RAMAIAH UNIVERSITY OF APPLIED SCIENCES

ASSIGNMENT

Course Code	20CSC404A
Course Name	Internet of Things
Programme	B.Tech
Department	CSE
Faculty	FET

Name of the Student	K Srikanth
Reg. No	17ETCS002124
Semester/Year	7th Semester / 4th Year
Course Leader/s	Dr. Rinki Sharma

Declaration Sheet			
Student Name	K Srikanth		
Reg. No	17ETCS002124		
Programme	B.Tech	Semester/Year	7^{th} / 4^{th} Year
Course Code	20CSC404A		
Course Title	Internet of Things		
Course Date		to	
Course Leader	Dr. Rinki Sharma		

Declaration

The assignment submitted herewith is a result of my own investigations and that I have conformed to the guidelines against plagiarism as laid out in the Student Handbook. All sections of the text and results, which have been obtained from other sources, are fully referenced. I understand that cheating and plagiarism constitute a breach of University regulations and will be dealt with accordingly.

Signature of the Student	K Srikanth	Date	29/11/2021
Submission date stamp (by Examination & Assessment Section)			

Signature of the Course Leader and date	Signature of the Reviewer and date

Faculty of Engineering & Technology			
Ramaiah University of Applied Sciences			
Department	Computer Science and Engineering	**Programme**	B. Tech.
Semester	7^{th}		
Course Code	20CSC404A	**Course Title**	Internet of Things
Course Leader	Dr. Rinki Sharma		

Assignment					
Register No.		Name of Student			
Sections		**Marking Scheme**	**Max Marks**	**First Examiner Marks**	**Second Examiner Marks**
Question-1	1.1	Design details for the given application	08		
	1.2	Specification diagrams and explanation	15		
	1.3	Conclusion	02		
		Question 1 Max Marks	**25**		
		Total Assignment Marks	**25**		

Question 1

Q-1.1

Given Problem Statement

Consider a home automation IoT application for manual as well as remote control of the Door Knob to control permission to the visitors based on the video obtained from the camera at the entrance of the house.

Introduction

The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. A thing in the internet of things can be a person with a heart monitor implant, a farm animal with a biochip transponder, an automobile that has built-in sensors to alert the driver when tire pressure is low or any other natural or man-made object that can be assigned an Internet Protocol (IP) address and is able to transfer data over a network. Increasingly, organizations in a variety of industries are using IoT to operate more efficiently, better understand customers to deliver enhanced customer service, improve decision-making and increase the value of the business.

Now, let's start listing down few functional requirements for our product,

Functional Requirements

Table 1.1: Functional Requirement 1

Requirement Tag	FR1
Requirement Description	The system should have be able to communicate with the cloud server.
Dependent on Requirements	-
User/System interacting with the requirement	System

Table 1.2: Functional Requirement 2

Requirement Tag	FR2
Requirement Description	The system should have be able to unlock the door when user is detected.
Dependent on Requirements	-
User/System interacting with the requirement	System

Table 1.3: Functional Requirement 3

Requirement Tag	FR3
Requirement Description	The system should have an interface where user can upload the faces of who can access the door
Dependent on Requirements	-
User/System interacting with the requirement	User

Table 1.4: Functional Requirement 4

Requirement Tag	FR4
Requirement Description	The system can decide if the door has to open automatically or manually based on settings of the user.
Dependent on Requirements	-
User/System interacting with the requirement	System

Now that we have basic functional requirements we can go ahead and design a high level architecture to see how our system works overall.

