SafeAssign Originality Report

SOFTWARE DESIGN • Creating a Class diagram and design pattern selection (30%)

ADNAN IRFAN POTRIK -

Total Score: High risk 70 %

Submission UUID: f5e70b12-626a-2859-d7a0-847beaf340fa

Total Number of Reports	Highest Match	Average Match	Submitted on	Average Word Count
1	70 % Software Design Task 3 (FINAL).docx	70 %	11/18/22 06:44 PM GMT+8	1,042 Highest: Software Design Task 3 (FINAL).d...

Attachment 1 70 %

Word Count: 1,042
Software Design Task 3 (FINAL).docx

Institutional database (3) 64 %

1 *Student paper*
2 *Student paper*
3 *Student paper*

Internet (2) 6 %

5 *wikipedia*
4 *jbaysolutions*

Top sources (3)

1 *Student paper*
2 *Student paper*
5 *wikipedia*

Excluded sources (0)

Task 3 Adnan Irfan

1 Creating a class diagram and selection of a design pattern

Task 3

1 INTI International College Penang School of Engineering and Technology 3+0 Bachelor of Science (Hons) in Computer Science, in collaboration with Coventry University, UK

Coursework cover sheet

2 Section A - To be completed by the student

Full Name: Adnan Irfan Potrik

1 CU Student ID Number: 13446776

Semester: 2

Session: August 2022

Lecturer: 1 Nadhrah Abdul Hadi (nadhrah.abdulhadi@newinti.edu.my)

Module Code and Title: 4067CEM Software Design

Assignment No. / Title: 1 Continuous Assessment % of Module Mark: 50

1 Hand out Date: 6th September 2022 Due Date: Task 1: 1 30 September 2022, by 11.59pm. Task 2: 1 18 November 2022, by 11.59pm

Task 3: (2) 4 November 2022, by 11.59pm. Task 4: (2) 4 November 2022, by 11.59pm. Task 5: (2) 4 November 2022, by 11.59pm.

Penalties: (1) No late work will be accepted. (2) If you are unable to submit coursework on time due

to extenuating circumstances, you may be eligible for an extension. (1) Please consult the lecturer.

Declaration: (1) I/we the undersigned confirm that I/we have read and agree to abide by the University regulations on plagiarism and cheating and Faculty course-work 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):

(2) Section B - To be completed by the module leader

Intended learning outcomes assessed by this work: 1. (1) Understand and apply appropriate concepts, tools and techniques to each stage of the software development

1. (1) Understand and apply design patterns to software components in developing new software

1. (1) Demonstrate an understanding of project planning and working to agreed deadlines, along with professional, interpersonal skills and effective communication required for software production

5. (1) 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

(2) the Malaysian National Computer Confederation.

(1) Marking scheme Max Mark

1. (3) User Story Mapping 20

2. (2) Setting up a GitHub

Repository 10

3. (2) Creating a Class diagram and

(3) design pattern selection 30

4. (2) Creating a Prototype User

(3) Interface and Usability Testing 20

5. (2) Discuss the ethical issue

(3) related to the software 20

Total 100

Creating a Class Diagram A class diagram was created with the help of Draw.io tool

(1) Figure 1.1 – Class Diagram for the Buddy application

The Figure 1.1 shows the Class Diagram of the entire Buddy System. (4) It is seen that; the entire system is divided into different classes. Each class has its own function and role. Some class have subclasses while some do not. First, is the Person Class. This class is the base class for the user and buddies to be distinguished. As it is seen, the person can only Login and Register into the system, by entering a few details, like the name, email, phone number, and password. Once the user logins, they become the main user of the application on their phone. Whatever details are saved from the login and registration page, it is now saved under the Student Class. The student can do numerous tasks, starting from customizing their profile, to viewing other's profile, to sending them messages, to Rating the application and finally logging out. The Buddy Class and the Buddy Profile Class represents the details of other people that the student can see in the application. Their Name and Picture is just shown at first, but more details and information can be known if the student visits their profile. To Chat and interact with these buddies, Chat class was made as well. The chat feature allows the student to send texts message or even some file or picture. The creation of subclasses under Chat Class will be explained further in the next topic. In order to maintain the application and ensure that it catches up with the student's needs, a rating and feedback section is needed. Hence, a System Class was created which would record all the ratings and feedbacks given by the student.

Selection of Design Pattern Programming in object-oriented languages is incomplete without design patterns. There are a few classes that make up this software infrastructure. It is used to solve technical problems. There are many common problems that can be solved with the help of Design Patterns. Many people are already familiar with them, which makes them easy to maintain. For the Buddy Application, Factory Design Method is used. Factory Methods are creational patterns that provide a way for users to create objects in a superclass while allowing subclasses to change what type of objects will be created. (5) A factory method is called rather than a constructor when creating objects by either specifying one in an interface or implementing one in a base class and optionally overriding it in derived classes. Among the 23 popular design patterns, Factory Method describes a method to fix recurring design problems and develop, modify, test, and reuse flexible and reusable object-oriented software. Problem – The problem arises in the chat section.

There was an option of chatting, but what if the user wanted to create a group and talk to multiple people at once. Solution – To fix this issue, the chat class was divided into two, one being the Personal Chat, where the user can talk to an individual person in their personal chat. Second is the Group chat, where the user could

add multiple people in the group chat and talk with them all together.

