



VIT[®]
Vellore Institute of Technology
 (Deemed to be University under section 3 of UGC Act, 1956)

IoT DOMAIN ANALYST

ECE3502

Winter Semester 2021-22

J COMPONENT PROJECT REPORT

MESS POLLING AND BOOKING SYSTEM **USING ANDROID APP**

GROUP MEMBERS

HARJAP SINGH DANDIWAL	19BEC0595
SHIVAM KUMAR OJHA	19BEC0602
SATYAM UPADHYAY	19BEC0612
ASHUTOSH DEWANGAN	19BEC0628
SHRUT MAKDE	19BEC0656
SLOT	L7 + L8
FACULTY	PROF. ARVIND KUMAR
DATE	24-04-2022

CONTENTS

Abstract	3
Introduction	4
Hardware Required	5
Software Required	6
Methodology	8
Block Diagram	10
Code & pictures/screenshots of the working project	11
Pros & Cons of the project	22
Future Work	23
Conclusion	23
References	24

ABSTRACT

In our hostel days, before the breakout of Covid-19, we used to have our meals in hostel's mess, each month and each day there used to be a different menu and loads of food is wasted everyday because some students don't come to have their food. So to overcome such situations, five of us have come up with an idea to make such project to help everyone in the campus and create a food mess booking system which would allow students to poll and vote for the particular food item of their liking and also book their food accordingly whether it be breakfast, lunch or dinner using an app in which would also display the booking done by students on the counter of the mess which will the mess staff that the particular student will come and have food, so, they will keep the plate ready. This way food wastage can be exponentially reduced and a new modern mess system can be started.

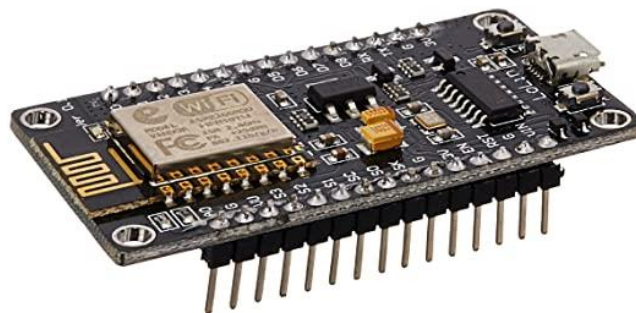
INTRODUCTION

Our project mainly focuses on what customers poll and would want to book and staff providing accordingly with least waste so that everything is efficient and up to point. In this we are using **Android Studio** to design and run the application and using **Firebase** to feed data and **Adruino IDE** code the content of the application.

There will be two switches, one for 'polling' and other for 'booking'. Users can poll and then later book their needs accordingly and later staff can work on it. Furthermore, we worked on hardware side to display booking/polling system status for staff side so that it will be more convenient for them to figure out the needs of the customers

HARDWARE REQUIRED

1. **Node MCU:** NodeMCU is an open source IoT platform. It includes firmware which runs on the ESP8266 Wi-Fi SoC. NodeMCU has 128 KB RAM and 4MB of Flash memory to store data and programs. NodeMCU can be powered using a Micro USB jack and VIN pin (External Supply Pin).



2. **Switches** (For Polling and Booking)

SOFTWARE REQUIRED

1. **ARDUINO IDE:** The open-source **Arduino** Software (**IDE**) makes it easy to write code and upload it to the board. This software can be used with any **Arduino board**.



2. **FIREBASE:** Google Firebase is a Google-backed application development software that **enables developers to develop iOS, Android and Web apps**. Firebase provides tools for tracking analytics, reporting and fixing app crashes, creating marketing and product experiment.



3. ANDROID STUDIO: Android Studio provides a unified environment where you can **build apps for Android phones, tablets, Android Wear, Android TV, and Android Auto**. Structured code modules allow you to divide your project into units of functionality that you can independently build, test, and debug.



METHODOLOGY

An integration between different devices has been done in this project using the Firebase. The 2 Android Apps - one for students and one for management, and hardware which will be at the hostel mess controlled by the mess admin are connected to firebase. In order to connect the hardware in mess to firebase a NodeMCU module has been interfaced which will connect the hardware to firebase via Wifi whose credentials are coded into NodeMCU.

