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

**3+0 Bachelor of Science (Hons) in Computing,**

**in collaboration with Coventry University, UK**

Coursework cover sheet - Section A

|  |  |
| --- | --- |
| Full Name: Ong Zheng Yang | |
| CU Student ID Number: P21013454 | |
| Semester: 1 | |
| Session:  **August 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:  **6th September 2022** | Due Date:  **Task 1: 30 September 2022, by 11.59pm.**  **Task 2: 18 November 2022, by 11.59pm**  **Task 3: 4 November 2022, by 11.59pm.**  **Task 4: 4 November 2022, by 11.59pm.**  **Task 5: 4 November 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): A picture containing text  Description automatically generated | |

# Class diagram and design pattern selection Assignment 3

# 3.1 Class Diagram

A picture containing timeline

Description automatically generated

Figure 3.1 Friendly Buddy System Class Diagram

# 3.2 Design Pattern – Mediator

# A picture containing timeline Description automatically generated

# Figure 3.2 Friendly Buddy System Design Pattern

# A behavioral design pattern called mediator enables you to lessen complicated relationships between items. The design pattern prevents things from interacting directly and compels them to work together exclusively through intermediary components. There is a home dialogue on the Friendly Buddy system. It includes a variety of form controls, including profile, location, chat, follow, discover, and create post. The Mediator design recommends that Friendly Buddy systems cease any direct contact between parts, so separating them from one another. Instead, these parts must work together covertly by calling intermediary objects that reroute calls to the proper parts. As a result, instead of being dependent on hundreds of coworkers tied to them, these components simply need one intermediary class. In our example of the Friendly Buddy system, the home page itself can serve as an intermediary. The Friendly Buddy system doesn't even need to add a new dependency to this class because it's likely that the main page already knows about all of its group attached. The Friendly Buddy system may incorporate a complicated network of relationships between diverse items into a single Mediator object thanks to the Mediator pattern. It is simpler to change, extend, or reuse a class the less dependencies it has. (Mediator 2022)

# 3.3 Reference

Mediator (2022) Refactoring.Guru. Available at: https://refactoring.guru/design-atterns/mediator (Accessed: November 18, 2022).