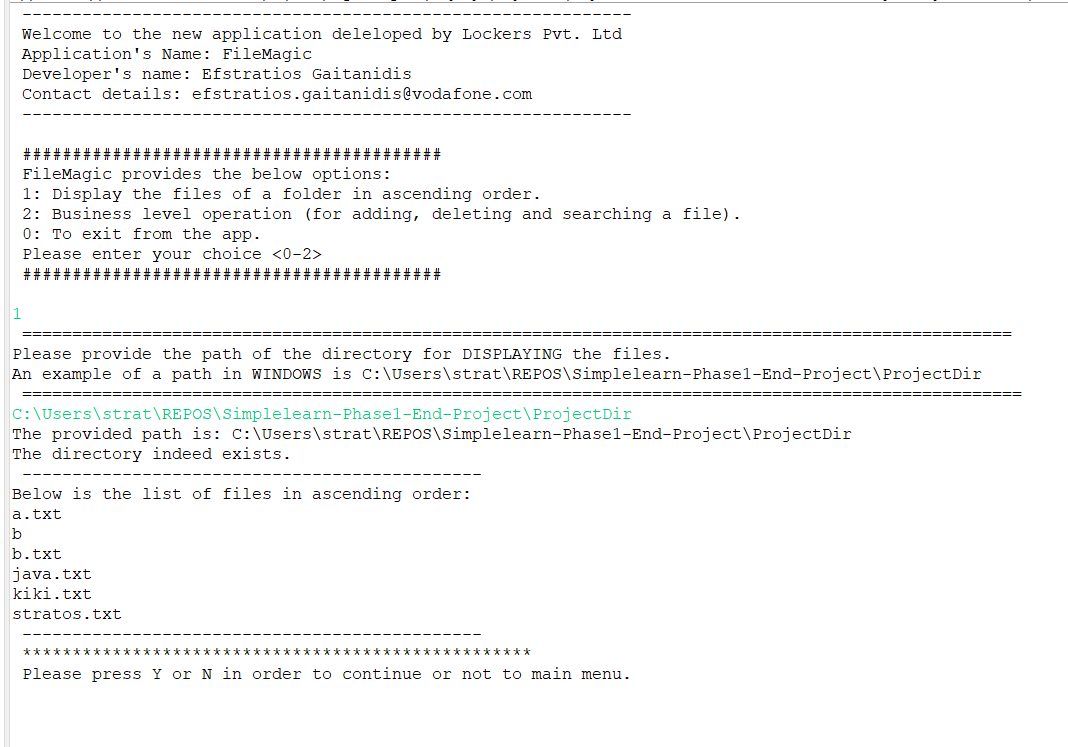
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | App | **Welcome** | **MainMenu** | **SubMenu** | **Service** |
| **Public** | main() | displayWelcome() | displayMainMenu | displaySubMenu() | serviceApp |
| **Private** |  |  | invalidMainMenuOptionMessage()  exitMainMenuMessasge() | invalidMainMenuOptionMessage()  exitSubMenuMessasge() | validateDirectory()  displayFiles()  retrieveFilename()  createFile()  deleteFile()  searchFile() |

# User Guide

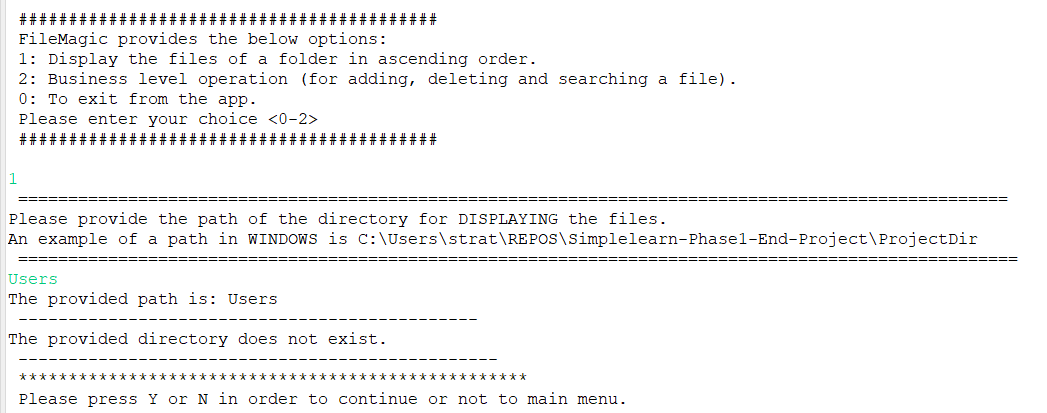
This user guide provides a description on how the user can use the application.

