VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JnanaSangama, Belgaum-590014

An Internship Report Based On Software Part Of

"TEMPERATURE MONITORING ANDROID APP WITH GLOBAL REAL TIME DATA ACCESSIBILITY USING IoT"

Submitted in Partial fulfillment of the Requirements for the VII Semester of the Degree of

Bachelor of Engineering

In

Computer Science & Engineering

By

KRUPA D

(1CE17CS052)

Under the Guidance of

Mrs. DEEPIKA R

Asst. Professor, Dept. of CSE

CITY ENGINEERING COLLEGE

Doddakallasandra, Kanakapura Road,

Bengaluru-560061

CITY ENGINEERING COLLEGE

Doddakallasandra, Kanakapura Road, Bengaluru-560061

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the Internship Report based on the Software Part of work entitled **"TEMPERATURE MONITORING ANDROID APP WITH GLOBAL REAL TIME DATA ACCESSIBILITY USING IoT"** has been carried out by **KRUPA D (1CE17CS052),** bonafide student of City Engineering College in partial fulfilment for the award of **Bachelor of Engineering in Computer Science and Engineering.** of the Visveshvaraya Technological University, Belgaum during the year **2020-2021**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The Internship Report based on the Software Part has been approved as it satisfies the academic requirements in respect of project work prescribed for the said Degree.

21/12/2020

Mrs. Deepika R
Internal Guide, Dept. of CSE

Mr. Sreenivasa Setty
External guide, CTO
SST TECHNOLOGIES
Bangalore

Mr. **B Vivekavardhana Reddy**
Head, Dept. of CSE

Dr. V. S Rama Murthy
Principal

External Viva

Name of the Examiners	Signature with Date
1.	
2.	

ABSTRACT

"TEMPERATURE MONITORING ANDROID APP WITH GLOBAL REAL TIME DATA ACCESSIBILITY USING IoT" is a Project which aims in Developing an Android Application which is built using MIT App Inventor, a real time database which is built using Firebase, Node MCU, Temperature Sensor and IoT Hardware Components. This system can be made use in many domains such as Digital temperature monitoring in laboratories, Food safety Compliances, Warehouse and Inventory Management, Monitoring of different Equipment and many more domains as such.

This System mainly aims in Global data Accessibility which can be achieved by using Globally accessible Real time Database that is Firebase and the Temperature data is then retrieved using an Android Application using MIT Android Application Inventor. And an IoT system that is designed to collect data from surroundings and send it to the Real time Database.

ACKNOWLEDGEMENT

While presenting this Internship Report based on the Software Part of work on "**Temperature Monitoring Android App With Global Real Time Data Accessibility Using IoT**", I feel that it is my duty to acknowledge the help rendered to me by various persons.

Firstly I thank God for showering his blessings on me. I'm grateful to my institution **City Engineering College** for providing me a congenial atmosphere to carry out the project successfully.

I would like to express my heartfelt gratitude to **Dr. V S Ramamurthy,**
Principal, CEC, Bangalore, for extending his support.

I would also like to express my heartfelt gratitude to **Prof. Vivekavardhana Reddy**, HOD, Computer Science and Engineering whose guidance and support was truly invaluable.

I'm very grateful to my guide, **Mrs. Deepika R**, Asst. Prof., Department of Computer Science, for her able guidance and valuable advice at every stage of my project which helped me in the successful completion of my project.

I would also have indebted to my Parent and Friends for their continued moral and material support throughout the course of project and helping me in finalize the presentation.

My hearty thanks to all those who have contributed bits, bytes and words to accomplish this Project.

KRUPA D (1CE17CS052)

TABLE OF CONTENTS

line Spg

LIST OF FIGURES

Line Spacing : 1.5
Text : 12 Pts

Chapter 1

INTRODUCTION

IoT and Embedded System

The Internet of Things (IoT) refers to the ever growing network of technologies connecting and communicating via the internet to send and receive data in the absence of human-to-human or human-to-computer interaction. IoT consists of a network of smart devices, sensors, and actuators interconnecting with each other over the internet. IoT is rapidly evolving throughout the embedded industry. It is projected that there will be about 50 billion IoT devices connected to the internet by 2030.

An embedded system is a microprocessor-based computer hardware system with software that is designed to perform a dedicated function, either as an independent system or as a part of a large system. At the core is an integrated circuit designed to carry out computation for real-time operations.

Cisco Packet Tracer

Packet Tracer is a cross-platform visual simulation tool designed by Cisco Systems that allows users to create network topologies and imitate modern computer networks. The software allows users to simulate the configuration of Cisco routers and switches using a simulated command line interface. Packet Tracer makes use of a drag and drop user interface, allowing users to add and remove simulated network devices as they see fit. Packet Tracer can also be run on Linux, Microsoft Windows, and macOS. Similar Android and iOS apps are also available.

IoT Hardware Components

Internet of Things (IoT) devices enable formerly unimaginable levels of remote monitoring and control. From locomotives to baby monitors, home appliance controls, wearables and interactive lighting, new applications are sparking the imaginations of designers. This article reviews the anatomy of wireless sensor nodes, explains how they usually function, and provides guidance regarding important considerations for selecting components for these types of applications. Whether you're designing a wearable device, an interactive lighting system, or even a jet engine, the building blocks of an IoT device are remarkably similar. Here are the three main components for a wireless sensor node:

- Sensors – gather information about the environment and condition signals before transmitting to the microprocessor.
- Microcontrollers – process the signal from sensors, determine appropriate responses, manage power consumption and local memory.
- Communication – wireless chips, radio modules and protocols needed to transmit the information between devices and to the cloud.

The components in an IoT node will vary in sophistication, depending on the application. But the basic topology of a wireless sensor node always includes these elements.

Arduino and Arduino Programming

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amount of accessible knowledge that can be of great help to novices and experts alike.

Firebase

The Firebase Realtime Database lets you build rich, collaborative applications by allowing secure access to the database directly from client-side code. Data is persisted locally, and even while offline, realtime events continue to fire, giving the end user a responsive experience. When the device regains connection, the Realtime Database synchronizes the local data changes with the remote updates that occurred while the client was offline, merging any conflicts automatically.The Realtime Database provides a flexible, expression-based rules language, called Firebase Realtime Database Security Rules, to define how your data should be structured and when data can be read from or written to. When integrated with Firebase

Authentication, developers can define who has access to what data, and how they can access it. The Realtime Database is a NoSQL database and as such has different optimizations and functionality compared to a relational database. The Realtime Database API is designed to only allow operations that can be executed quickly. This enables you to build a great realtime experience that can serve millions of users without compromising on responsiveness. Because of this, it is important to think about how users need to access your data and then structure it accordingly.

MIT App Inventor

MIT App Inventor is a web application integrated development environment originally provided by Google, and now maintained by the Massachusetts Institute of Technology (MIT). It allows newcomers to computer programming to create application software(apps) for two operating systems (OS): Android, and iOS, which, as of 8 July 2019, is in final beta testing. It is free and open-source software released under dual licensing: a Creative Commons Attribution ShareAlike 3.0 Unported license, and an Apache License 2.0 for the source code.