Architecture

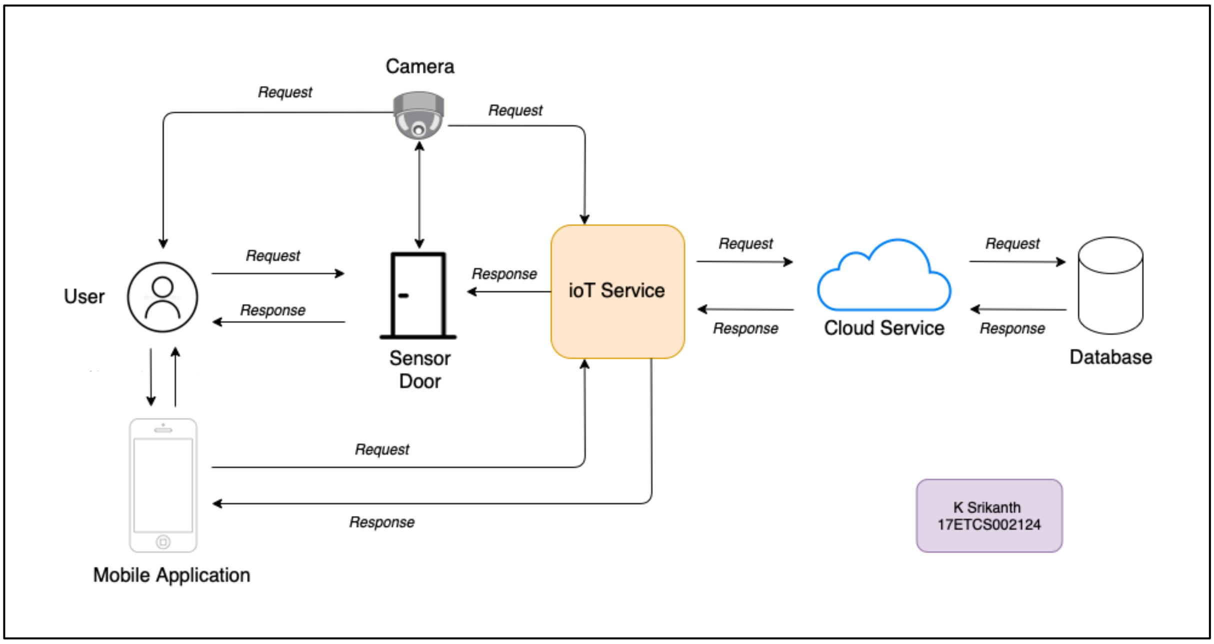

Figure 1 High-level Architecture Design of the given problem statement

Now that our architecture is finished. Let me tell you step by step to check how it works.

Walkthrough

1. The user comes home and waits for the door to open so here the door is waiting for someone to make the request and then sensor door and camera communicate using any of the ioT protocols such as MQTT, Zigbee Bluetooth. Then the camera gets activate and requests user to show his/her face to get it mapped to open the door.
2. After collecting the data from the camera, it sends a request to ioT Service asking to confirm the face with the face data in the database and waits for the response. The ioT service can keep a cache of the face data for a quick response.
3. After a request that is been received to the ioT service from the camera then the request is made to a cloud service to get the data where the iOT module can be authenticated using **unique identifiers (UIDs)** from the device and if it has the access then it cloud confirm the data if it match's with one in the database. If yes, it sends a success response else it sends a failed response.

4. That response is carried to ioT Service and then if the response is yes, then the ioT service lets the sensor door to open else it stays closed.
5. Now that we know how the data is been sent and received between the ioT devices the use of mobile application is to add the user family members faces and interact / customize the ioT devices in the home. If user wishes to add the new face the request is made to ioT Service using Wi-Fi in the home and then added in the database for future requirements. If yes, it sends a success response else it sends a failed response saying not been added.

Q-1.2

Okay, so we have our model of how our devices works now let's go deep to see w.r.t to theory of how it works.

Q-1.2a

A process specification is a method used to document, analyze and explain the decision-making logic and formulas used to create output data from process input data. Its objective is to flow down and specify regulatory/engineering requirements and procedures. High-quality, consistent data requires clear and complete process specifications.

