# COMP6733 Research Project, Term 2, 2025

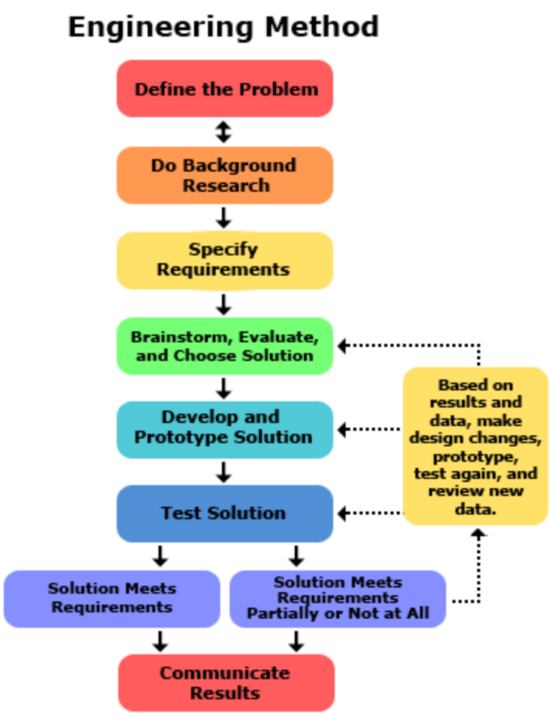
Version 1.0, 1 June 2025.

**Team formation and preferences due: 17:00, Friday 20 June 2025 (Week 3)**
**Class Presentation (1): lecture hours, 30 June, 2 July 2025 (Week 5)**
**Preliminary Report Due: 17:00, Friday, 4 July 2025 (Week 5)**
**Class Presentation (2): lecture hours, 21, 23 July 2025 (Week 8)**
**Intermediate Report Due: 17:00, Friday, 25 July 2025 (Week 8)**
**Report and Code Due: 17:00, Sunday 10 August 2025 (Week 10).**
**Demo date: 11 August 2025 (Week 11).**

**NOTE:** Updates to this assignment, including any corrections and clarifications, will be posted on the course website. Make sure that you check the website regularly for updates.

## 1. Overview

The project component of COMP6733 aims to give you an opportunity to solve a real-life problem in IoT. The format of the project aims to emulate the classical engineering design process which is depicted in the below figure. This project will therefore test your ability to solve problems, propose solutions and analyse their suitability as well as your skills in implementing a solution, which includes planning, testing and programming. The scope of the project is fairly open to allow you to explore the subject area.




Section 2 of this document describes a number of projects that you may choose to work on, however you are free to propose your own project. You are expected to do the project in a team of four students and bid for these projects by expressing your preferences. The Lecturer-in-Charge (LiC) will then allocate the projects according to your expressed preferences by Week 3. More details on the bidding process are described in Section 3.2.

The project consists of two stages. In the first stage, you should consider different options that you can use to solve the problem. You do this by analyzing the problem and perform research on possible solutions. Based on your research and analysis, you are required to propose how you plan to tackle this problem. You are required to submit a preliminary report and give a project presentation (in Week 5) to explain your proposed solution. Your report should include:

1. A discussion on how you plan to tackle the problem with your justification.

2. The implementation that you plan to do with timeline.

3. The goals of your demonstration. You will need to specify goals on what you will be

demonstrating in the final demonstration.

The second stage of the project will be devoted to working towards the goal that you have set for yourselves.

The LiC will go through your preliminary report and provide feedback on the work that you plan to do. In all cases, you should be aware the University expects students taking a 6 UoC course to devote about 14 hours per week to each course. Given the above expectations on the amount of time you will be spending on the project and to ensure that all projects meet a common standard, the LiC may choose to add, remove or change tasks from your proposed plan.