It uses a graphical user interface (GUI) very similar to the programming languages Scratch (programming language) and the StarLogo, which allows users to drag and drop visual objects to create an application that can run on android devices, while a App-Inventor Companion (The program that allows the app to run and debug on) that works on iOS running devices are still under development. In creating App Inventor, Google drew upon significant prior research in educational computing, and work done within Google on online development environments.

1.1 Company Profile

The Company is involved in IoT and Embedded Systems based Services. Current Status of SST TECHNOLOGIES is – Active.

Product Sales & Distribution – provides personalized sales and distribution related Services, Creating experiences and Guiding Customers for better spend.

Technical Support & Services – provides extended support to the business for smooth functioning of their IT infrastructure through customized Engineering outsourcing services.

SST Technologies, is a leading provider of IoT and Embedded System Technologies and related Services. They are known for our Industry Knowledge, worker's quality, compliance, customer-centric approach and flexibility, and Today, SST Technologies has evolved as a premier provider of manpower for their Junior/middle & Amp; executive talent management Staffing needs.

- Vision : We believe that SST Technologies is the platform for all Engineering Graduates, where we transform their knowledge into SKILLS, where we bridge the gap between classrooms to work stations and where our clients meet their Expectations.
- Mission : @ SST Technologies, build the best IoT and Embedded System based professionals for IT industry. We have reinvented the ways of Technical Hiring and On Job Training for the aspiring engineers towards reshaping them into stronger a team player.

1.2 About The Company

Recognize the importance of implementing the right solution for your Domain. We offer a wide range of services to build a solution that is right for your Domain:

Every Domain, no matter the size, needs advice and support. We have several years of Technical Experience and have accumulated a wealth of IT Infrastructure knowledge. Our Services helps you to establish your requirements. We will be with you every step of the way, from product selection through to configuration and Installation.

To stay Relavent to our Customers, it is important that we continuously demonstrates our 'Value Creating Ability' by :

1. Helping our Customers to choose the right Hardware.
2. Bring competitive commercials to the discussion table.
3. Remaining predictable in our execution.
4. Backing it with excellent post-sales services.

Since established in 2018, SST Technologies has become a pioneer in providing distinguished end-to-end IT Infrastructure solutions to its customers through our business functions maximizing customer engagement with personalized services. We believe that today more than ever, business are dependent on technology solutions.

Chapter 2

TRAINING CONTENT

"TEMPERATURE MONITORING ANDROID APP WITH GLOBAL REAL TIME DATA ACCESSIBILITY USING IOT" is an IoT based System, which provides users to monitor temperature from any part of the world. Users can directly download the APK file available or QR code available to download the Android Mobile Application and start retrieving the Real time data on the Smartphone.

This project is mainly aimed to provide Real time Temperature data in the fields of Digital temperature monitoring in laboratories, Food safety Compliances, Warehouse and Inventory Management, Monitoring of different Equipment and many more domains as such. Data cab be accessed globally as Firebase is used to store the Real time data in this project.

This entire System ensures the its Users will be guaranteed their desired needs in the particular fields. As it has Global data accessibility, it finds a vast Applications in all of the main Domains including Agricultural Domain.

2.1 IoT – Internet Of Things

Internet of Things (IoT) is a network of physical objects or people called "things" that are embedded with software, electronics, network, and sensors that allows these objects to collect and exchange data. The goal of IoT is to extend to internet connectivity from standard devices like computer, mobile, tablet to relatively dumb devices like a toaster. IoT makes virtually everything "smart," by improving aspects of our life with the power of data collection, AI algorithm, and networks

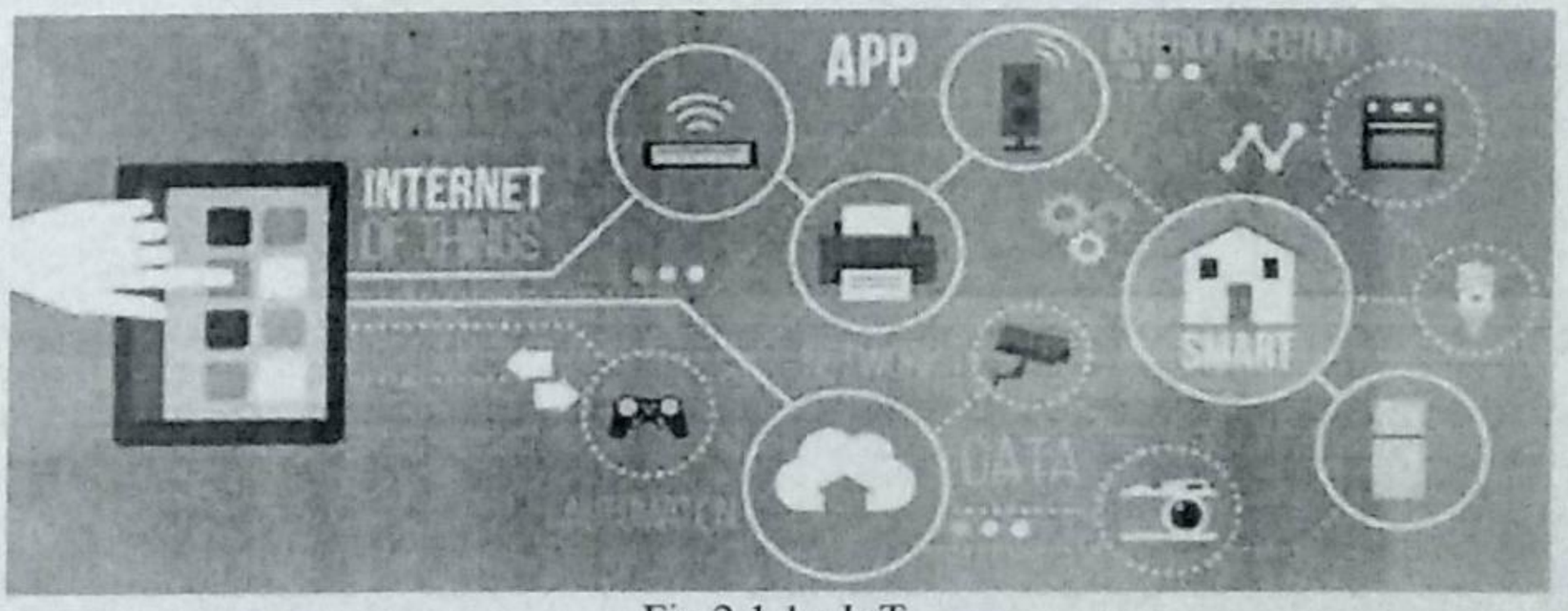

Fig 2.1.1 : IoT

Working :

1) Sensors/Devices: Sensors or devices are a key component that helps you to collect live data from the surrounding environment. All this data may have various levels of complexities. It could be a simple temperature monitoring sensor, or it may be in the form of the video feed.