Design

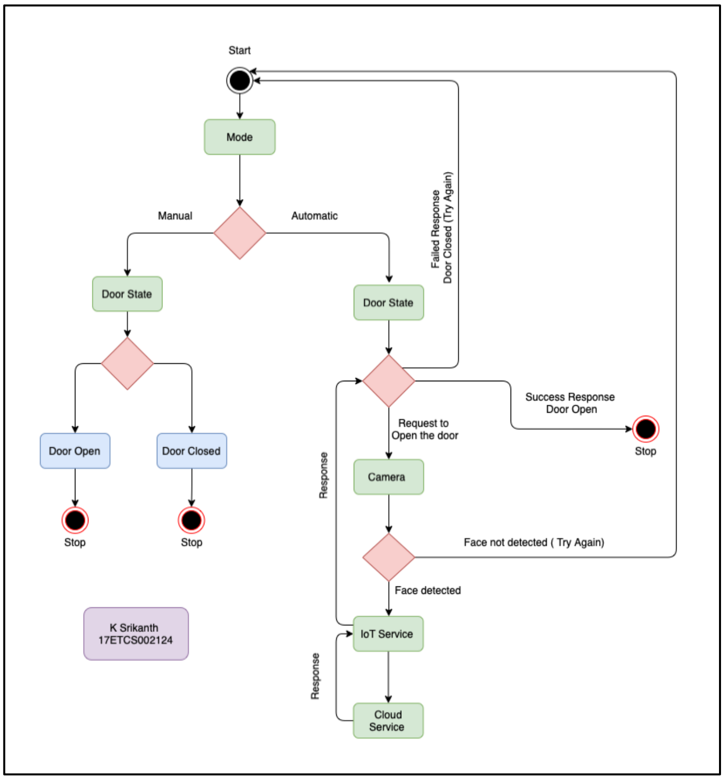

Figure 2

Process Specification Model for the given problem statement

Explanation

As we can see in the above diagram, we start with the start state then followed by the mode state where we have seen if our input is automatic or manual (*this input can be set using the mobile application) based on the input,

1. **Automatic Input**

   With our input being automatic,

   a. We move to door state where the door is closed and then we make an input to the door and that makes an input to camera.
   b. If face is detected then it sends an input to ioT Service.
   c. Then ioT Service sends an input to Cloud Service and the output of the Cloud Service is sent as the input for ioT Service
   d. Which then sends an input to door to which decides to open the door or let it be in the same state if the sends a success response, then it opens the door else if the input received is failed response, then it lets the user to try again

2. **Manual Input**
   a. If the door is open then end the state
   b. If door is closed then open the door and end the state

<u>Q-1.2b</u>

Information Model defines the structure of all the information in the IoT system, for example, attributes of Virtual Entities, relations, etc. Information model does not describe the specifics of how the information is represented or stored. To define the information model, we first list the Virtual Entities defined in the Domain Model. Information model adds more details to the Virtual Entities by defining their attributes and relations.