## Display the files of a directory

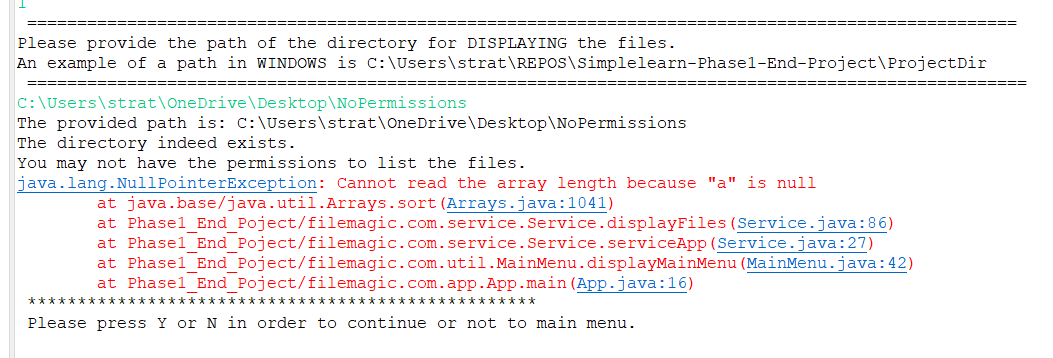
The user needs to press 1 in main menu and provide a valid directory in which he has the appropriate permissions.



If user provides an improper directory, then app will inform him that directory does not exist.

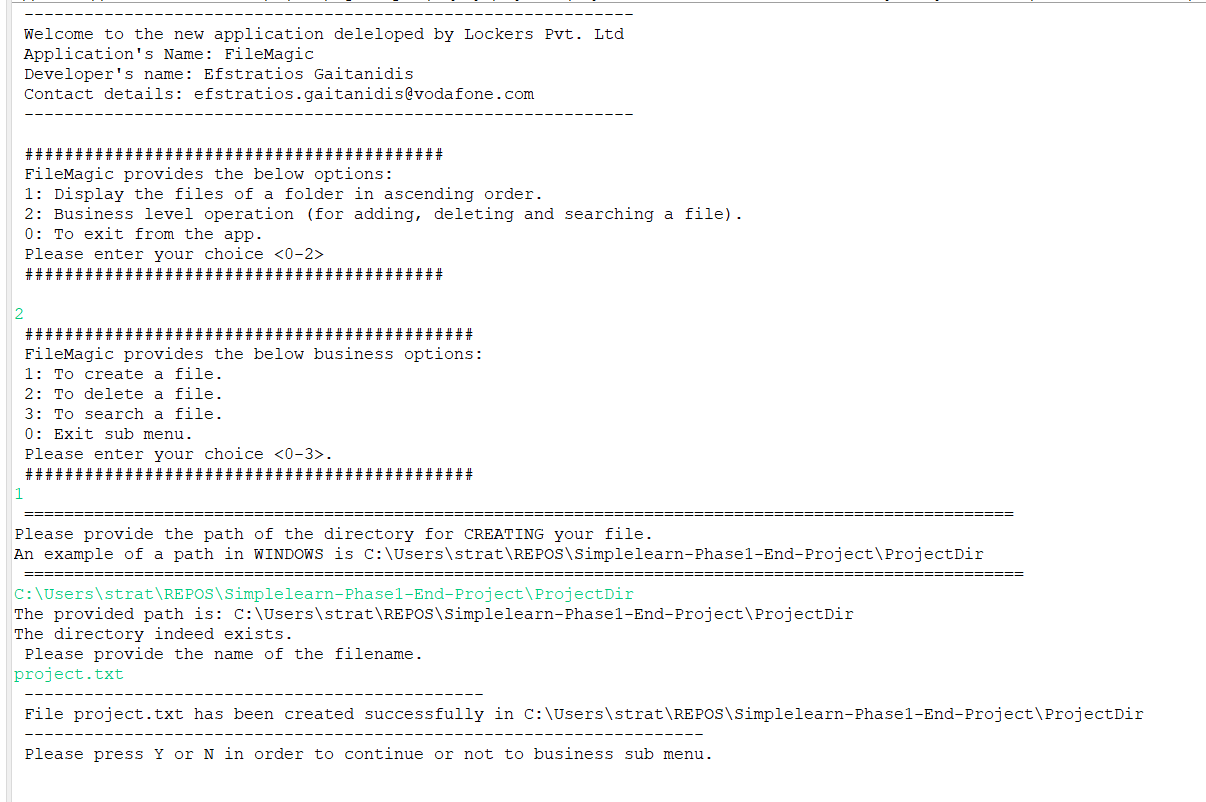


if user does not have the permission the following message will appear.