2) Connectivity: All the collected data is sent to a cloud infrastructure. The sensors should be connected to the cloud using various mediums of communications. These communication mediums include mobile or satellite networks, Bluetooth, WI-FI, WAN, etc.

3) Data Processing: Once that data is collected, and it gets to the cloud, the software performs processing on the gathered data. This process can be just checking the temperature, reading on devices like AC or heaters. However, it can sometimes also be very complex like identifying objects, using computer vision on video.

4)User Interface: The information needs to be available to the end-user in some way which can be achieved by triggering alarms on their phones or sending them notification through email or text message. The user sometimes might need an interface which actively checks their IoT system.

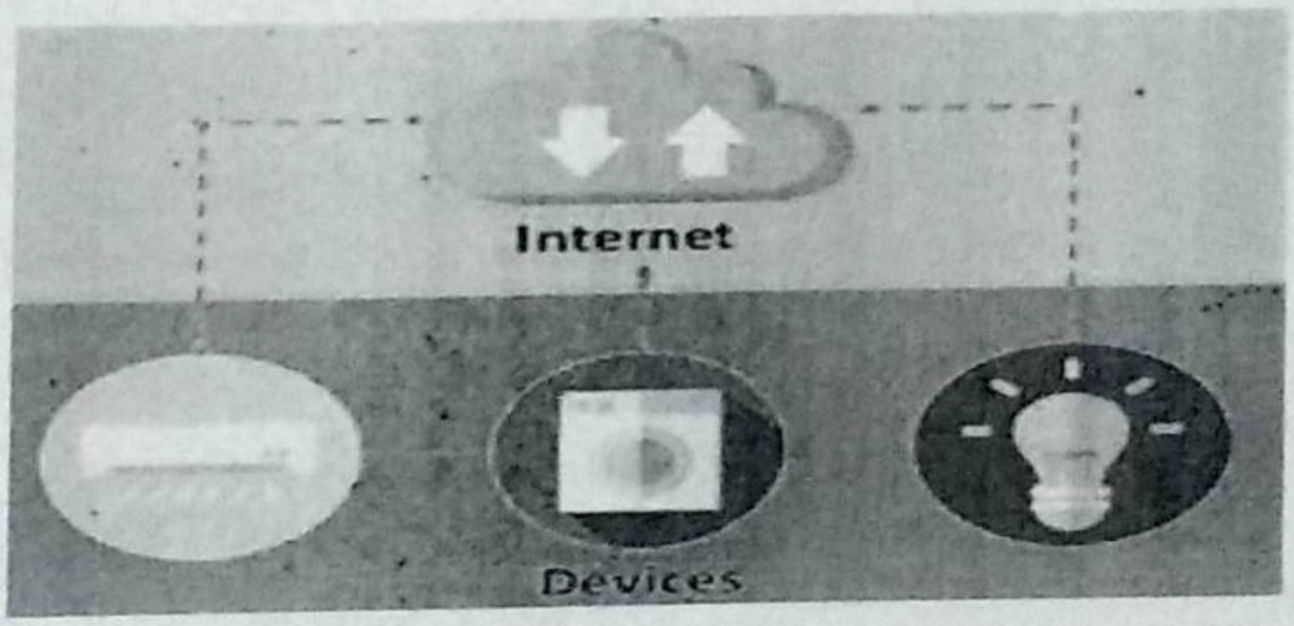

Fig 2.1.2 : WORKING OF IoT

Advantages Of IoT :

- **Technical Optimization:** IoT technology helps a lot in improving technologies and making them better. Example. with IoT, a manufacturer is able to collect data from various car sensors. The manufacturer analyzes them to improve its design and make them more efficient.

- **Improved Data Collection:** Traditional data collection has its limitations and its design for passive use. IoT facilitates immediate action on data.
- **Reduced Waste:** IoT offers real-time information leading to effective decision making & management of resources. For example, if a manufacturer finds an issue in multiple car engines, he can track the manufacturing plan of those engines and solves this issue with the manufacturing belt.
- **Improved Customer Engagement:** IoT allows you to improve customer experience by detecting problems and improving the process.

Disadvantages Of IoT:

- **Security:** IoT technology creates an ecosystem of connected devices. However, during this process, the system may offer little authentication control despite sufficient security measures.
- **Privacy:** The use of IoT, exposes a substantial amount of personal data, in extreme detail, without the user's active participation. This creates lots of privacy issues.
- **Flexibility:** There is a huge concern regarding the flexibility of an IoT system. It is mainly regarding integrating with another system as there are many diverse systems involved in the process.
- **Complexity:** The design of the IoT system is also quite complicated. Moreover, it's deployment and maintenance also not very easy.
- **Compliance:** IoT has its own set of rules and regulations. However, because of its complexity, the task of compliance is quite challenging.

