UOW MALAYSIA KDU PENANG UNIVERSITY COLLEGE
PART OF THE UNIVERSITY OF WOLLONGONG AUSTRALIA GLOBAL NETWORK

ASSIGNMENT COVER PAGE

Validated by UNIVERSITY OF LINCOLN UNITED KINGDOM

Programme	**Course code and title**
Bachelor of Computer Science (Hons)/ Bachelor of Computer Science (Hons) in Computer and Network Technology/ Bachelor of Information Systems (Hons)/ Bachelor of Software Engineering (Hons)	CET3063/N/CET3064 Internet of Things
Student's name / student's ID	**Lecturer's name**
CHAN SEOW FEN / 0207368	Dr. Khoo Hee Kooi

Date issued	**Submission deadline**	**Indicative weighting**
25th September 2023 (Week 3)	20th October 2023 (Week 6)	30%
Assignment 1 title	Implement an IoT system (400 words)	

This assessment assesses the following course learning outcomes

# as in course guide	**UOW Malaysia KDU Penang University College learning outcomes**
CLO1	Discuss the various Internet of Things (IoT) components required for various environment.
CLO2	Evaluate IoT system architecture for a real world application.
CLO3	N/A
CLO4	N/A
# as in course guide	**University of Lincoln learning outcomes**
CLO1	Critically evaluate the strengths, weaknesses, and resource constraints of IoT computing systems in comparison to traditional computing models
CLO2	Design and implement a connected prototype IoT software system that utilises sensor data
CLO3	Implement industry standard IoT messaging protocols
CLO4	N/A

Student's declaration

I certify that the work submitted for this assessment is my own and research sources are fully acknowledged.

Student's signature:

Submission date: 20/10/23

Assignment 1

ORIGINALITY REPORT

3%	2%	1%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	docs.google.com Internet Source	1%
2	www.asktheeu.org Internet Source	1%
3	Elahe Naserian, Xinheng Wang, Keshav P. Dahal, Jose M. Alcaraz-Calero, Honghao Gao. "A Partition-Based Partial Personalized Model for Points-of-Interest Recommendations", IEEE Transactions on Computational Social Systems, 2021 Publication	1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		

