Inha University in Tashkent

INHA UNIVERSITY TASHKENT

DEPARTMENT OF CSE & ICE

Internet of Things

Group: “InSight”

Project title: “*Smart Shop*”

Team members:

Yuliya Durova U1710104 (002)

Sevara Abdullaeva U1710059 (001)

Mukhammad Valiev U1710172 (004)

Madina Akhmadjanova U1710182 (002)

Javlonkhuja Eshonkhodjaev U1710206 (003)

Bekhzod Aliev U1710270 (004)

Spring 2021

Abstract

As the world today has shifted, we are faced with less human interactions. Thus, we decided to apply this approach to our product purchasing process. We understand that online shopping already exists, but still for some people it is inconvenient. People like to take the products, to hold them and feel by their hands in order to choose better one. For the product quality control, we introduced temperature and humidity sensors. With their help, the products will be kept safe for entire shelf life and will not lose their appearance.

What we propose for them is cashier-free shop. Due to the development of IoT, we can now implement this idea by using various network protocols, sensors, devices and actuators. In addition to this, cloud computing is becoming more and more popular and its power and convenience are much higher than before.

We are going to track the products taken with camera and weight sensors, as well as validate them with RFID tags attached to every product. The process of payment will also become much easier as we introduce a personalized account for our customers, which are connected to the credit card. Thus, whenever the customer exits our shop through RFID reader and the product list is validated, the transfer of money will occur.

We are looking forward to introduce better technologies in our smart shop to provide full security and comfort both for customers and enterprises.

Table of Contents

1. Introduction 4

1.1 Problem 4

1.2 Goal setting for intelligent digital transformation 4

1.3 Team roles 4

2. Design end-to-end IoT network architecture for your application 5

2.1 Specification of conceptual architectural functions (SPNDSe) 5

2.2 Three tier architecture diagram for our application 6

2.3 High level design for end to end IoT network architecture based on ITU-T Y2060 ARM 7

2.4 Specification of oneM2M infrastructure and field domain architecture entities that can be deployed on the network elements in a distributed fashion 7

2.5 Potential security threats from end to end architecture perspective and methodologies to protect from the threats for each functional architecture component 8

3. Choose the optimal platforms for IoT devices and connectivity for your IoT service application 10

3.1 Important functions of CSE(Common Service Entity) 10

3.2 The distribution of CSE over oneM2M architecture entities 12

3.3 Optimal connectivity of IoT service application 12

3.4 Table of Requirements 13

4. Design open service platform enabling creation of mash-up IoT services 14

4.1 Sensors/Actuators, IoT devices, OSHW, device software(OS), and their roles 14

4.2 Connectivity of devices 15

4.3 Open Service Platform Architecture 16

4.4 IoT protocol stack of Network Layer 16

4.5 IoT protocol stack of Application Layer 17

4.6 Implement an end to end IoT service scenario based on solid results of all homework assignments 19

5. Consumer Value-oriented IoT Service 20

5.1 Our high level design for business model of the consumer value oriented IoT service 20

5.1.1 Our business model 20

5.1.2 Graceful maturity progress business model for intelligent (AI integrated) IoT service 20

5.2 Design a service model for our application 21

5.3 Overcome the chasm of IoT market 23

6. Conclusion 23

1. Introduction

1.1 Problem

Define [X] and [Y] to set up objectives for creating new value [Y] (consumer value-oriented IoT service).

1.2 Goal setting for intelligent digital transformation

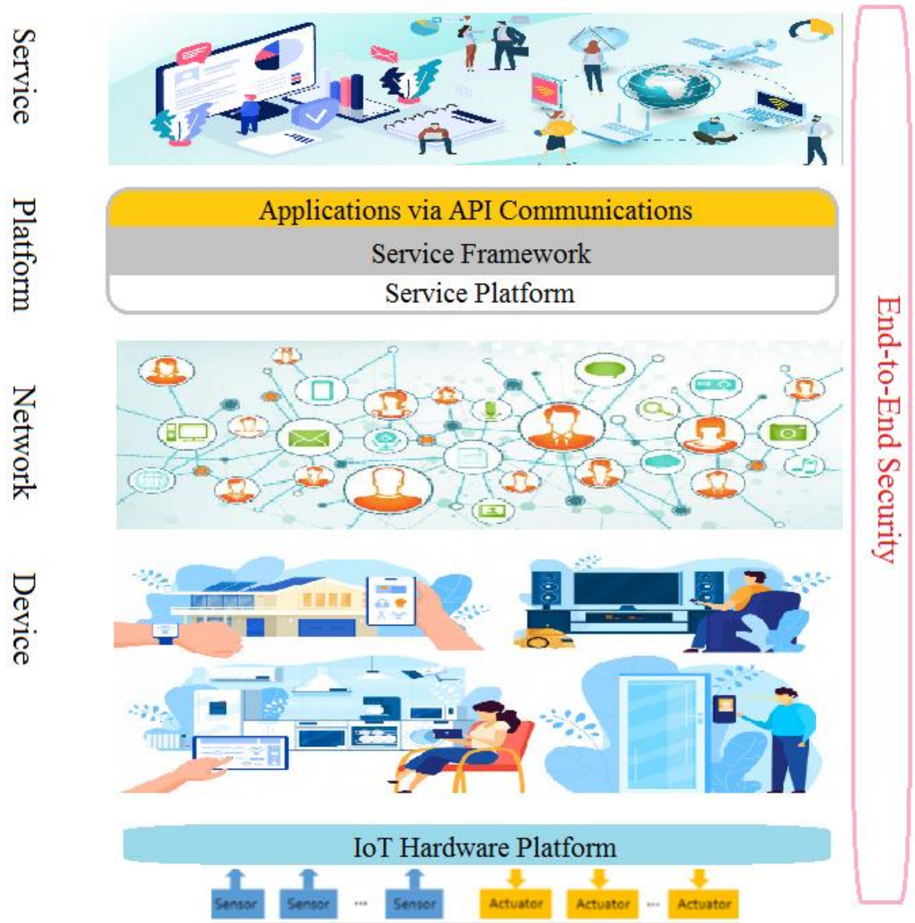
The main goal of today's technologies is to make people's life simpler and more convenient. Connectivity between various devices creates new opportunities for science and businesses. After COVID 19 pandemic the world will not be the same. People now understand the importance of social distancing and how technologies may help. Our team proposes a smart shop system where customers do not need to worry about queues, interactions with cashiers, assistants. We would like to create a shop with IoT system where Artificial Intelligence controls all interactions between customers and products excluding almost all health risks. The implementation includes cameras, sensors, smart carts and assistants.

1.3 Team roles

Full Name	ID (Section)	Title	Responsibility
Yuliya Durova	U1710104 (002)	Team Leader, Researcher	Form a team, Propose the problem. Identify potential solutions. Divide tasks between developers. Task checking after completion.
Sevara Abdullaeva	U1710059 (001)	Main Developer, Researcher	Make research on given problem, find different ways to solve them. Product development.
Mukhammad Valiev	U1710172 (004)	Developer	Product development, developing various solutions given by PM.
Madina Akhmadjanova	U1710182 (002)	Designer	Style creation, Develop UI and UX.
Javlonkhuja Eshonkhodjaev	U1710206 (003)	Full Stack Developer	Front-end, Back-end development. Developing solutions given by PM.
Bekhzod Aliev	U1710270 (004)	Project manager, Researcher	Making research on given problem, identify potential solutions. Cheer up teammates.