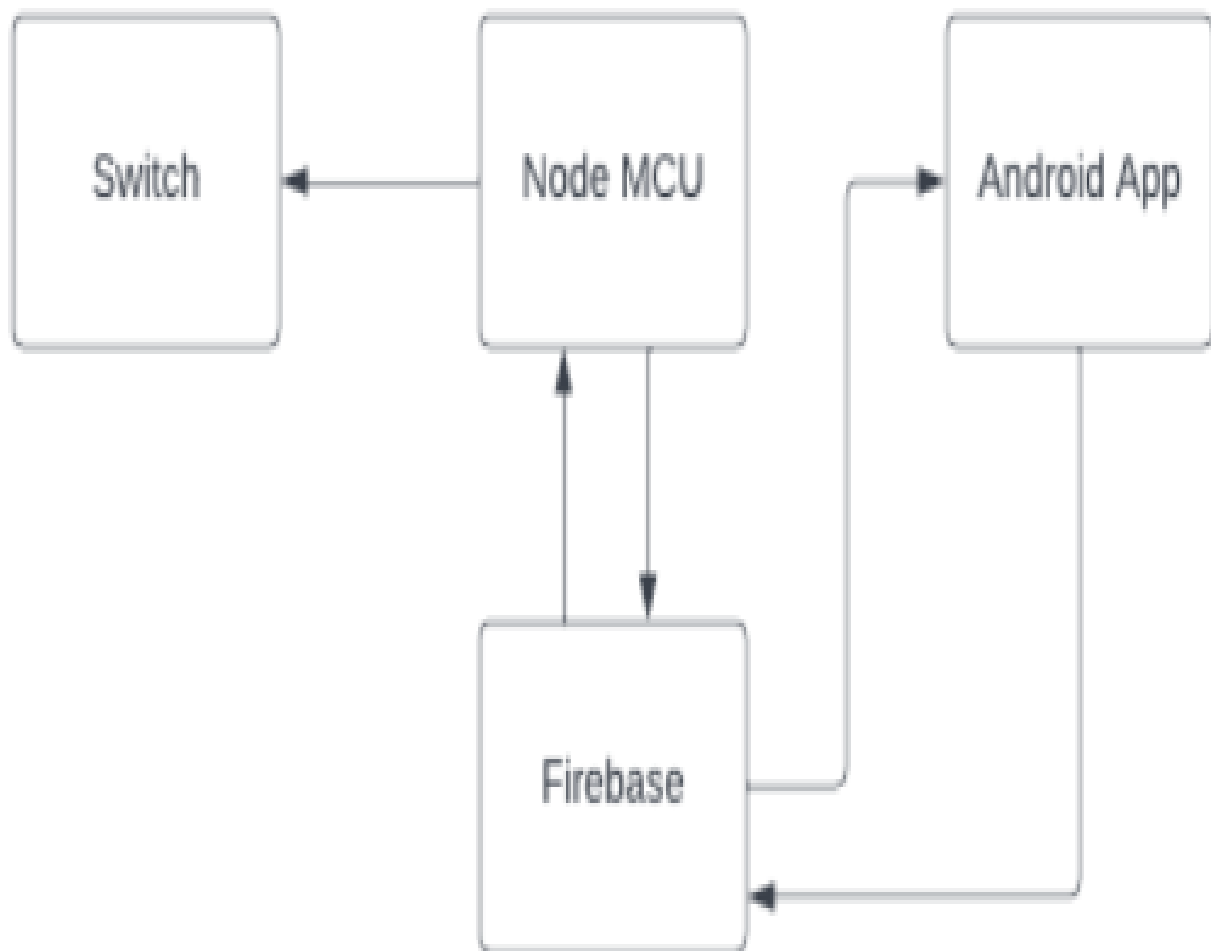
The 2 Android apps are developed using Java Language in Android studio. Addition and deletion of menu items by mess admin plus getting the details about students' poll in his interests with exact quantity are the main features of the mess management app. Whereas in Students' App, ability to poll to the menu items added by mess admin plus feature of specifying the exact quantity to be eaten are the main highlights of students' app.

There is an extensive use of recyclerviews and adapters in both of the mobile apps. These views are used to set the data in app screen. These allow the apps to show the

items in app screen dynamically i.e., no matter how many datas are there in the database, just by adding a recyclerview those datas can be accommodated to mobile screen. If we don't use recyclerview and adapter in the app then we need to create as number of views as the number of elements present in database, which is not a good practice. That's why recyclerview with adapters needs to be implemented.

Student will be able to give his poll or book his meal only when the mess admin allows. When the mess admin presses the poll/book switch, data will be sent to firebase and then that data will be sent to students' app which will decide whether to allow polling or booking or stop the process. And this status of whether booking or polling will be displayed to the LCD display attached to NodeMCU to mess admin.

BLOCK DIAGRAM



CODE, PICTURES/SCREENSHOTS OF THE PROJECT AND ITS WORKING

Code in NodeMCU using ARDUINO:

Link for Code's drive folder is provided:

<https://drive.google.com/drive/folders/1wmSwyan-qYohMSmU8DOZz42zOK8S9RgB?usp=sharing>



```
IoT_Domain_Project | Arduino 1.8.19 (Windows Store 1.8.57.0)
File Edit Sketch Tools Help

IoT_Domain_Project
#include <ESP8266WiFi.h>
#include <FirebaseArduino.h>
#include <Wire.h> // library for I2C protocol
#include <LiquidCrystal_I2C.h> // library for I2C LCD
LiquidCrystal_I2C lcd = LiquidCrystal_I2C(0x27,16,2); // set the LCD address to 0x27 for a 16 chars and 2 line display

#define FIREBASE_HOST "iotproject-6483b-default-rtdb.firebaseio.com"
#define FIREBASE_AUTH "xMpRk8I10SRaMSQXkqnvcoHe1STENfa6U4VHQVg0"
#define WIFI_SSID "realme 8"
#define WIFI_PASSWORD "Realme88"

void setup()
{
  lcd.begin(16,2);
  lcd.init(); // initialize the lcd
  lcd.backlight(); // backlight ON

  Serial.begin(9600);
  pinMode(D4,INPUT); // For Booking
  pinMode(D3,INPUT); // For Polling

  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Connecting to ");
  Serial.print(WIFI_SSID);

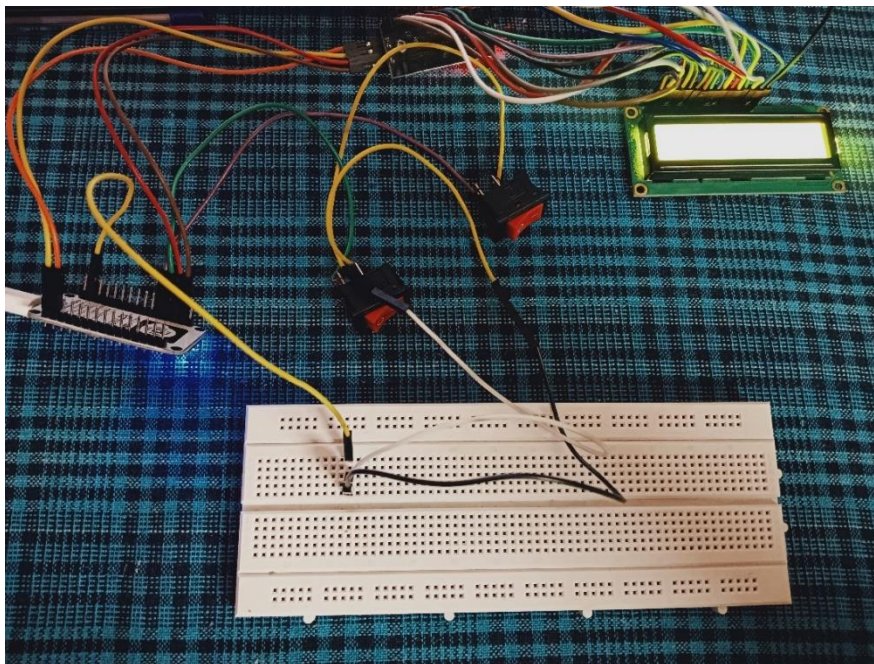
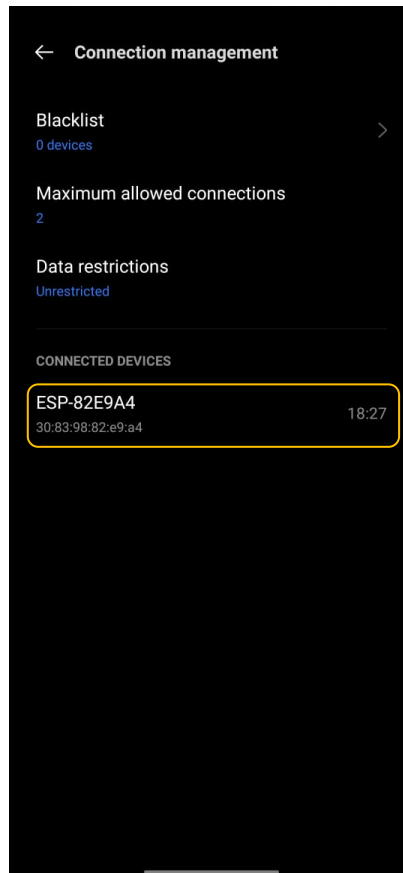
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
```

