University of Westminster & Informatics Institute of Technology

Degree: BEng(Hons) in Software Engineering

Unit Code and Description:
(2023) 5COSC019C Object Oriented Programming and Design

Module Leader: Mr.Saman Hettiarachchi.

Assignment: Course Work 2022/23

Assignment Type: Individual Course Work.

Tutorial Group : B

Student Name: S.Mohanaranjan.
Student ID
      IIT : 20200607
      UOW : 18705841 / W1870584

Coursework Report – 5COSC019C Object Oriented Programming

Student Name: Mohanaranjan
Student ID: w1870584

Have you submitted the <u>video with the demonstration</u> of your system?   ✔ Yes   ☐ No

Phase 1 – Design and classes implementation

Task	Did you attempt the task?	Student's comments (To which extent you implemented the task? Have you encountered any problems or issue?)
Design a UML Use Case Diagram of your system (submitted in a separate file).	✔ Yes ☐ No	Completed
Design a UML Class Diagram of your system (submitted in a separate file).	✔ Yes ☐ No	Do up to known knowledge
Implementation Class Person	✔ Yes ☐ No	Implemented
Implementation Class Doctor	✔ Yes ☐ No	Implemented
Implementation Class Patient	✔ Yes ☐ No	Implemented
Implementation Class Consultation	✔ Yes ☐ No	Implemented
Implementation Interface WestminsterSkinConsultationManager	✔ Yes ☐ No	Implemented

Phase 2 – Console menu implementation

Task	Did you attempt the task?	Student's comments (To which extent you implemented the task? Have you encountered any problems or issue?)
Add a doctor in the system with all the relative information (max 10 doctors)	✔ Yes ☐ No	Did the task

Delete a doctor from the system selecting the medical licence number. Display a message to confirm he/she has been removed and the total number of doctors in the centres.	✔Yes ☐ No	Did the task
Print on the screen the list the doctors in the centre with all the relative information. The list should be ordered alphabetically.	✔Yes ☐ No	Did the task
Save in a file entered by the user so far. The user should be able to load back the information running a new instance of the application.	✔Yes ☐ No	Did the task

Phase 3 – GUI Implementation

Task	Did you attempt the task?	Student's comments (To which extent you implemented the task? Have you encountered any problems or issue?)
Doctor list visualisation. Sorting alphabetically.	✔Yes ☐ No	Sort by firstname done through the sort button
The user can select a doctor and add a consultation.	✔Yes ☐ No	Did the task
In the consultation the user can add all the patient details.	✔Yes ☐ No	Did the task
The user can select the date/time of the consultation considering that a doctor cannot have more than one consultation at the time.	✔Yes ☐ No	Did the task as option to type time may have select by other at same time
The user can enter and save the cost for the consultation. (£25 per hour and only the first one £15).	✔Yes ☐ No	Did the task as using patient id and cost is determined
The user can add some notes (text information or images). This information has been encrypted.	✔Yes ☐ No	Notes can be added but not images or encrypted

Phase 4 – Testing and system validation

Task	Did you attempt the task?	Student's comments (To which extent you implemented the task? Have you encountered any problems or issue?)
Test plan. (Submitted in a separate file).	✔Yes ☐ No	Tested in GUI and Console
Implementation of an automated unit test for each scenario in the console menu.	☐Yes ✔ No	Not done
Error Handling across all the code, input validation and code quality.	✔Yes ☐ No	Try with handling errors

Consultation

- month : int
- date : int
- hour : int
- minutes : int
- consultationCost : Double
- consultationDate : LocalDate Time
- notes : string

+ consultationDate() : LocalDate Time
+ consultationCost() : Double
+ getConsultationDate() : LocalDate Time
+ getConsultationCost() : Double
+ getDoctor() : string
+ getPatient() : string
+ getNotes() : string
+ setConsultationCost() : Double
+ setConsultationCost() : Double
+ setDoctor() : string
+ setPatient() : string
+ setNotes() : string

Person

- name : string
- surname : string
- dateOfBirth : LocalDate
- mobileNumber : int

+ getName() : string
+ getSurname() : string
+ getDateOfBirth() : LocalDate
+ getMobileNumber() : int
+ setName() : string
+ setSurname() : string
+ setDateOfBirth() : LocalDate
+ setMobileNumber() : int

SkinConsultationManager

+ serializingDocObject()
+ serializingConObject()
+ deserializingConObject()
+ deserializingDocObject()
+ addDoctor()
+ deleteDoctor()
+ displayDoctorList()

Patient

- uniqueId : int

+ getUniqueId() : int
+ setUniqueId() : int

Doctor

- medicalLicenseNumber : int
- specialisation : string

+ getMedicalLicenseNumber() : int
+ getSpecialisation() : string
+ setMedicalLicenseNumber() : int
+ setSpecialisation() : string

WestminterSkinConsultationManager

- doctorList : Doctor[]
- consultationList :
- displayAranged :
- displayDoctorList :

+ serializingDocObject()
+ serializingConObject()
+ deserializingConObject()
+ deserializingDocObject()
+ addDoctor()
+ deleteDoctor()
+ displayDoctor()
+ getConsultationList()
+ setConsultationList()
+ getDoctorList()
+ setDoctorList()
+ displayArranged()

Westminster Skin Consultation Manager

Display Error Message

<<includes>> Add Doctor

<<includes>>

Delete Doctor

<<includes>>

<<extend>>

Save Doctor <<includes>> Manage Doctors

<<includes>>

Providing Details

Display sorted / load Doctor

<<includes>>

Details of Doctor

<<includes>>

Manage Consultations

View Doctor <<includes>>

<<includes>>

Book Consultation

<<includes>>

View Consultation

Doctor

Person

Patient

Skin Consultation Center