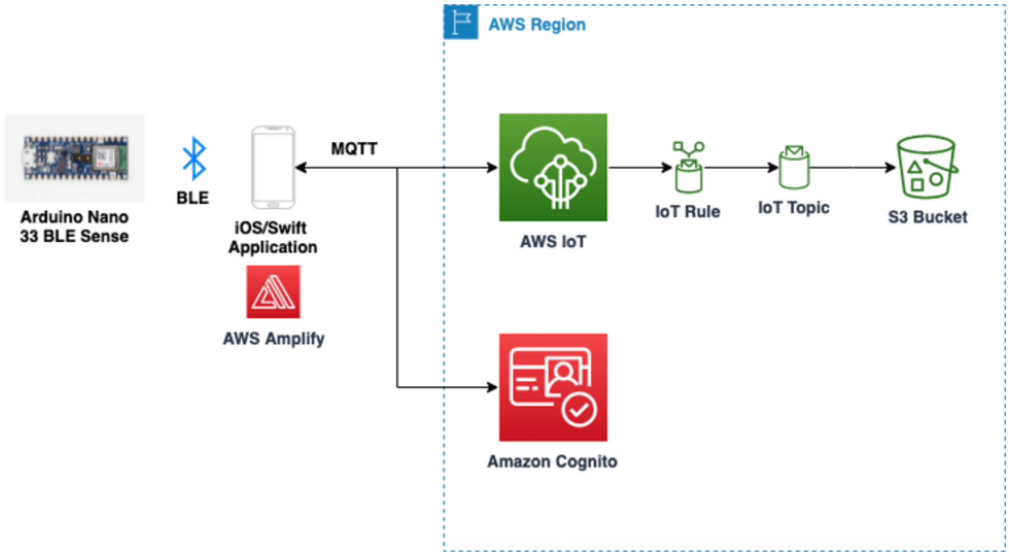
## 2. Project description

### 2.1 Project 1: ***Android/iOS smartphone AWS IoT (MQTT)/Bluetooth Low Energy (BLE) gateway***

In the labs, you have used a notebook computer as a (MQTT) gateway (e.g., Bleak) to bridge the communications between BLE-based IoT devices and the AWS IoT cloud service, and use a smartphone app (e.g., nRF Connect) to communicate with the IoT devices via BLE.

In this project, you will design and implement an Android OR an iOS smartphone gateway to bridge the communications between BLE-based IoT devices and the AWS IoT cloud service via MQTT messaging by referring the design and functionality of Bleak and nRF Connect, and investigate its performance such as throughput, latency, resource (energy) consumption and cost. You may also consider how to choose the WiFi or cellular interfaces in the smartphone phones for Internet connection.

Please see below of one possible solution architecture and steps.




1. Configure Bluetooth connection to the BLE device:

- On Android, use such BluetoothAdapter and BluetoothGatt classes to scan, connect, and communicate with BLE devices. On iOS, use the CoreBluetooth framework (or refer to tutorial 4). Establish a BLE connection and handle data exchange with the BLE devices.

2. Exchange data from the BLE devices:
    - Implement the necessary code to exchange data from the connected BLE devices. This could involve reading, writing, synchronisation, etc.
3. Install and configure MQTT client on the smartphone:
    - Install an MQTT client library or application on your smartphone. For Android, you may use Eclipse Paho. For iOS, you may use libraries such as MQTT-Client-Framework or Moscapsule.
    - Set up the MQTT client with the appropriate connection parameters (e.g., AWS IoT endpoint, client ID, certificates/keys).
4. Exchange data with the AWS IoT core via MQTT

5. Develop smartphone GUI for a user to input BLE Generic ATTribute profile (GATT) for their BLE services (please refer to the design of nRF Connect in Tutorial 4) and AWS IoT CORE connection parameters such as AWS IoT endpoint, client ID, certificates/keys.

We have experience with Eclipse Paho MQTT in Android Studio and there are also multiple ways to design the project such as the AWS Amplify in the figure above.

[1] Tutorials 4 and 5.

