**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: Ko Chi Ling | |
| CU Student ID Number: 13446950  INTI Student ID Number: P21013435 | |
| 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): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |

**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

The complete class diagrams

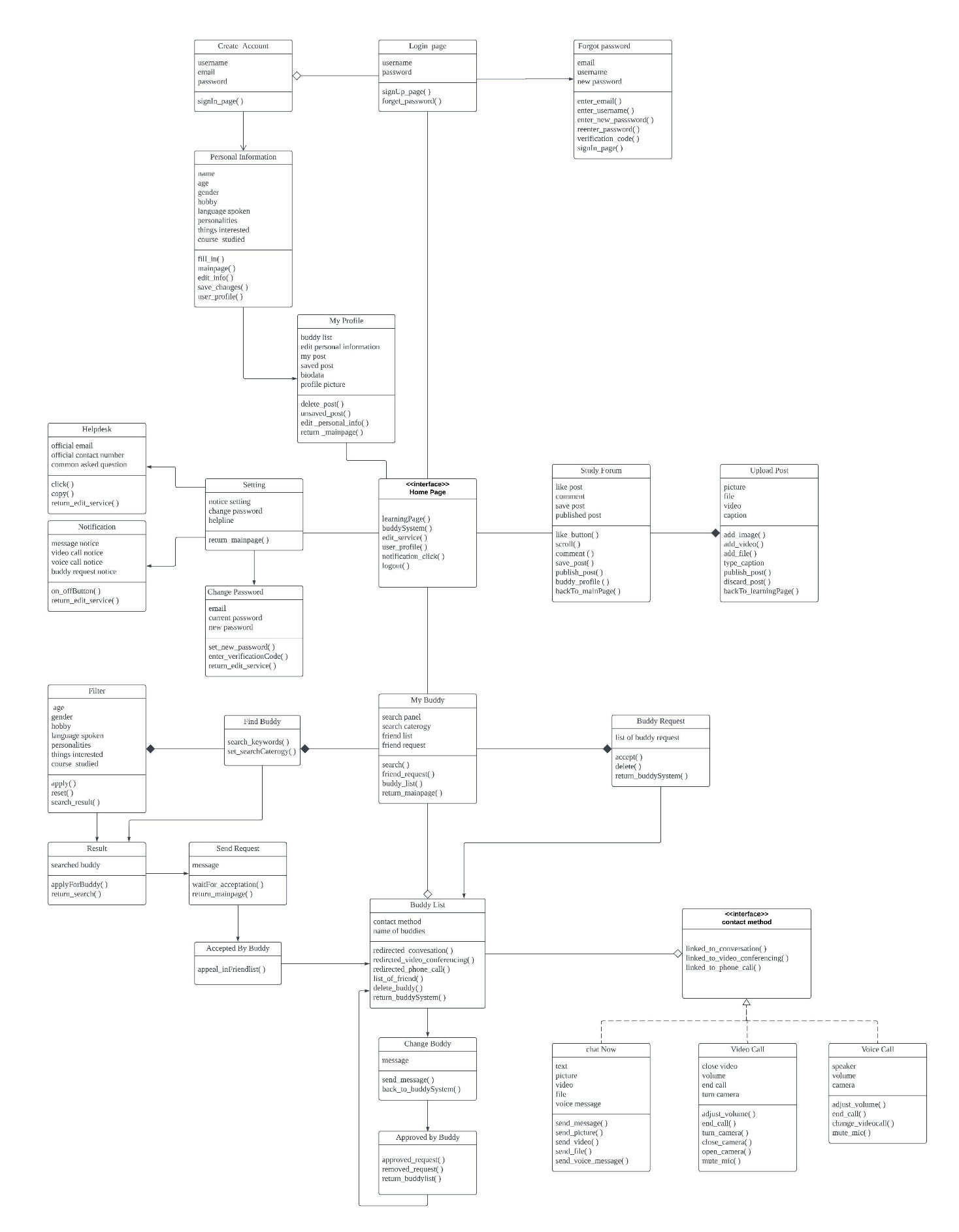


Figure 1 Class Diagram

Figure above shows the class diagram of this buddy system. There are 24 classes and 2 interfaces in this class diagram. This class diagram can be divided into four parts. The first part is login part. The login page class is inherited by the forget password class and aggregated by the create account class. The create account has a directed association relationship with the personal information class. Then, the personal information class is inherited by my profile class. The home page interface has an association relationship with the setting class, my profile class, my buddy class, and study forum class. The second part is study forum part. The study forum class has a composition relationship with the create post class. The third part is setting part. The setting class has an inheritance relationship with the change password class, notification class and helpdesk class.

The fourth part is the buddy system part. My buddy class has a composition relationship with the buddy request class and find buddy class. The find buddy class has a composition relationship with the filter class. The find buddy class and filter class have an inheritance relationship with the result class. The result class has an inheritance relationship with the send request class. The send request has an inheritance relationship with the accept by buddy class without the attributes. The accept by buddy class has inheritance relationship with the buddy list class. My buddy class has an aggregation relationship with the buddy list class. The buddy request class has an inheritance relationship with the buddy list class. The buddy list has an aggregation relationship with the contact method interface. The contact method interface has a realization relationship with the chat now class, video class, and voice call class. The buddy list has inheritance relationship with the change buddy class. The change buddy class has a relationship with approved by buddy class. The approved by buddy class has an inheritance relationship with the buddy list class.

