SafeAssign Originality Report

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

MAH CHUN-HOE -

Submission UUID: 7a297f14-aa9f-05a0-2171-68f22ed92bbc

Total Score: ⊗ High risk 94 %

Total Number of Reports	Highest Match	Average Match	Submitted on	Average Word Count
1	94 % 4067CEM_AUG2022_ContinuousAssessm...	94 %	11/18/22 05:27 PM GMT+8	660 Highest: 4067CEM_AUG2022_Continuous...

⊗ Attachment 1   94 %

Word Count: 660
4067CEM_AUG2022_ContinuousAssessment Task3.docx

Institutional database (4)   89 %

1 Student paper   2 Student paper   3 Student paper
4 Student paper

Internet (2)   5 %

5 java2novice   6 coengoedegebure

Top sources (3)

1 Student paper   2 Student paper   3 Student paper

Excluded sources (0)

---

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

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: MAH CHUN-HOE

1 CU Student ID Number: P22014268

Semester: 1

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: 2 6th September 2022 Due Date: Task 1: 2 30 September 2022, by 11.59pm. Task 2: 2 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. 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: (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): _____

(1) 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

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

3. (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 the Malaysian National Computer Confederation.

Marking scheme Max Mark

1. (3) User Story Mapping 2. Setting up a GitHub Repository 3. (1) Creating a Class diagram and design pattern selection

4. (3) Creating a Prototype User Interface and Usability Testing 5. (1) Discuss the ethical issue related to the software 20

10

30

20

20

Total 100

(4) Task 3- Class Diagram

Class Diagram

Figure 1: Student Buddy System Class Diagram

Design pattern selection A design pattern is a solution to some problems in software design. Design patterns are descriptions or templates on how to solve problems faced that can be used in different situations when developing software. (5) There are three types of design patterns, mainly creational, structural and behavioral design patterns. (6) Creational design patterns create objects in a manner that is suitable to the situation. Structural design patterns identify an easy way to visualize relationships between entities. Finally, behavioral design patterns classify the common communication patterns between objects to increase the flexibility in carrying out the communication. The design pattern chosen to be implemented in the class diagram is the command pattern from the behavioral design patterns. Command pattern is useful because it separates the object that calls on the operation from the object that executes the operation. It also makes it easy for users to add new commands while keeping the existing classes unchanged. This design pattern groups the requests that perform in a similar way into a concrete class of commands. The problem of the class diagram in Figure 1 is that there are multiple classes with similar commands connecting to the main class which is the Student Buddy System. By implementing the command pattern, the classes that function similarly like calls, chat features and stories will be grouped into a single class called the homepage class. By breaking the similar commands into layers, the software design looks much cleaner and organized. Figure 2 below shows the class diagram after inserting the command method into the class diagram.

Figure 2: Part of Student Buddy System Class Diagram after Design Pattern implementation

Source Matches (26)

| 1 | *Student paper* | 100% |

Student paper

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.

Original source

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

---

| 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

---

| 1 | *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* | 100% |

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 the Malaysian National Computer Confederation. Marking scheme Max Mark

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 Marking scheme Max Mark

---

| 3 | *Student paper* | 100% |

Student paper

User Story Mapping 2. Setting up a GitHub Repository 3.

Original source

User Story Mapping 2 Setting up a GitHub Repository 3

	Student paper	100%
Student paper	**Original source**	
Creating a Class diagram and design pattern selection	Creating a Class diagram and design pattern selection	

	Student paper	100%
Student paper	**Original source**	
Creating a Prototype User Interface and Usability Testing 5.	Creating a Prototype User Interface and Usability Testing 5	

	Student paper	100%
Student paper	**Original source**	
Discuss the ethical issue related to the software 20	Discuss the ethical issue related to the software 20	

	Student paper	64%
Student paper	**Original source**	
Task 3- Class Diagram	Task 3 – Creating a Class diagram and design pattern selection	

	java2novice	88%
Student paper	**Original source**	
There are three types of design patterns, mainly creational, structural and behavioral design patterns.	The types of design patterns are Creational, Structural, and Behavioral design patterns	

	coengoedegebure	72%
Student paper	**Original source**	
Creational design patterns create objects in a manner that is suitable to the situation.	They try to create objects in a manner that is suitable to the situation	