[2] https://docs.aws.amazon.com/iot/latest/developerguide/iot-sdks.html#iot-mobile-sdks

[3] https://developer.android.com/guide/topics/connectivity/bluetooth/ble-overview

[4] https://developer.apple.com/bluetooth/

### 2.2 Project 2: *virtual fencing technology for rotational grazing*

Labour shortages make manual herd-moving costly. Virtual fencing lets graziers geo-fence cattle without physical wires. In this project you will implement an indoor WiFi/bluetooth localisation method based on RSSI fingerprinting with a smartphone and smart wristband (i.e., Arduino Nano) and detect the animal's location in real time. If an animal crosses the virtual fence, a buzzer will be triggered along with a message sent to the farmer. You will use Wi-Fi/Bluetooth fingerprinting that creates a radio map of a given area based on the RSSI data from several access points and generates a probability distribution of RSSI values for a given (x,y) location. Live RSSI values are then compared to the fingerprint to find the closest match and generate a predicted (x,y) location. You will evaluate such systems with real smart deployment in real environment (e.g., your residence).

The reference paper, *RADAR: An In-Building RF-based User Location and Tracking System* proposes one particular algorithm but you are free to implement other algorithms.

### 2.3 Project 3: *Gait-based authentication in wearable Devices*

A person may be identified from their walk pattern which can be recorded and modelled using an IMU IoT sensor (Arduino Nano 33 BLE Sense). In this project, you will develop and implement a biometric (gait) authentication method to ensure the identity of the wearer by making use of the IMU/motion sensor available in the Arduino Nano 33 BLE Sense. Specifically, your system will observe the walking pattern of a person (IMU measurements) and try to match it with the one stored in the database to decide the identification of the wearers.

Papers:

[1] Unobtrusive Gait Verification for Mobile Phones

[2] Gait-watch: A context-aware authentication system for smart watch based on gait recognition

### 2.4 Project 4: *Heartbeat-based authentication in wearable Devices*

The gait-based authentication system developed in Project 2.3 requires a person to walk to be identified, and it won't operate if a person is static (e.g., sleeping at home). In this project, you will use the sensor signals (e.g., IMU, pressure) recorded in Arduino Nano 33 BLE Sense to measure heartbeats of a person and to identify her/him.

Paper: *Unlock with Your Heart: Heartbeat-based Authentication on Commercial Mobile Phones*

### 2.5 Project 5: *LLM-guided, low-cost multispectral controller for crop health*

Vertical farms typically use pre-determined schedules for nutrient and light delivery to plants, and struggle to dynamically alter the resource inputs to optimise for photosynthesis efficiency. Industrial multispectral cameras cost >$1k; the AS7265x is a cost-effective sensor that gives 18 discrete VIS/NIR channels covering the plants photosynthesis light spectrum. Current practice involves a manual operated tool which is clamped on the leaf, which is labour-intensive and limits scalability. In this project, you will design an Arduino-based system to monitor the light spectrum of healthy/stressed leaves using an AS7265x spectrum sensor mounted next to a plant leaf. You will implement an end-to-end system by capturing reflectance spectra data every 30 min, then use a cloud-based LLM (e.g. ChatGPT API) to determine plant stress in real-time, and finally push the actionable results over MQTT gateway to AWS IoT.

Reference paper:
*What to Choose for Estimating Leaf Water Status—Spectral Reflectance or In vivo Chlorophyll Fluorescence?*

### 2.6 Project 6: *ThermoLeaf – FLIR-Lepton canopy-temperature analytics*

Leaf-to-air temperature difference ($\Delta T$) is a proven proxy for plant water stress, as the cooling effect from transpiration is reduced resulting in increased leaf temperature. Commercial IR cameras are costly and rarely used, however, recent IoT thermal cameras, such as FLIR Lepton, could be low-cost alternative.

In this project, you will build a deployable system (i.e., field-based node using a Raspberry Pi and FLIR Lepton), which can 1) display a thermal image of the plant and subtract ambient air temp (measured from BME280 or Arduino Nano BLE node) to yield a ΔT map. 2) Closed-loop demo where a heat-lamp briefly raises ΔT, and system triggers plant watering when defined threshold is reached. 3) Evaluation: correlation ($R^2$) between ΔT and soil-moisture sensor. 4) Stretch: build Edge-TPU model to predict ΔT 60 min ahead.