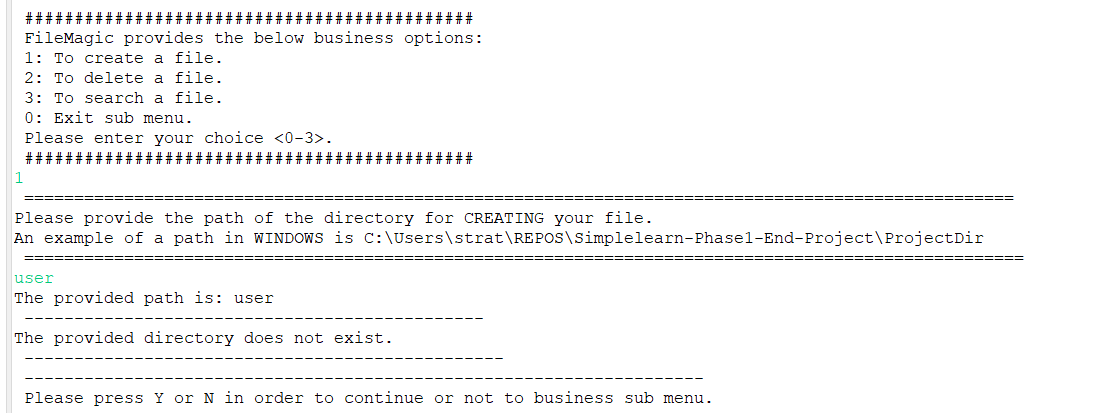


## Creating a file

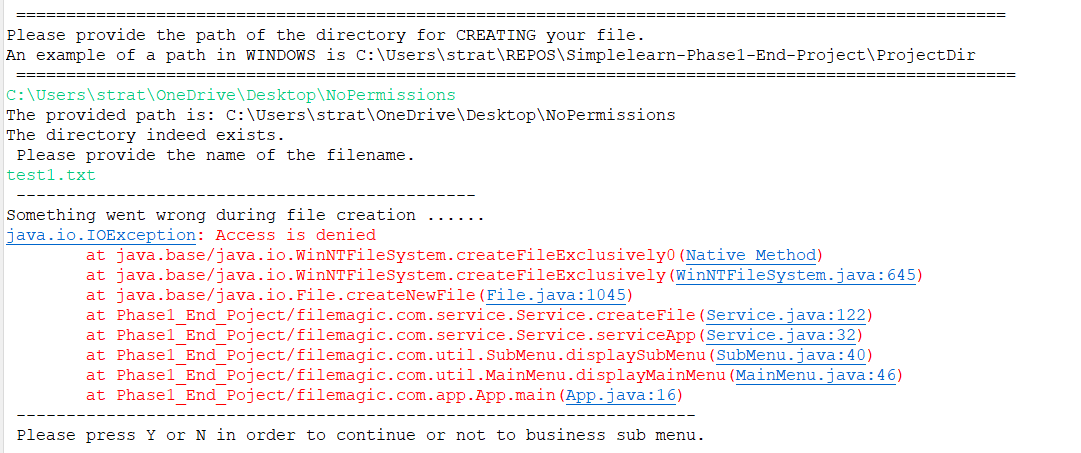
User needs to press 1, then 2, provide a valid directory and a filename