Figure 1.2 Factory Method on Chat Class

The above image depicts how the Chat class was divided into subclasses to fix the problem of chatting individually or in a group. As per the rules and structure of factory method, The chat page class is the Creator Class which is the main class that allows the user to chat and interact with people in the first place. The Personal and the Group Chat Class are the Concrete Creator Class A and B respectively. These classes override the main function of chat, and then returns a different product (Personal chat returns a chat page with only an individual in it, and the Group Chat returns a chat page with multiple selected people in it.

Source Matches (34)

1	*Student paper*	96%
Student paper	Original source	
Creating a class diagram and selection of a design pattern	Creating a Class diagram and design pattern selection	

1	*Student paper*	89%
Student paper	Original source	
INTI International College Penang School of Engineering and Technology 3+0 Bachelor of Science (Hons) in Computer Science, in collaboration with Coventry University, UK Coursework cover sheet	3+0 Bachelor of Science (Hons) in Computer Science, in collaboration with Coventry University, UK Coursework cover sheet	

2	*Student paper*	100%
Student paper	Original source	
Section A - To be completed by the student	Section A - To be completed by the student	

1	*Student paper*	100%
Student paper	Original source	
CU Student ID Number:	CU Student ID Number	

1	*Student paper*	100%
Student paper	Original source	
Nadhrah Abdul Hadi (nadhrah.abdulhadi@newinti.edu.my) Module Code and Title: 4067CEM Software Design	Nadhrah Abdul Hadi (nadhrah.abdulhadi@newinti.edu.my) Module Code and Title 4067CEM Software Design	

1	*Student paper*	100%
Student paper	Original source	
Continuous Assessment % of Module Mark:	Continuous Assessment % of Module Mark	

1	*Student paper*	100%
Student paper	Original source	
Hand out Date: 6th September 2022 Due Date:	Hand out Date 6th September 2022 Due Date	

1	*Student paper*	100%
Student paper	Original source	
30 September 2022, by 11.59pm.	30 September 2022, by 11.59pm	

1 Student paper	100%
Student paper 18 November 2022, by 11.59pm	Original source 18 November 2022, by 11.59pm

2 Student paper	100%
Student paper 4 November 2022, by 11.59pm.	Original source 4 November 2022, by 11.59pm

2 Student paper	100%
Student paper 4 November 2022, by 11.59pm.	Original source 4 November 2022, by 11.59pm

2 Student paper	100%
Student paper 4 November 2022, by 11.59pm.	Original source 4 November 2022, by 11.59pm

1 Student paper	100%
Student paper No late work will be accepted.	Original source No late work will be accepted

2 Student paper	100%
Student paper If you are unable to submit coursework on time due to extenuating circumstances, you may be eligible for an extension.	Original source If you are unable to submit coursework on time due to extenuating circumstances, you may be eligible for an extension

1 Student paper	100%
Student paper Please consult the lecturer.	Original source Please consult the lecturer

1 Student paper	100%
Student paper 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.	Original source 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

2 Student paper	100%
Student paper Section B - To be completed by the module leader Intended learning outcomes assessed by this work:	Original source Section B - To be completed by the module leader Intended learning outcomes assessed by this work

1 Student paper	100%
Student paper Understand and apply appropriate concepts, tools and techniques to each stage of the software development	Original source Understand and apply appropriate concepts, tools and techniques to each stage of the software development

1 Student paper	100%
Student paper Understand and apply design patterns to software components in developing new software	Original source Understand and apply design patterns to software components in developing new software

1 Student paper	100%
Student paper Demonstrate an understanding of project planning and working to agreed deadlines, along with professional, interpersonal skills and effective communication required for software production	Original source Demonstrate an understanding of project planning and working to agreed deadlines, along with professional, interpersonal skills and effective communication required for software production

1 Student paper	91%
Student paper 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	Original source 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

2 Student paper	100%
Student paper the Malaysian National Computer Confederation.	Original source the Malaysian National Computer Confederation

1 Student paper	100%
Student paper Marking scheme Max Mark	Original source Marking scheme Max Mark

3 Student paper	100%
Student paper User Story Mapping 20	Original source User Story Mapping 20

2 Student paper	100%
Student paper Setting up a GitHub	Original source Setting up a GitHub

2 Student paper	100%
Student paper Creating a Class diagram and	Original source Creating a Class diagram and

3 Student paper	100%
Student paper	Original source
design pattern selection 30	design pattern selection 30

2 Student paper	100%
Student paper	Original source
Creating a Prototype User	Creating a Prototype User

3 Student paper	100%
Student paper	Original source
Interface and Usability Testing 20	Interface and Usability Testing 20

2 Student paper	100%
Student paper	Original source
Discuss the ethical issue	Discuss the ethical issue

3 Student paper	100%
Student paper	Original source
related to the software 20	related to the software 20

1 Student paper	72%
Student paper	Original source
Figure 1.1 – Class Diagram for the Buddy application The Figure 1.1 shows the Class Diagram of the entire Buddy System.	Figure 1 Student Buddy System Class Diagram Figure 1 Student Buddy System Class Diagram

4 jbaysolutions	64%
Student paper	Original source
It is seen that;	And that is it

5 wikipedia	69%
Student paper	Original source
A factory method is called rather than a constructor when creating objects by either specifying one in an interface or implementing one in a base class and optionally overriding it in derived classes.	This is done by creating objects by calling a factory method—either specified in an interface and implemented by child classes, or implemented in a base class and optionally overridden by derived classes—rather than by calling a constructor