AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)

FACULTY OF SCIENCE & TECHNOLOGY

DEPARTMENT OF CSE

Programming in Python

Spring 2022-2023

Section: A

MID TERM PROJECT ON

Computer Lab Management Console Application using Python

Supervised By

Dr. Akinul Islam Jony

Submitted By:

Name	ID	Contribution
Muhammad Shahriar Zaman	**20-41840-1**	**All**

Date of Submission: **February 28th, 2023**

TABLE OF CONTENTS

Project Overview:

For this project we have been instructed to develop a console based python project of a Lab Computer Management System. It can be related to the issues we face in university computer laboratories.

Our program will provide standard functionalities like adding, updating and deleting records of computers. Users will also be able to search for specific computers and then perform functions on them. We have also added the functionality to store these data permanently.

Project Solution Design:

To accomplish our project goals firstly we will analyze the requirements set for this project. Then we will create models of our project using UML which will simplify our task plans.

After that we will keep developing this program incrementally until all the functionalities have been apparently fulfilled. Lastly the program will be put through a rigorous testing phase before final submission.

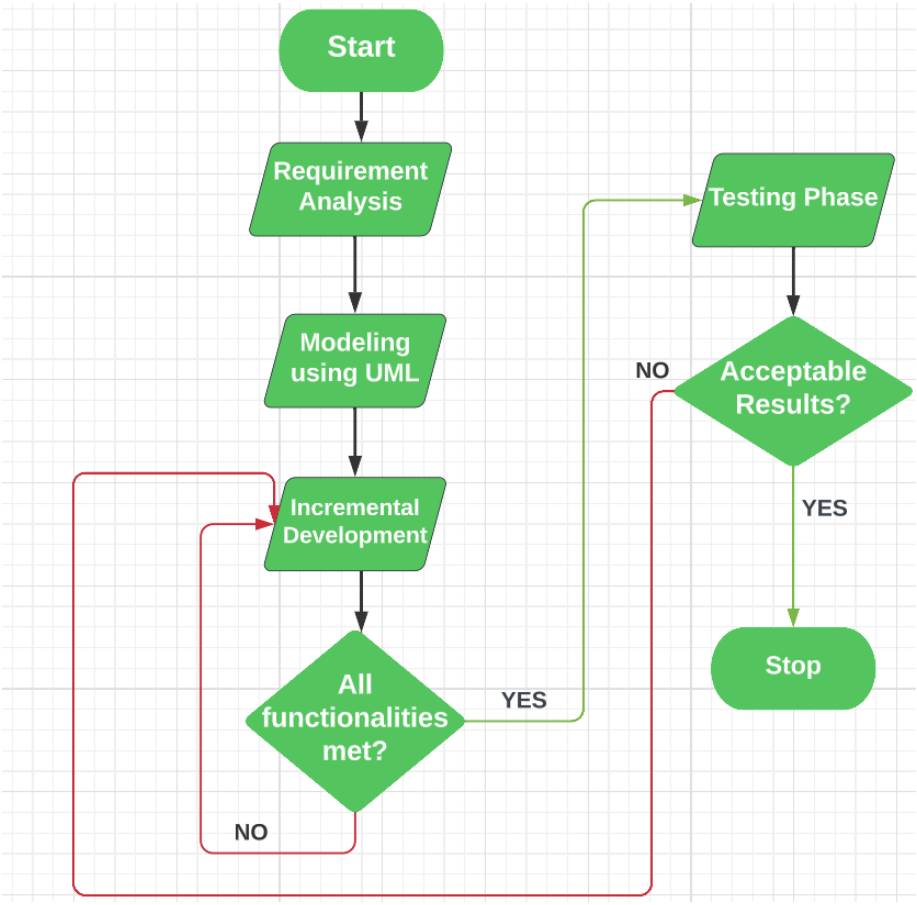


Fig. 1: Flow chart depicting our solution design

Project Implementation: Entire source code of our project=>

```
#Module containing the LabPC class

class LabPC:
    """This class will contain all the attributes and methods"""

    all_pc=[] #Class variable list to store all objects of this class



    def __init__(self,pc_number,pc_os,pc_status):
        """ This method works like a constructor
            and assigns the instance variables (number, OS, Status) """

        self.number=pc_number
        self.os=pc_os
        self.status=pc_status

        LabPC.all_pc.append(self) #Appending this object to the class variable list all_pc[]


        print(f"\n   New {self.os} PC registered with values")



    def add_pc(cls):
        """""" This method is for registering a new PC"""

        print("\n---------------------------------")
        print("       PC Registration")
        print("---------------------------------")  #Taking User Input
        pc_number=input("   Enter PC Number          : ")
        pc_os=    input("   Enter PC Operating System : ")
        pc_status=input("   Enter PC Status          : ")

        result_pc=LabPC.pc_number_checker(pc_number) #Checking if any existing PC has
the same number

        if result_pc==0:                        #In this case no existing PC has this number, so we
can create an object
            lpc=LabPC(pc_number,pc_os,pc_status)     #Creating object

        else:
```

```
            print("\n   Failure: PC adding failed")
            print("   Error: PC with same number already exists\n")
            print("      1) Update number of the existing PC")
            print("      2) Delete the existing PC")
            print("      3) Cancel adding this PC")