Table of Contents

1.0 Reviews on an IoT system

1.1 Finite-State Machine (FSM) Diagram

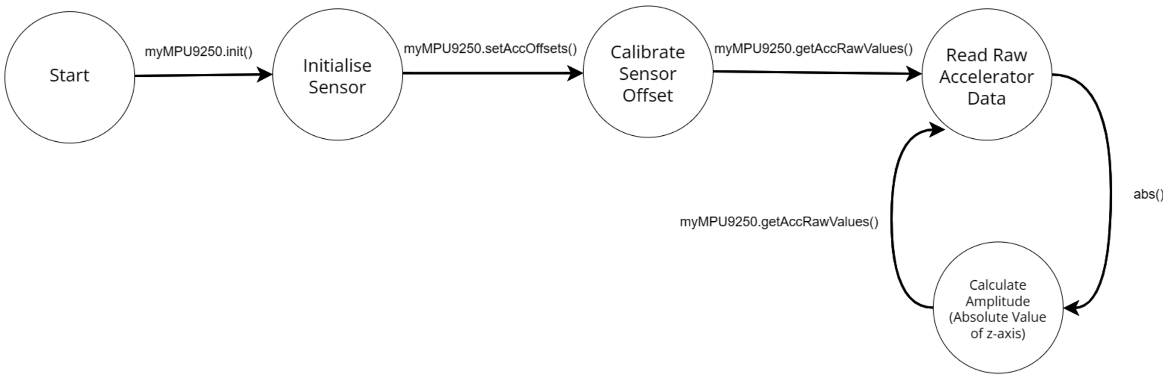

Figure 1 FSM Diagram of Sensor Layer

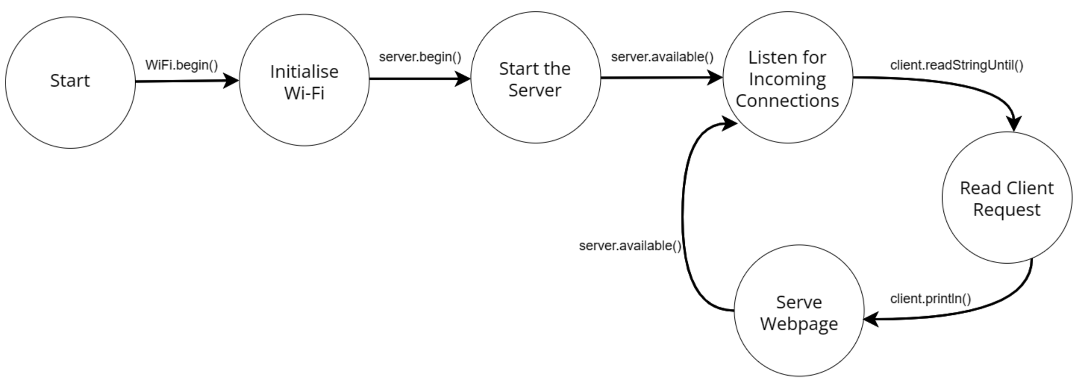

Figure 2 FSM Diagram of Network Layer

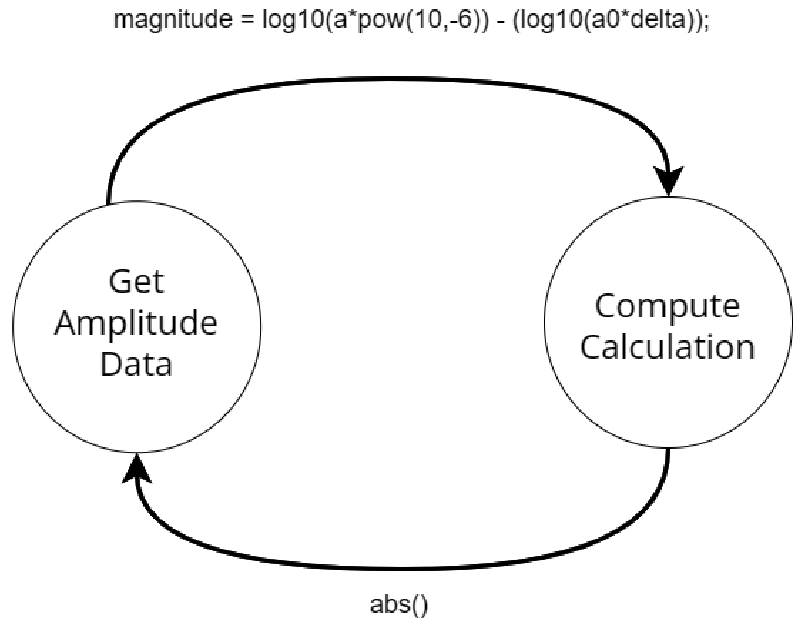

Figure 3 FSM Diagram of Platform Layer

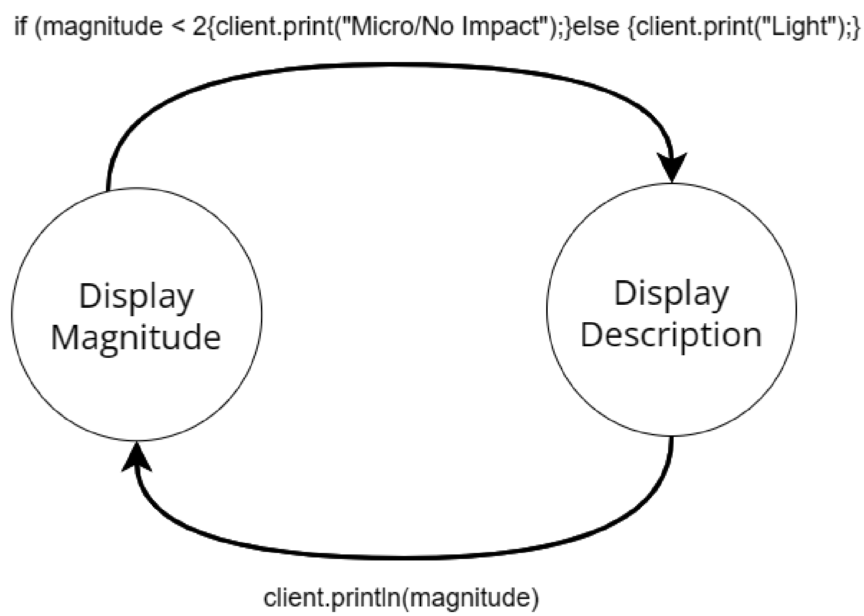

Figure 4 FSM Diagram of Application Layer

1.2 Sensors Justifications

1.2.1 Hardware Components

The hardware components which are required to build the earthquake monitoring system includes 1 accelerator (MPU9250), 2 NodeMCU (ESP8266) with USB cable, 1 breadboard, few female-to-male jumper wires, few female-to-female jumper wires and 1 vibration motor.