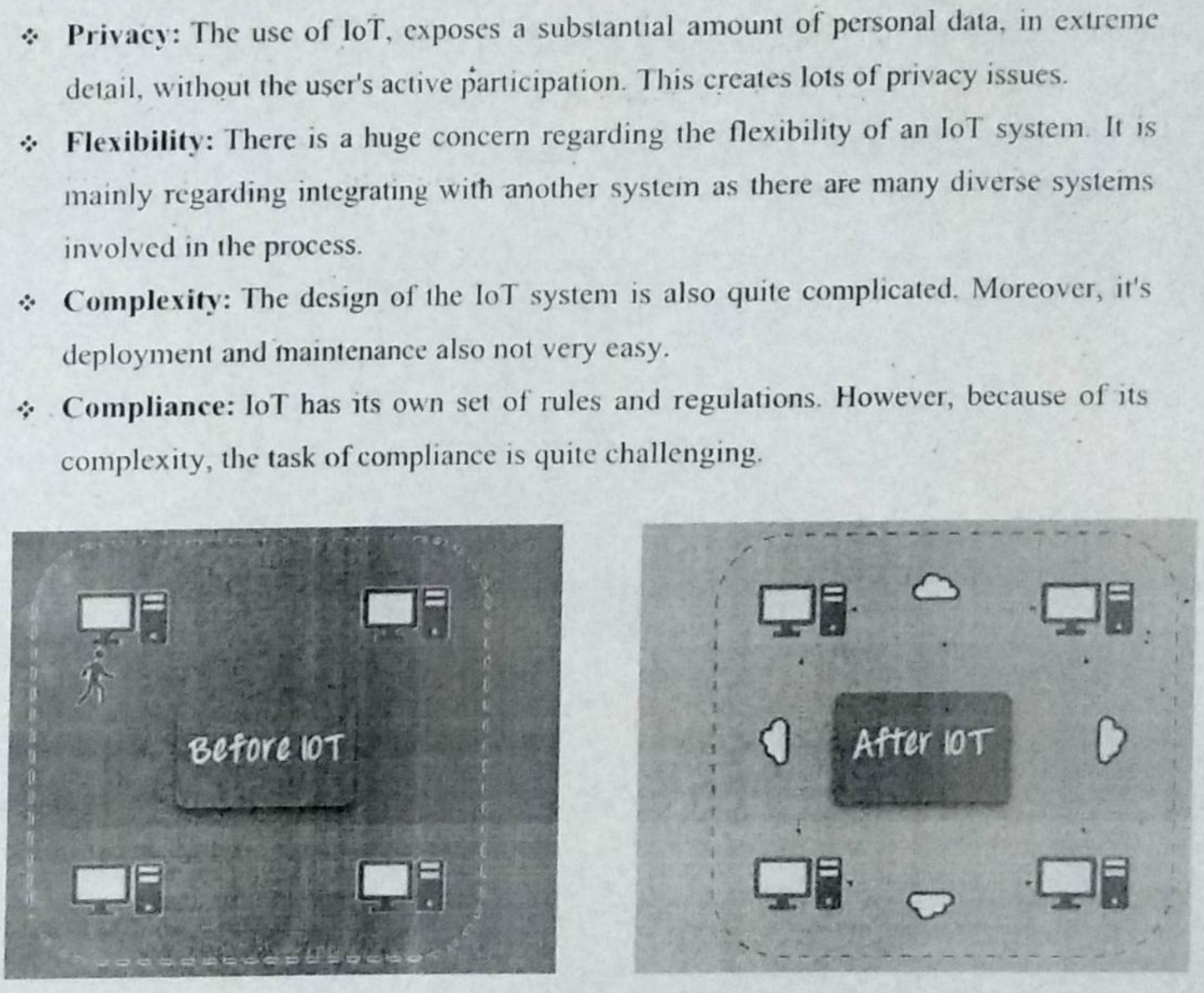

Fig 2.1.3 : ADVANTAGES AND DISADVANTAGES OF IoT

2.2 Embedded Systems

An **Embedded System** is a system that has software embedded into computer-hardware, which makes a system dedicated for a variety of application or specific part of an application or product or part of a larger system.

An embedded system can be a small independent system or a large combinational system. It is a microcontroller-based control system used to perform a specific task of operation.

An embedded system is a combination of three major components:

- **Hardware:** Hardware is physically used component that is physically connected with an embedded system. It comprises of microcontroller based integrated circuit, power supply, LCD display etc.
- **Application software:** Application software allows the user to perform varieties of application to be run on an embedded system by changing the code installed in an embedded system.
- **Real Time Operating system (RTOS):** RTOS supervises the way an embedded system work. It act as an interface between hardware and application software which supervises the application software and provide mechanism to let the processor run on the basis of scheduling for controlling the effect of latencies.

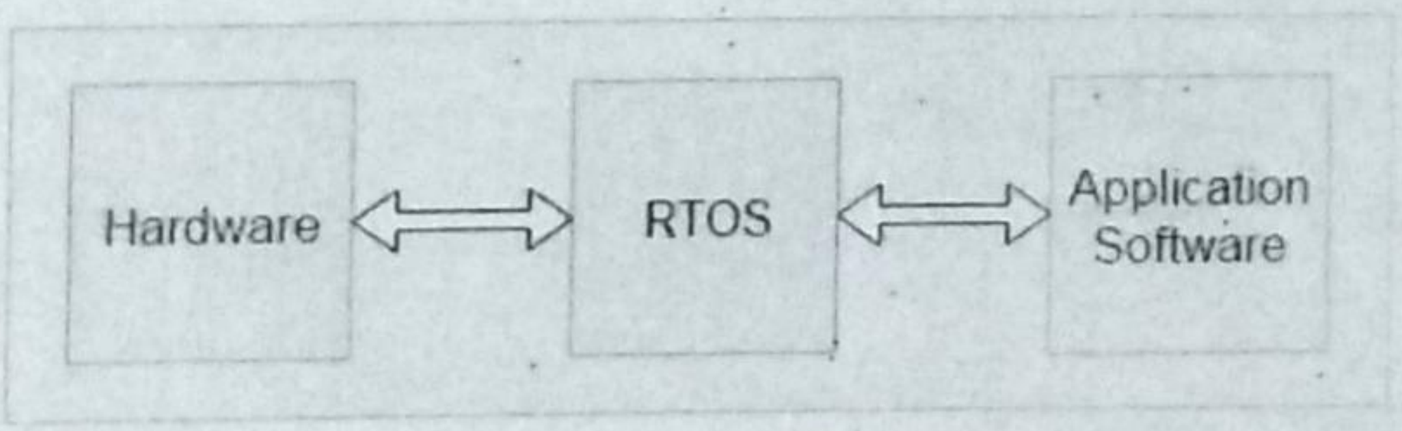

Fig 2.2.1 : EMBEDDED SYSTEMS

2.3 Cisco Packet Tracer

Cisco Packet Tracer is: "a powerful network simulation program that allows students to experiment with network behaviour and ask "what if" questions."

It further says: "The simulation-based learning environment helps students develop 21st century skills such as decision making, creative and critical thinking, and problem solving. Packet Tracer complements the Networking Academy curricula, allowing instructors to easily teach and demonstrate complex technical concepts and networking systems design."

Nothing could have defined that better. The biggest problem in learning networking is the need for multiple devices that from a network. Cisco Packet Tracer removes that hurdle altogether and that too simply. Although there are other network simulation programs (say GNS3), they need device images. Packet tracer has them inbuilt.

This software is available for free from Cisco's website provided you are a registered Networking Academy student, alumni, instructor, or administrator.

