

## COMP6324 - IoT Project Specifications

### SmartSeat –

- Navigate to seat
    - IoT Solution: GPS when far away, then switch to beacon when within range.
    - Map on smartphone
    - What do we need
  - Map of stadium
  - Smartphone (GPS)
  - Beacon on the seat
  - QR Codes for seat number on the ticket and on the seat
- 
- Order food/ snacks (pay online)
    - Fixed vendor
      - QR code for location
      - Website/app for making order, then scan QR code for location
      - Payment gateway (pay with credit card/ apple pay)
      - Track order
        - Stages of order - order sent, order preparation, order dispatched
        - Provide estimate of time from order sent to delivery
        - Once order leaves shop provide start tracking using beacon - tracking runner (runner has beacon) on the map

# IoT Solution Design

Team:

Team members:

1. Adam Golding
2. Jie Shang
3. Xinran Li
4. Michael Liyantama
5. Qian Cheng

## IoT Solution Design

### Requirements

#### Functional Requirements (Shang Jie)

[Identify **what** the IoT Solution should do]

1. Navigation
  - 1.1. The solution should be designed to navigate user from the outside area to the stadium as well as from the entrance to the seats.
2. Map of stadium
  - 2.1. The solution must can show the user's location on the map of the stadium.
  - 2.2. The solution should show the vendor's location when they get close to the user who order the food.
3. Website
  - 3.1. Website be able to show up after scan the QR code on the ticket or the seat.
  - 3.2. The solution must can show the menu for the food and able to order the food online. Payment gateway should be provided for the user to pay online.
  - 3.3. The solution should be able to show the rough waiting time for the food delivery and can track the order. The order details, including order ID, payment way, order time, food, user's seat number, should also be shown on the website.

- 3.4. The solution should be able to show the order stage: order sent, order preparation, order dispatched
4. QR code
  - 4.1. The QR code should be unique to each user, it contains the seat number and the location of the seat.
  - 4.2. The solution should be able to let user login the website to order the food service.
5. Smart seat
  - 5.1. The solution should be able to show the location of the seat

### **Non-functional Requirements (Adam)**

#### **1. Security**

The solution must be secure. It should be protected from unauthorised access. It should provide different levels of access depending on the type of user. The solution must also be protected against virus and malware attacks.

- Encrypted data over network using certificates
- Store data in a secure environment

#### **2. Confidentiality**

The data should be transmitted and stored in a secure and safe way. Data should only be accessed by authorised users. Data should only be created, edited, deleted and viewed by appropriate users.

- Implement appropriate security groups and roles for users of the solution, this includes user types who provide administration services
- Implement audit solution to track changes to and access of the data

#### **3. Accessibility**

The solution should be designed to allow use by all types of users. This includes people with disabilities and special needs. The solution should use appropriate assistive technology where required.

- Design for use with screen readers
- Use appropriate colour schemes for people with vision impairment
- Ensure fonts can be scaled

#### 4. Availability

Specific modules and functions classified as critical must always be available. Users must be notified if the system (or modules of the system) are unavailable at any time. SLAs must be in place with external providers.

- Critical functions need to be identified
- Ensure failover is operating for critical modules
- Define Service Level Agreement with external providers

#### 5. Reliability

The solution must work without failure for as long as possible. The solution development methodology and process should deal with coding problems and bugs. If an error is encountered it must be handled in an appropriate way. Any transactions involving external systems, e.g. payment gateway, must work appropriately.

- Ensure appropriate testing methodology has been used
- Provide a mechanism to track errors and bugs

#### 6. Usability

The solution should be easy to learn and use. The UI component should be easy to navigate. Users should be able to complete tasks without help in the minimum possible time and steps. The solution should handle errors gracefully.

- Utilise user experience methodologies to design the user interface
- Provide informative error messages to the user

#### 7. Safety

The solution should protect the user from harm.

- When utilising navigation modules, ensure the solution does not provide incorrect directions that can put them in danger

#### 8. Flexibility

The solution must be designed to be flexible and allow the introduction of new modules and functions.