1.2.2 Bill of Materials (BOM)

	Item Name	Description	Quantity	Unit Price (RM)	Subtotal (RM)
1.	Accelerator	Pre-soldered (MPU9250)	1	14.99	14.99
2.	NodeMCU	With Wi-Fi enabled (ESP8266, LoLin v3, CH340, with microUSB cable)	2	19.90	39.80
3.	Breadboard	170 holes	1	0.75	0.75
4.	Jumper Wires	120 pcs, 10 cm, Female-to-Male, Female-to-Female	2	3.31	6.62
5.	Vibration Motor	Arduino UNO MEGA2560 R3	1	2.22	2.22
		Total Quantity	7	Total (RM)	64.38

1.2.3 Justification

1. Accelerator (MPU9250): Accelerator will be the sensor involved for building this earthquake monitoring system. It is used to measure the earthquake parameters such as amplitude in the unit of

micrometers (µm) and converted into meter (m) to calculate the magnitude of the earthquake. The accelerator is placed on the epicenter of the earthquake to obtain the amplitude A_0 and code into the NodeMCU, then placed on the current location being impacted to obtain amplitude A from time to time. Both amplitudes will be obtained through the Z-axis of the acceleration value from the accelerator.

2. NodeMCU (ESP8266): NodeMCU will clearly be the controller for the sensor mentioned above. The reason for having two NodeMCU is because one is connected to the accelerator, and another is connected to a vibration motor. NodeMCU A that connected to the accelerator will use the amplitude A retrieve from the accelerator and fixed A_0 that coded in the NodeMCU to compute the calculation for Richter's magnitude, then, transmit the magnitude data to client's display. NodeMCU B will be the controller for vibration motor.

3. Breadboard: 1 breadboard is used to build the circuit for accelerator with NodeMCU.

4. Jumper Wires: Few female-to-male jumper wires are used to connect the NodeMCU with the accelerator and few female-to-female jumper wires are used to connect NodeMCU with the vibration motor.

5. Vibration Motor: Vibration motor is used to generate the vibration as simulating the earthquake for demonstrating the earthquake monitoring system.

1.3 Functional Requirements

1.3.1 Network Layer

The system should be able to handle network communications, setting up a Wi-Fi server and create a simple web interface for client to access earthquake monitoring information via a web browser. First, it needs to initialise Wi-Fi by defining Wi-Fi network credentials (SSID and password) as used to connect NodeMCU to a Wi-Fi network. Then, it needs to create a server object that listens on port 80 for incoming connections. Next, after the Wi-Fi connection is established, the system has to initialise the server using server.begin() and handle the client connection and request. After that, a HTTP response with the earthquake monitoring information should be constructed and send back to the client. The response status (HTTP 200 OK) has to set to indicate success and specify HTML as the content type. The system should use to loop function to check for new client connections and provide updated magnitude and descriptions.

1.3.2 Platform Layer

The system must be able to compute the calculation for the earthquake magnitude using the variables, which is the amplitude received from the accelerator, and the fixed amplitude and distance coded in the NodeMCU. The calculation should be based on the Richter's Magnitude formula, $M_R = \log_{10}(A) - \log_{10}(A_0\Delta)$, where A will be the maximum amplitude obtained through accelerator that placed at current location, A_0 will be the amplitude obtained through accelerator when it placed at the epicenter of earthquake and coded into the NodeMCU as fixed A_0, and Δ will be the distance between the current location and the epicenter of the earthquake which is also coded into the NodeMCU after measurement. The unit of amplitudes A and A_0 as well as the distance Δ should be converted into meter (m). For example, if A obtained is 0.0280 m and A_0 obtained is 0.0152 m, with Δ of 0.05 m, then, the Richter's magnitude of the earthquake will be 1.57.

1.4 Non-Functional Requirements

1.4.1 Application Layer

The system should be able to construct an HTML response to be sent back to the client's web browser in order to display the Richter's Magnitude based on the calculation in platform layer using real-time accelerator data from the accelerator. The magnitude should come with the description of the earthquake impact, such as Micro/No Impact or Light. The classification of the earthquake impact is shown in *Table 1*. Client should be able to access the information by connecting to the NodeMCU's Wi-Fi network. The magnitude and description should be updated continuously as new data is received from the sensor.