Design

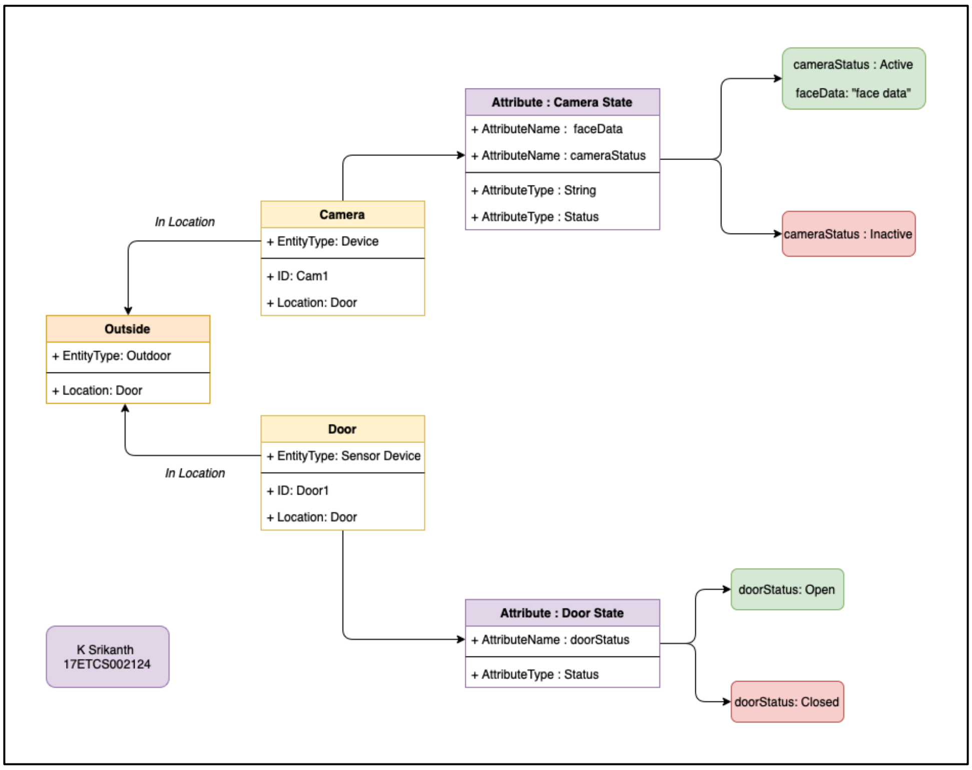

Figure 3 Information Model for the given problem statement

Explanation

As we can see in the above diagram, we can see that we have divided all out components/ modules into classes with their attribute now our super class i.e., **Outside** contains an **entityType and location** to map it with their subclasses. The subclasses here would be **Camera and Door.** In **Camera class** we have an attribute such as **entityType, ID and location** Where ID is used to verify if the device is authorized to make a request to the service and get the data or not and same case for the **Door Class.** Now coming to the Camera State attribute class, we can see that it has 2 attributes which are **facedata** and **cameraStatus** where facedata contains the data that was acquired from the camera and cameraStatus lets the user know if it’s available or not. After acquiring the data from the camera, it is further sent to the service for further processing of the data. Now coming to the Door State attribute class, we can see that it has only one attribute which is **doorStatus** which tells us if the door is opened or closed w.r.t to the given state.

Q-1.2c

Service specifications define the services in the IoT system, service types, service inputs/output, service endpoints, service schedules, service preconditions and service effects.

Design

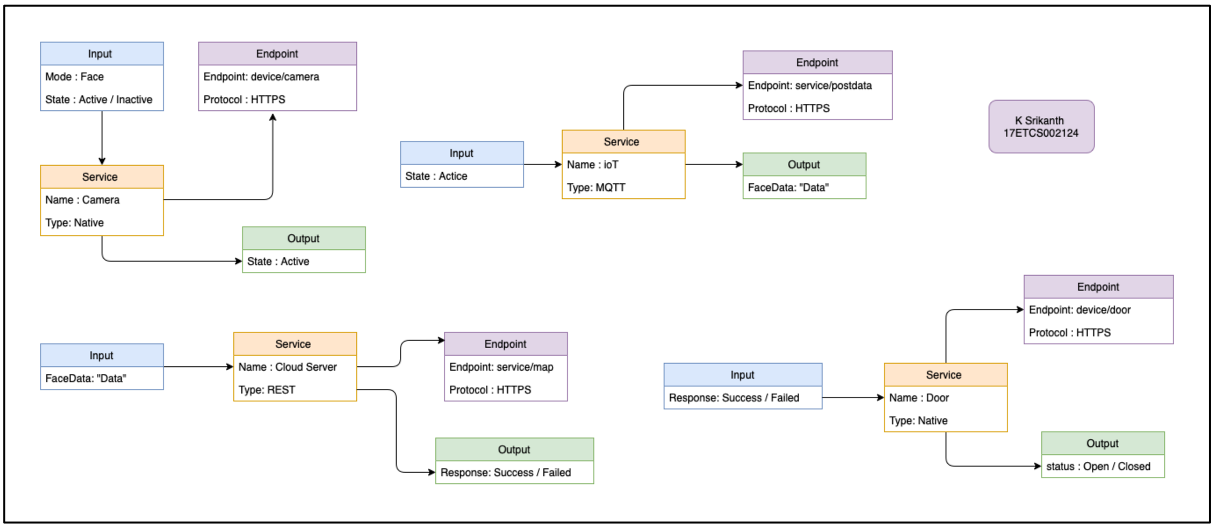

Figure 4 Service Specification Model for the given problem statement

Explanation

As we can see in the above diagram, here I'm going to explain it w.r.t to service,

1. **Camera**

   <u>Input</u>

   Here as we see from the diagram (4) the input is the face if the face is detected while the camera is active then the request is proceeded further.