If user provide an incorrect directory, he will receive the relevant response

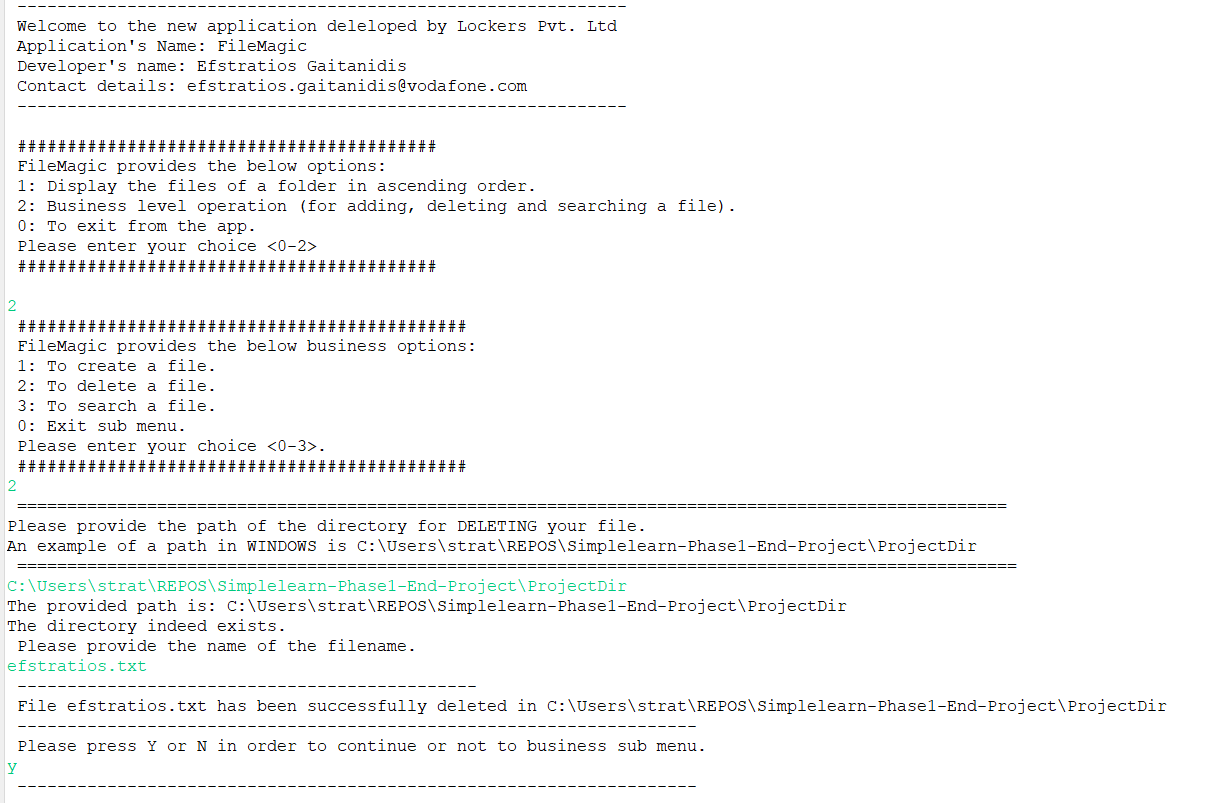


If user provides a directory with no permission, then an error message will be presented.



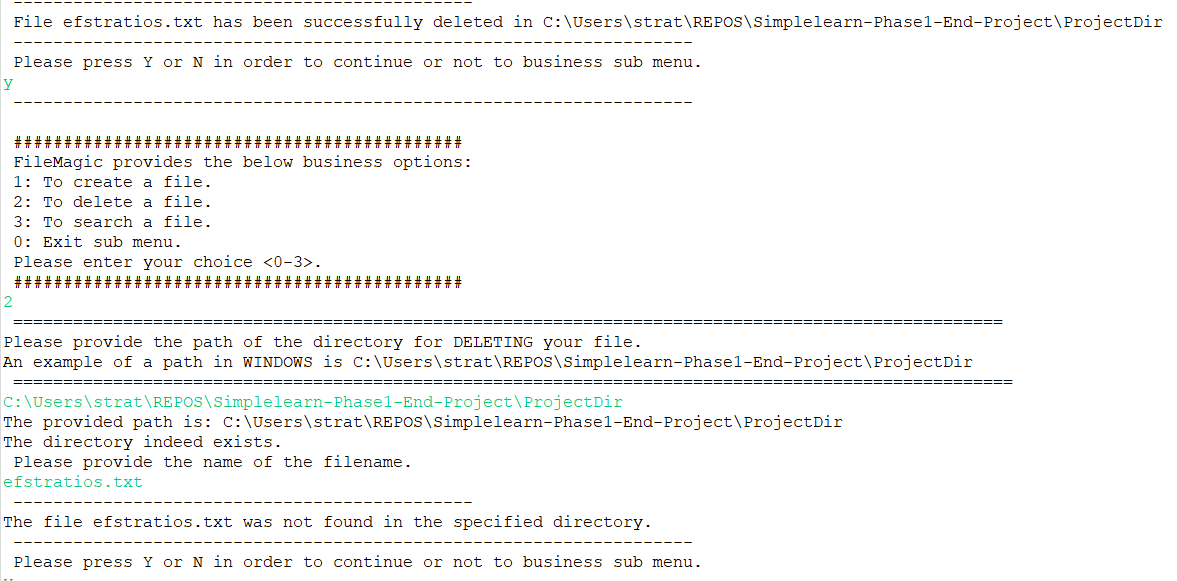
## Deleting a File

User needs to select 2 in main menu, then 2 in submenu, provide a valid directory and an existing filename.