Magnitude	**Description**
0.0 – 1.9	Micro/No Impact
2.0 and above	Light

Table 1 Classification of Earthquake Strength

2.0 Develop an IoT System

2.1 System Functionality Demonstration Video Link

https://uowmalaysia-my.sharepoint.com/:v:/g/personal/0207368_student_uow_edu_my/ESDOM9TIVXxKi_K-d-QrvFkBqXdSj5BYf-tDsx4L-OJF-w?nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOiJTdHJlYW1XZWJBcHAiLCJyZWZlcnJhbFZpZXciOiJTaGFyZURpYWxvZyIsInJlZmVycmFsQXBwUGxhdGZvcm0iOiJXZWIiLCJyZWZlcnJhbE1vZGUiOiJ2aWV3In19&e=VPyWh6

CET3063/N/CET3064 Internet of Things

MARKING RUBRIC

Assignment 1

Implement an IoT system (Weighted marks: 30%)

Question 1: Reviews on an IoT system (Score: 50%)

LEARNING OUTCOME	MARKING CRITERIA	SCALE					
		Failed (0% to 49%)	**3rd class (50% to 59%)**	**2nd lower (60% to 69%)**	**2nd upper (70% to 79%)**	**1st class (80% to 100%)**	**YOUR MARKS/COMMENTS**
CLO1: Discuss the various Internet of Things (IoT) components required for various environment.	**1(a) Finite-state machine (FSM) (10%)**	No FSM has illustrated for the IoT system.	Brief FSM has drawn for the IoT system. There are major flaws in the FSM.	Good FSM has drawn for the IoT system with minor flaws, however lack of technical details.	Good FSM has drawn for the IoT system with appropriate concepts and technical details are stated.	Excellent FSM has drawn for the IoT system with proper concepts and in-depth technical details are stated.	
	1(b) Sensors justifications (10%)	No discussion on the sensors involved.	Brief discussion on the sensors involved and there are misconceptions for certain terms.	Good discussion on the sensors involved, however lack of technical details.	Good discussion on the sensors involved with proper concepts and technical details are stated.	Excellent discussion on the sensors involved with proper concepts and in-depth technical details are stated.	
	1(c) Functional requirements (20%)	No discussion on the functional requirements for the IoT system.	Brief discussion on the functional requirements for the IoT system. There are misconceptions in the discussion.	Good discussion on the functional requirements for the IoT system, however lack of technical justifications.	Good discussion on the functional requirements for the IoT system with proper concepts and technical details are stated.	Excellent discussion on the functional requirements for the IoT system with proper concepts and in-depth technical details are stated.	
	1(d) Non-functional requirements (10%)	No discussion on the non-functional requirements for the IoT system.	Brief discussion on the non-functional requirements for the IoT system. There are misconceptions in the discussion.	Good discussion on the non-functional requirements for the IoT system, however lack of technical justifications.	Good discussion on the non-functional requirements for the IoT system with proper concepts and technical details are stated.	Excellent discussion on the non-functional requirements for the IoT system with proper concepts and in-depth technical details are stated.	
						Total (50%)	

Question 2: Develop an IoT system (Score: 50%)							
LEARNING OUTCOME	**MARKING CRITERIA**	**SCALE**					**YOUR MARKS/COMMENTS**
		Failed (0% to 49%)	**3rd class (50% to 59%)**	**2nd lower (60% to 69%)**	**2nd upper (70% to 79%)**	**1st class (80% to 100%)**	
CLO2: Evaluate IoT system architecture for a real world application.	**2(a) Sensor layer (10%)**	No development on sensor layer.	Erroneous in the program code for sensor layer, however there are some relevant codes are implemented.	Good developments of sensor layer with error free. However, there are some bugs occurred during runtime.	Good developments of sensor layer with error free. There are shortcomings for the flow during runtime.	Outstanding developments of sensor layer with error free. Efficient process flow during runtime.	
	2(b) Network layer (10%)	No development on network layer.	Erroneous in the program code for network layer, however there are some relevant codes are implemented.	Good developments of network layer with error free. However, there are some bugs occurred during runtime.	Good developments of network layer with error free. There are shortcomings for the flow during runtime.	Outstanding developments of network layer with error free. Efficient process flow during runtime.	
	2(c) Platform layer (10%)	No development on platform layer.	Erroneous in the program code for platform layer, however there are some relevant codes are implemented.	Good developments of platform layer with error free. However, there are some bugs occurred during runtime.	Good developments of platform layer with error free. There are shortcomings for the flow during runtime.	Outstanding developments of platform layer with error free. Efficient process flow during runtime.	
	2(d) Application layer (20%)	No development on application layer.	Erroneous in the program code for application layer, however there are some relevant codes are implemented.	Good developments of application layer with error free. However, there are some bugs occurred during runtime.	Good developments of application layer with error free. There are shortcomings for the flow during runtime.	Outstanding developments of application layer with error free. Efficient process flow during runtime.	
						Total (50%)	
						Overall score (100%)	
						Weighted marks (30%)	