            key=input("\n      Choose any one : ")

            if key=="1":
                new_pc_num=input("\n   Enter new PC Number    :")
                LabPC.update_pc(result_pc,new_pc_num,result_pc.os,result_pc.status)
                lpc=LabPC(pc_number,pc_os,pc_status)


            elif key=="2":
                LabPC.delete_pc(result_pc)
                lpc=LabPC(pc_number,pc_os,pc_status)


            elif key=="3":
                LabPC.main_menu()



    def all_pc_info(cls):

        """ This method will print information
            of all the pc from th all_pc[] list (class variable)"""

        if len(LabPC.all_pc)!=0:
            print("\n-------------------------")
            print("      All PC Info")
            print("-------------------------")

            for pc in LabPC.all_pc:
                print(f"    PC Number : {pc.number}")
                print(f"    PC OS     : {pc.os}")
                print(f"    PC Status : {pc.status}\n")
        else:
            print("No PC found, add some PC to this lab first!")


    def about(cls):
        """This method will simply print information
           of the developer and the course"""
```

```
    print("\n-------------------------")
    print("        About Us")
    print("-------------------------")
    print("   University : AIUB")
    print("   Course     : Programming in Python")
    print("   Section    : A")
    print("   Semester   : Spring 2022-23")
    print("   Teacher    : Dr. Akinul Islam Jony")
    print("   Developer  : Muhammad Shahriar Zaman")
    print("   ID         : 20-41840-1")

    key=input("\n   Press any key to go back: ")



def pc_number_checker(pc_number):

    """We are treating the number of a PC as its identity,
       so this method will check if it is unique"""

    status=1
    for pc in LabPC.all_pc:
        if pc.number==pc_number:
            status=0
            break
    if status==0:
        return pc
    else:
        return 0



def search_pc(pc_number):
    """Showig search results by matching with PC Number"""

    status=0
    for pc in LabPC.all_pc:
        if pc.number==pc_number:
            status=1
            print("\n-------------------------")
            print("      Search Result:")
            print("-------------------------")
            print(f"   PC number          : {pc.number}")
            print(f"   PC Operating System : {pc.os}")
```

```
            print(f"   PC Status          : {pc.status}")
            result_pc=pc
            break

    if status==1:
        return result_pc
    else:
        print(f"   No results found as PC number {pc_number}")
        return 0



def update_pc(pc,new_pc_num,new_pc_os,new_pc_status):

    """This method will update a PC
       with new values for attributes"""

    if LabPC.pc_number_checker(new_pc_num)==0:
        old_pc_number=pc.number

        pc.number=new_pc_num
        pc.os=new_pc_os
        pc.status=new_pc_status

        print(f"   Updated PC no. {old_pc_number} successfully\n")

    else: # PC number already taken
        print(f"Update failed, the number {pc.number} has already been taken\n")




def delete_pc(pc):

