Mobile App Development Assignment 1 2019-20

The Internet of Things – Theory and Implementation

This assignment involves the use of Mobile Device Development, Sensors and Location Awareness, and the Internet of Things. The idea is to develop a system to interact with some sensors and output devices; to interact with them via the Internet using a mobile device and server, and to show an understanding of the design of systems for the Internet of Things.

Module Learning Outcomes Assessed:

- LO1: Understand, develop and use a variety of advanced mobile applications and location aware mobile development technologies, operating systems and environments.
- LO3: Create and assess intelligent location aware mobile applications
- LO4: Research and demonstrate knowledge and practical application of current and novel mobile device techniques

Tasks to be completed for the assignment
There are several tasks to complete, each graded separately, which can be developed independently. They can be completed in any order in the main.

Task 1 – Sensor recording/actuation - 40 marks
Develop a sensor recording system to record the current status of a door lock (using a servo motor) in a database table.

- The table is to be on a remote server to the actual lock/motor.
- A separate class should be developed to open the lock using an RFID tag, opening for authorized tags. Tags and doors should be linked so that a tag only opens the appropriate door (and not, for example, every door)
- The table should record the tag id used to attempt the open the lock, the time, and success/fail of the attempt. You will need other table(s) for tag/room linking
- Appropriate users should be notified of events.
- You may use MQTT or a database approach to the lock/RFID communication, or a combination.
- Data should ideally be sent in json format.

Task 2 – Mobile Phone App Integration – 20 marks
Integrate the tasks 1 above to display/activate the state of the lock, inside an Android application. This task relies on you completing the majority of Task 1.

- A notification should appear on the Android app when the lock is accessed (whether successful or failed).
- The app should have a button to open the lock.
- Some form of user verification should be in place to ensure the correct phone opens the correct door.
- Data should ideally be exchanged in json format.

Task 3 – Design of a sensor network to control water valves for garden sprinklers – 30 marks
This task involves designing a sensor network of nodes to create a Smart Valve System to supply garden sprinklers, and allow smart remote control. Each node should consist of a motorised solenoid valve for water supply (see reference for sample), a processor (single board) and wireless communications. There should be a centralized hub node to communicate with the valve nodes and communicate over the internet to allow remote control via an app on a phone. Other relevant kit may be used/investigated.

The task is to design and price such a system.

Draw an annotated block diagram (systems type chart) showing the valve and hub nodes with their corresponding sensors/actuators, hub and gateway. This will be similar to system diagrams demonstrated in the course lectures for control of heating systems, voice control, lighting systems etc.

- You should indicate the protocols for communication between each sensor-hub-gateway-internet, and any extra hardware that may be needed
- You should indicate the power supplies used
- You should include any information on the type of data sent between node/hub/internet
- You should describe the software that will be in place on each device/server, and where that device/server will be located. You do not have to write the software, just describe what it will process, the input/output, data formats.
- Show how you could make such a system accessible via voice control, using (for example) Amazon Echo or similar.
- Show the pricing to build such a system for a minimum of one node for the valve controller, and the hub node.
- Comment on any security/privacy aspects of your design, and how these may be alleviated.

Sample solenoid valve: (many others available)
https://www.amazon.co.uk/Electric-Solenoid-Valve-Water-Air/dp/B00BVESSJY

Task 4 – Code and Documentation Quality – Incorporated into each task
You will be assessed on the quality of your code in each section, so it should encompass good software development techniques, including (but not exclusively)

- Code refactoring, code reuse etc
- Removal of redundant code
- Code extensibility – ease of extending your code in future
- Code layout, variable naming, self documenting code

Task 5 – Report – 10 marks
Document your tasks in **a single pdf document**, explaining each task, how far you have completed that task and **showing screenshots of the relevant sections in action**. Your code show have suitable console type messages indicating the progress of messages/data through your system. The document will include your design from task 3.

Screenshots should include:

- Android screen captures of the software running at relevant stages
- Messages from each module as the data passes through, showing what is received/sent
- Database table values before and after update attempts to the lock

What to hand in

A single ZIP file (not RAR or other compression) with your PDF report for Task 5, which will include Task 3, (not Word, Open Document etc.) inside; Also include project(s) zip files to import for testing, for both Eclipse and Android Studio.

Your ZIP file should be named with your Surname, Forename, Student_id, eg Smith_Bill_18374562

There will be a link on Moodle to upload your zip file.

Marking Scheme

Tasks 1, 2 and 4 (60 marks) will be assessing your coding skills for the IoT type sensor network. Marks are allocated according to the task marks above, and will include factors such as code quality, expandability, usability/readability of the code, correct functionality and following IoT features/concepts. Additional features will be given credit once the main tasks are complete.

Task 3 (30 marks) will assess your ability to produce well designed IoT system, employing IoT techniques.

Task 5 (10 marks) will assess your ability to competently discuss and present your work and findings.

