UNIVERSITY OF WESTMINSTER

INFORMATICS INSTITUTE OF TECHNOLOGY

# 5COSC019C: Object Oriented Programming

# Coursework Report

Student Name: Irushi Nicoline  Perera

Student ID: 20200216

UoW Id : w1867413

Link for the video:

https://drive.google.com/file/d/1_unAZwavkc6EnYdQgIV_aYNmliFNxcl9/view?usp=sharing

# Coursework Report – 5COSC019C Object Oriented Programming

Student Name: Irushi Nicoline Perera
Student ID: w1867413

Have you submitted the video with the demonstration of your system? - Yes

Link for the video:
https://drive.google.com/file/d/1_unAZwavkc6EnYdQgIV_aYNmliFNxcl9/view?usp=share_link

## 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 | The UML Use Case Diagram was designed and submitted in a separate file (It is also attached in this document) |
| Design a UML Class Diagram of your system (submitted in a separate file). | Yes | The UML Use Class Diagram was designed and submitted in a separate file.<br>(It is also attached in this document) |
| Implementation Class Person | Yes | Class "Person" is implemented successfully. |
| Implementation Class Doctor | Yes | The Class "Doctor" is implemented successfully. |
| Implementation Class Patient | Yes | The Class "Patient" is implemented successfully. |
| Implementation Class Consultation | Yes | The Class "Consultation" is implemented successfully. |
| Implementation Interface WestminsterSkinConsultationManager | Yes | "WestminsterSkinConsultationManager" Is successfully 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 | The programme adds a doctor by getting user input, once the "centre" has 10 doctors the programme will display a message to the user. (Which shows that the centre already has 10 doctors) |
| 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 | The programme deletes a doctor, by the licence number given by the user, the programme also displays a message to confirm that the relevant doctor has been removed and the current number of doctors will also be displayed. The programme satisfies all the given requirements successfully. |
| Print on the screen the list the doctors in the centre with all the relative information. The list should be ordered alphabetically. | Yes | The programme prints all the doctors alphabetically. The programme satisfies the given requirement successfully. |
| 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 | The programme saves all the doctors information.<br>The programme satisfies the given requirement successfully. |

## 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 | The doctor list will be sorted alphabetically one the "SORT" button is clicked.<br>The programme satisfies the given requirement successfully. |
| The user can select a doctor and add a consultation. | Yes | The programme satisfies the given requirement successfully. |
| In the consultation the user can add all the patient details. | Yes | The user can add all the patient details successfully. The programme satisfies the given requirement successfully. |
| The user can select the date/time of the consultation considering that a doctor cannot have more than one consultation at the time. | Yes | The user can only select the date in this programme. The programme satisfies the given requirements partially |
| The user can enter and save the cost for the consultation. (£25 per hour and only the first one £15). | Yes | The programme satisfies the given requirements successfully. |
| The user can add some notes (text information or images). This information has been encrypted. | Yes | User Can add notes. The programme satisfies the given requirement successfully. |

## 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 | The test plan is submitted as a separate file. |
| Implementation of an automated unit test for each scenario in the console menu. | No | This part was not completed. |
| Error Handling across all the code, input validation and code quality. | Yes | I/O Exception was used throughout the programme. |