   <u>Output</u>

   Here as we see from the diagram (4) the output is to check if the device is active or not.

   <u>Endpoints</u>

   device/camera: This endpoint will return us the response if the device is active or inactive according to the response, we continue the process further

2. **ioT Service**

   Input

   Here as we see from the diagram (4) the input is the **status** that was acquired from the camera and if the status is active, we get the data from the camera using MQTT protocol.

   Output

   Here as we see from the diagram (4) the output is the data that is received from the camera device.

   Endpoints

   service/postdata: This endpoint will send the data to the cloud server for further processing of the data that we got from the camera.

3. **Cloud Server**

   Input

   Here as we see from the diagram (4) the input is the **facedata** that was acquired from the ioT Service and we verify if the data is mapping in the **database** with our input data.

   Output

   Here as we see from the diagram (4) the output is response on the data being verified from the input in the cloud database if data is mapped then it returns **Success** else **Failed**

   Endpoints

   service/map: This endpoint will return us the response if the if data mapped is **Success or Failed** according to the response, we continue the process further

4. **Door**

   Input

   Here as we see from the diagram (4) the input is the response that we got from the cloud server i.e., **Success or Failed.**

   Output

   Here as we see from the diagram (4) the output if the input is success, then open the door else stay in the close state.

   Endpoints

   device/door: This endpoint will return us the response if the device is open or closed according to the response, we continue the process further.

Q-1.2d

Service specifications define the services in the IoT system, service types, service inputs/output, service endpoints, service schedules, service preconditions and service effects. This step in the IoT design methodology is to define the IoT level for the system.

Design

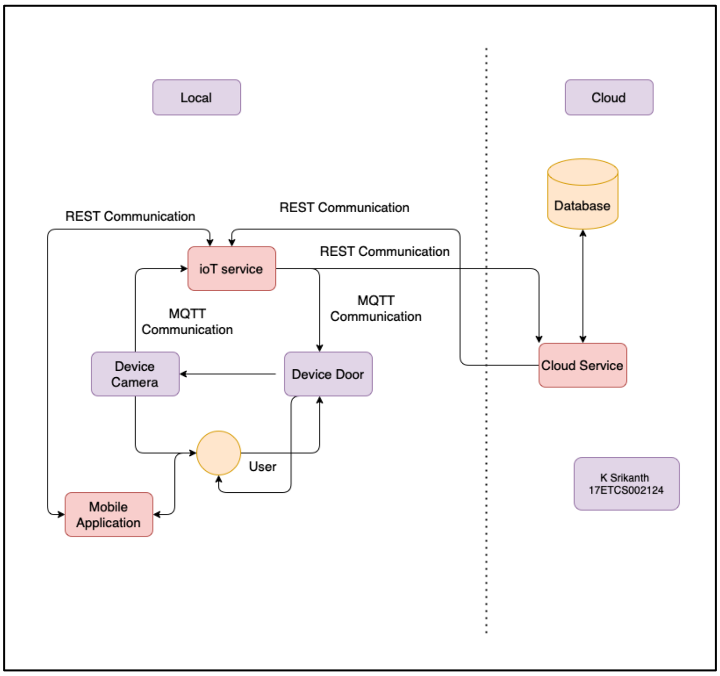

Figure 5 IoT Level Specification Model for the given problem statement

Explanation

As we can see in the above diagram, we get a picture about processing in local and cloud. I'm just going to explain it w.r.t to **local and cloud,**

Local

Here as we see from the diagram (5) the services and components that are present are

1. **User:** The person who is using this service.
2. **Device Camera:** The ioT device that captures the face and then sends it to the ioT Service using MQTT Protocol.
3. **Device Door:** The ioT device that interacts with user then sends a request to camera to get the data of the person and waits for the response from the ioT Service using MQTT Protocol.

4. **ioT Service:** The Service which uses MQTT to communicate with the devices in the environment i.e., Camera and Door. And after the data being received, it is then sent to cloud for further processing.
5. **Mobile Application:** To monitor all the services in the local environment and update or add or delete the members of the family this uses REST Commination.