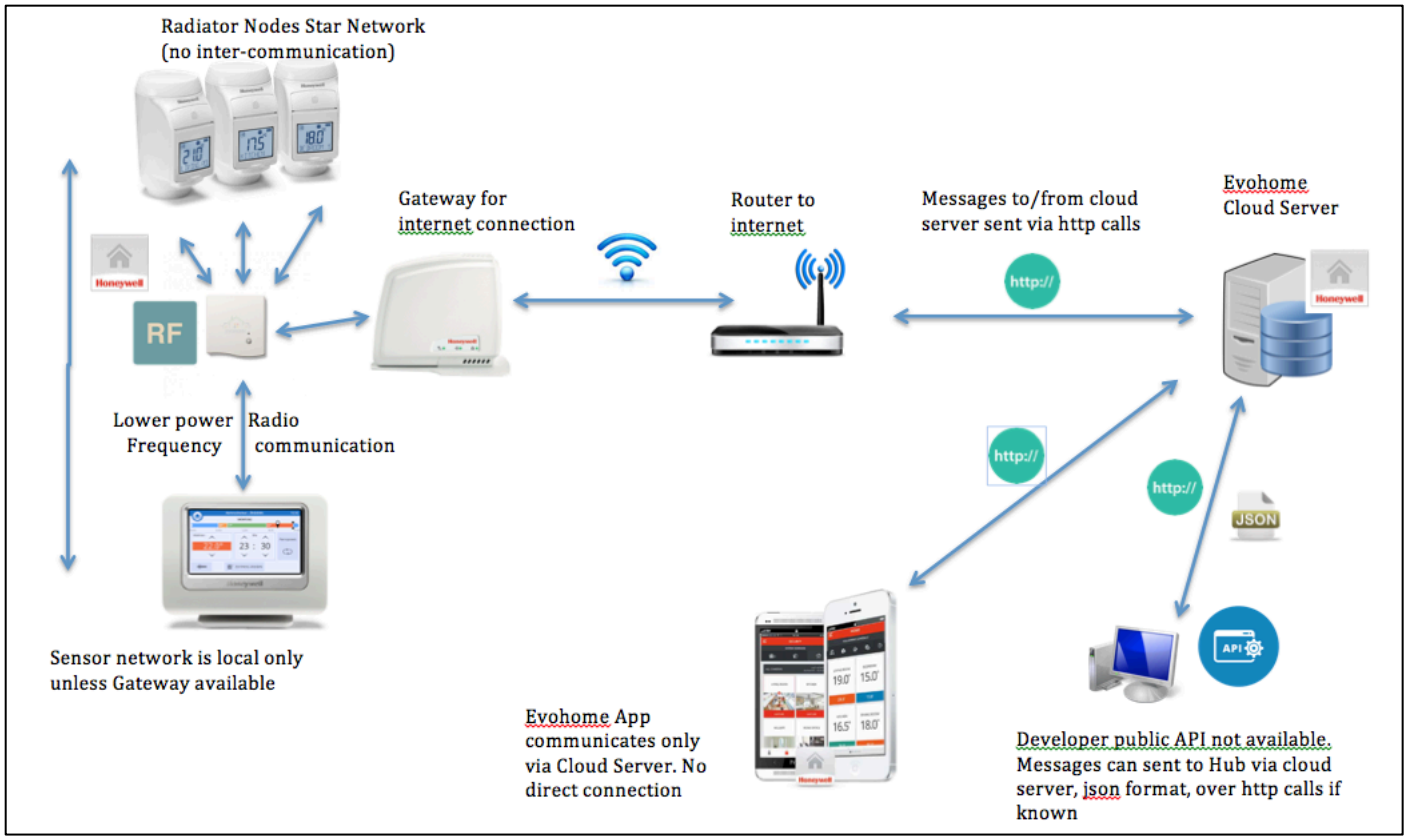
Grade Range	**Task 1&4 Sensor/Actuator (40 marks)**	**Task 2&4 Mobile Device (20 marks)**	**Task 3 System Diagram of Nodes (30 marks)**	**Task 5 Report and Results (10 marks)**
86-100%	As below plus: Code quality is excellent. Code is well structured and easily amended / expanded. Evidence of further development of functionality.	As below plus: Code quality is excellent. Code is well structured and easily amended / expanded. Evidence of further development of functionality. GUI is well designed and standard for platform.	Diagram, pricing and voice add-on completed, well laid out, software fully described and fully annotated diagram. Good discussion of security/privacy aspects.	Excellent presentation, all components complete.
70-85%	As below plus: Good attempts made at improving quality of code. Locks and Tags linked to avoid multiple doors opening.	As below plus: Good attempts made at improving quality of code. Phones/locks are linked to ensure phone only opens correct lock.		
60-69%	As below plus: full details stored in db, with id, success/fail, timestamp. Json exchange fully in place	As below plus: Mobile app sends/receives data to/from database.	Pricing complete, attempt made at including voice control. Software described, better annotation. Some discussion of security/privacy aspects	As below plus: Conformation of each task completed referenced to code and screenshots
50-59%	As below plus: Some attempt made at data storage. Some attempt made at data exchange in json.	As below plus: Mobile app successfully connects to mqtt or server to send / receive data for lock	As below, but more complete. Pricing details attempted. Minimal discussion of security/privacy aspects	As below plus: Screen shots showing code in action included in report
40-49%	Program sends data between RFID device and motor. Values not permanently stored	Mobile app developed that has widgets to display and activate lock	Diagram completed, missing some components and partially described / annotated	Report submitted in required format with project code and report included
30-39%	Code almost working but incomplete	Code almost working but incomplete	An attempt made but not enough detail	Report not submitted or inadequate
20-29%	Valid attempt made but no working software	Valid attempt made but no working software	An attempt made but not enough detail	
0-19%	No attempt or program has major errors	No attempt or program has major errors	No attempt made or insubstantial work	