Reference paper: *Feasibility study on Lepton 3.5 in terms of accuracy for measuring leaf temperature of crops*

Thermal camera: *https://www.flir.com.au/products/lepton/*

### 2.7 Project 7: *Monitoring Building Occupancy*

CSE building managers want to check occupancy numbers for teaching spaces and need a minimum distance of 1m between people for building safety compliance. Therefore, every room has a maximum number of occupancy limit depending on its size. In this project you will implement a system for monitoring building occupancy. Your system will use WiFi and/or Bluetooth signals (transmitted by user's devices such as smartphones) as indicators of the amount of people present. Almost everyone carries a smartphone today, and these devices will periodically attempt to connect to WiFi and/or Bluetooth networks (often successfully e.g. when you are on campus your devices connect to the Uniwide network) in the surrounding by broadcasting beacon frames. Your system will consist of a network of IoT nodes that sniff and record the WiFi and/or Bluetooth messages exchanged between these devices and the WiFi access and/or Bluetooth beacon points. Your system should fuse data from multiple devices to arrive an estimate of the occupancy levels in a given area. You may also consider using triangulation to uniquely identify and locate each device within the building.

Reference paper: *Estimating Crowd Densities and Pedestrian Flows Using Wi-Fi and Bluetooth*.

### 2.8 Project 8: *Hand washing quality monitoring with mmWave Radar*

Hand washing is one of the best ways to stop viruses spreading, if done correctly and properly. Unfortunately, not all of us wash our hands frequently. Out of those that do, it is estimated only 5% of us know how to do so properly.

In this project, you will build a privacy-preserving device free system (i.e., radio sensing with TI mmWave radar), which can detect 1) how many times per day a person washes her hands, 2) the quality of these hand wash events. You will also develop an associated mobile app that can report these statistics to the user in real time. Furthermore, the app can coach a user to wash her hand properly via a step-by-step guild and corrections.

Reference paper: ***RFWash: a weakly supervised tracking of hand hygiene technique***

### 2.9 Project 9: *Accurate body motion monitoring system.*

You will build an accurate body motion monitoring system using a network of Arduino Nano 33 BLE Sense worn in different parts of body (wrist, arm, leg, waist etc.). The system also comes with a mobile app that can show the body motion via an avatar in real time. The potential applications include different sports coaching and activity detection.

See Zepp Tennis (http://gadgetsandwearables.com/2015/04/04/zepp-tennis-swing-analyser/) and golf (http://www.zepp.com/en-us/golf/smart-coach/) swing analyzer for one example.

### 2.10 Project 10: *Bringing IoT to Sports Analytics*

You will explore the possibility of bringing IoT to sports analytics, particularly to the game of Cricket or any other ball games. You will develop solutions to track a ball's 3D trajectory and spin with inexpensive sensors (e.g., IMU, depth sensor such as Intel RealSense and cameras such as iPhone 12 Pro) by addressing the challenges of localization and motion tracking accuracy.

Your system may fuse disparate sources of partial information – wireless, inertial sensing, and motion models – into a non-linear error minimization framework, which achieves centimeter accuracy compared to ground truth.

Reference papers: [1] *Condor: Mobile Swing Tracking via Sensor Fusion using Conditional Generative Adversarial Network.*
[2] *SwingNet: A Ubiquitous Fine-Grained Swing Tracking Framework via Neural Architecture Search (NAS) and Adversarial Learning*

### 2.11 Project 11: Tooth brushing Monitoring using Wristwatch

Daily tooth brushing is essential for maintaining oral health. However, there is very limited technology to monitor the effectiveness of tooth brushing at home. In this project, you will build a system to monitor the brushing quality on all 16-tooth surfaces using a manual toothbrush and an off-the-shelf wristwatch. You may modify the toothbrush by attaching small magnets to the handle, so that its orientation and motion can be captured by the magnetic sensor in the wristwatch.

Your system will recognize the tooth brushing gestures based on inertial sensing data from the wristwatch. As the acoustic signal collected from the watch is correlated with the motion of tooth brushing stroke, you may use an acoustic sensing algorithm to assist in recognition. User-specific tooth brushing order may also be utilized to improve the surface recognition.

Reference paper: *Toothbrushing Monitoring using Wristwatch*

### 2.12 Project 12: *Real-time body mesh with mmWave radar.*

You will build a real-time body motion monitoring system using mmWave radar. The system also comes with a mobile app that can show the body motion via an avatar in real time. The potential applications include different sports coaching and activity detection.