Cloud

Here as we see from the diagram (5) the services that are present are,

1. **Cloud Service:** The REST Commination Service which communicates with the **ioT Service** for requested data processing in the cloud i.e., for the face data mapping.
2. **Database:** The Place to store all the faces of particular user and perform queries like
    a. **Insert:** To add a new face.
    b. **Update:** To update the data of an existing face.
    c. **Delete:** To delete the data of an existing face.
    d. **Get:** To get the data of an existing face.

Q-1.2e

The Functional View (FV) defines the functions of the IoT systems grouped into various Functional Groups (FGs). Each Functional Group either provides functionalities for interacting with instances of concepts defined in the Domain Model or provides information related to these concepts.

Design

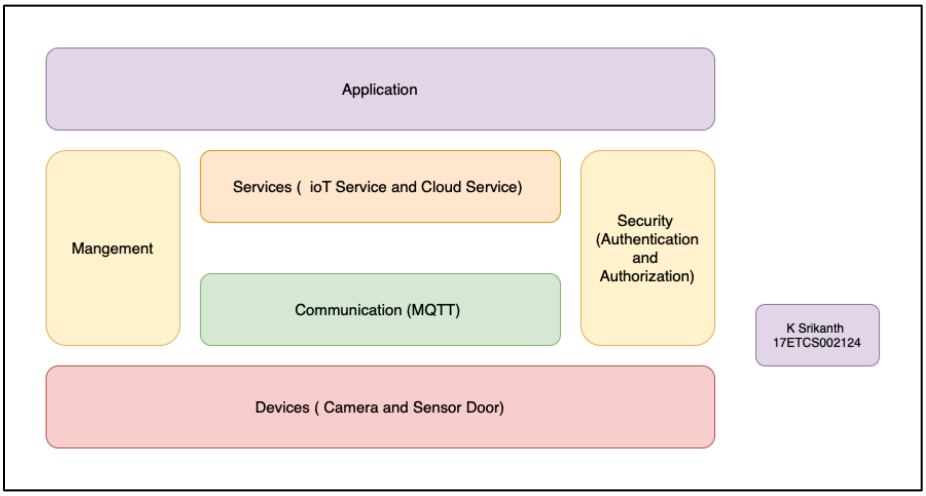

Figure 6 Functional View Specification Model for the given problem statement

Explanation

As we can see in the above diagram, this is a stranded ioT Architecture Diagram. Now let's look at what are services and components are going to communicate with the component and modules in the ioT Architecture.

1. **Device Layer**

In this layer we have components like **camera and sensor door** which are our core components for our architecture to make the things work and collect data from the environment.

2. **Communication Layer**

In this layer we have protocols like **MQTT, Zigbee etc.** which are our core way to communicate with the components and collect the data in the device layer from our architecture.

3. **Services Layer**

In this layer we have services like **MQTT Broker, IoT Service and Cloud Service** with connects and ties knot with other layers for example **MQTT Broker** is used to talk with the MQTT Protocol devices and helps connect to **internet using REST API's**

4. **Security Layer**

In this layer we check if the device is an authorized device that made the request using ioT Service to Cloud Service to get the data for further processing using unique identifiers (UIDs) in the device.

5. **Management Layer**

In this layer we check if the devices are active or inactive using the help of MQTT Broker, IoT Service and manage the sessions for the user requests.

6. **Application Layer**

In this layer we have a User Application to control the devices i.e., Camera and Door in his house using an mobile application which communicates with the **IoT Service.**

Q-1.3

Conclusion

With increase in IOT and cloud technologies, it is necessary for us to adapt to these new things and make use of the advantages that is provided by these new technologies. Here we used these services in-order to automate the security system by taking the image of the person at the arrival and then sending the image to the IOT service which then communicates with our backend service (Image processing) in the cloud and returns us the authenticity of the person. Below are the some of the advantages and disadvantages of the above used approach,

Advantages:

1) Accuracy
2) Speed
3) Secure
4) Scalability
5) Portability

Disadvantages:

1) Needs 24*7 internet connectivity
2) Single point of failure may cause the whole system to shut down
3) Response time can be bit more sometimes leading to bad user experience
4) Can be further improved by using Caching at IoT service
5) High prone to cyber attacks