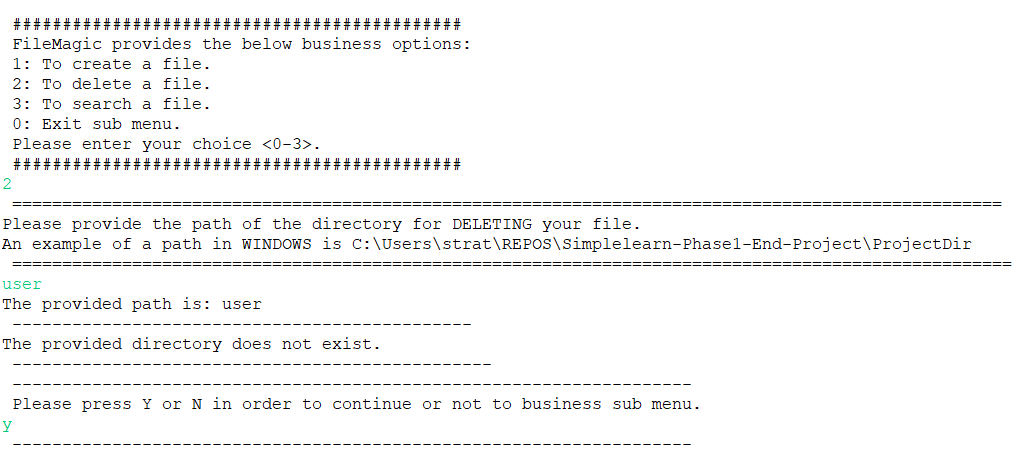


If user provides a file that does not exist it will appear an appropriate message, the same applies with the directory.

