CLOUD BEACON

Electronics Club

Ashwath Mukesh Bhat

Sundar Raman P

Cloud Beacon

Arvind Menon, Sundar Raman, Ashwath Bhat,
Ameya Ainchwar, Tamil Sudaravan M

Inter IIT Tech Meet 8.0

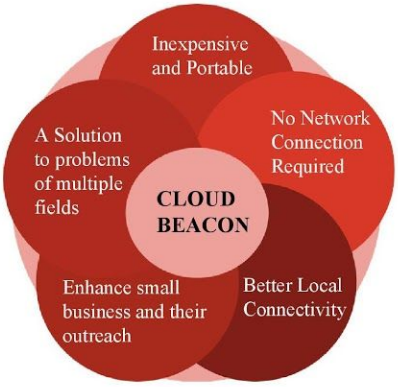
Centre For Innovation

Abstract:

The aim of the project is to make local communication much easier so as to provide personal services in a faster, secure and efficient manner.

Introduction:

The problem addressed in the lack of the ability to communicate locally in a more efficient and non intrusive manner,i.e. Shopkeepers/Local businesses still have to use big billboards or banners to advertise.The proposed technology tackles this issue and provides personal service locally.



Poster can be made redundant. Find how?

HOW IT WORKS?

Host

NodeMCU ESP8266/ESP32

- The host uses the application to upload the data, to cloud storage, that are publicly available to the user. After uploading the data, using unique information related to data's location in cloud, the host creates access point with the use of ESP8266/ESP32. The access point name is encoded in way, it has information known to the host which links the user to the data which the hosts wants to display.

User

- This module acts as a Wi-Fi access point, covering a 27 m radius around it. We tested the product in a restaurant and it successfully dispensed information up-to a distance of 27meters, which is an optimal distance for local information sharing.
- The access points created by the module act as a beacon of information.
- The user application extracts all the access points created by the Wi-Fi module among all the access point available. Those access points are listed out in the user application and the user can choose whichever information he/she wishes to read. This information can be in any format.

User

Novelty:

- Transfer of information without even having a wireless connection to the network.
- Since, the users are not connecting to the Wi-Fi, there won't be any security issues.
- If there are lot of user in a locality, this way of transferring information would not create a traffic in sending information to the user

Applications:

The following are some of places where this project can be implemented

- Vendors, small business looking to spread the details of their products or offers and discounts.
- Museum that wants to send information about an artifact directly to your phone as you get near it
- A bus or a train that wants to provide detailed info to its passengers or people located at the platform

Future Works:

- During CFI Open House 2019, a private Hotel Owner wanted to install our product in his hotel
- DeTect Technologies asked us to integrate our project with their drone. We are currently working on this.

Reference:

1. ESP32/ESP8266 official documentation.
2. Firebase official documentation.
3. Android Studio official documentation.

Our project represented IIT Madras in Engineer's Conclave Exhibition event, Inter-IIT Tech meet where participants will have to showcase their projects so that the students get to know about the strengths, resources, and projects undertaken by different IITs. Every IIT is permitted to present a maximum of 5 projects with participation points for each project.

Team Members

<table>
<tr><td rowspan="2">  </td><td>EP17B017</td><td>ARVIND SATISH MENON</td></tr>
<tr><td colspan="2">Third-year Engineering Physics undergraduate
Interests: Coding
Contribution : Project mentor to mini server network
Email id: ep17b017@smail.iitm.ac.in</td></tr>
<tr><td rowspan="2">  </td><td>ME18B122</td><td>AMEYA AINCHWAR</td></tr>
<tr><td colspan="2">Second-year Mechanical Engineering undergraduate
Interests: Android App Development, IoT
Contribution : Developed the administrator app, configured ESP using app
Email id: ameya.potter7@gmail.com</td></tr>
<tr><td rowspan="2">  </td><td>EE18B061</td><td>PRAVEEN KUMAR M</td></tr>
<tr><td colspan="2">Second-year Electrical Engineering undergraduate
Interests: Microcontrollers, Wifi modules and what not!
Contribution : Working with microcontrollers, wifi modules and integrating wifi module with app
Email id: praveenkumar2k1@gmail.com</td></tr>
</table>

CE18B122	SAGNIK DATTA

- Second-year Civil Engineering undergraduate
- **Interests**: Android Application Development, programming and IOT
- **Contribution**: Worked with the application development part. Worked on Fusion 360 for the case.
- **E-mail Id:** sagnikdatta175@gmail.com