## 9. Maintainability

The solution should be easy to maintain. Clear roles and responsibilities should be adopted to ensure issues are dealt with quickly.

- Define roles and responsibilities for all parts of the solution
- Establish levels of support (Level 1, Level 2 and Level 3)
- Hardware (sensors) must be able to be replaced quickly and with minimum effort and skill.

## 10. Scalability

- The solution must scale to support 1,000s of users.
- The software must be designed to handle increases in transactions and users.
- The solution must be able to add more sensors without affecting performance.

## 11. Interoperability

- The solution must be able to integrate to external systems such as a payment gateway. All APIs should be easy to implement and access.
- The solution must allow data exchanged between systems from different organisations.

## 12. Reusability

- The solution must be designed that allows the reuse of components. This includes software components and hardware components.

## 13. Performance

- The solution must be able to handle 1,000s of users and respond in real time.
- The response times must be as close to real time as possible.
- The solution must be designed to minimise resource utilisation (CPU, memory, disk).

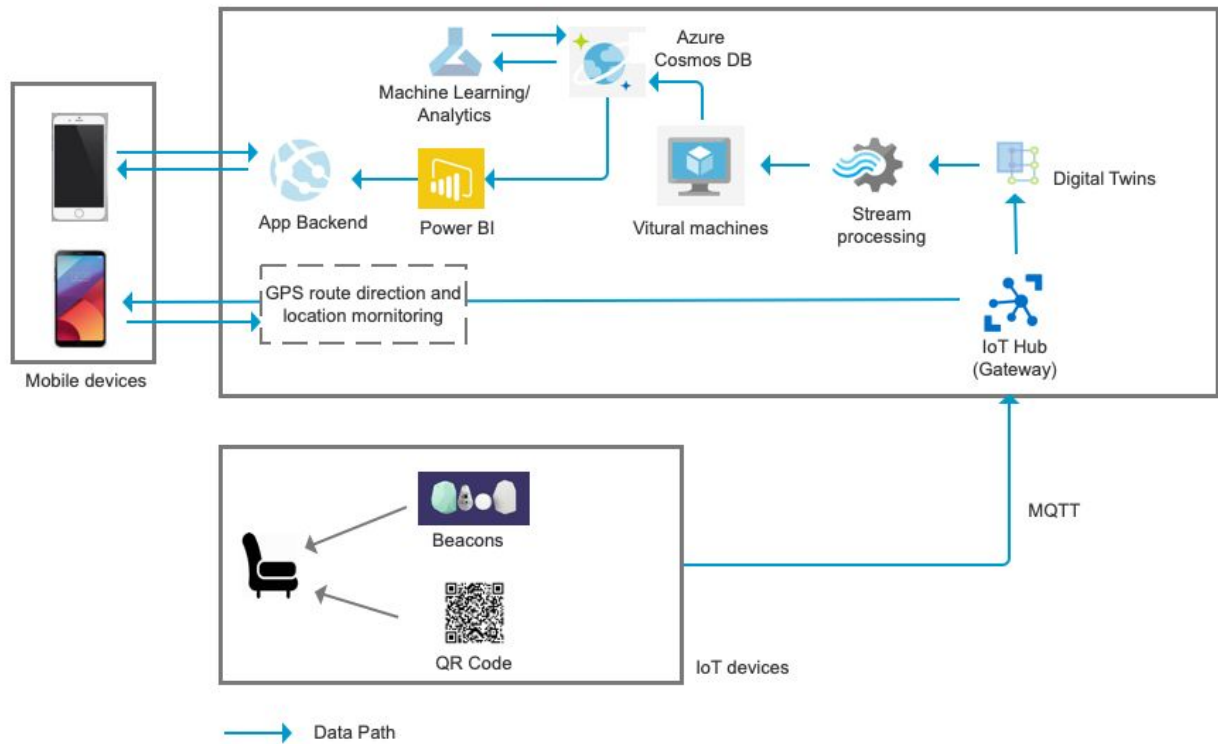
## 14. Legal and Regulatory

The solution must abide by all legal and regulatory requirements.