Here user tried to delete again the same file

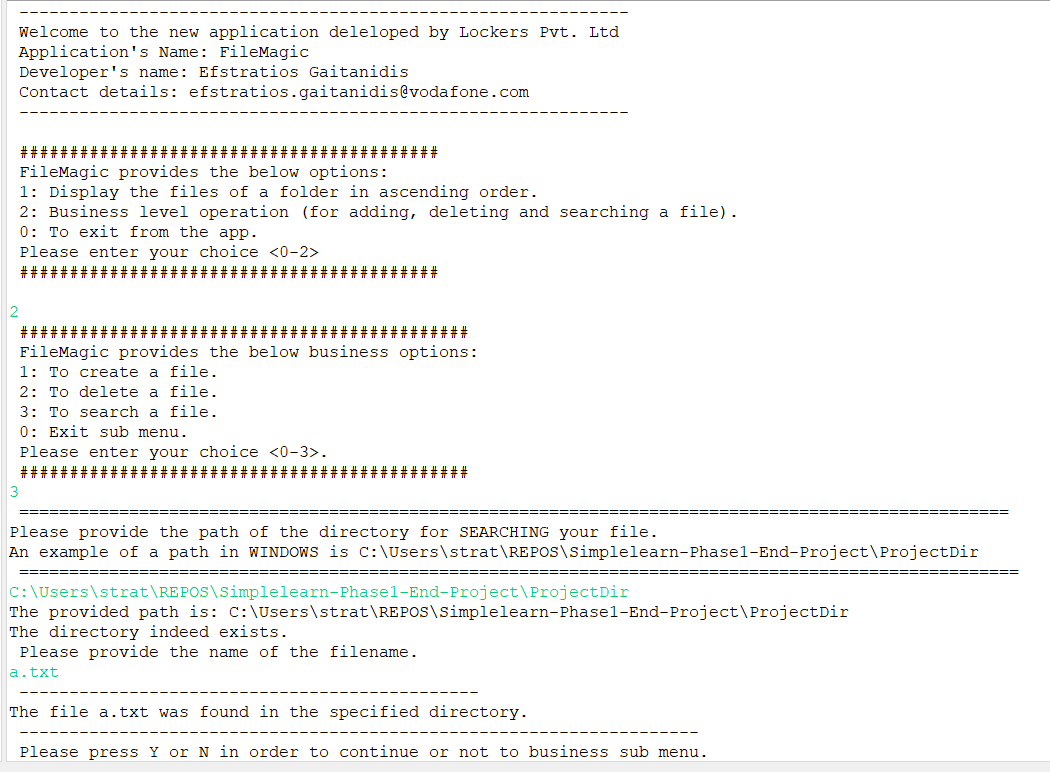


Here user provided an incorrect directory

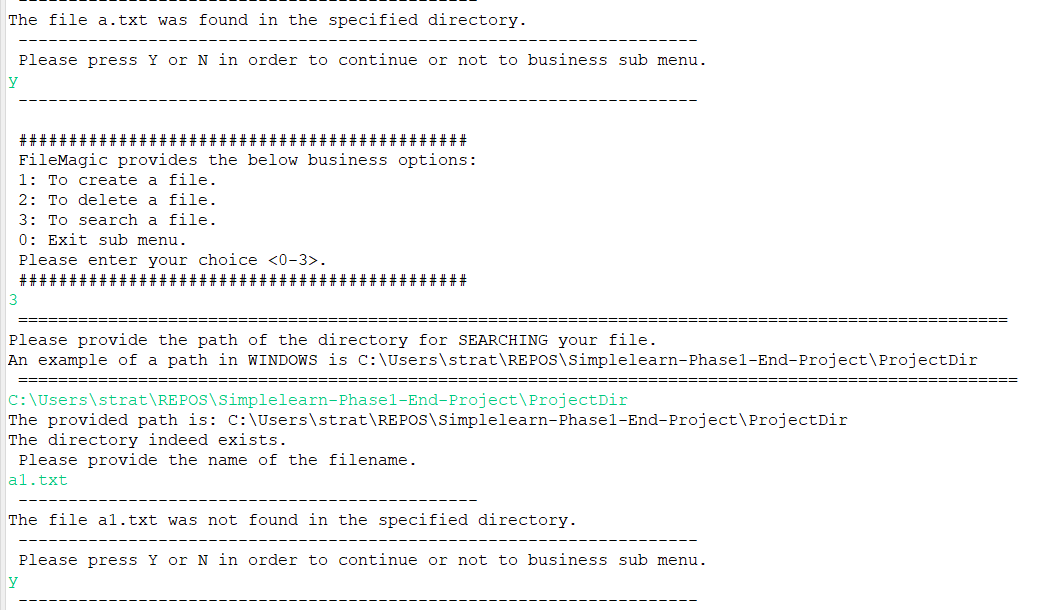


## Searching a file

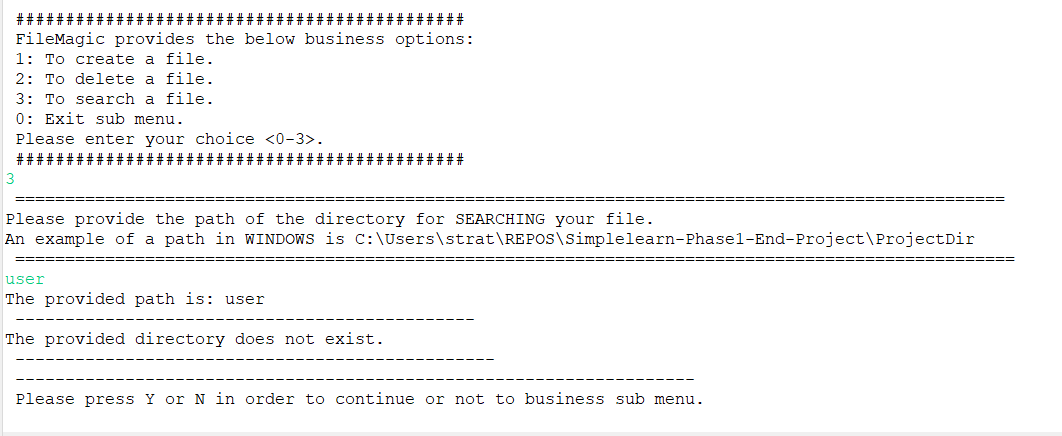
User needs to press 2 and then 3 and provide a valid directory and the filename to search.



In this case the file was not found.