EE18B119	TAMIL SUDARAVAN M

- Second-year Electrical Engineering undergraduate
- **Interests:** App-Dev, ML , DL, Electronics
- **Contribution** : worked in the App Development part of the project.
- **Email id:** tamilsudaravan@gmail.com

EE18B030	SHAIK RESHMA RUKSHINDA

- Second-year Electrical Engineering undergraduate
- **Interests:** Interested in coding working with modules
- **Contribution** : Worked on making ESP module access point
- **Email id:** reshmarukshinda@gmail.com

Table of Contents

1. Title of Invention

Cloud Beacon

2. Field of Invention

Our project Cloud Beacon is one of it's kind, an innovative idea which made it difficult to be classified into one of the existing fields. But it can be identified as an IoT device.

3. Background of Invention

Once upon a time, when QR codes are not widespread, imagine the difficulty of sharing data to people, without knowing or storing their contacts. It became much more convenient to use QR codes as the are fast, easy to generate, and error-free. But still, why use an old idea, which has problems such as, non reusable, displaying the QR code, etc. So, we need a means to share data anywhere within proximity and to reuse it multiple times. This resulted in our project Cloud Beacon, sharing data encoded over wifi, hence it's reusable and can be transferred within a proximity.

4. Object of Invention

The first object of invention is sharing data anywhere within proximity. This saves a lot of work in search for the place where data is shared. Another object of information is that there's no need for an established connection between data provider and receiver. This is the most important feature, as it eliminates all the

threats because of connecting to unknown networks. Another object of the invention is that multiple requests can be handled at the same time, and there does not exist an upper limit for number of people.

5. Design History and Overview

Give an overview of your current design. Include all the prototypes or models made before arriving to the present one, issues with the previous design. Also include the images of previous and current designs.

The current design consists of two android apps and a wifi module. The wifi modules used are ESP32 and ESP8266. The algorithm used has two major parts, administrator and user.
The administrator uploads the information to be shared to module with the help of administrator app. First the module is connected to the mobile device(with app) via wifi, then the data is sent from mobile to esp. The received information is processed and Wifi is setup by esp(like a router) which has encoded information in it. The user app scans and lists all the encoded wifi networks, each corresponding to different data uploaded by different admins. Selecting a particular option leads to the file that admin has uploaded, which is accessed from the name of the wifi.

A prototype module was first developed which acts as a web server and the administrator should upload his file by accessing the web page created by the module. Later, this was discarded because of the difficulty in accessing webpage from app.

The administrator app connects the host to the ESP module through the WiFi network created by the module. The app requests the host to enter the host name. Then the app prompts the host to upload the required file (only PDF as of now).

Then it uploads the file to the Firebase cloud. It retrieves the URL of the file in the cloud and shortens it using TinyUrl API. This shortened URL is sent to the ESP after adding certain identifiers. The ESP then creates a WiFi network with the host name as it's name and the shortened URL encoded in the network name.
The administrator app is also designed well. The process of initially connecting the mobile to the ESP is also automated. As soon as the app is opened, it checks if the WiFi is switched on. If the WiFi is off, it automatically switches it on without user intervention. Then the app automatically connects to the ESP without having to do anything manually.
The user application is made in a way, it gets continuous feed about the access point made by the particular module(ESP32/ESP8266) there around it. The access points created by the module act as a source of communication (more or less as a beacon). The access points are listed out in the user application and the user can choose whichever information he/she wishes to read. This information can be in the form of a picture, video or a pdf document.

6. Market Analysis

Customers for the project is almost everyone who goes to a public place.

For example,

- Someone who visits a mall to shop for a product is a potential customer of our project. He can use our project to get list of shops in which he can get the product he wishes
- Someone who is travelling in public transport is a potential customer of our project. We can provide a personalised map system for him.
- Someone who is taking a seminar, wants to share a file will whoever attending his seminar with no time using our product.

A slightly modified version of this project from its core version can be in many fields. So the common public people in public places are potential customers.

So customer identification is not a major problem.

This project can be simply implemented using a ESP32/ESP8266 modules. So, investment for the product is economical compared to other means.

For a simple case of advertising, this product can be an alternative to large size bill-boards and banners. This cuts the rent charges for advertising. The money which we spend for advertising can be very less compared to the money spent on conventional Bill-Boards and Banners. So, in this case our project provides an innovative and cost-effective way of advertising. Similarly there are many ways in which this project can be economical depends on the area of its use.