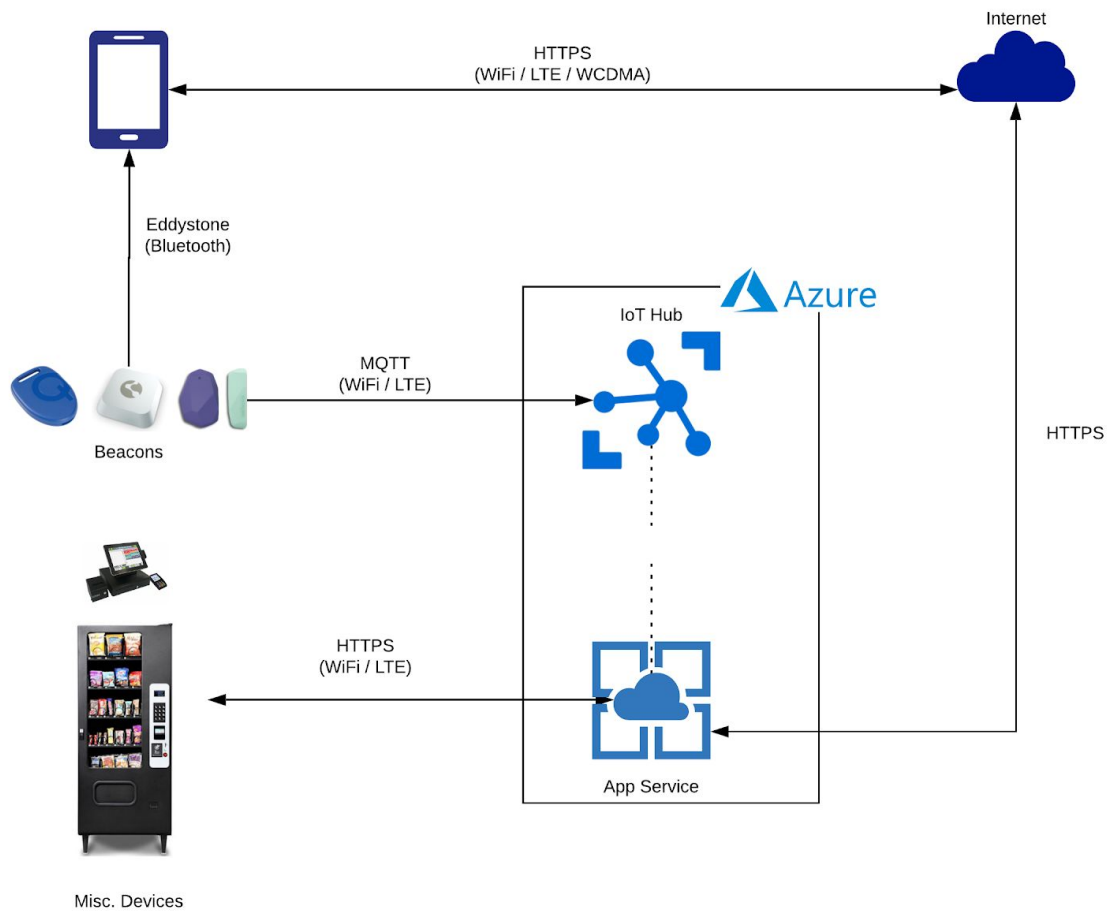
## **Solution**

### **Software Design (Sharon)**

[Create a diagram to show the major software components and how they process and store the data]