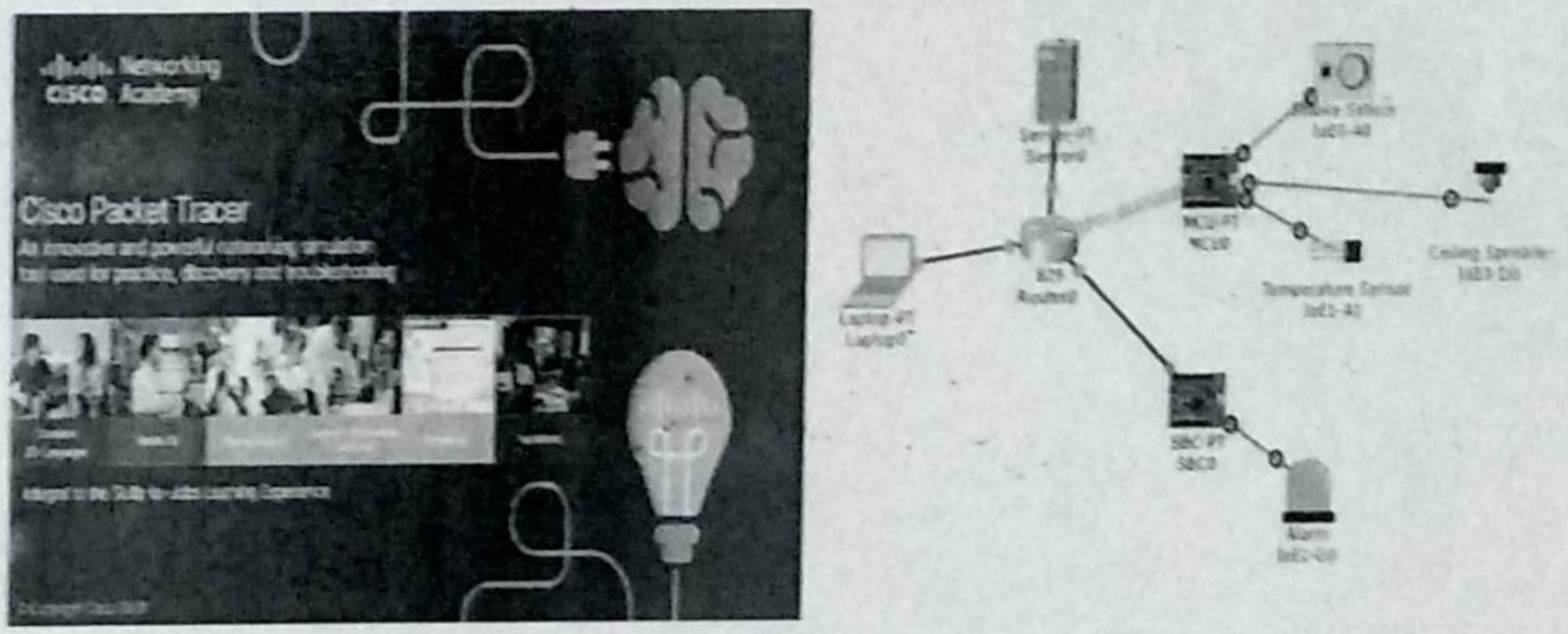

Fig 2.3.1 : CISCO PACKET TRACER

2.4 IoT Hardware Components

The set of devices that respond and have the capabilities to capture data, follow the instructions can be considered as the IoT Hardware. The following fall into such categories where they not only collect data but also respond to instructions based on the processed data.

Chips : This is much a broader classification that contains all the electrical and electronic appliances such as microcontrollers, chips, integrated circuits, radio frequency systems, etc.

Sensors : Sensors, which are one of the base components of an IoT system, have three modules to it - Power Management modules, Sensing modules, and the Energy modules.

Actuators : These devices provide the motion to a data collection system such as the solenoids, comb drives etc to fetch details based on movements.

Standard devices : Standard devices constitute of the generally used devices such as Tablets, Smartphones, Switches, Routers and etc. Each of these devices have their own set of settings that allow them to collect data.

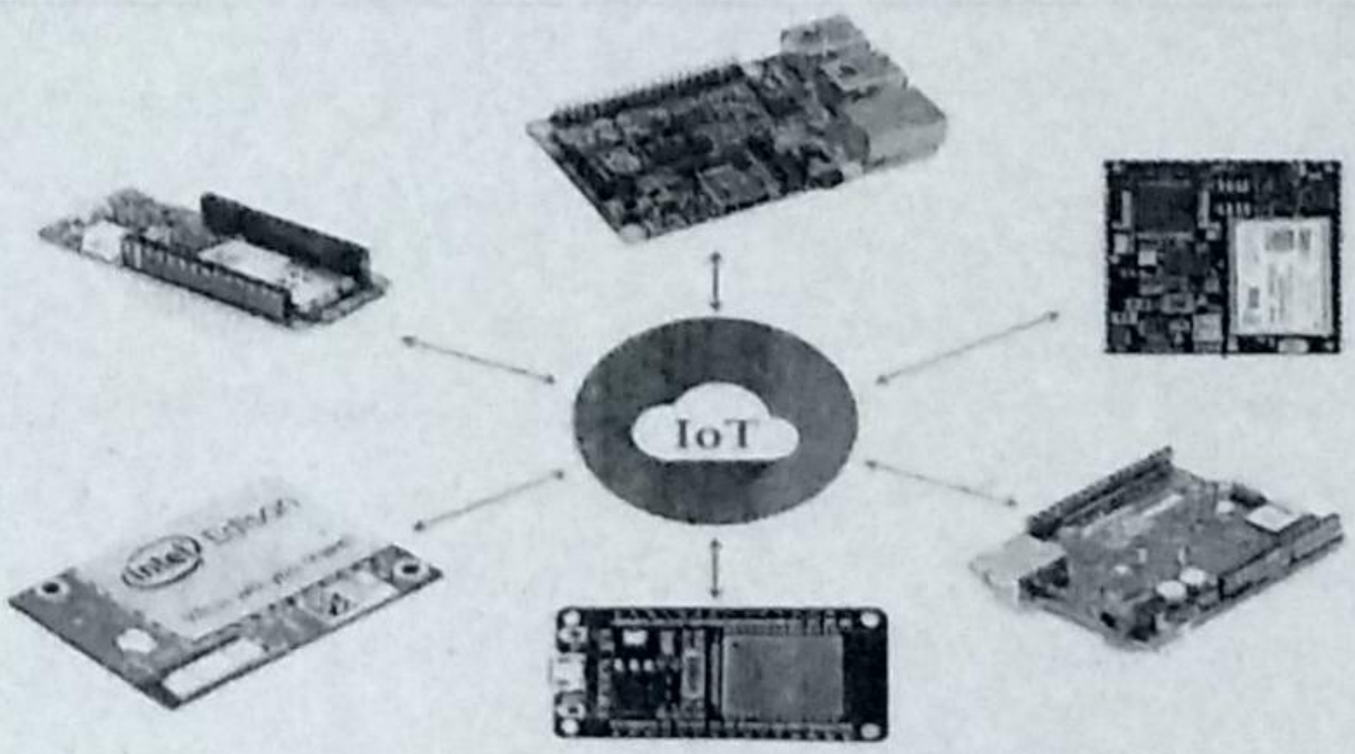

Fig 2.4.1 : IoT HARDWARE COMPONENTS

2.5 Arduino Programming

Arduino programs are written in the Arduino Integrated Development Environment (IDE). Arduino IDE is a special software running on your system that allows you to write sketches for different Arduino boards. The Arduino programming language is based on a very simple hardware programming language called processing, which is similar to the C language. After the sketch is written in the Arduino IDE, it should be uploaded on the Arduino board for execution.

The first step in programming the Arduino board is downloading and installing the Arduino IDE. The open source Arduino IDE runs on Windows, Mac OS X, and Linux.

The structure of Arduino program is pretty simple. Arduino programs have a minimum of 2 blocks,

Preparation & Execution

Each block has a set of statements enclosed in curly braces:

void setup( )

```
{
statement-1;
.
statement-n;
}
void loop ( )
{
statement-1;
.
statement-n;
}
```

Here, setup ( ) is the preparation block and loop ( ) is an execution block.