Reference paper: *mmMesh: Towards 3D Real-Time Dynamic Human Mesh Construction Using Millimeter-Wave*

### 2.13 Project 13: Leaf Moisture Monitoring with mmWave radar

Accurately monitoring plant health is key for high crop yield. New sensors are required to remotely measure plant physiology. Current techniques require attaching sensors onto individual leaves which limits large-scale deployment. Radar is a new sensing technique to determine plant health by examining the reflected signal to extract water content in the plant. You will build a system to track the growth of plant using a camera with environmental sensors (temp, humidity, light etc) along with a new radar sensor.

Reference papers: *Leafeon: Towards accurate, robust and low-cost leaf water content sensing using mmWave radar*

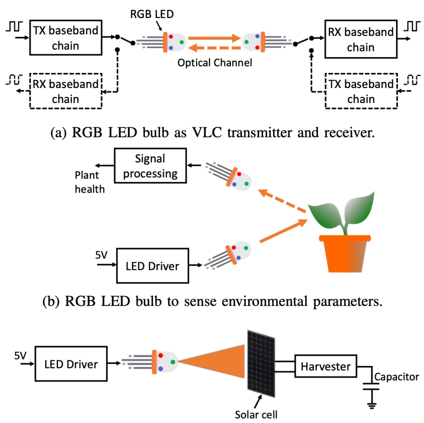
### 2.14 Project 14: Large Language Model based Zero-Shot Human Activity Recognizers

There is an ongoing debate regarding the potential of Large Language Models (LLMs) as foundational models seamlessly integrated with Internet of Things (IoT) for interpreting the physical world. In this project, you will carry out a case study to answer the following question: Are LLMs capable of zero-shot human activity recognition (HAR)? Your study, called HARGPT, will demonstrate that LLMs can comprehend raw IMU data and perform HAR tasks in a zero-shot manner, with only appropriate prompts. HARGPT inputs raw IMU data into LLMs and utilizes the role-play and "think step-by-step" strategies for prompting. You will implement an end-to-end system by collecting IMU data from Arduino Nano and interpret them in ChatGPT in real-time to recognise human activities from raw IMU data.

Reference papers: *HARGPT: Are LLMs Zero-Shot Human Activity Recognizers?*

### 2.15 Project 15: RGB LED Bulbs for Communication, Harvesting and Sensing in smart farming

RGB LED bulbs have entered the market as a promising alternative to traditional phosphor coating LEDs to meet illumination standards. In this project, your will explore RGB LED lighting for the Internet of Things smart farming applications, performing communication, harvesting energy and sensing tasks (see below figure).



Reference papers: *RGB LED Bulbs for Communication, Harvesting and Sensing*

### 2.16 Project 16: *Micro-climate analytics for indoor vertical farms*

Maintaining uniform temperature, humidity, and light intensity at canopy level is critical to maximising yield and preventing diseases in vertical farming systems, yet most commercial growers rely on just a few wall-mounted sensors that miss shelf-level variations. In this project, you will design a distributed sensing network to measure and map micro-climate conditions across each tier of a vertical farm in real-time. Each sensor node will capture temperature, humidity, and photosynthetic active radiation (PAR) data, transmitting measurements every minute via BLE to a Jetson-based gateway. The gateway aggregates this data into 15-minute summaries, which are then analysed locally by a compact 7-B parameter LLM (e.g., Llama). The LLM identifies hot, dry, or insufficiently lit areas and issues JSON-formatted control commands (e.g., { "fanB": 90, "ledRow3": -15 }) over MQTT to fan and LED controllers. The system also uploads daily data to the cloud for deeper analysis by GPT-4o, providing system reports to growers. Evaluation: show the edge pipeline flagging an induced hot-spot and correcting it within 5 min.

**Reference paper: *Fast: A Ubiquitous Computational Model for Temperature and Humidity Sensing in Industrial Environments***

### 2.17 Project x: *Proposed your own IoT project*

### 2.20 Scope of the project

1. In your proposal, you will need to state how you want to demonstrate your results. For example, you want to demonstrate that the physical distancing algorithm that you have implemented is working. You may program the nodes so that they will light the (red) LED up if the distance between two nodes is less than 1.5m. You may change the radio environment a few times in the demonstration and show that it works for different environments (e.g., outdoor and indoor).