And here the user provided an incorrect directory path

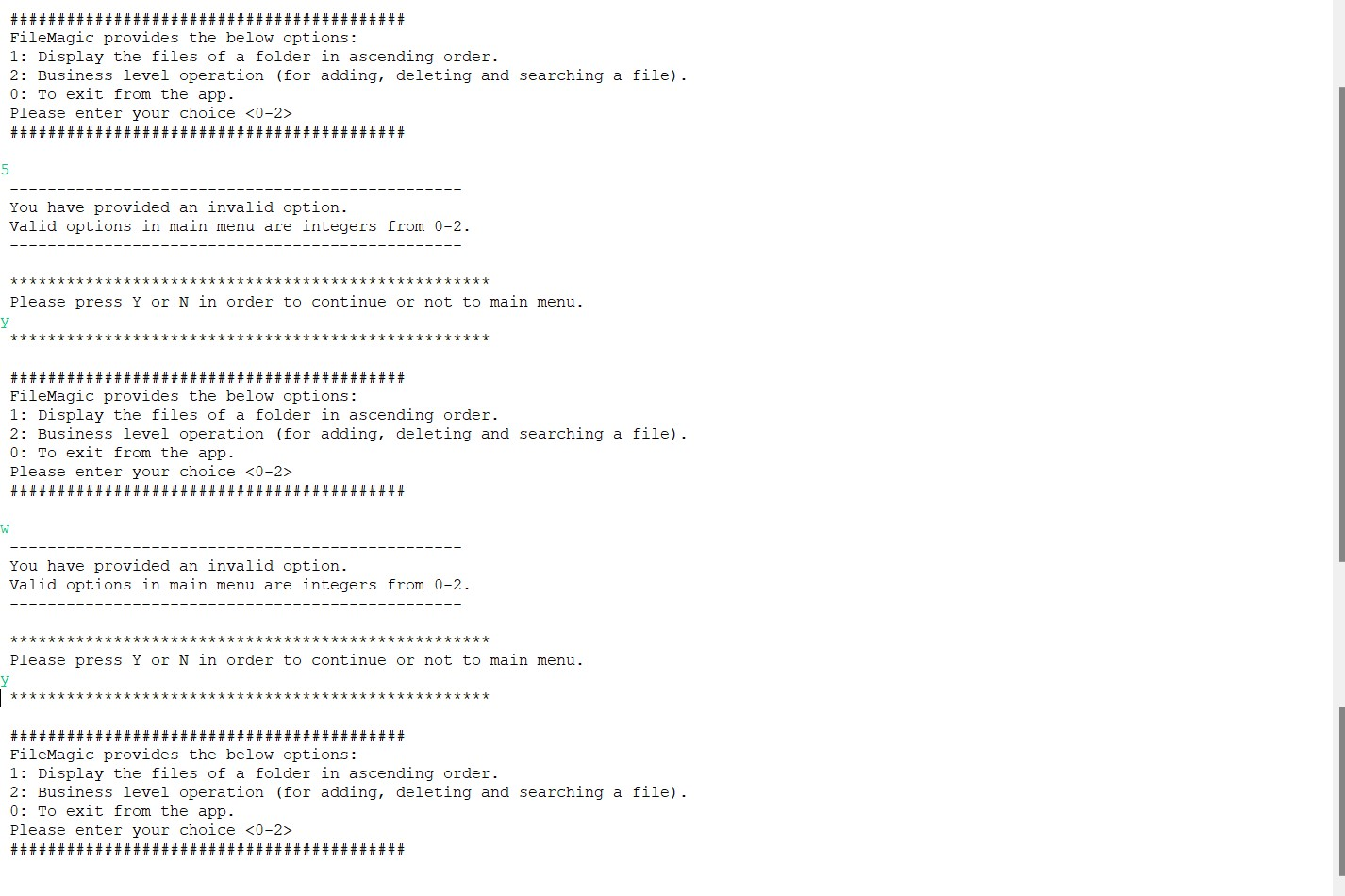


# Menus

**User provided a wrong option in main menu** (string or integer other than 0,1 and 2)

The App informs the user for the invalid option and ask him if he wants to stay in main menu

In this case the app will ask him if he wants to continue or not in main menu



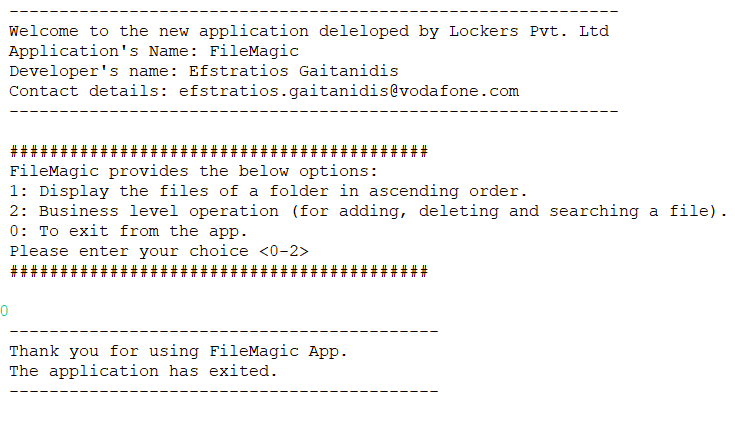
**User provided an incorrect input in business menu.**

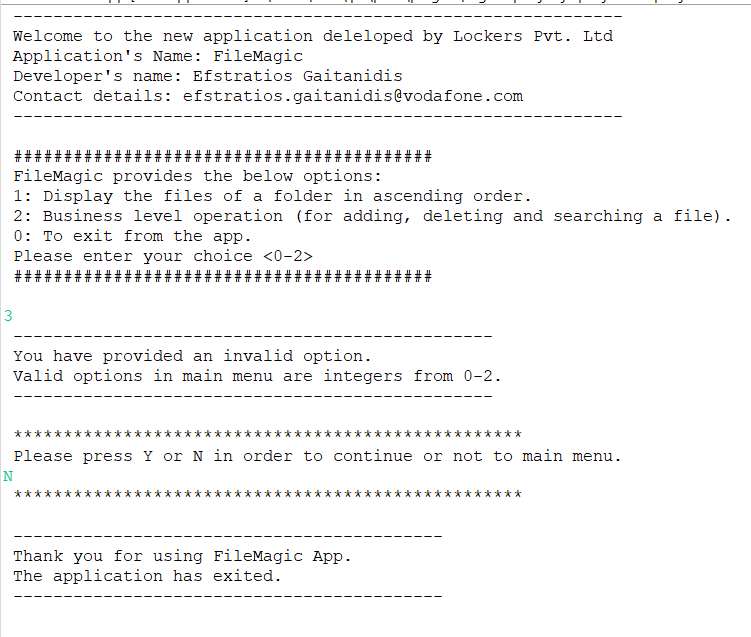
For example, he entered a sting or integer other than 3,2,1 and 0.