The setup function is the first to execute when the program is executed, and this function is called only once. The setup function is used to initialize the pin modes and start serial communication. This function has to be included even if there are no statements to execute.

```
void setup ( )
{
pinMode (pin-number, OUTPUT); // set the 'pin-number' as output
pinMode (pin-number, INPUT); // set the 'pin-number' as output
}
```

After the setup ( ) function is executed, the execution block runs next. The execution block hosts statements like reading inputs, triggering outputs, checking conditions etc..

In the above example loop ( ) function is a part of execution block. As the name suggests, the loop( ) function executes the set of statements (enclosed in curly braces) repeatedly.

```
void loop ( )
{
digitalWrite (pin-number,HIGH); // turns ON the component connected to 'pin-number'
delay (1000); // wait for 1 sec
digitalWrite (pin-number, LOW); // turns OFF the component connected to 'pin-number'
delay (1000); // wait for 1sec
}
```

2.6 Firebase

Firebase is a backend platform for building Web, Android and IOS applications. It offers real time database, different APIs, multiple authentication types and hosting platform.

Prerequisites

We will need some JavaScript knowledge to be able to follow this tutorial. Knowledge about some backend platform is not necessary, but it could help you to understand the various Firebase concepts.

Firebase Features

- Real-time Database – Firebase supports JSON data and all users connected to it receive live updates after every change.
- Authentication – We can use anonymous, password or different social authentications.
- Hosting – The applications can be deployed over secured connection to Firebase servers.

Firebase Advantages

- It is simple and user friendly. No need for complicated configuration.
- The data is real-time, which means that every change will automatically update connected clients.
- Firebase offers simple control dashboard.
- There are a number of useful services to choose.

Firebase Limitations

- Firebase free plan is limited to 50 Connections and 100 MB of storage.

Steps To Create Firebase Account :

- Navigate to https://firebase.google.com/ and sign in in the top right hand of the page.
- After creating an account and signing in, create a project by clicking the "Add Project" button.

- Fill out the highlighted fields in the window that pops up. Name your project anything you want. Click the "Create Project" button.
- Once the project has been created, you'll be taken to the page below to configure various settings to enable push notifications in your white label apps for Android and iOS.

2.7 MIT App Inventor

MIT App Inventor is a free, cloud-based service that allows you to make your own mobile apps using a blocks based programming language. You access App Inventor using a web browser (Chrome, Firefox, Safari).

Steps To Create MIT App Inventor :

Step 1a : From the User Interface palette, drag and drop the Button component to Screen1 .

Step 1b : To give the button the image of the bee, in the Properties pane, under Image, click on the text "None..." and click "Upload File..." . A window will pop up to let you choose the image file. Click "Browse" and then navigate to the location of the codi.jpg file you downloaded earlier . Click the codi.jpg file, click "Open", and then click "OK".

Step2 : ChangetheButton's Text property : Delete "Text for Button1", leaving the Button's text property blank so that there is no writing over the bee image.

Step3 : From the User Interface palette, drag and drop the Label component to the Viewer, placing it below the picture of the bee. It will appear under your list of components as Label1.

Under the Properties pane, change the

- Text property of Label1 to read "Touch the Bee". You'll see the text change in the Designer and on your device.
- FontSize to 30.
- BackgroundColor of Label1 by clicking on the box.
- TextColor to any color you like.

Step 4. Under Palette, click on the Media drawer and drag out a Sound component and place it in the Viewer. Wherever you drop it, it will appear in the area at the bottom of the Viewer marked Non-visible components. Under the Media pane, Click Upload File. Browse to the

location of the mp3 file that you downloaded earlier and upload it to this project. Under the Properties pane, see that the Source property currently says None.... Click the word None... to change the Sound1 component's Source to Bee-Sound.mp3.

Programming with the Blocks Editor

To start programming the behavior of the app, you need to go to the Blocks Editor. Click the Blocks button in the upper right of your screen to go to the Blocks Editor. Once you have the Blocks Editor in front of you, continue to the next step to start programming your app with blocks.

Packaging your app

While your device (emulator or phone/tablet) has been connected to App Inventor, your app has been running in real time on your device. If you disconnect the emulator/device from the Blocks Editor, the app will vanish. You can always make it return by reconnecting the device. To have an app running without being connected to App Inventor, you must "package" the app to produce an application package (apk file).

To "package" the app to install on your device or to send to someone else, click the Build tab at the top of the screen. Under Build, there are two options available for you to choose from:

1. App (provide QR code) : You can generate a Barcode (a QR Code), which you can use to install the app on a mobile device that has a camera, with the aid of a barcode scanner, like the ZXing barcode scanner (freely available in Google Play).

2. App (save to my computer) : You can download the app to your computer as an apk file, which you can distribute and share as you like by manually installing it on other devices.

Chapter 3

IMPLEMENTATION

3.1 HARDWARE AND SOFTWARE INTEGERATION

After completing the implementation of hardware and software we integrate both of it, as follows:

3.1.1 CONNECTING FIREBASE WITH MIT APP INVENTOR

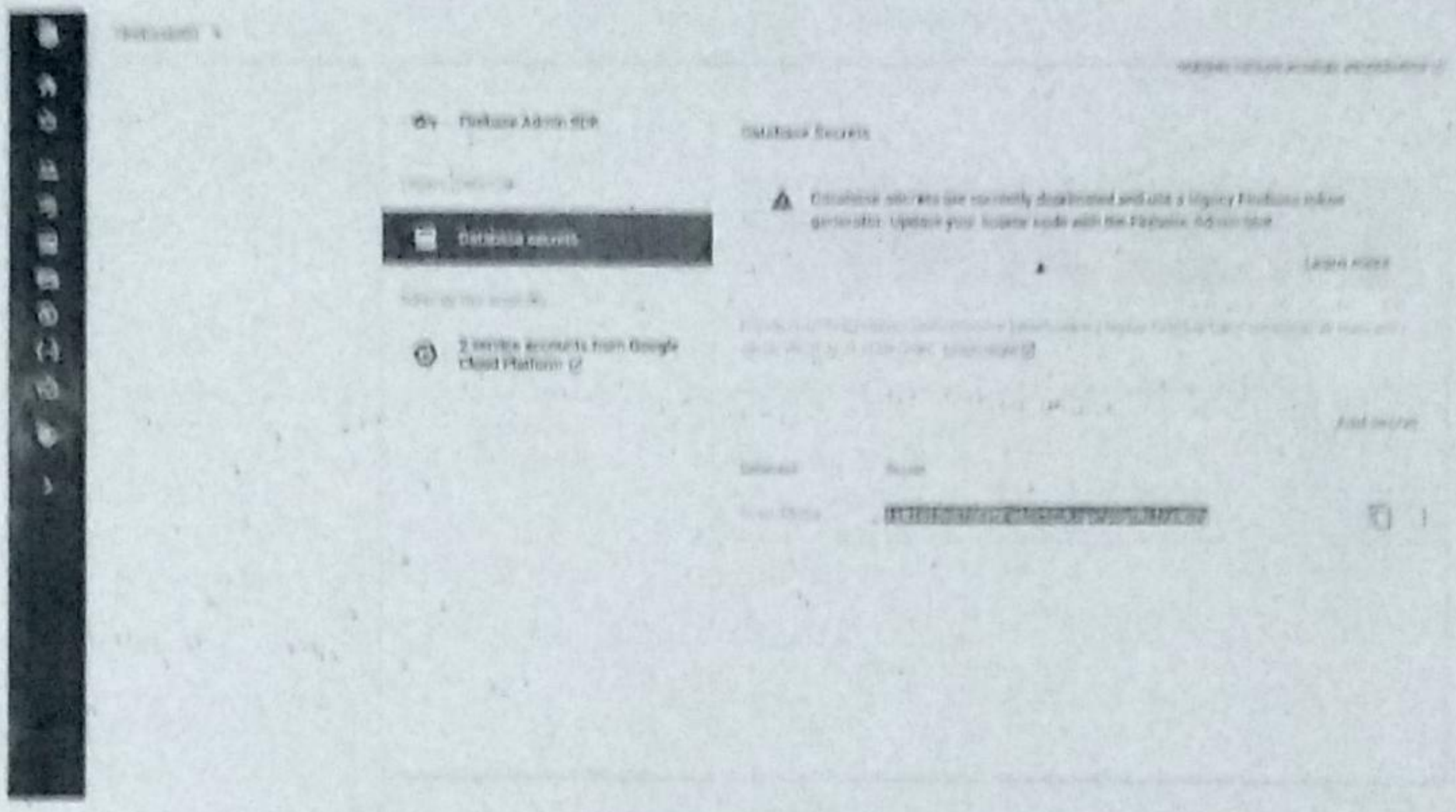

Fig 3.1 Firebase: copying the firebase token

Step 1: Go to firebase, open the settings then select project settings, go to service accounts where u can see database secrets select it and copy the token.

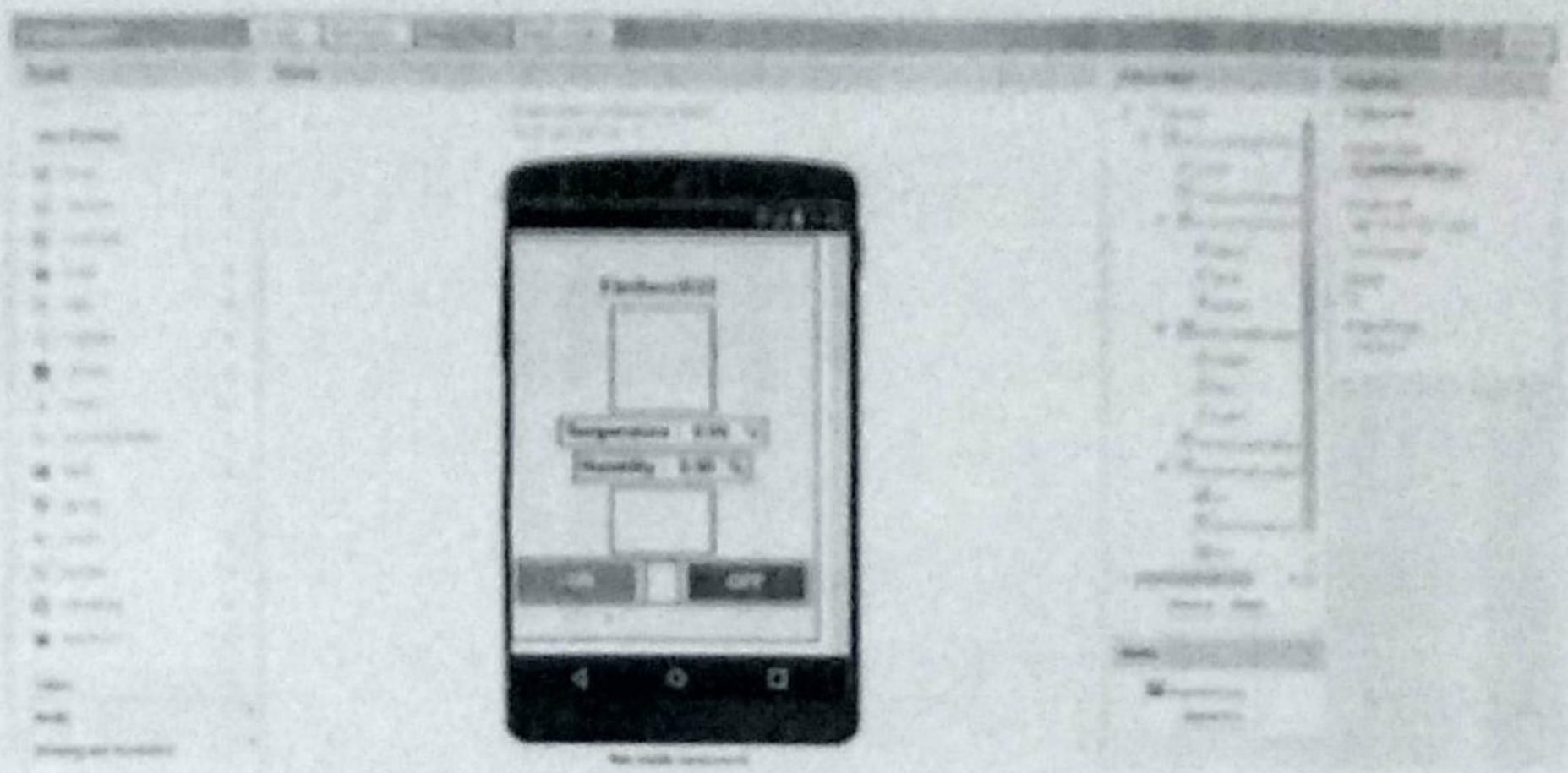

Fig 3.2 MIT App inventor: paste the token copied from firebase.

Step 2: now open the MIT app inventor, select the database where you can find firebase token in properties, paste the token copied from the firebase.