2. If you require additional hardware, please discuss your requirements ASAP with the LiC. We can purchase some additional sensors provided that they are not very expensive.

3. The preliminary report should include a discussion on your research, how you plan to tackle the problem, work plan and milestones for Week 8 and final demonstration.

4. There are a number of good conferences and journals on the IoT.

   1) Both the digital libraries of ACM and IEEE will be excellent starting point of your research. Their URLs are respectively:

a. http://portal.acm.org/portal.cfm

b. http://ieeexplore.ieee.org/Xplore/dynhome.jsp?tag=1

2) ACM has serval IoT, mobile and ubiquitous systems conferences: IPSN (part of CPS-IoT Week), SenSys, MobiSys, MobiCom and Ubicomp.

## 3. General Requirements

### 3.1 Choice of project

You can do one of the proposed projects or propose your own project. If you want to propose your own project, please make an appointment to speak with the LiC with your proposal in Week 2 or 3.

### 3.2 Formation of project teams and bidding process

You are expected to do this project in a group of students. You are free to form your own team. Please enter your team membership using the WebCMS function available at the course website.

Please email cs6733@cse.unsw.edu.au your team membership or list it in webcms (https://webcms3.cse.unsw.edu.au/COMP6733/25T2/groups/, please make your group visible) by **17:00, Friday 20 June 2025**. In your email, please also include your preferences for each of the proposed projects. You are asked to express your preference by giving each project a score of 1, 2 and 3 where '1' means a project that you most want to do and '3' means a project that you least want to do. Please express your preference for all the projects. You can have any combination of 1 to 3, e.g. you can give all projects a '3' or you can give 2 projects a '3' and a project a '1'.

The LiC will attempt to allocate the projects according to your preference. The aim is to attempt to balance the number of groups doing each project.

If we do not receive your team membership by this time, we will do a random team formation for you.

### 3.3 Assessment

Note the following assessment and submission requirements (***one report from each group please***):

1. You should submit a preliminary report (with your proposed solution, project plans, milestones for both Week 8 and final demonstration) by **17:00 Friday 4 July 2025** via the *give* system. The report will be assessed based on the quality of the research that you have done, your explanation on why your proposed solution best fit the problem and the quality of your work plan. The give command to use is: "give cs6733 projectPlan". Your project report must be in PDF format. Please tar your report and call it "project0.tar". This is the only filename accepted by the system.

2. In Week 5, each team will give a presentation on their proposed solution and their proposed work plan to the whole class. Each team member should do a part of the presentation. LiC and fellow students will assess the presentations according to predefined marking criteria.

3. In Week 8, each team will give a presentation on their project progress and demonstrate its

project milestones to the whole class. The aim is to ensure that you have made progress on the project. Each team member should do a part of the presentation. LiC and fellow students will assess the presentations according to predefined marking criteria. In Week 10, each team will need to negotiate with the LiC on a time to demonstrate. The milestone report is to be submitted by **17:00 Friday 25 July 2025** via *give*. The give command to use is: "give cs6733 projectMilestone". Please tar your final submission and call it "project1.tar". This is the only filename accepted by the system.

4. The project report and code are to be submitted by **17:00 Sunday 10 August 2025** via *give*. The give command to use is: "give cs6733 projectFinalReport". Your project report must be in pdf format. Please tar your final submission and call it "project2.tar". This is the only filename accepted by the system.

5. A final demonstration and interview will be held on **11 August 2025**. Each team is given approximately 30 minutes for the demo, and this will also include a short interview with each student. You will be asked to express preference on the interview day later on.

6. The project in total is worth 60% of your final mark in this course. The breakdown is as follows:

1. Project Plan and preliminary report + Class Presentation in Week 5: 10%
2. Project progress demo + class presentation in Week 8: 10%
3. Project final report: 15%
4. Project final demo: 25%

**The deadlines are hard. No extensions will be offered.**