2. Design end-to-end IoT network architecture for your application

2.1 Specification of conceptual architectural functions (SPNDSe)



Devices:

- RFID Tags – stickered to every products to track its state
- RFID Reader –located on the shop exit to validate the products bought
- Sensors – located on shelves, provide data of product's status.
- Camera – located around shop for product and consumers tracking.

Connection Network:

- devices and sensors are interconnected through network (Wi-Fi, 4G ,5G) based on oneM2M standard.

Data Processing Platform:

- Sensors gather the data, send it to cloud
- Artificial Intelligence analyzes received information.
- Shop security status analysis.

- Edge Cloud
- Big Data Analytics

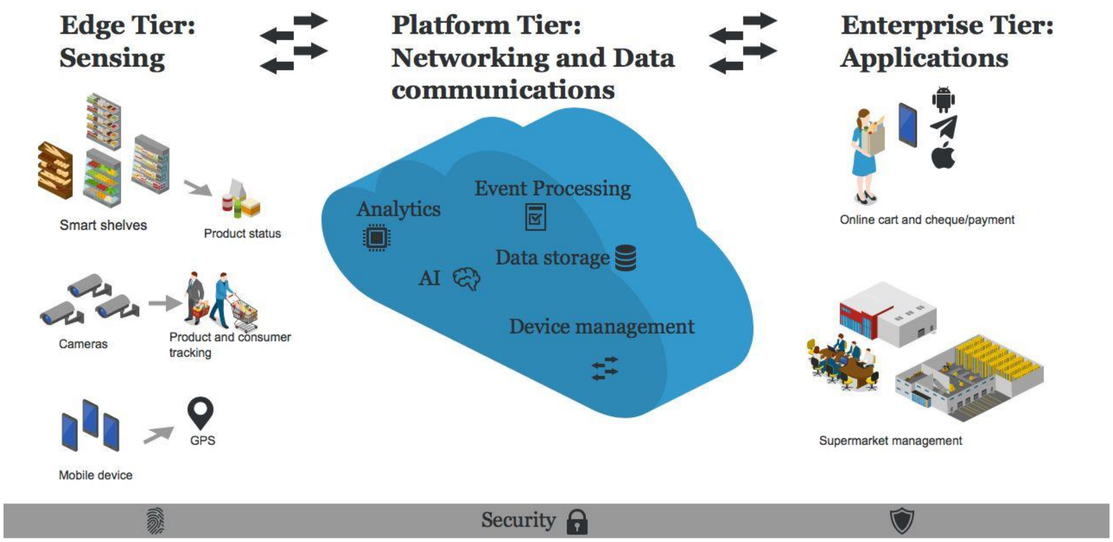
Service Framework:

- Systems know what products are taken from shells by user and users get the list of their purchases on their phone
- After leaving the shop they charged for their purchases.
- Targeting personalized sales, discounts and offers.

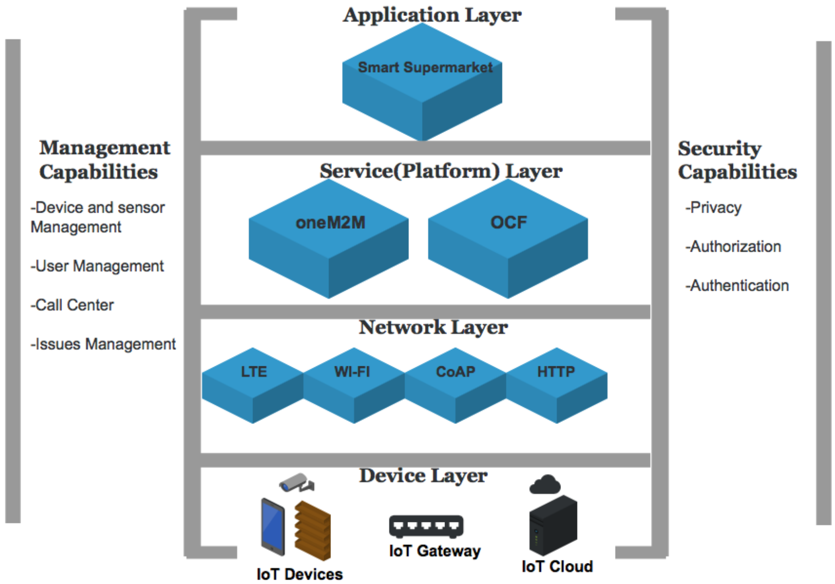
Security:

- Context-Aware Intelligence.
- oneM2M standards.

2.2 Three tier architecture diagram for our application



2.3 High level design for end to end IoT network architecture based on ITU-T Y2060 ARM



2.4 Specification of oneM2M infrastructure and field domain architecture entities that can be deployed on the network elements in a distributed fashion

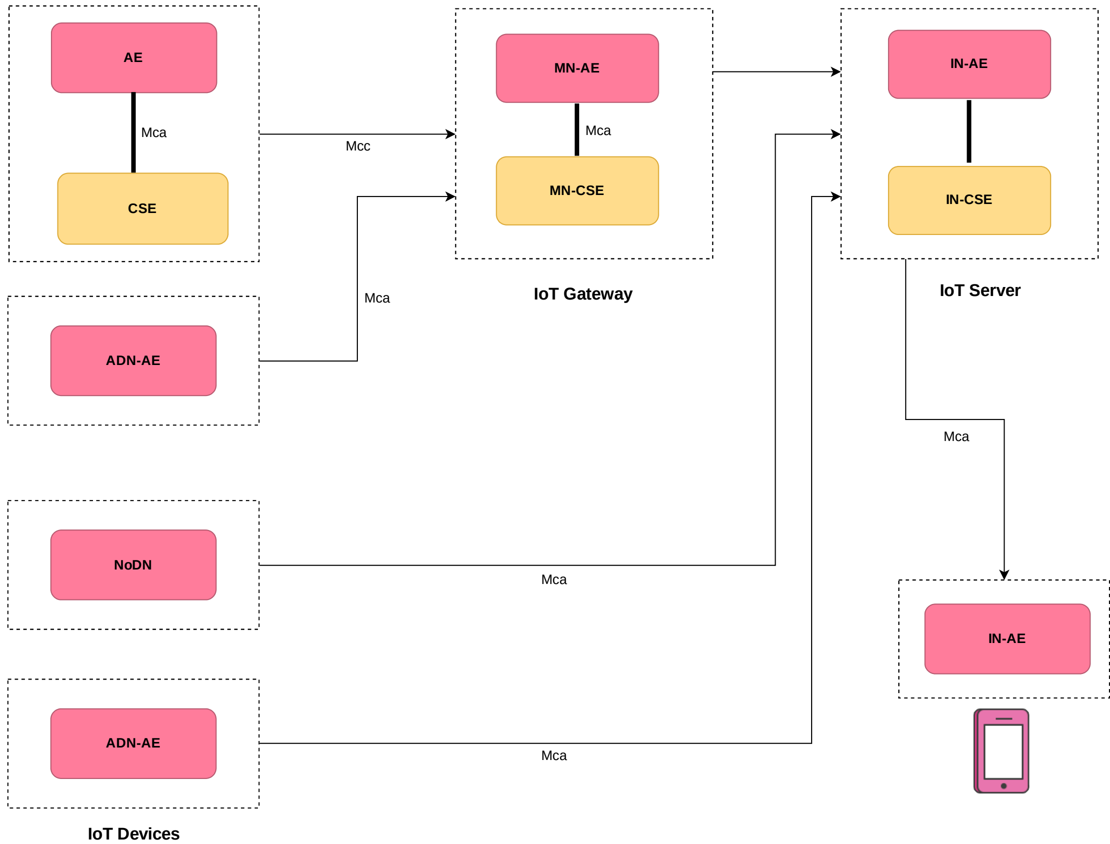
According to oneM2M infrastructure, the sensors, smartphones, cameras host on AE (Application Entity). The CSE (Common Service Entity) provide "service functions". Moreover, it applies for distributed intelligence (device, gateway, cloud apps).