    print(f"\n   Deleted PC no. {pc.number} successfully\n")
    LabPC.all_pc.remove(pc)
```

```
def store_all_data():

    try :
        filename ="all_pc_info.txt" #Storing all the PC in this text file in CSV format

        with open(filename,'w') as file_object:
            for pc in LabPC.all_pc:
                file_object.write(f"{pc.number},{pc.os},{pc.status}\n")
        print("\n   All PC information has been stored in a text file successfully")

    except FileNotFoundError:
        print("\n   Writing on text file failed")




def main_menu():
    """This the main menu, user will access everything of this program through this"""

    print("\n---------------------------------------------------")
    print("     Computer Lab Management Application")
    print("---------------------------------------------------")
    print("Choose any option:\n")
    print("1. Add a New PC") # To create a new PC object
    print("2. Search existing PC") # To search among all the existing PC objects by PC
number
    print("3. Update existing PC") # To update any of the existing PC objects
    print("4. Show All PC Information") # To view information of all the existing objects
    print("5. Store All PC's Information")  # To store all the information
    print("6. Delete existing PC") # Delete the object of an existing PC
    print("7. About this Project")   # Background information of this project
    print("8. Quit Application") # To quit this program


    key=input("\n  Select any one: ") # Which option the user has selected

    if key=="1": #Adding a new PC
        LabPC.add_pc(LabPC)

    elif key=="2": # Searching for a PC
        pc_num=input("\n   Enter PC Number to be Searched :")
        LabPC.search_pc(pc_num)
```

```
elif key=="3": #Updating a PC
    pc_num=input("\n   Enter PC Number to be Updated :")
    result_pc=LabPC.search_pc(pc_num)

    if result_pc!=0:
        print("   For this selected PC,\n")
        new_pc_num=   input("   Enter new PC Number          :")
        new_pc_os=    input("   Enter new PC Operating System :")
        new_pc_status=input("   Enter new PC Status          :")

        LabPC.update_pc(result_pc,new_pc_num,new_pc_os,new_pc_status)



elif key=="4": # Showing information of all PC
    LabPC.all_pc_info(LabPC)


elif key=="5": # Storing all the data in a text file
    LabPC.store_all_data()


elif key=="6": #Deleting a PC"
    pc_num=input("\n   Enter PC Number to be Deleted :")
    result_pc=LabPC.search_pc(pc_num)

    if result_pc!=0:
        LabPC.delete_pc(result_pc)


elif key=="7":  # About developers
    LabPC.about(LabPC)

elif key=="8": #Quitting the program
    quit()
```

```
# Separate file containing the main code

import lab_pc_module

#lab_pc_module is the module containing the class
#LabPC is the class in this module, we have imported it over here

lpc= lab_pc_module.LabPC("1","Windows 7","Running") # Creating some sample objects
lpc2=lab_pc_module.LabPC("2","MacOS 10.12","Running")
lpc3=lab_pc_module.LabPC("3","Windows 10","Damaged")

while 1<2: # Calling the main menu repeatedly
    lab_pc_module.LabPC.main_menu()
```