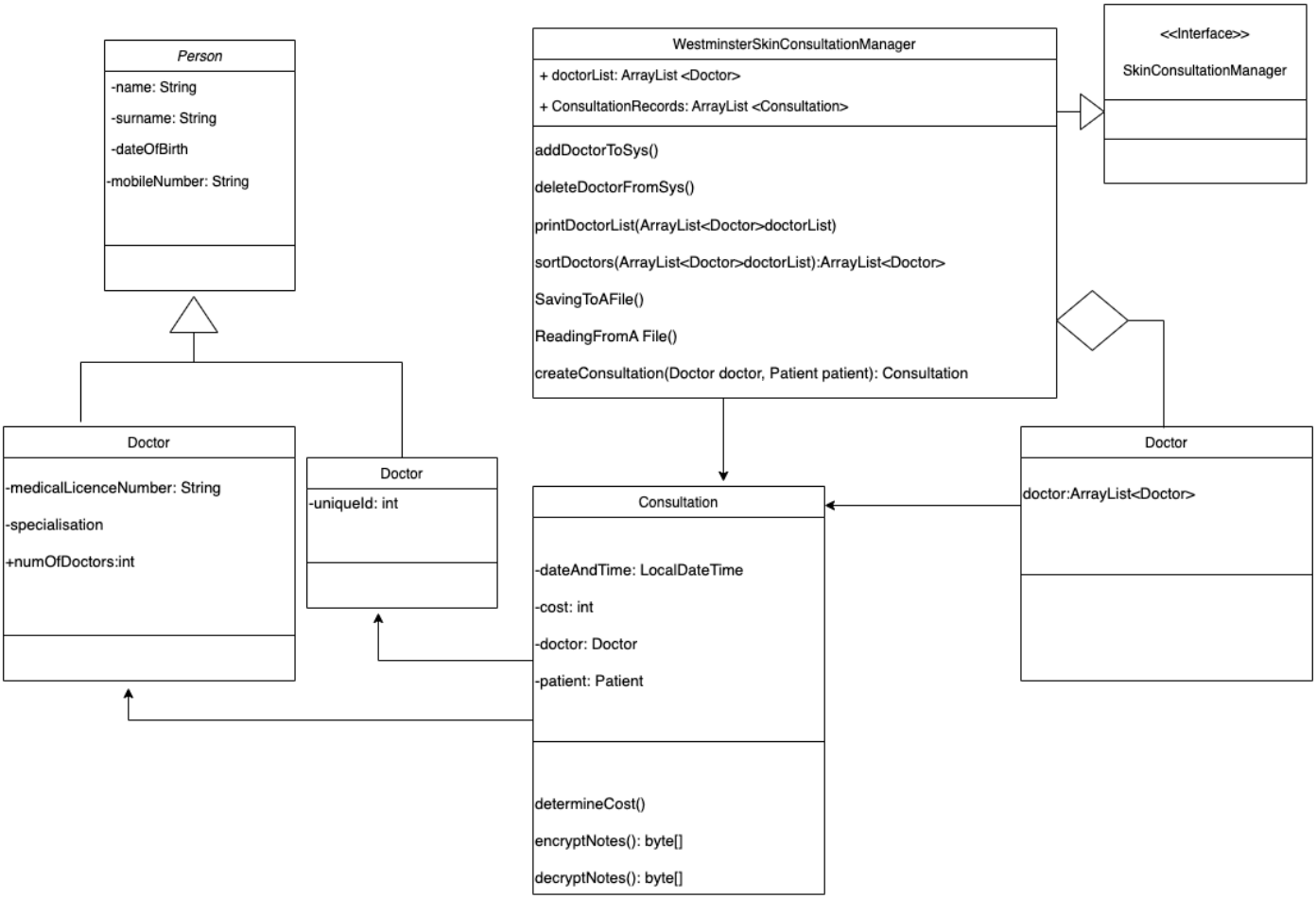


**Figure 1| UML Class diagram**

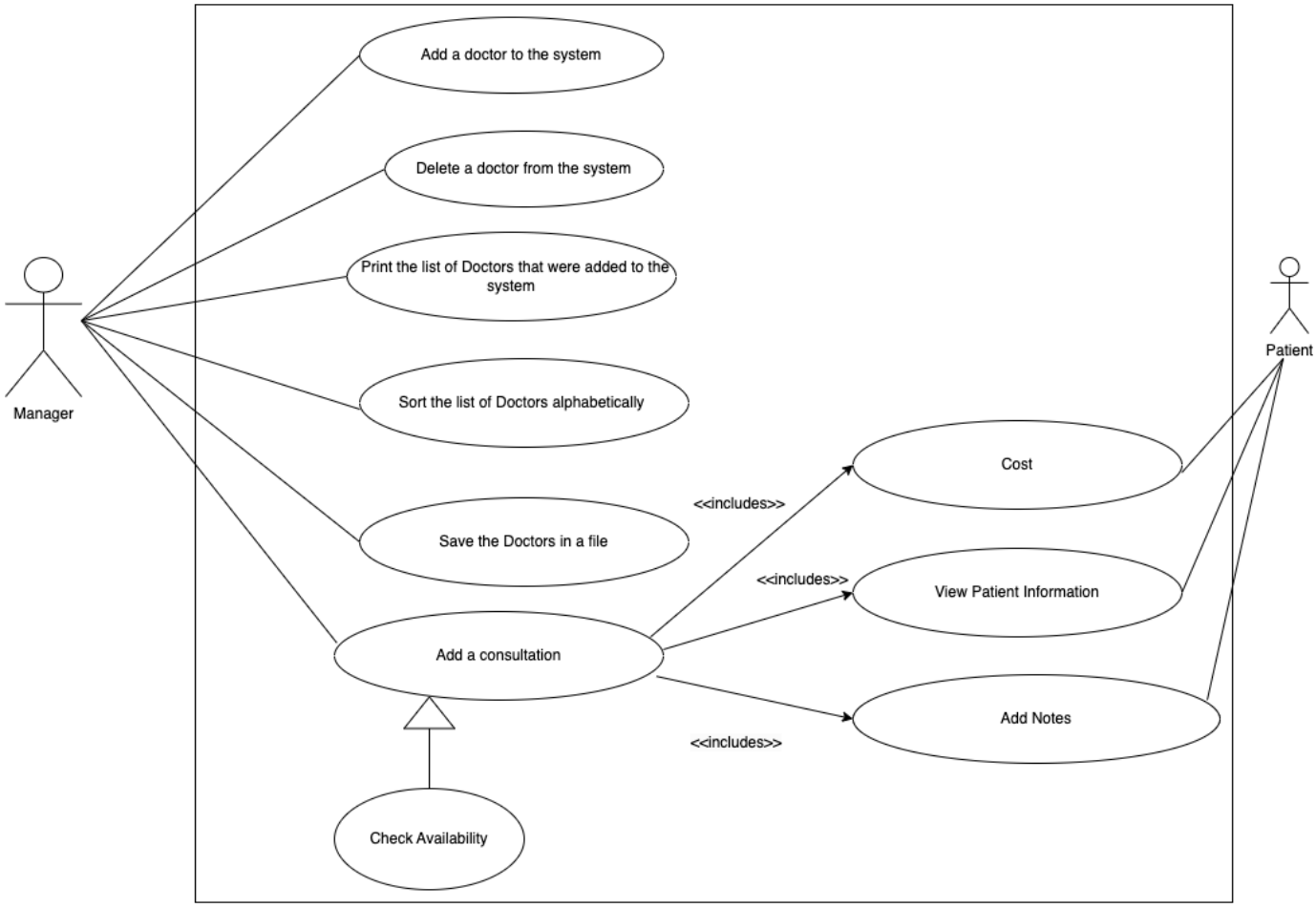


**Figure 2| UML Usecase diagram**

# References

Java Cryptography - Encrypting Data. (no date). Available from https://www.tutorialspoint.com/java_cryptography/java_cryptography_encrypting_data.htm [Accessed 16 December 2022].

Java Swing | Simple User Registration Form. (2019). GeeksforGeeks. Available from https://www.geeksforgeeks.org/java-swing-simple-user-registration-form/ [Accessed 28 December 2022].

Java Swing Tutorial - Java Swing Font. (no date). Available from http://www.java2s.com/Tutorials/Java/Java_Swing/1520__Java_Swing_Font.htm#:~:text=To%20set%20the%20font%20for,setFont(f4)%3B [Accessed 26 December 2022].