Infrastructure includes IN-CSE (short for Infrastructure Node CSE) is hosted in the cloud. and a. The reference point used between the gateway MN-CSE and IN-CSE is Mcc.

Field domain consists of AND (Application Dedicated Note) is implemented on a rather resource constraint device that may not have access to rich storage or processing resources and – therefore – may be limited to only host a oneM2M AE and not a CSE, Application Service Node (ASN) is implemented on a range of different devices ranging from rather resource constraint devices up to much richer HW, Middle Node(MN) is hosted on the Gateway.

Sensors and cameras are connected to a shop gate which communicates with a oneM2M server. Because our system uses Cloud computing using Artificial Intelligence that work on IN-CSE.

There is simple architecture of our oneM2M view for our system.



2.5 Potential security threats from end to end architecture perspective and methodologies to protect from the threats for each functional architecture component

As today systems become more complex and interconnected, the concern about security at each level of system comes to front ground. Breakage at any level may bring damage to the whole system and users.

Because devices as sensors are limited with processing capabilities and memory, they can be easily compromised. Users also can bring damage through the smartphones, where their personal account may be stolen and used against them and the system.
On the network side, without appropriate identification, privacy and integrity, the chances of break to the system are very high.

Below, we would like to indicate the security threats for each level:

Devices:

- Hacking of low-performance devices
- Device management vulnerability

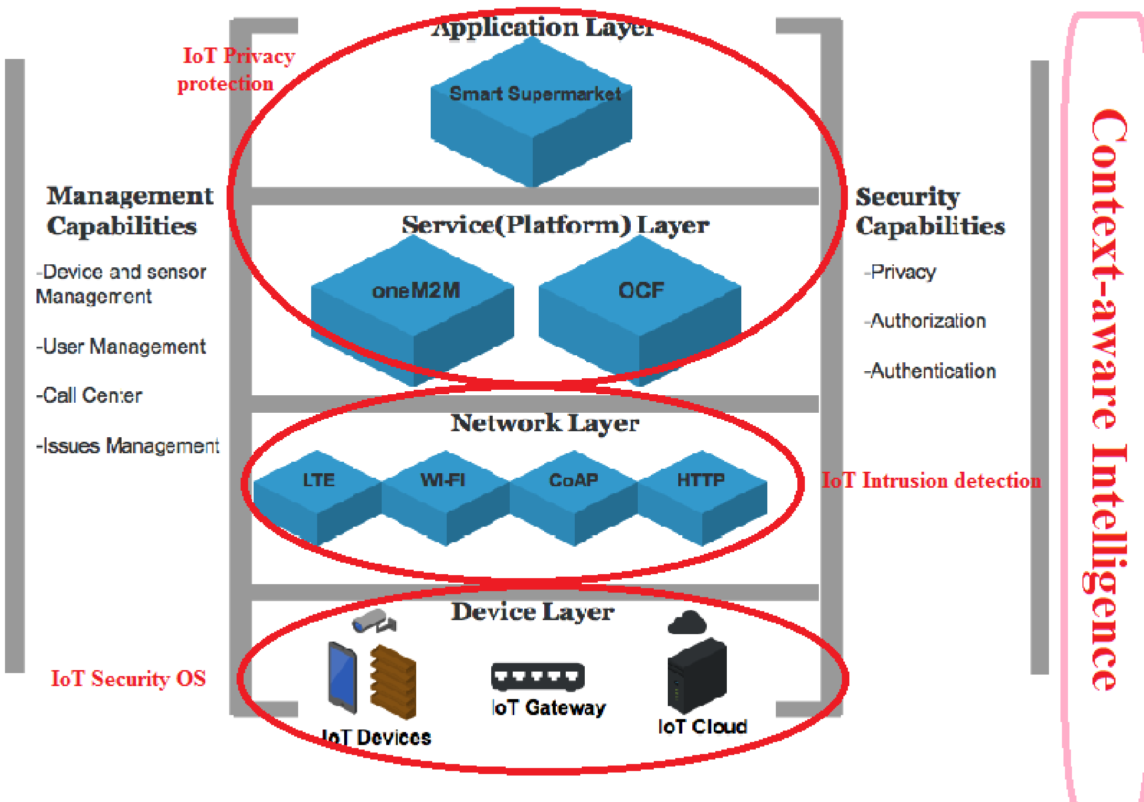
Network:

- Vulnerability of wireless network
- Explosion of network traffic

Service/platform:

- Vulnerability of open service platform
- Personal Information leakage

For better IoT protection, we propose to use IoT Security OS to keep data in devices safe, IoT Intrusion detection technology that will detect physical/behavioral anomalies of devices and networks in real time, IoT Privacy protection technologies that detects and removes sensitive information exposure risk by analyzing unstructured big data for Platform/Service. All these measure may be combined with Artificial Intelligence and Big Data creating Context-aware Intelligence that will detect anomalies and propose measures to solve the problems.



3. Choose the optimal platforms for IoT devices and connectivity for your IoT service application

3.1 Important functions of CSE(Common Service Entity)

Common Service Entity is a core of IoT platform that is responsible for registration, discovery, security, communication management, etc. functions. These functions are exposed to other entities through Mca (exposure to AEs) and Mcc(exposure to other CSEs) points.
For our IoT platform the main functions CSE should perform are:

- Authentication and authorization – to allow users enter the shop and have access to products.
- Security – to protect user's personal data and the shop's data.
- Location – to track consumer's movements in shop and synchronize this data with cameras and sensors.
- Service Charging & Accounting – After leaving the shop, users are charged for their purchases.
- Connectivity Management – to provide better connectivity between devices and efficient data transport.
- Data Management – to improve the data storage, processing and transport.
- Group Management – to improve communication and data exchange between groups of devices.
- Device Management – to improve performance of devices.

For shop service we defined the following functions:

- Authentication and authorization of customers
- Tracking the sensors' and cameras' data
- Tracking the sensors' conditions
- Analyze products status
- Analyze purchases and provide billing services
- Detect anomalies
- Detect and analyze suspicious behavior of customers
- Provide consulting to customers if needed

For our RESTful API structure sensors, cameras, carts and mobile phones are resources that transmit their current state to the gateway and then to the server where the data updated. The basic idea is that using URI (Uniform Recourse Identifier) we indicate resource for interaction and notification of its current state (i.e. sensor on shelf send signal that the product is taken, camera synchronizes its data to identify the user and sends update to server). Then using such CRUD+ Notify methods as POST, GET, PUT, DELETE we will be able to read, update, delete and create required data for our system and provide service to users.