Application Overview:

i) Main Menu: Whenever our program is opened, the user will firstly view this page and all the functionalities can be selected from here.

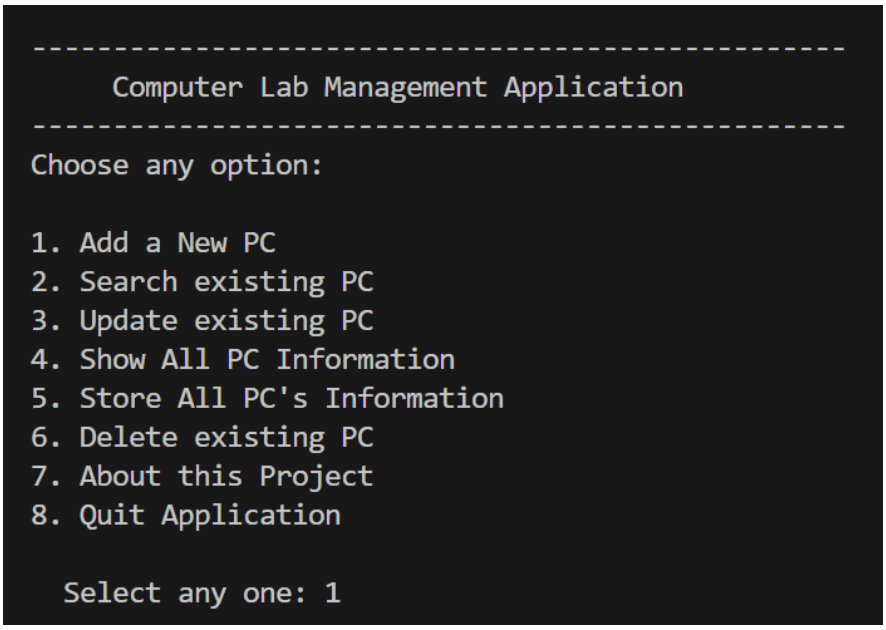


Fig. 2: Main Menu of the program

ii) Adding a New PC: The user will be able to register new PCs to the program. While registering, attributes such as PC Number, OS and Status have to be input.

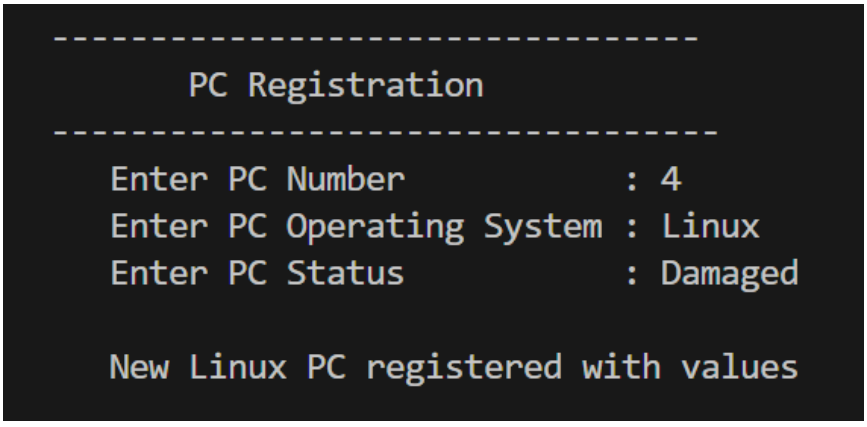


Fig. 3: Adding a PC

iii) Trying to add a PC with a number of an Existing PC:

We are treating the PC number as unique, so duplication won't be allowed. When the user is trying to register a PC with an already existing PC number, the following options will be shown.