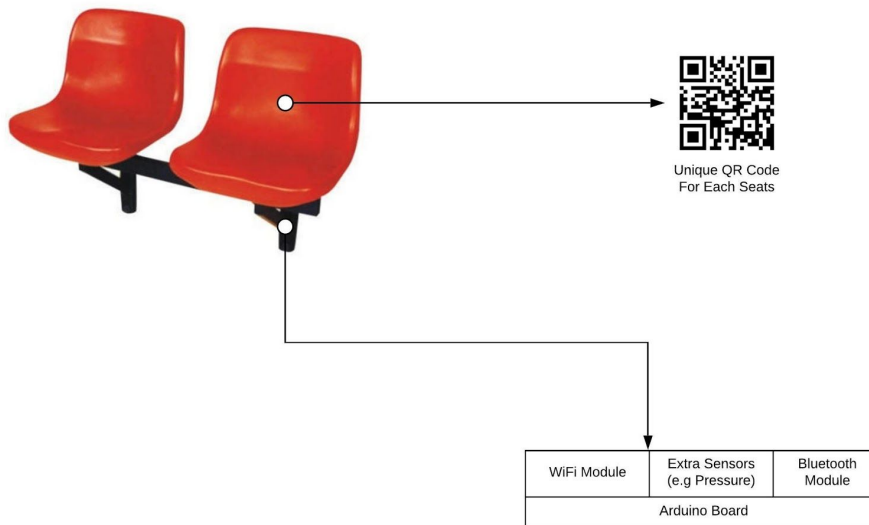


## Deployment Design Diagram



## Hardware Design

(Not Production Design)



## User Acceptance Testing

[Create some test plans based on the functional requirements which validate that when the IoT Solution accepts inputs, that the IoT Solution produces the expected outputs. Then execute the test scripts and validate the results]

### User Acceptance Testing Checklist

There are many variables that must be considered when performing a user acceptance test (UAT). We are prepared to execute test cases using the UAT checklist and decrease the number of mistakes.

The following checklist's purpose is to ensure that the appropriate steps have been taken to execute a user acceptance test (UAT), to wrap up the process and ensure proper documentation has been completed. Simply choose "Yes, No or N/A" as it applies to the UAT process in the organization.

Criteria	Completed?
<b>Team Awareness / Preparations</b>	
Has the team agreed that the test cases provides comprehensive and effective coverage of all aspects of functionality of the application?	



Has the project team been made aware of its role in providing support for all testing issues?	
Will the users feel comfortable in challenging requirements where required? They should be.	
Are the user acceptance tests based around proving the business requirements?	
Does the team understand the responsibilities and required actions for each category of problem identified during testing?	
<b>Navigation</b>	
Is the navigation system easy for users to use?	
Does the navigation system find the best way for both entry and exit?	
Does the navigation system quick enough for response?	
Does the navigation system show the user's location on the map of the stadium?	
Does the navigation system show the vendor's location when they get close to the user who order the food?	
Does the navigation system work well when it comes to a high visitors flow rate?	
Does the navigation system work well when it comes to an accident (e.g: fire, flood..)?	
Does the navigation system have it own using description?	
Is there a support process in place for users who have difficulties with function?	
Is there a clear process in place for the users to categorise and report issues, and for those issues to be fed back to the appropriate are for quick resolution?	
<b>Website</b>	
Is the website easy for users to use?	
Does the website interface show up after scan the QR code on the ticket or the seat?	
Does the payment gateway which provided for the user to pay online work well and ensure the pay safety?	
Does the website show the rough waiting time for the food delivery and can track the order?	
Does the website show the order status?	
Does the website work well when it comes to a high visitors flow rate?	
Does the website quick enough for response?	
Does the website have a maintain plan?	
Does the website has it own using description?	
Is there a support process in place for users who have difficulties with function?	
Is there a clear process in place for the users to categorise and report issues, and for	

those issues to be fed back to the appropriate are for quick resolution?	
<b>Smart Seat with QR-code</b>	
Is the Smart Seat with QR-code easy for users to use?	
Does QR code be unique to each user?	
Does Smart Seat show the location of the seat?	
Does the Smart Seat with QR-code work well when it comes to an accident (e.g: fire, flood..)?	
Does the Smart Seat with QR-code quick enough for response?	
Does the Smart Seat with QR-code have a maintain plan?	
Does the Smart Seat with QR-code has it own using description?	
Is there a support process in place for users who have difficulties with function?	
Is there a clear process in place for the users to categorise and report issues, and for those issues to be fed back to the appropriate are for quick resolution?	
<b>Test Execution and Evaluation</b>	
Were all steps of the test run documented?	
Did the users review the test results?	
Are the services provided by the system in compliance with user requirements?	
Were all identified defects and issues resolved?	
Are the users going to show other people in the business what they have seen?	