RESTful API structure:

User Registration: /users/	[POST]
Get user state:/users/{id}	[GET]
Update users state:/users/{id}	[PUT]
Retrieve all customers: /users/	[GET]
Sensor Registration: /sensors/	[POST]
Get sensor state: /sensors/{id}	[GET]
Update sensor's state:/sensors/{id}	[PUT]
Create a new order for customer:/users/{id}/orders/	[POST]
Update user's order:/user/{id}/orders/{id}	[PUT]
Retrieve all orders of customer:/users/{id}/orders/	[GET]
Delete customers' orders:/users/{id}/orders/{id}	[DELETE]
Create a product in order:/orders/{id}/products/	[POST]
Delete product from order:/orders/{id}/products/{id}	[DELETE]

An example of RESTful API:

User registration:

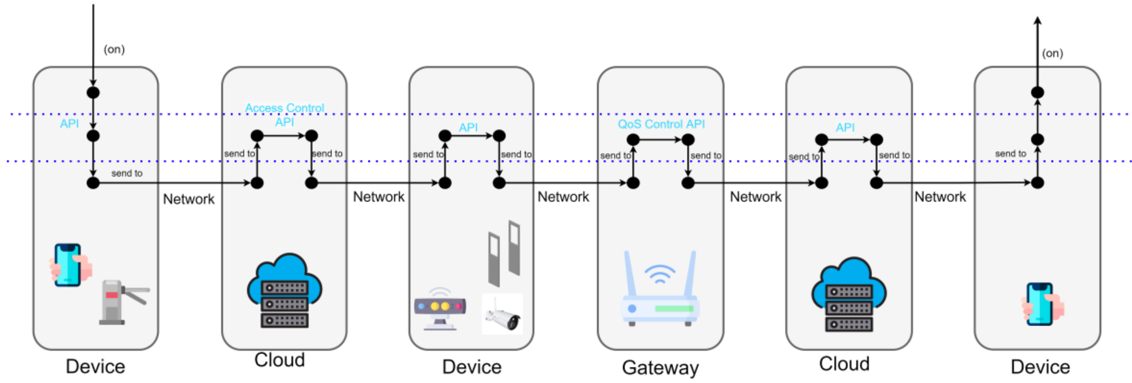
Request:

```
POST https://insight-shop.com/users/ HTTP/1.1
Host: insight-shop.com
Content-Type: application/json; charset=utf-8
Accept: application/json
{
    "Name":"Sevara Abdullaeva",
    "Address": "Chirchik, Tashkent Provinvce",
    "PhoneNum": "+998917769846",
    "CreditCard": "8600900010322181"
}
```

Response:

```
HTTP/1.1 201 Crerated
Content-Type: application/json
{
    "userID":"u1710059"
}
```

3.2 The distribution of CSE over oneM2M architecture entities



Application Layer:

- Works on devices and sensors

Service Layer

- Authentication and authorization
- Security
- Location
- Service Charging & Accounting
- Connectivity Management
- Data Management
- Group Management
- Device Management

Network Layer:

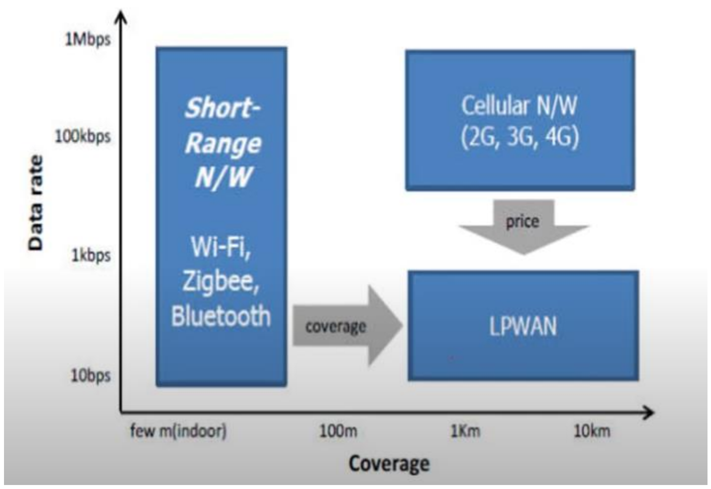
- Cloud performs analysis and returns results on mobile applications.
- Gateway gathers data and send it to server.

3.3 Optimal connectivity of IoT service application

There are several requirements for network performance in our IoT system:

- Range – our customers use our application in different parts of our shop, and additionally we also track our customer's locations, so we need around 100 meters of coverage.
- Data Rate – For connection of UI Devices, IoT Gateway and Cloud we need fast exchange of data. For example: Customers can't decide whether to buy item or not, each of his movements are tracked, sent to cloud, processed and returned back.
- Power – Our system is proposed to work constantly around 10 years, since all alternatives technologies (unlicensed, licensed) are working not less than 10 years.

- Frequency – For our data transfer it is enough to deploy unlicensed frequency range, since it is just a grocery store.
- Security – We need our information to be protected, because we track everything and everyone in our shopping store. We are gathering data about our customers to use it next time they come. And this might be their personal data.
- Smartphone compatibility – Since, it is technological market, each customer needs to have a smartphone with android/iOS/Linux version not earlier than 2013.



3.4 Table of Requirements

Requirements/Data Flow	UI Devices and IoT Gateway	UI Devices and Cloud
Range	Range between them should be <100m inside the shop	Distance between IoT Gateway and Cloud <100m
Data Rate	Maximum Payload is 2Mbps	Maximum Payload is 2Mbps
Power	Up to 5W on main with a battery as an alternative source of power	Up to 5W on main with a battery as an alternative source of power
Frequency	2.4GHz unlicensed for short-range wireless communications (Ours are 3G/4G, Wi-Fi and RFID). Also it resists interference.	Wi-Fi solves the problem of connectivity network. 2.4GHz unlicensed for short-range wireless communications (Ours are Wi-Fi and RFID)

Security	Encryption	Encryption
Smartphone compatibility	Android/iOS/Linux, at least 2013	Android/iOS/Linux, at least 2013
Cost	Cost of technology and its support should not exceed 5-10% of annual profit.	Cost of technology and its support should not exceed 5-10% of annual profit.

4. Design open service platform enabling creation of mash-up IoT services

4.1 Sensors/Actuators, IoT devices, OSHW, device software(OS), and their roles

Our IoT platform will provide shop services where there is no need for cashiers, assistants.
Sensors for “Smart shop”:

- Weight scales sensors: When customers take a product from shelf, sensor will send signal of weight change to the gateway, which will collect data from cameras and send to server, where synchronization and analysis will be processed. Weight Sensor HX711 may be used for this task.
- Thermal sensors: For fruits and vegetables department we offer these sensors for better conservation and temperature controlling. DS18B20 Temperature Sensor may be used.
- Humidity sensor: Some products are better to store with a certain humidity level. DFT22 humidity sensor is an example for use.
- QR Scanner: When customers enter the market, they have to scan QR code from their mobile application in order to get access to products. Scanner registers that user entered the shop and notify server about starting purchases. ESP32-CAM-QR Code Reader may be implemented.
- RFID passive tag: Products will be tagged with RFID for scanning in the entrance in order to perform better security.
- RFID Reader: It will be placed in entrance for scanning products and their status.