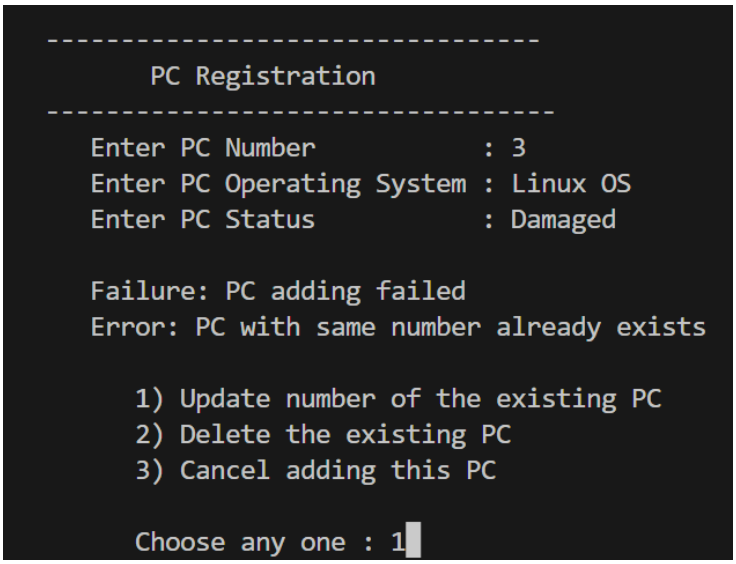


Fig. 3: Adding a PC with an existing PC number

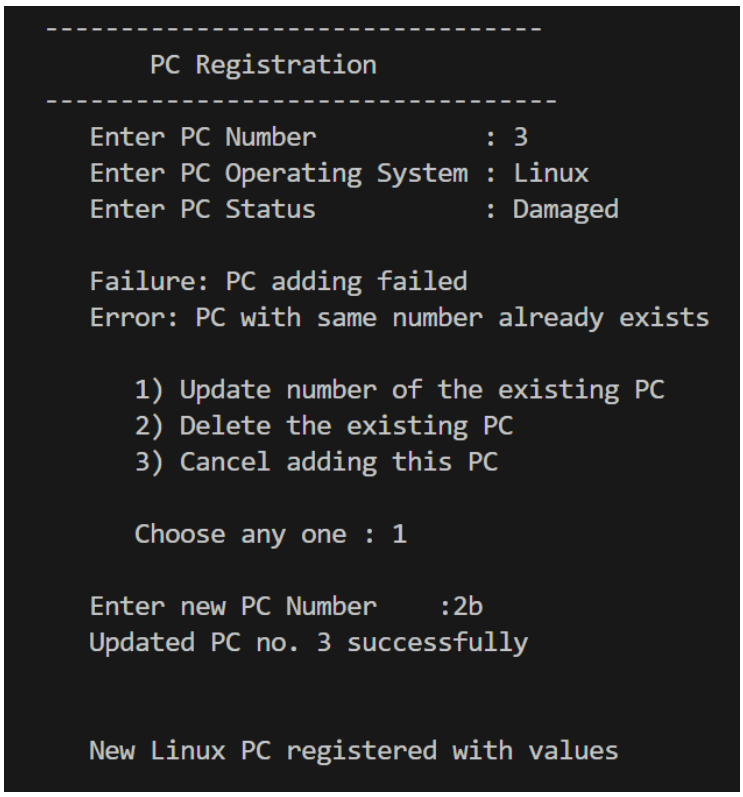


Fig. 4: Updating the existing PC's number to add a new PC

```
--------------------------------
      PC Registration
--------------------------------
   Enter PC Number           : 3
   Enter PC Operating System : Windows 11
   Enter PC Status           : Slightly Damaged

   Failure: PC adding failed
   Error: PC with same number already exists

      1) Update number of the existing PC
      2) Delete the existing PC
      3) Cancel adding this PC

      Choose any one : 2

   Deleted PC no. 3 successfully


   New Windows 11 PC registered with values
```

Fig. 5: Deleting the existing PC to add a new PC

iii) Searching a PC by number: Every PC we register is stored in a class variable list. When searching, our program will use linear search algorithm.

```
  Select any one: 2

   Enter PC Number to be Searched :2b

--------------------------
     Search Result:
--------------------------
   PC number           : 2b
   PC Operating System : Windows 10
   PC Status           : Damaged
```

Fig. 6: Searching existing PCs

iv) Updating a PC: When updating, the user first has to enter the PC number of his desired PC, search result will be shown for this number and then the user will be able to update the attributes.

```
   Enter PC Number to be Updated :3

-------------------------
      Search Result:
-------------------------
   PC number           : 3
   PC Operating System : Windows 11
   PC Status           : Slightly Damaged
   For this selected PC,

   Enter new PC Number           :3a
   Enter new PC Operating System :Windows 10
   Enter new PC Status           :Working
   Updated PC no. 3 successfully
```