The association relationship means any logical connection between classes. The inheritance relationship means the child class has inherits the parent class functionality. The aggregation relationship means child class can exist independently without the parent element. The composition relationship means the child class could not exist without the parent class. The realization relationship means another class's implementation of functionality defined in one class.

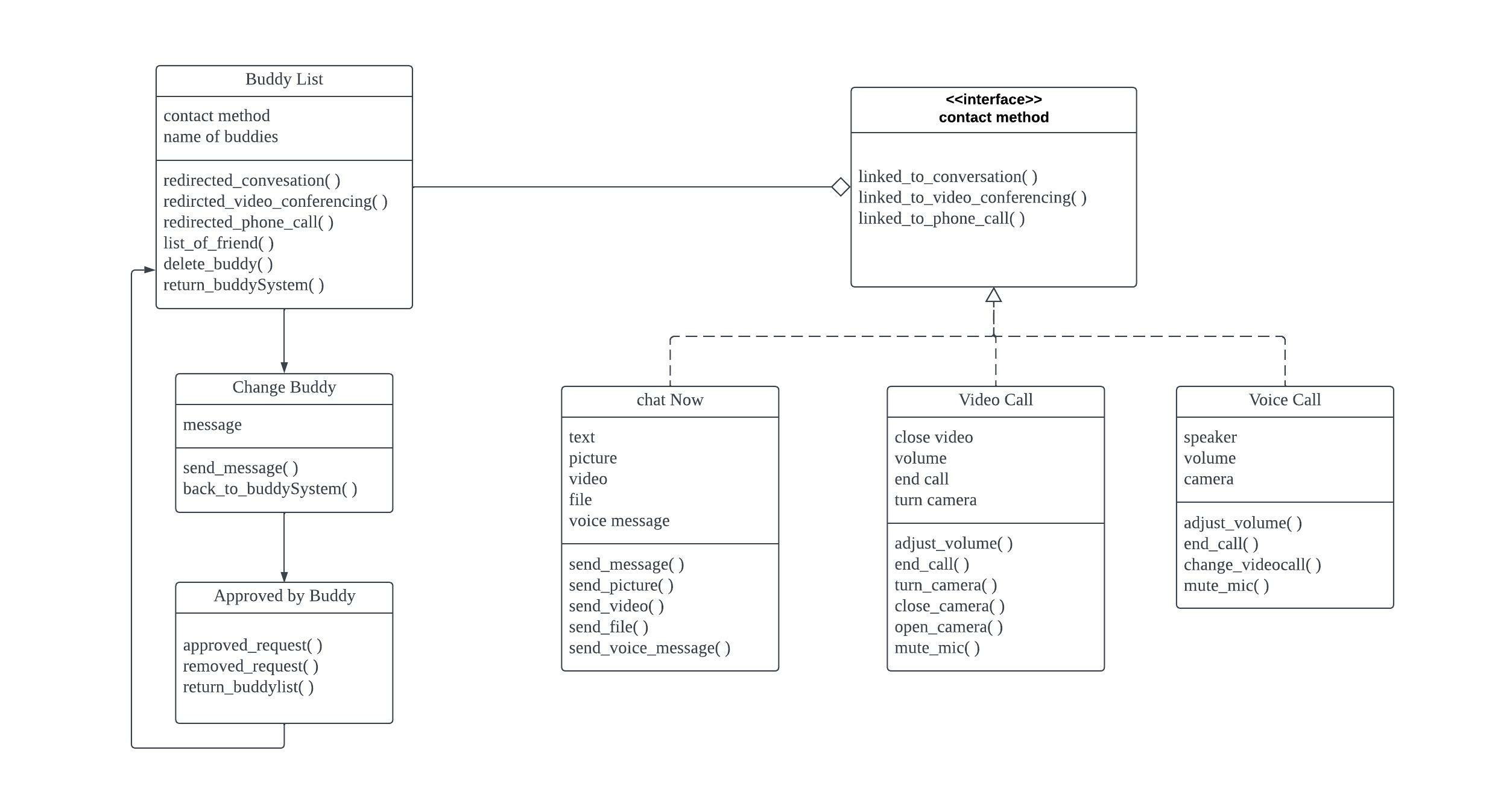


Figure 2 Class Diagram ( Bridge Patterns)

This class diagram has used the bridge patterns in the structural patterns. This bridge patterns have been used in this class diagram. Buddy list class act as the abstraction and has some same methods in implementation. The contact method act as the interface to help the user to have communication with the buddies. The chat with buddy class, video call and voice call classes has linked to each other and these classes have some same different operations . The problem that has been found in this system is there are many contacts method in the buddy system and the contact method also has linking to other contact method. To solve this problem, the contact method has become an interface and linked to the chat now function, video call function, and voice call function. These functions have included some different operations and attributes. So, this bridge pattern has help to separate one of the features into a separate class hierarchy. The buddy system can refer to an interface that is contact method in the new hierarchy rather than have all of the states and behaviours in the class. The contact method has implemented with three of the concrete implementation and link together. The three of the concrete implementations are chat now, video call and voice call . The classes which are chat now, voice call and video call has the same operation that is they all link together. The chat now has attribute of text, picture, video, file, and voice message. The chat now class has the operation that is send message , picture, video, file and voice message. The voice call has close video, volume, end call, turn camera as the attribute. The video call has adjusted volume, hang up call , turn camera, close and open camera, and mute mic. The voice call has attributes of speaker, volume, and end call. The voice call has operation of the adjust volume, change to videocall, mute mic and end call.