Formative Feedback
You should make use of labs to discuss your progress, as well as emailing the tutor (N.Whittaker@mmu.ac.uk) or book personal appointments via the availability link on the Moodle page for the module. There will be lots of help provided in lectures and online via the forum on Moodle.

CAUTION:
Phidget Kits are available to borrow from the technicians in room C208. If you abuse the lending system by not returning on time, there is a penalty of having marks removed from your assignment.

There are a number of online implementations of this and similar concepts. While it is OK to review these for ideas, it is not OK to copy whole or parts of these code examples and pass them off as your own. We will use an AUTOMATED PLAGIARISM CHECKER to compare your submissions against other students' work and online examples. In the past, this has led to students being awarded 0 marks.

Sample Systems Diagram for a Honeywell Node Connected to Radiator Control Valves



Department of Computing and Mathematics

ASSIGNMENT COVER SHEET

Unit title:	6G6Z1104 Mobile Application Development
Assignment set by:	Nick Whittaker
Assignment ID:	1CWK50
Assignment title:	IoT Theory and Implementation
Assignment weighting:	50%
Type: (Group/Individual)	Individual
Hand-in deadline:	Friday 13 December 2019, 21:00
Hand-in format and mechanism:	Zip file via Unit area on Moodle
Support:	*Support in all labs after week 5. Formative feedback on Forum/ Office hours shown on main Moodle page for personal meetings.*

Learning outcomes being assessed:

- LO1: Understand, develop and use a variety of advanced mobile applications and location aware mobile development technologies, operating systems and environments.
- LO2: Critically assess concepts of wireless network security
- LO4: Research and demonstrate knowledge and practical application of current and novel mobile device techniques

Note: it is your responsibility to make sure that your work is complete and available for marking by the deadline. Make sure that you have followed the submission instructions carefully, and your work is submitted in the correct format, using the correct hand-in mechanism (e.g. Moodle upload). If submitting via Moodle, you are advised to check your work after upload, to make sure it has uploaded properly. Do not alter your work after the deadline. You should make at least one full backup copy of your work.

Penalties for late hand-in: see Regulations for Undergraduate Programmes of Study (http://www.mmu.ac.uk/academic/casqe/regulations/assessment.php). The timeliness of submissions is strictly monitored and enforced.

All coursework has a late submission window of **5 working days**, but any work submitted within the late window will be capped at 40%, unless you have an agreed extension. Work submitted after the 5-day window will be capped at zero, unless you have an agreed extension.

Please note that individual tutors are unable to grant extensions to coursework. Extensions can only be granted on the basis of a PLP, or approved Exceptional Factors (see below).

Exceptional Factors affecting your performance: see Regulations for Undergraduate Programmes of Study (https://www.mmu.ac.uk/academic/casqe/regulations/assessment/docs/ug-regs.pdf). For advice relating to exceptional factors, please see the following website: https://www2.mmu.ac.uk/student-case-management/guidance-for-students/exceptional-factors/ or visit a Student Hub for more information.

Plagiarism: Plagiarism is the unacknowledged representation of another person's work, or use of their ideas, as one's own. Manchester Metropolitan University takes care to detect plagiarism, employs plagiarism detection software, and imposes severe penalties, as outlined in the Student Handbook (http://www.mmu.ac.uk/academic/casqe/regulations/docs/policies_regulations.pdf and Regulations for Undergraduate Programmes (http://www.mmu.ac.uk/academic/casqe/regulations/assessment.php). Bad referencing or submitting the wrong assignment may still be treated as plagiarism. If in doubt, seek advice from your tutor.

As part of a plagiarism check, you may be asked to attend a meeting with the Unit Leader, or another member of the unit delivery team, where you will be asked to explain your work (e.g. explain the code in a programming assignment). If you are called to one of these meetings, it is very important that you attend.

Assessment Criteria:	Indicated in the attached assignment specification.
Formative Feedback:	Lecture/Lab discussion and interactive with tutor onwards from week 5
Summative Feedback Format:	You will be given individual feed back via Moodle, as well as common feedback for all the class.