Fig. 6: Searching existing PCs

v) Storing all Information: This functionality will store all the information of the computers into a text file

```
Select any one: 5

 All PC information has been stored in a text file successfully
```

Fig. 7: Storing information of all existing computers

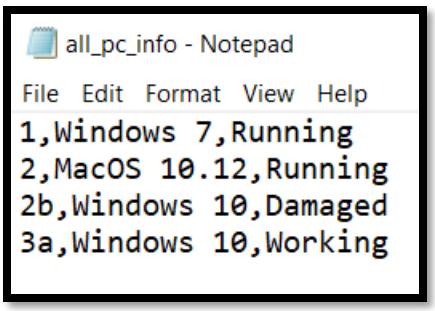


Fig. 8: PC information stored in a text file

vi) Viewing All PCs: All the PCs and their information stored in the class variable list will be shown through this functionality.

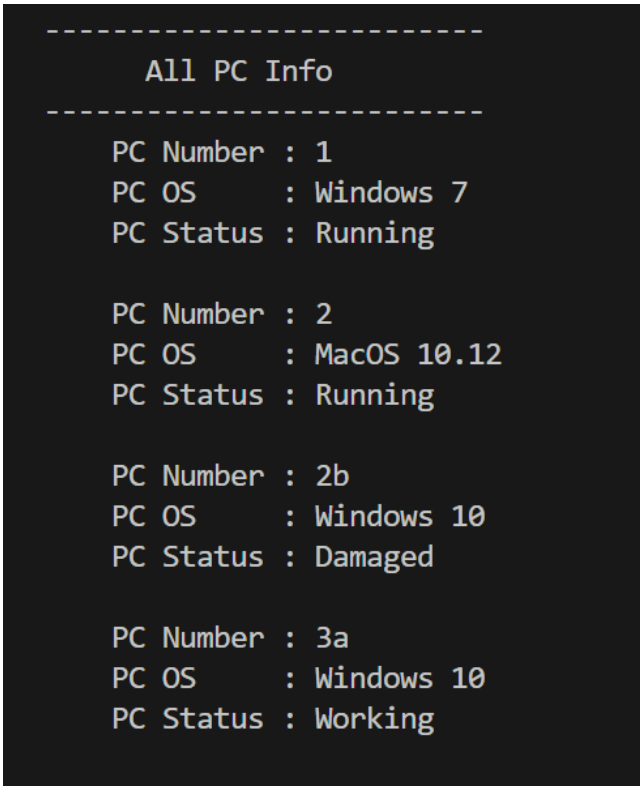


Fig. 9: Viewing information of all Computers

vii) Deleting a PC: Like updating, users will first have to search for their desired PC and then delete it.