Code for students' and staff's app using android studio:

Link for Code's drive folder:

<https://drive.google.com/drive/folders/1xZnNWSysFu6ckeUg-W3pE6NsTikoDLV?usp=sharing>

Connect NodeMCU to Wifi, in this project we are connecting it to our mobile hotspot whose credentials are already coded into the NodeMCU module:



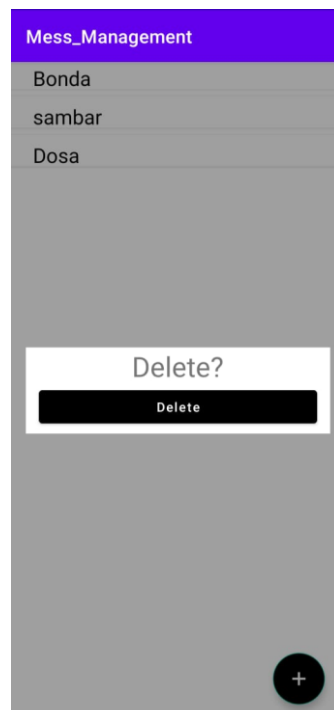
Snapshot of Hardware implementation of our project

As soon as mess admin enters the mess, his first Task is to update the Menu for that particular meal time on the database using Mess_management app:

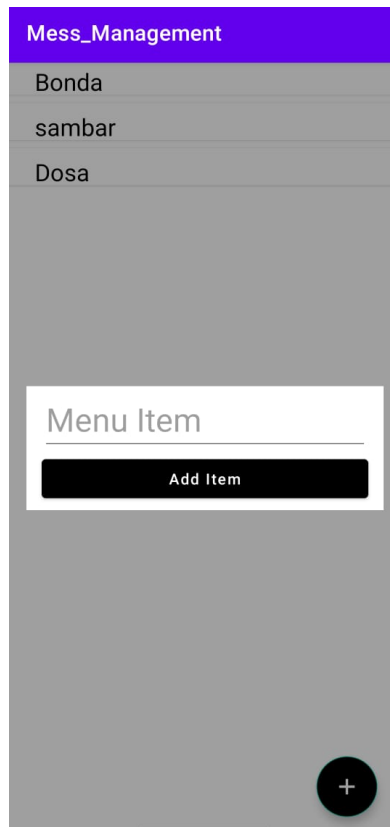
Mess_Management
Bonda
sambar
vada
Pav Bhaji
biryani
Dosa



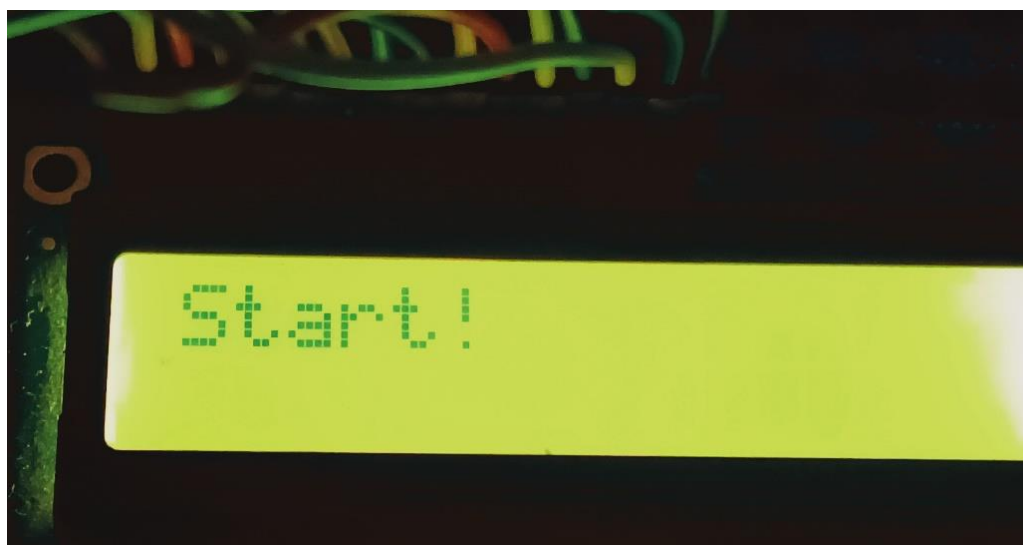
Already added items can be deleted by long click on those items:



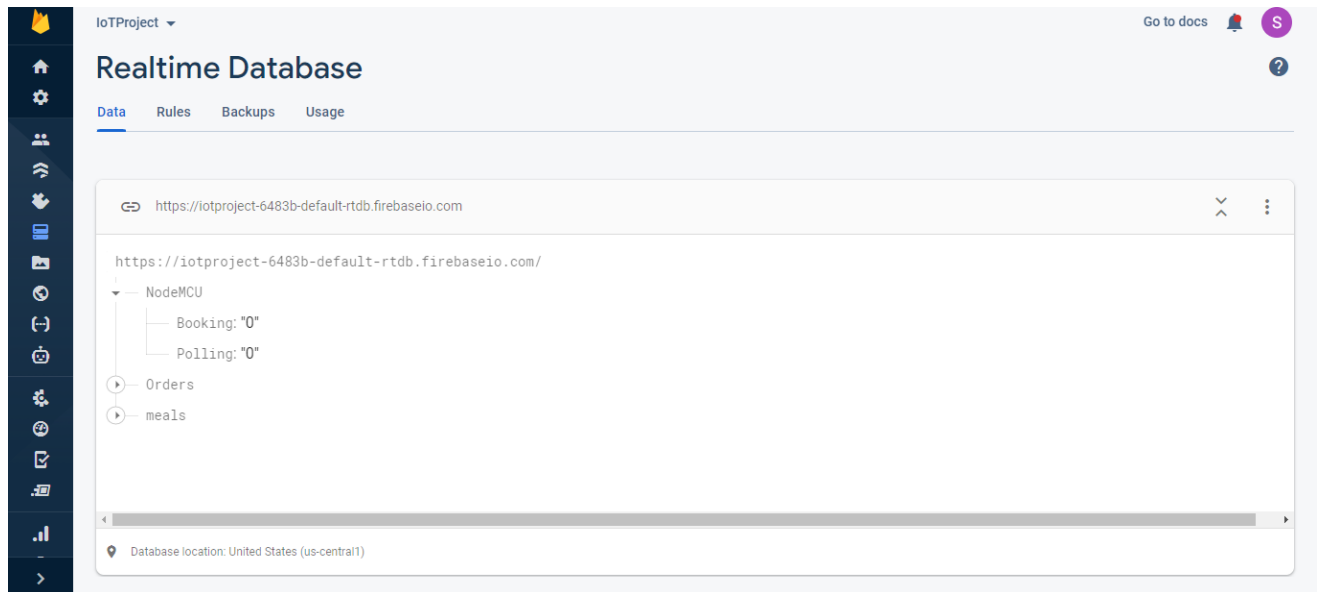
And new items can be added by clicking on the “+” floating button:



After the menu updating is done, when the admin turn on the device with circuit as shown above, then he will observe the following on the LCD display:



And in the backend i.e., in the database which is Firebase in this project, the scenario will be like:



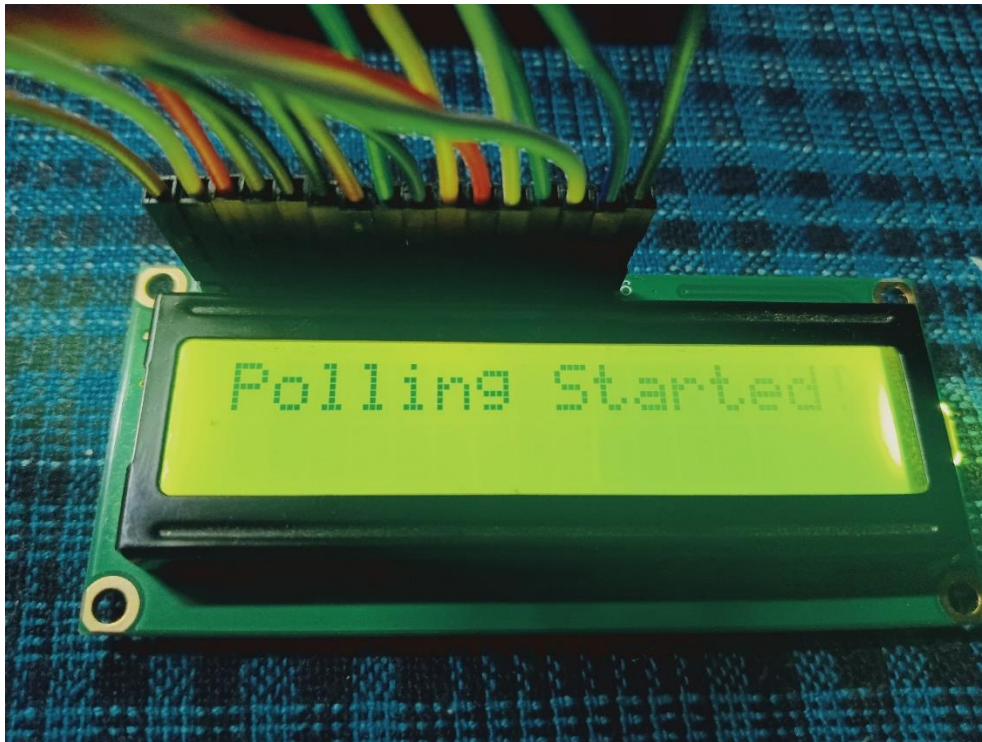
From the above snapshot, it can be observed that the neither the Polling nor booking process has been initiated by the mess admin and similarly, according to this data Buttons for poll and booking will be disabled in the students' app:

IoT Project

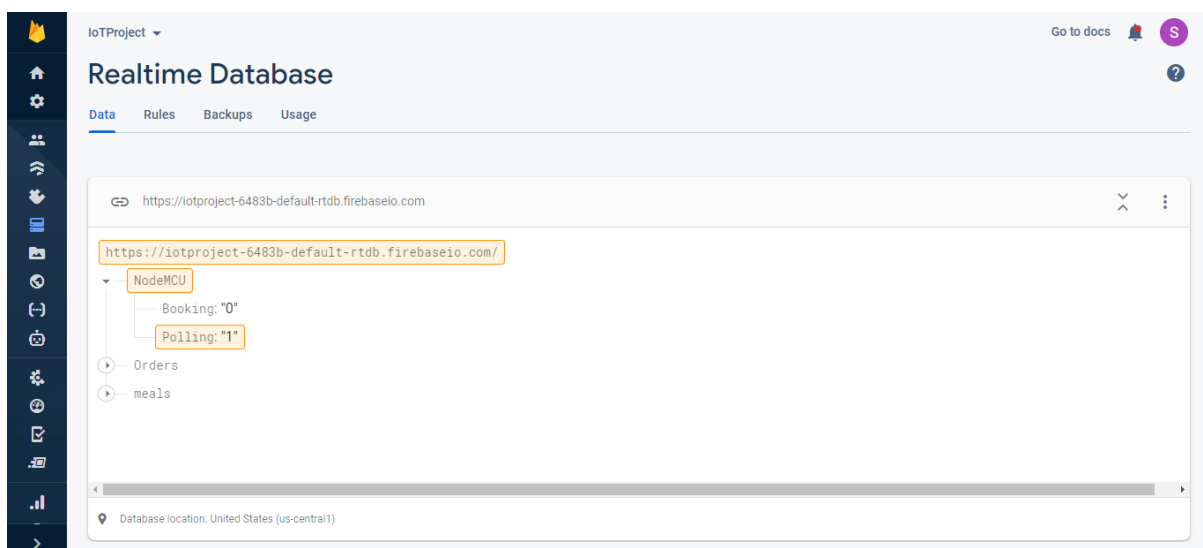
BOOK YOUR MEAL

POLL

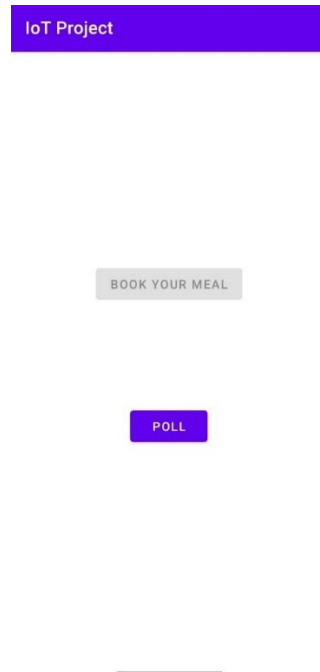
When the admin switches on the polling switch, it will be displayed in the LCD:



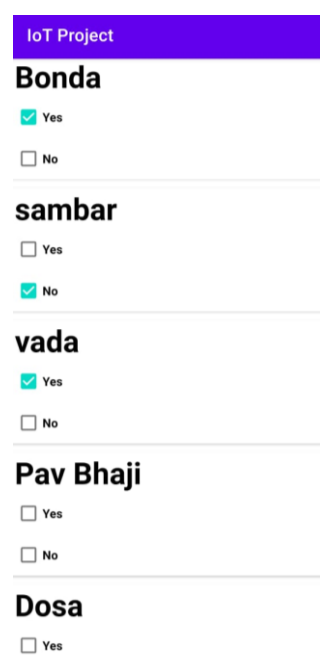
In addition to this, NodeMCU will update the data in firebase to “1” indicating that the Polling has been initiated:



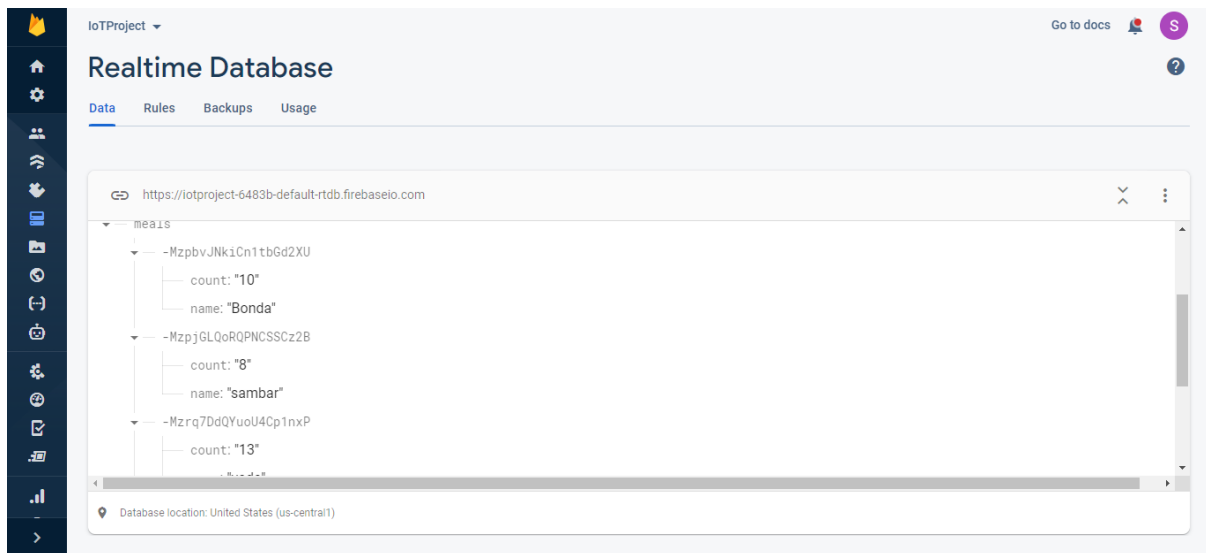
As soon as the polling status is updated in firebase, same will be reflected into the Students' App:



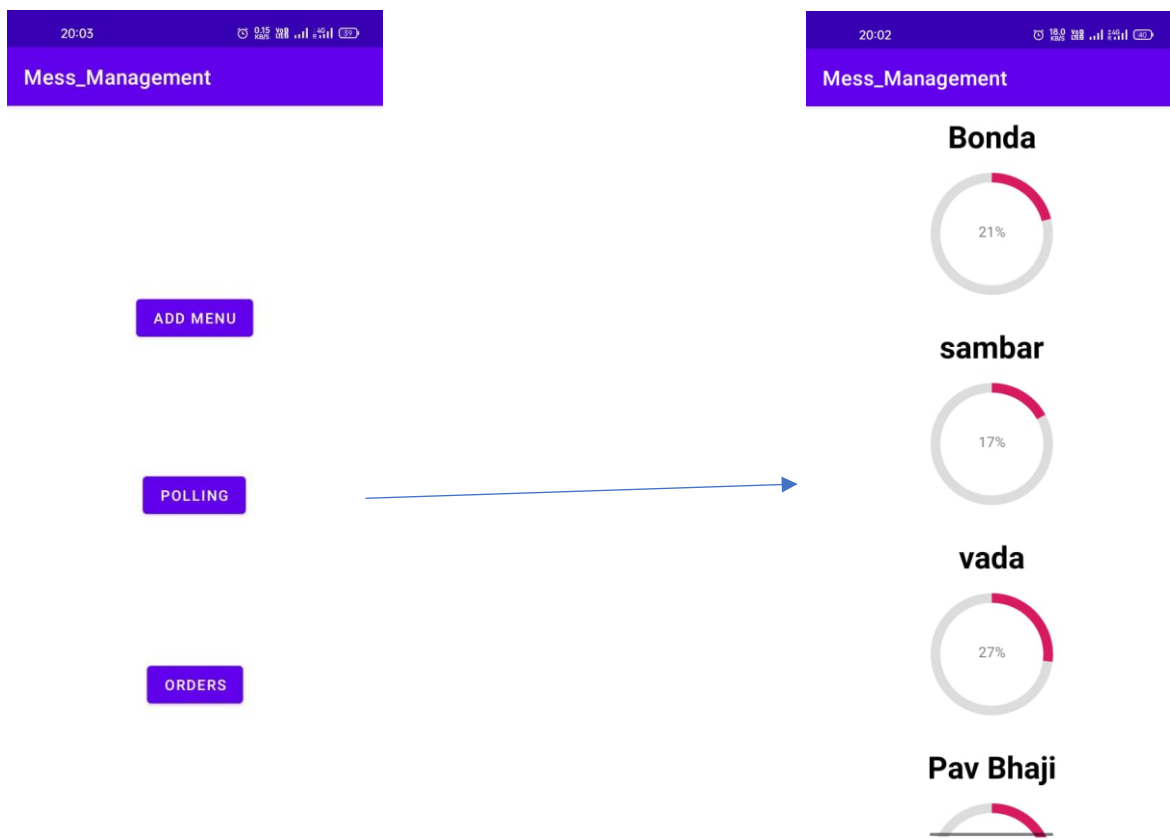
Now the student can give his poll by checking the boxes “Yes” or “No” and the number of users interested in specific menu item will be updated in the firebase:



As the students' keep on giving their poll, the data will also keep updating in firebase:

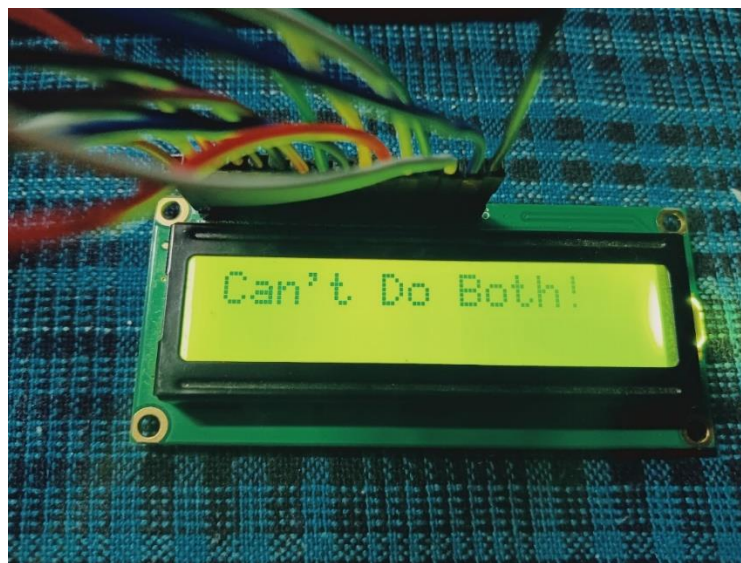


This poll results can be observed in the mess management app:

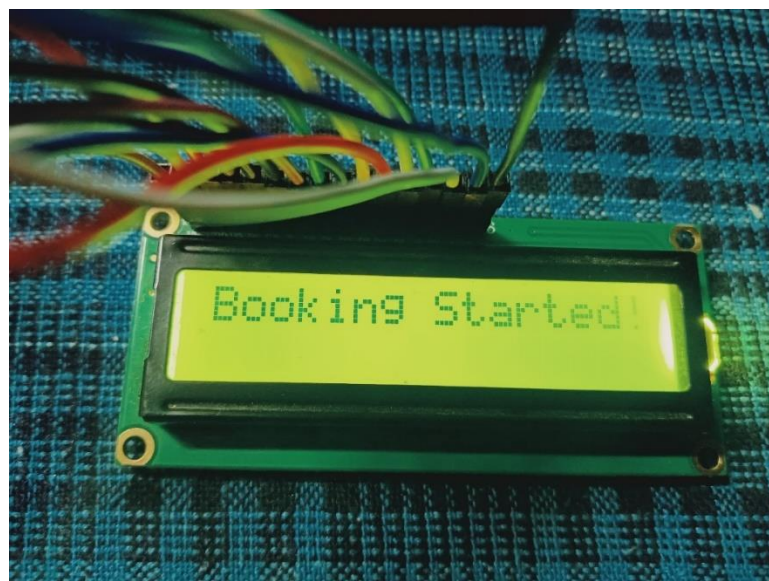


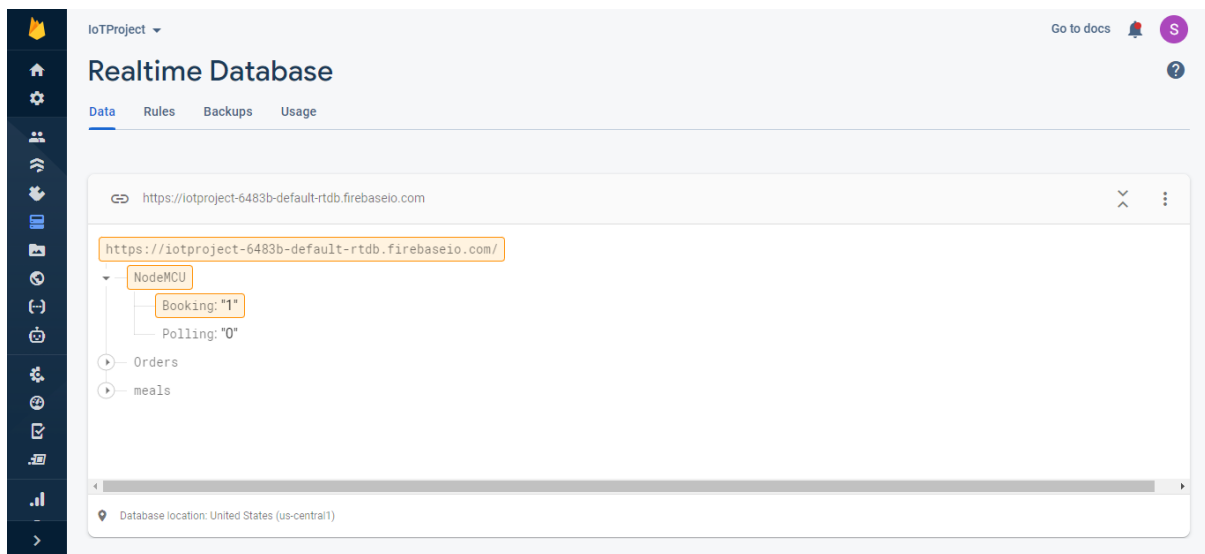
Note that here, percentage of people interested in that particular item are displayed and not the quantity to be made. The quantity to be eaten/ordered by students will be displayed in Orders screen.

If by chance, admin switches on both the buttons i.e., poll and booking at the same time, the process will be blocked and LCD will display:

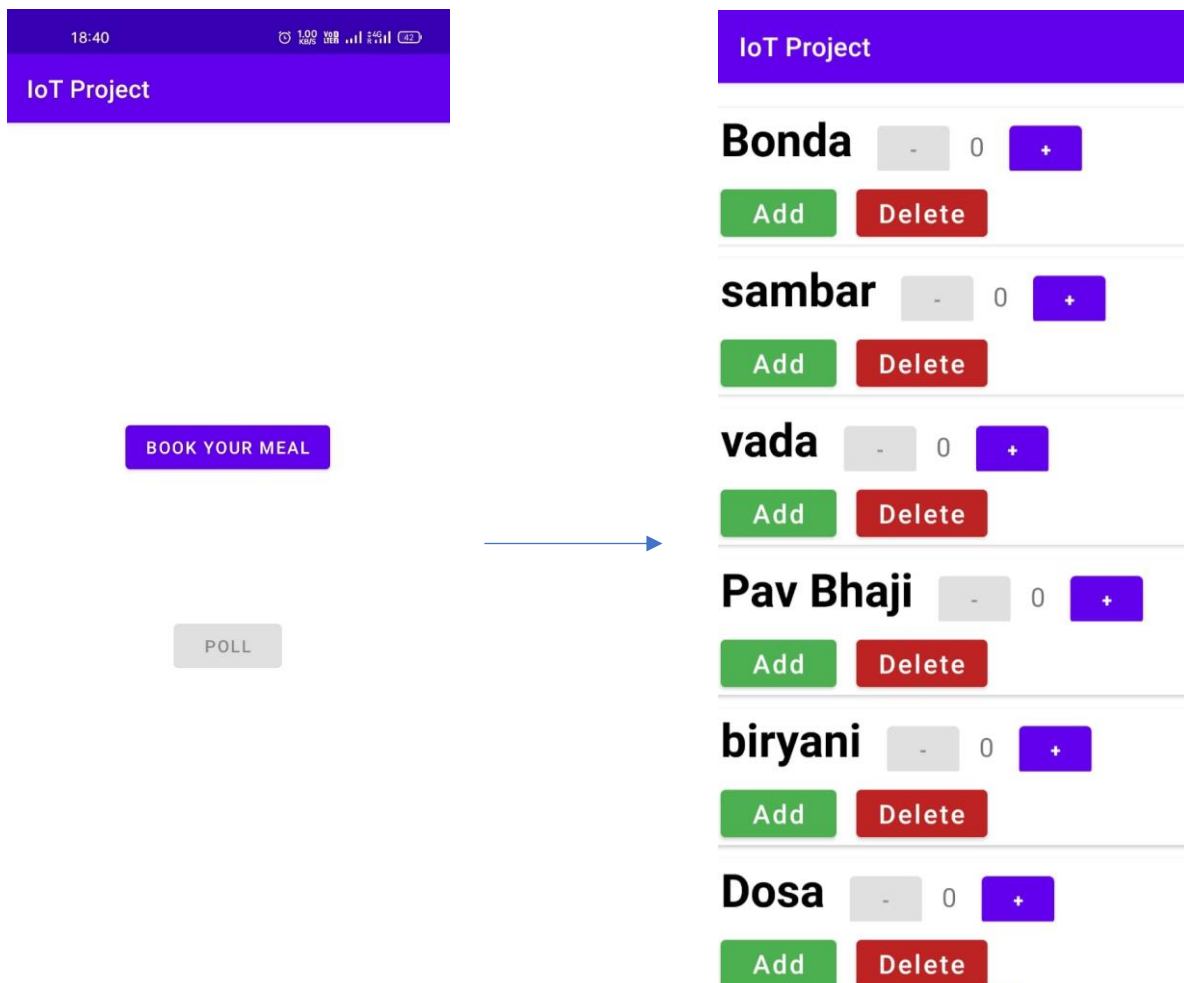


Now after some time when the admin switches off the poll switch and turns on the booking switch, LCD will display the same and accordingly the data will be updated in firebase:

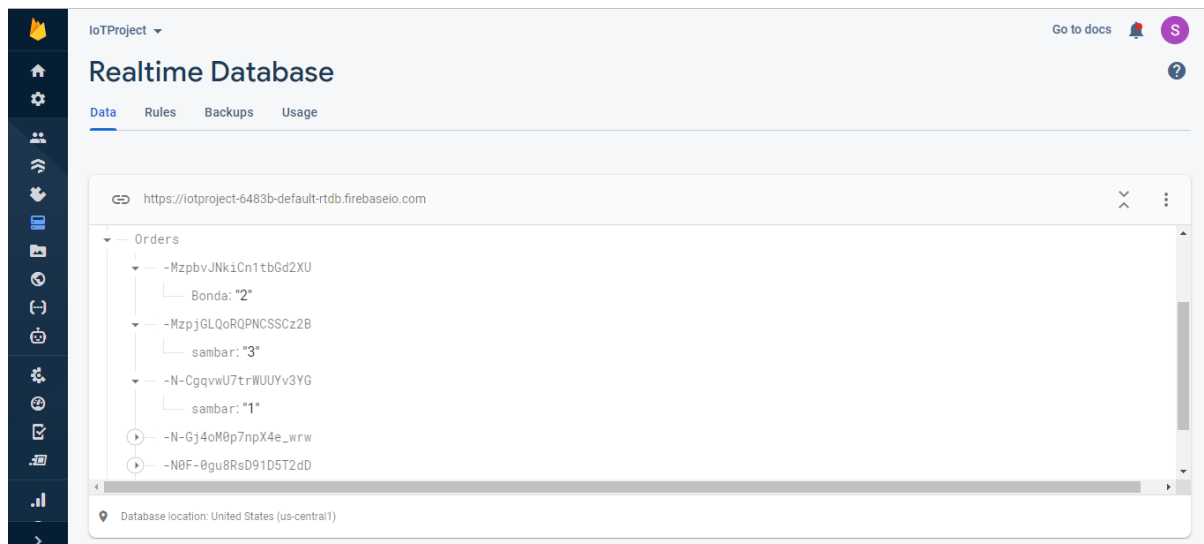




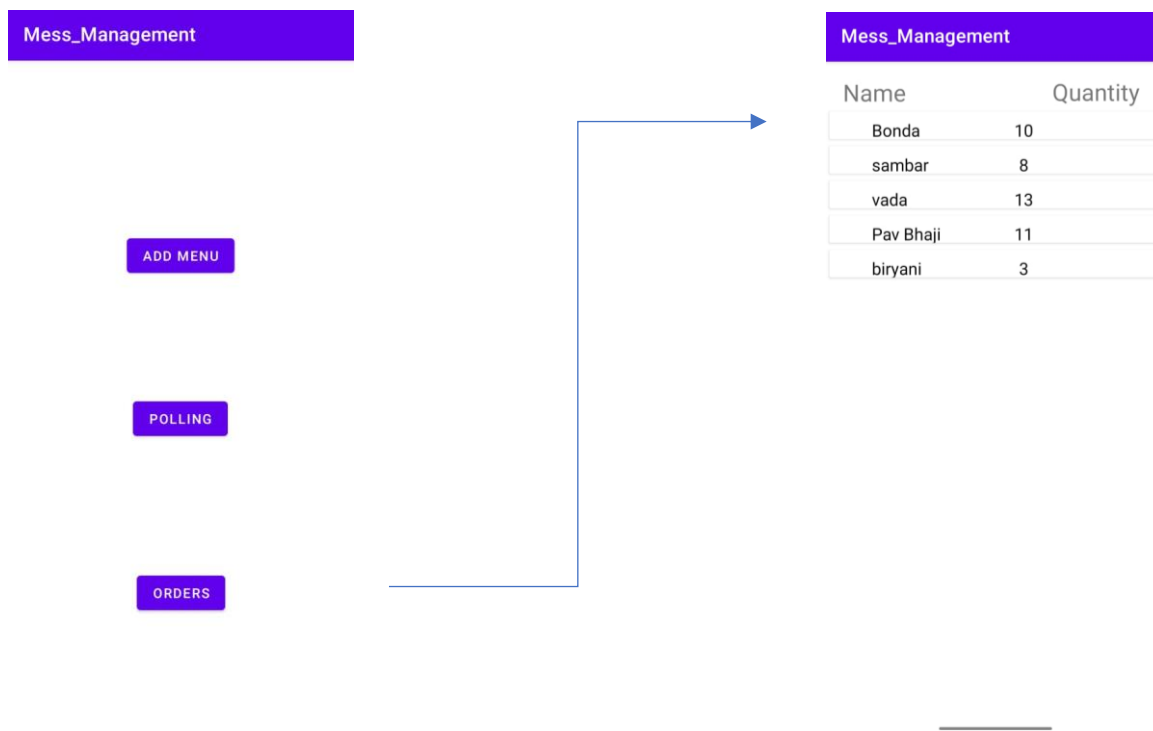
And now the student give the specific number for the menu items he/she is interested in:



As the students give their orders, the data will be updated on the firebase and the same will be reflected to the mess management app:



After certain time, when the mess admin checks the app, he can get the exact info about the quantity of each item to be eaten by students:



PROS OF THE PROJECT

1. As we can see, because of the polling system the staff can figure out the percentage of users/customers that would prefer to eat a specific food item over other items, hence, staff can freely adjust the amount of food items to prepare and ultimately reducing wastage of food.
2. The instalment of LED or the hardware part as a whole helps the staff side a lot in terms of handling and would make the process a lot easier to figure out the needs of the user and status of the booking or the polling system.

CONS OF THE PROJECT

1. If any of the switch stops working, it will stop the whole system of either booking or the polling respectively, hence, hindering its flow and ultimately causing inconvenience to the users and the staff.
2. All the hardware parts are to be handled with care because if they are damaged it would stop the whole application's working hence causing inconvenience.

FUTURE WORK

We can also add notification system in the application to know when the polling is going to start or has started or has ended so that the users can log in and poll their needs accordingly and conveniently. Same for booking part too. We will have to set time for the notification to pop up as for when the polling or booking is going to start or end so that the users can poll or book.

CONCLUSION

We have successfully implemented our main goal motive for this project that is developing an app for booking mess food to accommodate for hostels and workplaces with least wastage of food at basic stage.

Also, we have implemented the project on hardware by using NodeMCU, switches and LED to display the status of booking and polling, hence, making it easier for the staff to carry out the work without any workload and making it convenient for them to figure out when to start polling or booking.

REFERENCES

- <https://firebase.google.com/docs/android/setup>
- <https://randomnerdtutorials.com/esp8266-nodemcu-firebase-realtime-database/>
- <https://developer.android.com/guide>