**INTI International College Penang School of Engineering and Technology**

**3+0 Bachelor of Science (Hons) in Computer Science, in collaboration with Coventry University, UK**

**3+0 Bachelor of Science (Hons) in Computing, in collaboration with Coventry University, UK**

**Coursework cover sheet**

**Section A - To be completed by the student**

|  |  |
| --- | --- |
| Full Name: Chow Ching Huey | |
| CU Student ID Number: 12673195 | |
| Semester: 2 | |
| Session:  **April 2022** | |
| Lecturer:  **Nadhrah Abdul Hadi (nadhrah.abdulhadi@newinti.edu.my)** | |
| Module Code and Title:  **4067CEM Software Design** | |
| Assignment No. / Title:  **Continuous Assessment** | % of Module Mark:  **50** |
| Hand out Date:  **22nd April 2022** | Due Date:  **Task 1: 13 May 2022, by 11.59pm**  **Task 2: 1 July 2022, by 11.59pm**  **Task 3: 19 June 2022, by 11.59pm.**  **Task 4: 19 June 2022, by 11.59pm.**  **Task 5: 19 June 2022, by 11.59pm.** |
| Penalties: No late work will be accepted. If you are unable to submit coursework on time due to extenuating circumstances, you may be eligible for an extension. Please consult the lecturer. | |
| Declaration: I/we the undersigned confirm that I/we have read and agree to abide by the University regulations on plagiarism and cheating and Faculty coursework policies and procedures. I/we confirm that this piece of work is my/our own. I/we consent to appropriate storage of our work for plagiarism checking.  Signature(s): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |

**Section B - To be completed by the module leader**

|  |  |  |
| --- | --- | --- |
| Intended learning outcomes assessed by this work:  1. Understand and apply appropriate concepts, tools and techniques to each stage of the software development  2. Understand and apply design patterns to software components in developing new software  3. Demonstrate an understanding of project planning and working to agreed deadlines, along with professional, interpersonal skills and effective communication required for software production  5. Demonstrate an awareness of, and ability to apply, social, professional, legal and ethical standards as documented in relevant laws and professional codes of conduct such as that of the Malaysian National Computer Confederation. | | |
| Marking scheme | Max | Mark |
| 1. User Story Mapping 2. Setting up a GitHub Repository 3. Creating a Class diagram and design pattern selection 4. Creating a Prototype User Interface and Usability Testing 5. Discuss the ethical issue related to the software | 20  10  30  20  20 |  |
| Total | 100 |  |

# TASK 3: CREATING A CLASS DIAGRAM AND DESIGN PATTERN SELECTION

# Simple Class Diagram

Diagram, engineering drawing

Description automatically generated

/updated by

/updated by

Figure 1

* This is the simple class diagram of the College Event System in INTI International College Penang.
* 12 Classes:

1. Admin
2. Account
3. Student
4. Club
5. Member
6. Committee
7. Post
8. Event
9. Facility Booking
10. Participant
11. Feedback
12. Analytics
13. Notifications

* One to many **Account(s)** are created/deleted/updated by one to many **Admin(s)**, and there may be a group of admins. The admins use students’ emails for student accounts and clubs’ emails (given by club presidents) to create club accounts. They can delete once they are of no use.
* After that, there are one to many **Student(s)** and one to many **Club(s),** and each of them has one account.
* One to many **Student(s)** can join/unjoin zero to many **Club(s)** to becomes club members, which leads to one to many clubs having one to many members as a club will exist only if there is someone is in the club. Four to many **Committee(s)** are then picked from among the **Member**s which is four to many. They are set as four to many as there are president, vice president, secretary, treasurer, and trainees to be passed on the positions to.
* One to four **Committee(s)** organize one to many **Event(s)** in the background.
* But for managing the club account, only one to two of them (club president and vice president) have the password to access the account. The rest of the committee team are the same level as club members with fancy committee positions in their club page. Hence, one to two **Committee(s)** (club president and vice president) can use their only one club account to perform tasks as a club:

1. Create/delete/update zero to many **Post(s)**, as the club may not have posts yet, and delete zero to many **Post(s)**
2. Organize zero to many **Event(s),** as the club may not have events yet

* All students can either view, like or save posts posted by clubs or do nothing about them. Hence one to many **Student(s)** view/like/save zero to many **Post(s).**
* Events may have facility/room booking, hence one to many **Event(s)** has zero to many **Facility Booking(s)**. One to many **Event(s)** and one to many **Facility Booking(s)** are approved/disapproved by one to many **Admin(s)**.
* Students (non-members) and members attend to events, hence one to many **Student(s)** and one to many **Member(s)** become as one to many **Participant(s)** This implies that one to many **Participant(s)** attends one to many **Event(s)**.
* After attending the event or after the event ends, it is needed to give ratings and feedback to that event. One to many **Event(s)** have one to many **Feedbacks(s)**. Hence, there is one to many **Participants(s)**, and each of them gives one **Feedback**.
* One to many **Feedback(s)** are stored in one **Analytics**.
* Each account has one to many **Notification(s)**.

# UML Class Diagram Representing the Design Pattern

The design pattern used for the class diagram is Facade Pattern. It is a structural design pattern that encapsulates a subsystem to hide the subsystem’s complexity, and acts as a point of entry into a subsystem without adding more functionality inside itself. The class diagram representing the design pattern is as shown below:

Diagram, engineering drawing, schematic

Description automatically generated

Figure 2

* There are hierarchical, multilevel, and single inheritances which would combine as a hybrid inheritance shown in the class diagram in Figure 2. **Admin**, **Student** and **Club** are accounts inherited from **User**. S**tudent** also has its own 2 child classes which are **Member** and **Participant**. Since students need to become a club member to become a club committee, **Committee** is inherited from **Member**. For posts, as **Event Post** is another version of **Post**, **Event Post** is the child class of **Post**.
* There are classes where the object cannot exist without their parent class object. Without a **User** account, there will be no **Notifications**. Without **Event**, there will be no **Analytics, Facility or Room Booking** and **Feedback**. Without a **Club** account, there will be no **Event, Post** and **Member** and no **Analytics** too.
* All the relations are the same as explained in the simple class diagram.