App will inform him and ask him to continue or not in business menu



User can exit with 0 in main menu or press N when app asks user if he desires to continue or not.





Similarly happens with submenu

With 0 the user exits to main menu and N when app asks user to continue or not in submenu (also known as business menu)





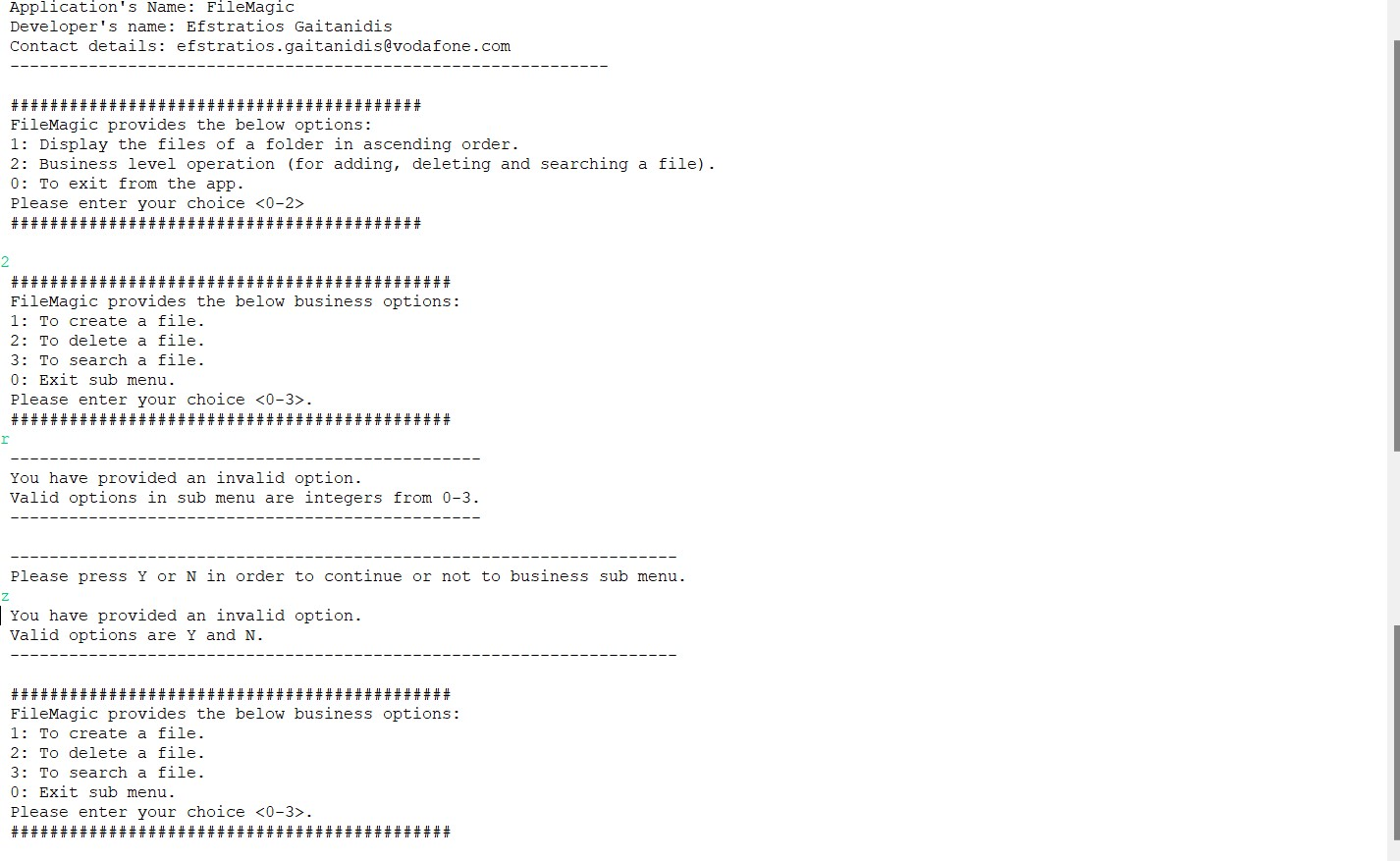
After the completion of an action, the app will ask in main menu and sub menu if user wants to continue or not. Valid input are: Y,y and N,n.

If user provides an invalid input, the app with present the appropriate message and he will remain in the menu

Invalid confirmation to continue in main menu



Invalid confirmation to continue in submenu



.