During CFI Open House 2019, a private Hotel Owner wanted to install our product in his hotel. Detect Technologies (DeTect) is an IIT Madras incubated startup. DeTect wanted to integrate our prototype with their drones. We are working on both of this. The prototype was reviewed by the professors of our institute who did confirm the helpfulness of the module in the classrooms for a better teacher – student interaction. Our project was also involved in sessions conducted by NIRMAAN, the pre-incubation cell in IIT Madras.

Considering the above observation as sample survey we got to know that our product can also be implemented in Hotel/Lodge market, Drone market for transferring information without internet. So the market for this product is vast and customers are also willing to buy the product.

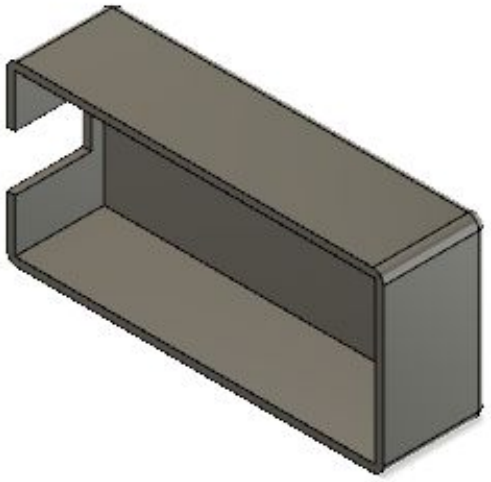
7. What did we learn?

To begin with, our project had a very good idea. But that made things tougher, as we had to find a problem statement for which we had to modify our idea, so that it can be used as a solution. In the above process, we learnt the process of ideation, which was refined as we had to find ways in which we can implement our solution. The solution required us to work with apps and wifi modules. So, we learnt the structure of Android apps and how to develop them. Also, we learnt to work with microcontrollers, especially the ones with WiFi option available. In the following process, we learnt the basics of web server and web pages. Finally, we learnt how to integrate microcontroller with android apps.

8. Acknowledgement

We would like to express our gratitude to CFI for taking up this project, without which I fear the project could have achieved this milestone. We are obliged to give special thanks to Electronics club for their help and guidance and also to the love and support from the cool club heads. We would also like to thank PM team for their valuable advice. We learnt a lot of new stuff and working as a team was especially fun. The overall journey with CFI and Electronics Club was something that we would recommend every techie to have.

Appendix - A : CAD Models



This case is used protect the micro-controller from yhe external environment. The grove is used for connecting the micro-controller to power supply.

Appendix - B : Source Code

Source Code for User Application:

- Link for Frontend code: https://github.com/SudaravanM/Cloud_Beacon/blob/master/app/src/main/res/layout/activity_main.xml
- Link for Backend code: https://github.com/SudaravanM/Cloud_Beacon/blob/master/app/src/main/java/com/example/wifi_scanner/MainActivity.java
- Link for Manifest code of the application:

https://github.com/SudaravanM/Cloud_Beacon/blob/master/app/src/main/AndroidManifest.xml

Source Code for Administrator Application:

- Link for Frontend code: https://github.com/SudaravanM/Cloud_Beacon/blob/master/Administrator_Application/app/src/main/res/layout/activity_main.xml
- Link for Backend code: https://github.com/SudaravanM/Cloud_Beacon/tree/master/Administrator_Application/app/src/main/java/com/example/firebasefileupload
- Link for Manifest code of the application: https://github.com/SudaravanM/Cloud_Beacon/blob/master/Administrator_Application/app/src/main/AndroidManifest.xml
- Video Link of working User Application: https://drive.google.com/drive/u/2/folders/1t9QM6obEV9oN6XsMliisMure_5K8A1pE

Source Code for Microcontroller:

- Code for Web-Server: https://github.com/SudaravanM/Cloud_Beacon/blob/master/MCU_Source_Code/webserver_code.ino
- Code integrating administrator application with MCU: https://github.com/SudaravanM/Cloud_Beacon/blob/master/MCU_Source_Code/source_code.ino

Bibliography

- https://github.com/MorbidEntree/android-tinyurl/blob/master/screenshots/created.png
- https://stackoverflow.com/questions/8818290/how-do-i-connect-to-a-specific-wi-fi-network-in-android-programmatically
- https://tttapa.github.io/ESP8266/Chap01%20-%20ESP8266.html
- https://www.udemy.com/course/complete-android-n-developer-course/
- https://github.com/esp8266
- https://tttapa.github.io/ESP8266/Chap01%20-%20ESP8266.html
- https://github.com/lucadentella/esp32-tutorial
- http://tiny-url.info/open_api.html
- https://developers.google.com/url-shortener/v1/getting_started