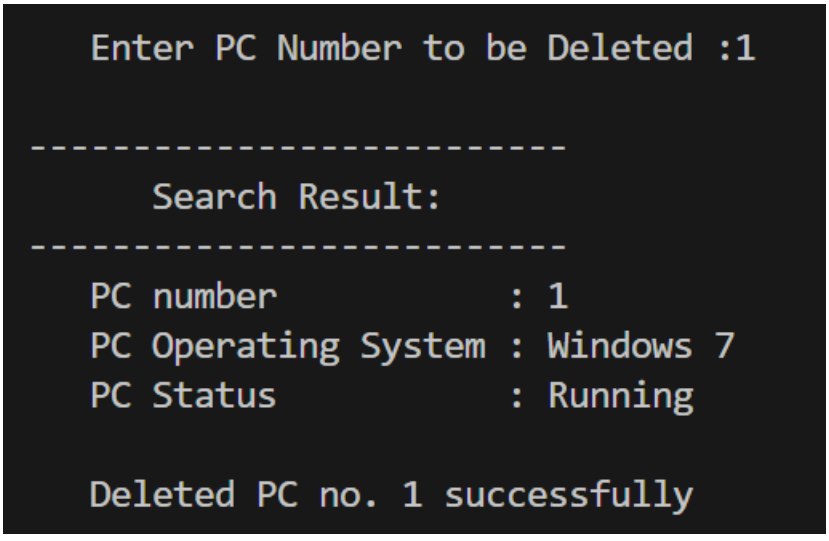


Fig. 10: Searching and deleting a PC

viii) Showing About Information: This will show the information of the institution, course, teacher, student etc. behind this project.

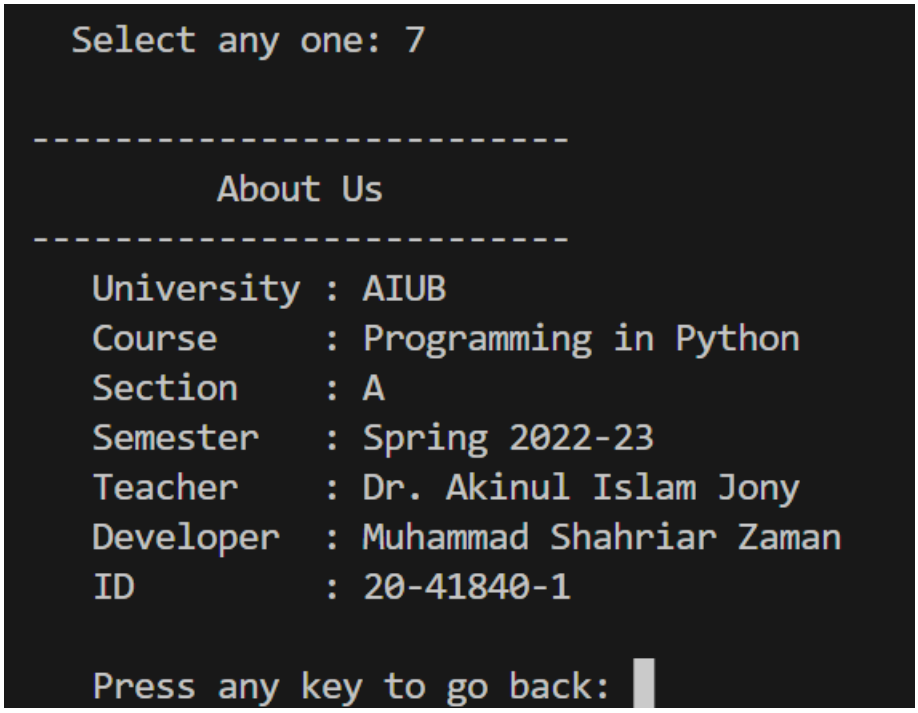
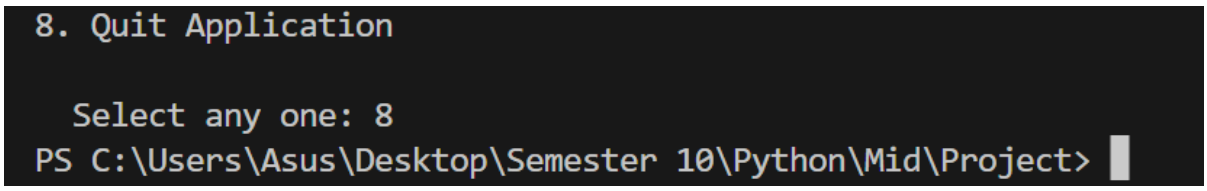


Fig. 11: Viewing background information of this project

ix) Quitting the application: This functionality will close the application



Note: Our project can be downloaded from the following link.

https://drive.google.com/drive/u/0/folders/1YxzyBDhS_tQxhx1Q58XhE5OQ9kGoYHMU