Actuator:

- Thermal controlling system: changes the temperature in department if server detected wrong temperature in the area.

IoT devices:

- Mobile phone: provides information from server to customers.
- Cameras: Cameras will collect the movement data of users, their behavior and send it to server.
- Gateway Node: Raspberry Pi to collect data within Wi-Fi connections and send to server for further analysis.

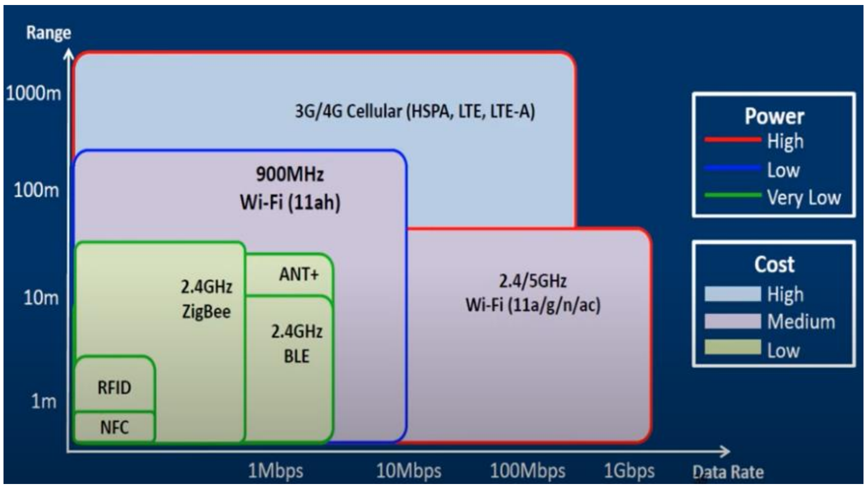
OSHW:

- To send data to gateway ESP32-S may be used. It can be easily connected to sensors and has low cost.
- Raspberry Pi: Interacts with devises and react to notifications about state changes on sensors, starts gathering data for sending to the server.

Devise Software:

- Arduino or Raspberry Pi software for weight scales sensors, thermal and humidity sensors, and QR reader.

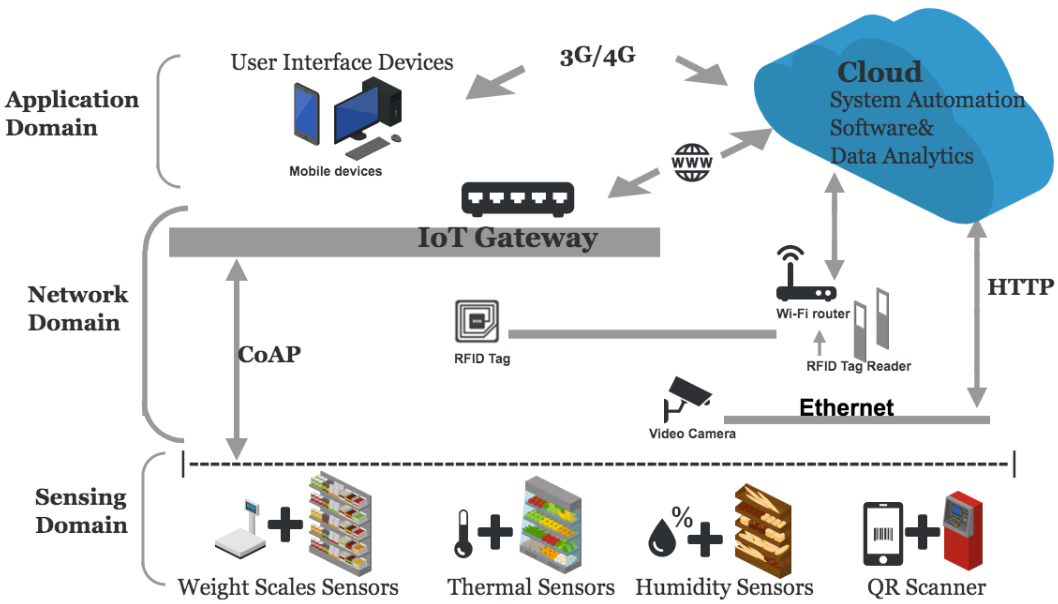
4.2 Connectivity of devices



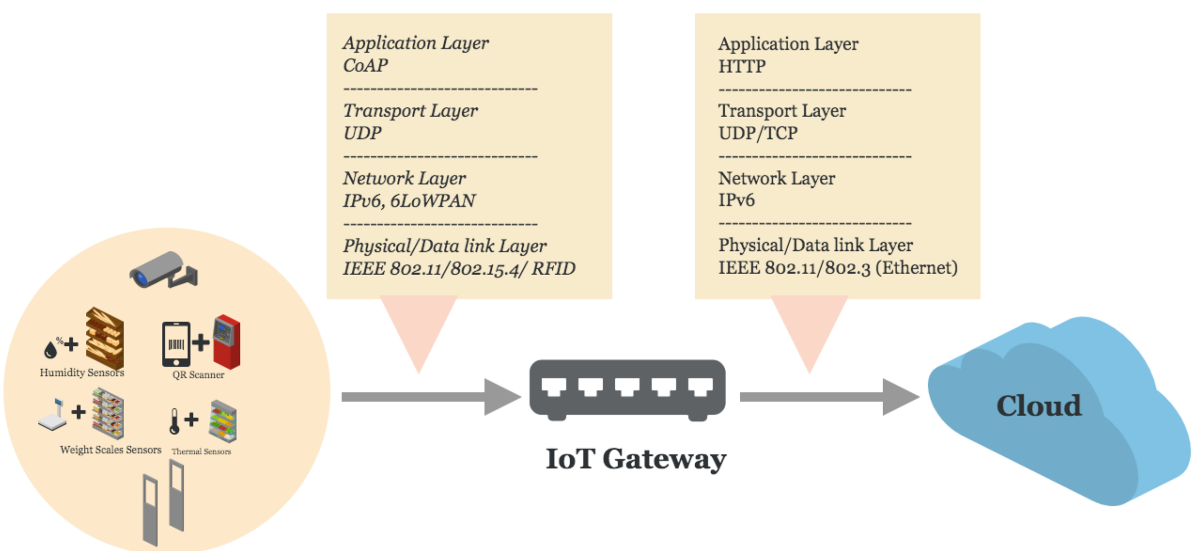
For our grocery shop, the best wireless technologies to connect the IoT devices, cloud and processes would be ZigBee, Wi-Fi and RFID. Since, our main aim is to set up grocery store and develop it to queue-free mode. So, this should be cheap as well. 3G/4G is used by our customers to connect to internet and to be online during the shopping process.

The connection between sensors' points and IoT Gateway Nodes require n-to-one connection light and with not high bandwidth. That is why we use a ZigBee technology to connect sensors to gateway. Also, our grocery items are scanned and monitored by RFID technology. The RFID reader will be connected to Ethernet through Wi-Fi router and send results to cloud directly. Cameras will also send data to cloud through Ethernet connection. Concluding all these functionalities, we decided that it is much better to use Wi-Fi and RFID technology in our store. Of course there are other many technologies which are also suit, but others are for long distances. So, there is no need to set up technology which applied for connections in kilometers and pay much more.

4.3 Open Service Platform Architecture



4.4 IoT protocol stack of Network Layer



In the scope of our project we use to communications:

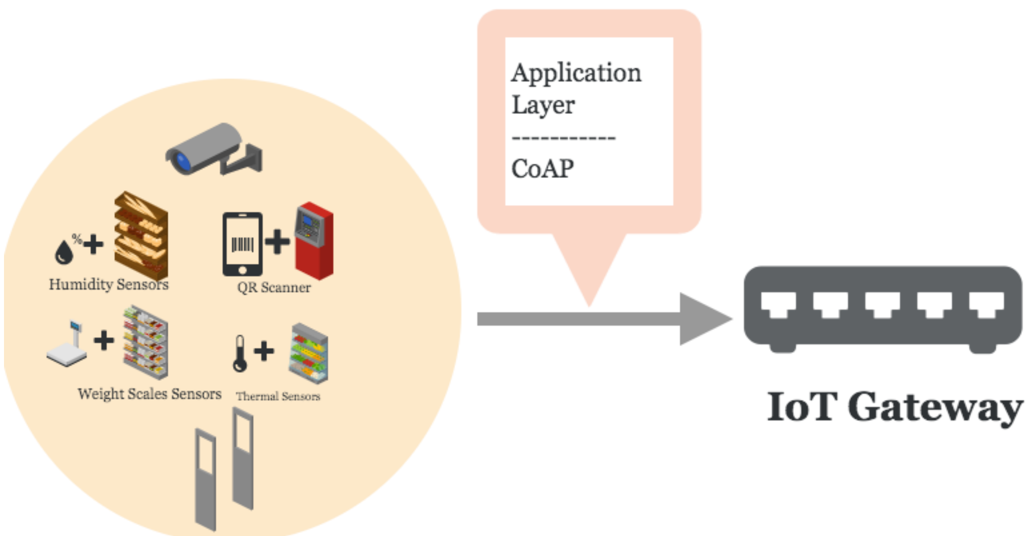
- Between sensors and IoT Gateway.
- Between IoT Gateway and Cloud.

First type of communication has IoT protocol stack as we defined. Our sensors are different in connection type. Let's start with RFID, at the physical/datalink layer we define it itself because all data is captured through reader. Our cameras will be gathered in the middle node and connected to Ethernet. Other sensors can either be connected with the ZigBee, thus we use the down layer protocol of IEEE 802.15.4. At the network layer we are using IPv6 protocol and for the IEEE 802.15.4 integration – adaptation layer with 6LoWPAN, in order to allow our devices to interconnect. As we need low power consumption, we use UDP protocol at a transport layer. Logically, we use only CoAP at application layer as relevant for us and suitable with UDP.

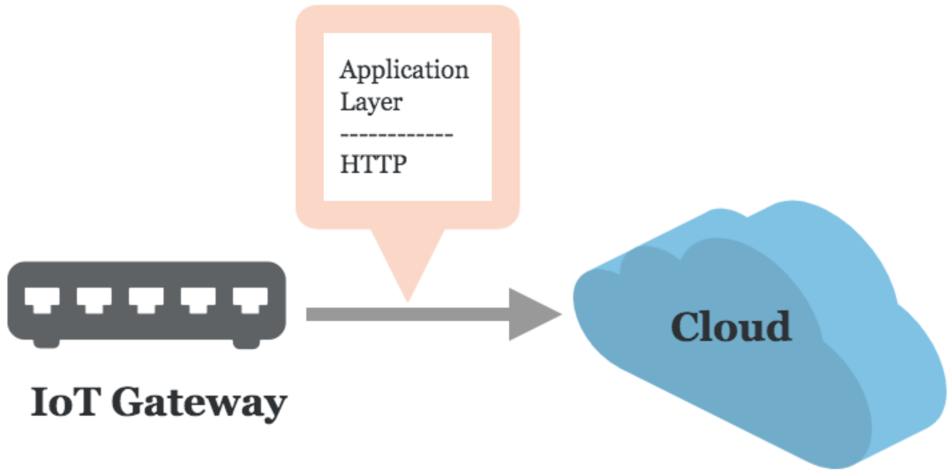
Concerning second, our sensors are scattered across huge supermarket and we can say that we use LAN network. With it we all devices may share same resources. For this purpose, we integrated Ethernet at physical/data link layer and IEEEE 802.11(Wi-Fi). It will run over IPv6 on the network layer. Both UDP and TCP transportation protocols can be used, as our gateway-to-cloud should be more reliable, or in other words connection-oriented- TCP is good. Because it happens on the level of Internet, we use simple HTTP application layer protocol.

4.5 IoT protocol stack of Application Layer

Our solutions for application layer protocols are RESTful: GET, POST, PUT, DELETE.



For the majority of our sensors we decided to use CoAP because it is low cost and secure, platform independent and easy in use. It is based on request/response and works well with resource-constrained devices. It operates over UDP. A sensor device acts as a Server and Gateway as a CoAP Client. CoAP is helpful when it comes to translate and interoperate with HTTP web protocol, exactly what we need. CoAP provides four message types: Confirmable, Non-confirmable, Acknowledgement, Reset. CoAP also has the solution of Observation to the problem when we need to get data periodically, e.g. get temperature and humidity every 5 seconds, and give warning if something goes wrong.



As we have a simple web stack on the side of cloud-to-gateway we will use HTTP/2.0 protocol. It provides less bandwidth consumption due to the binary data format. Also, by implementing parallel transmission based on TCP connection, it increases the speed. One more advantage of HTTP/2.0 is that it provides multiplexing for multiple files only requires one TCP connection, compared to HTTP/1.1 that needs new connection for each new file. Bases on the HTTP/2.0 our clients-meaning the phones of customers, will also interact to provide the check list of the items taken.

Let's first look at the scenario of QR sensor, here customer will either open our app or go on the web and scan the QR. By this, our HTTP-client will send a message that will be converted through CoAP proxy to CoAP and sent to Server (scanner), to match and initiate the connection to virtual cart.

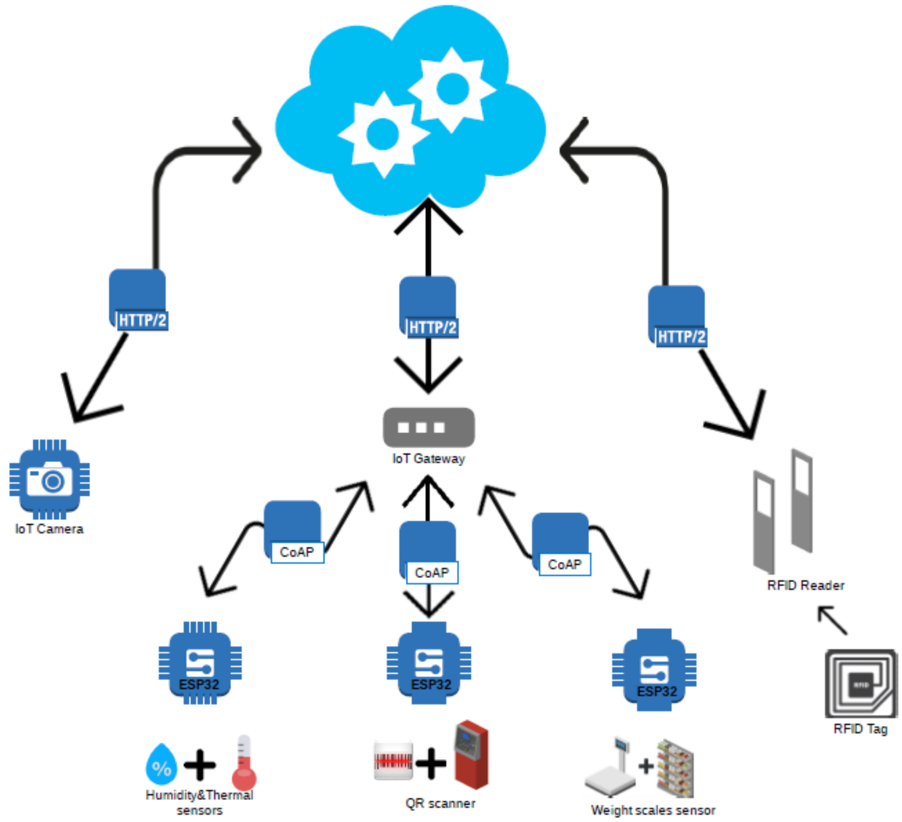
The Humidity and Thermal sensors will get message from the cloud (HTTP-go to proxy) or gateway (CoAP) and report their state.

Our IoT camera will go over HTTP as it is more reliable and frequently used for video streaming, so only for it we will use TCP also at this level.

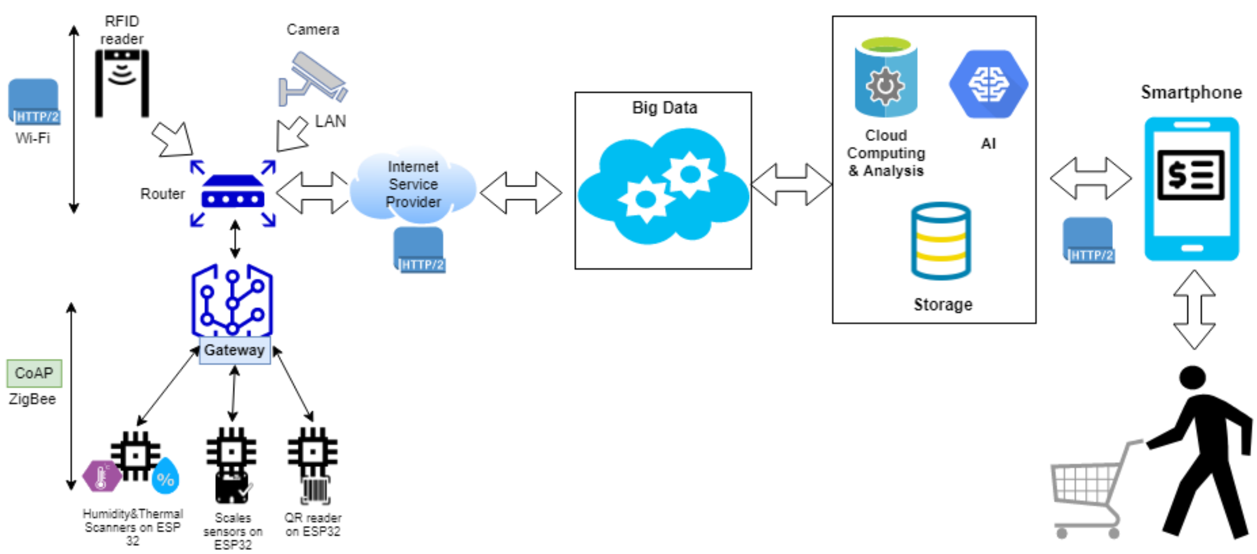
Regarding the RFID reader, it is specially designed to support HTTP connection, thus it will also interact with cloud on the HTTP base. RFID reader communicates with RFID tag on the RF communication. In other words, tag gets the power from the RF field (with antenna) and reader send tag commands and then get response from the tags. We need this to confirm the products that our customer bought at the exit of the shop to prevent any type of theft and exclude any errors of our camera and weight sensors.

Weight Scale sensors will work in the same manner as thermal and humidity. By the mechanism on the cloud we will match the data from the camera, that some item is taken from the shelve and by obtaining data from the weight sensors we may confirm the type of item taken, and also was it actually taken, or just taken and put back on the shelf again.

Actually, in order to correctly define our application layer protocols we need to go through the sensor types available in the market and their support. It is worth to mention that before writing this report we did this investigation.



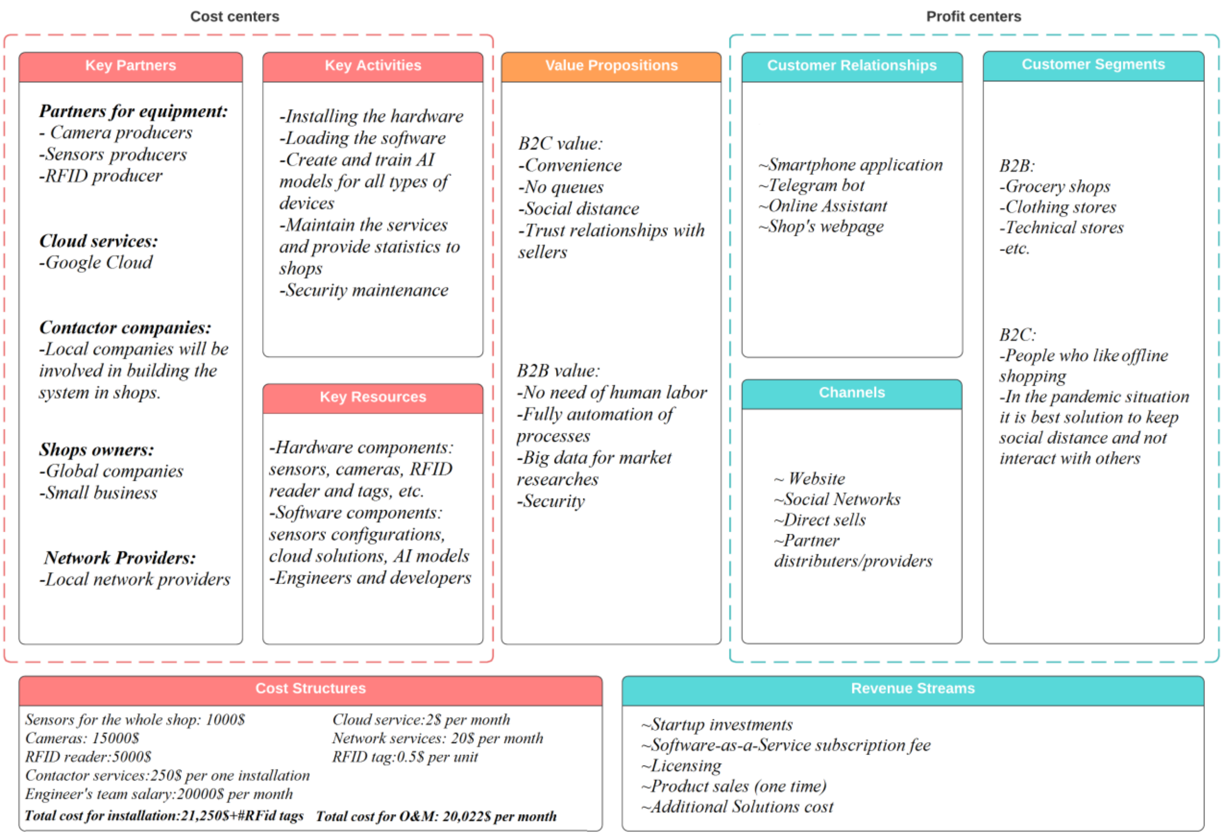
4.6 Implement an end to end IoT service scenario based on solid results of all homework assignments



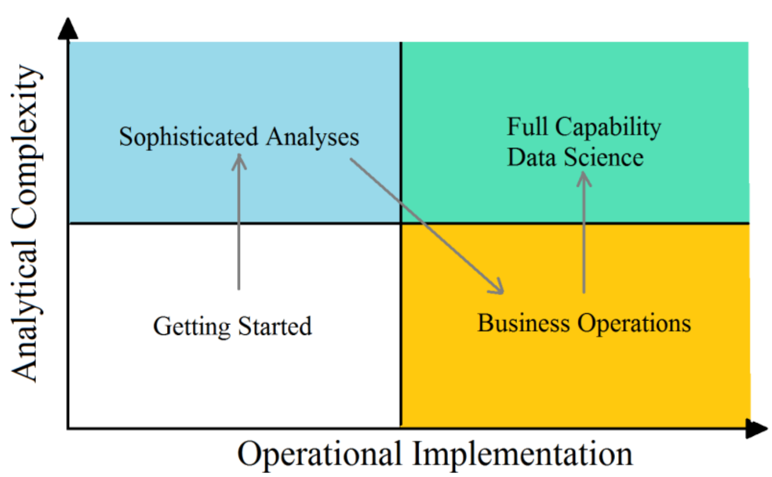
5. Consumer Value-oriented IoT Service

5.1 Our high level design for business model of the consumer value oriented IoT service

5.1.1 Our business model



5.1.2 Graceful maturity progress business model for intelligent (AI integrated) IoT service



1. **Getting Started:**

   In this first step, we identify how data is collected and what it's type of and try to find opportunities for further use. Our sensors and cameras are the main source of information that grows from little data to the Big one. We will gather such data as: thermal and humidity states, notifications when scales detect changes in weight of shelves, cameras' video data, RFID data.
   All this information will be send to the cloud for analysis and providing service for customers.

2. **Sophisticated Analyses:**

   One of the major aims for cloud analysis and operations is that we need to achieve is to train the face recognition and object detection models. When scales sensor sends notification that weight of shelf is changed, video from camera in sector of this sensor must be analyzed in order to identify the customer and assign the price for product.

   Also, our AI will be trained to control the temperature and humidity levels in the shops.

   Therefore, conditions analysis model must be developed.

3. **Business Operations:**

   The result that we want to achieve is that our customer will be able to enter the shop, authorize, take products they want and leave the shop. Services as billing, checking and security will be fully automated and achieved by IoT devices and Artificial Intelligence analysis.

4. **Full Capability Data Science:**

   Many product producers and economic universities spend a lot of money for market research. From obtained Big Data we may make their life easier by training AI to make market predictions. It also will help to the shop owners to supply more popular products.
   Moreover, further analysis of received data will allow to learn more about our customers, their preferences and behavior. These aspects are important in economic, psychological and social sciences.

5.2 Design a service model for our application

At the first stage, customer register in our shop client's database providing their credit card and personal information. Easy process of shopping will lead to more customers, so more revenue. Customers may come to other branches being registered in the same database. Products are monitored in real-time as well as payments are made online. So

products with the help of sensors and informative tags and reduced expenses on staff and product waste will provide more revenue, than ordinary shop.
Therefore, we are able to represent our service model from all sides, namely: Product-As-a- Service, Data-As-a-Service and Process-As-a-Service.

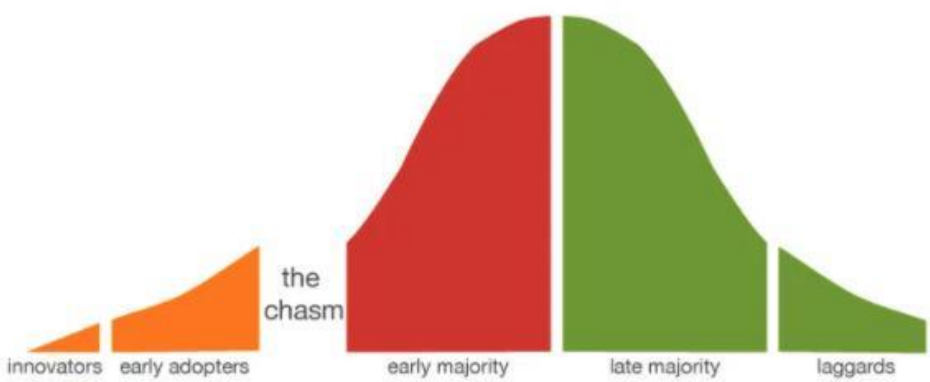
- With the help of weight sensors, camera and RFID tag we can monitor the product state in real-time.
- With the integration of AI, ML and Deep learning capabilities, our system can analyze most frequently needed products, or the products that are stale. This enables to reduce expenses on unpopular things, understand buying patterns and auto scale the buyouts in the market.
- IoT-as-a-service in our case gives the enterprise an opportunity to lead predictive managing, instead of preventive, where everything was based on the guesswork.
- As interaction with customer happens without human intervention(staff) and is stored under personalized data piece, we can collect more data about the user and provide better performance. For example, if customer buys something every time in bakery section in market in his visit to our shop, we can provide sales and discounts, or special offers for him. Thus, both we and customers are getting benefits.
- With thermal and humidity sensors we extend the life of our products and reduce the waste, especially in grocery.

5.3 Overcome the chasm of IoT market

The product life cycle is the process of adoption. First, we will sell our product (first MVP) to innovators. Innovators are those who are interested in IoT technologies, that means our first clients are owners of techno markets and people who follow IoT technology development beside their different background. Next step is developing our application, by adding features which include (ABC) and provide second step of our MVP to early adopters.

After successfully crossing early adopters stage where we possess 16%(Early Market) of the whole market, we face a chasm. The chasm appears because we would like to increase our sales.

So, to overcome the chasm we need to bring our MVP to final level, means we need completed product with all viable features inside. We analyzed “The KANO Analysis” and decided to use his idea of maximizing consumer satisfaction with minimal investment. First our mission is market investigation. We need to compare the prices in the existing market, and we are going to put our price a bit less than our competitors. So, final step to overcome the chasm, is to provide the advertisement of product to those who may highly appreciate our product, means huge markets like Havas, Makro and Korzinka). By following all these steps, we are assured to pass the chasm and reach Early Majority (Pragmatists).



6. Conclusion

All in all, our main goal was to create a smart IoT system that will make people’s life more comfortable and less stressful. IoT systems are the future of industries and soon they will be everywhere. Today technologies become more powerful in computing and decision making, and connectivity between devices becomes stronger and more effective, our Smart Shop may be really deployed to provide qualitative services and perform Big data analyses.