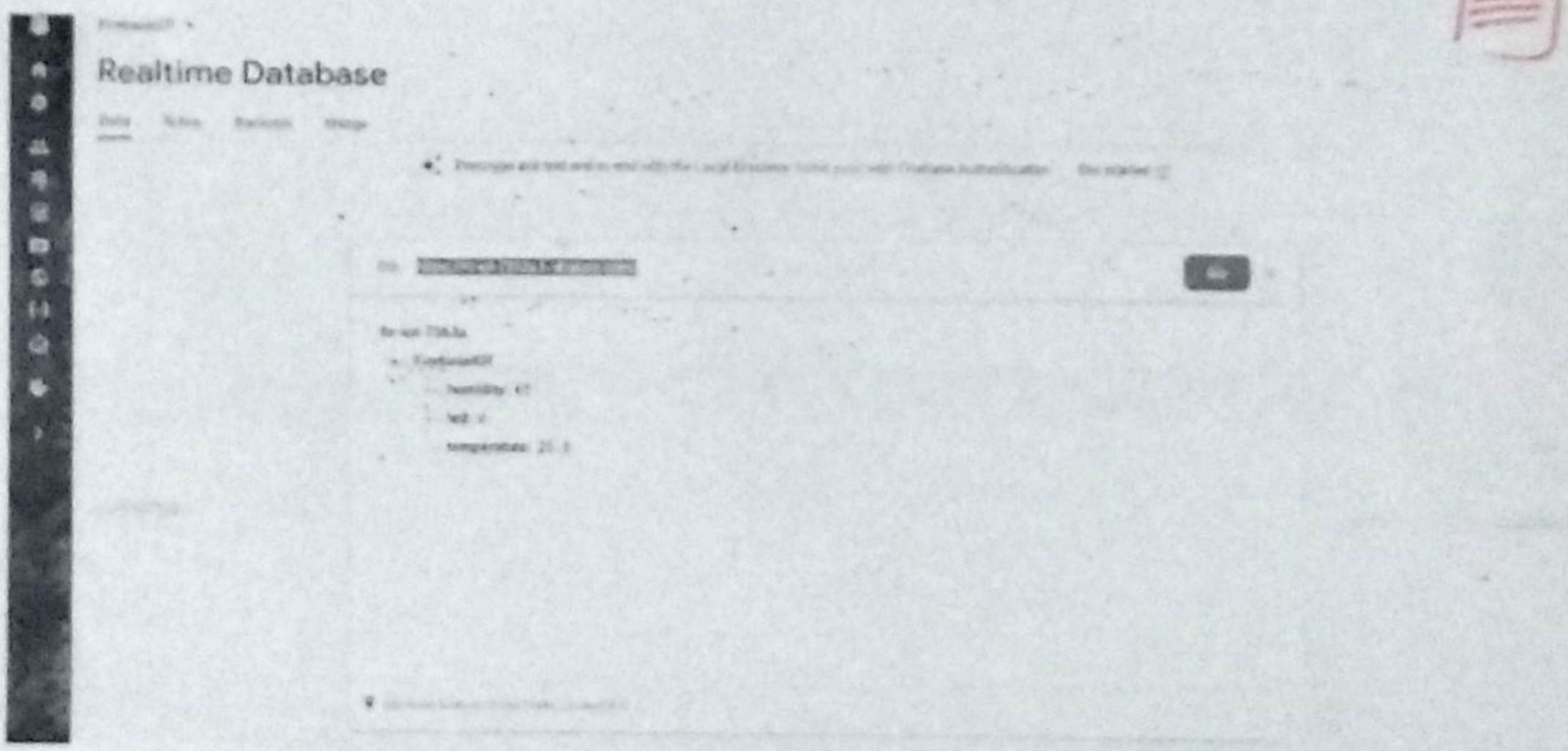

Fig 3.3 Firebase: go to Realtime database to copy the URL.

Step3: from Realtime database copy the URL, to paste it to MIT App inventor.

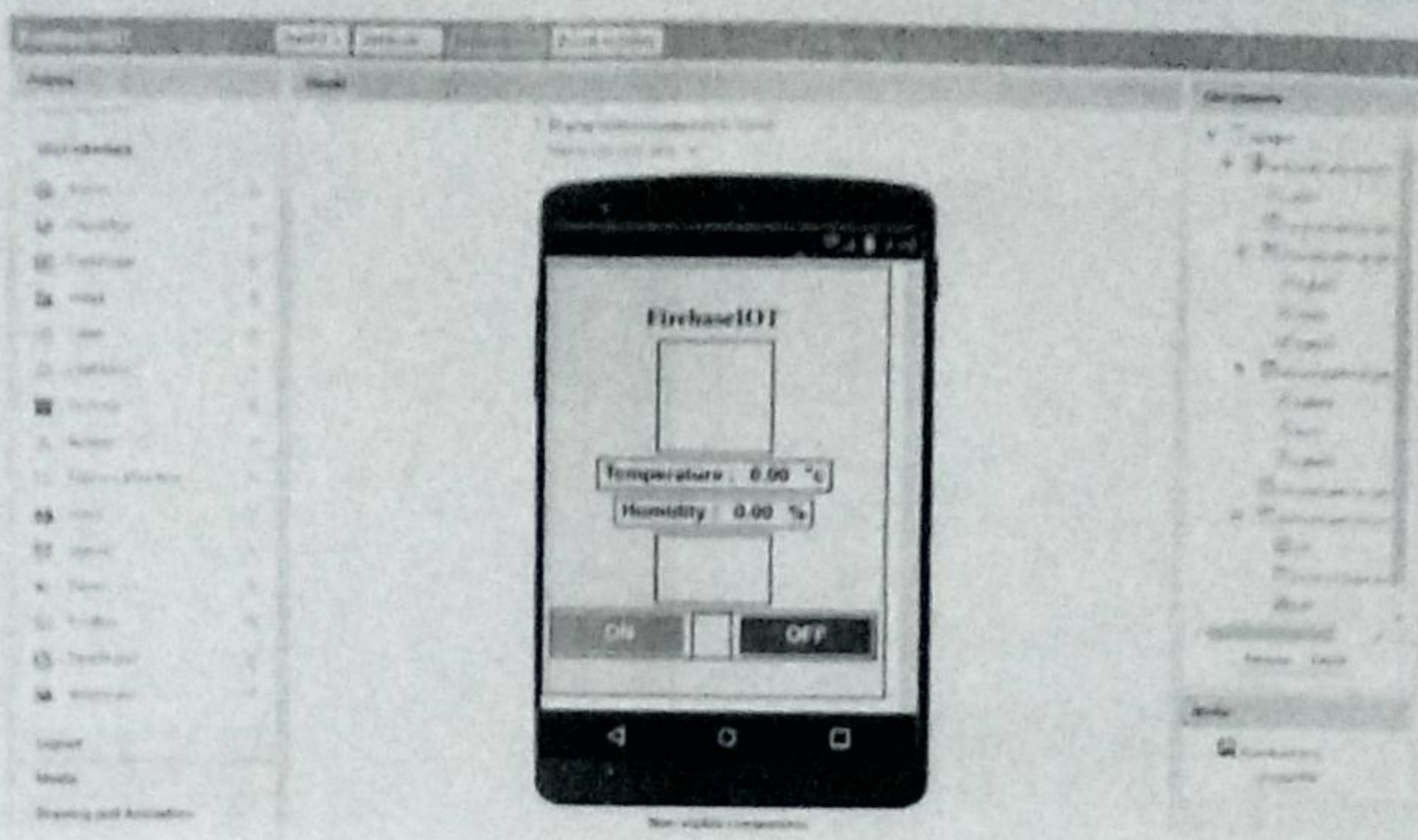

Fig 3.4 MIT App inventor: paste the URL.

Step4: after copying the URL from firebase we must paste the URL in MIT app inventor, go to properties paste the URL on firebase URL.

This is how we connect firebase to MIT App inventor.

3.1.2 CONNECTING FIREBASE TO ARDUINO

Step 1: first copy the token and URL from the firebase, as show above.

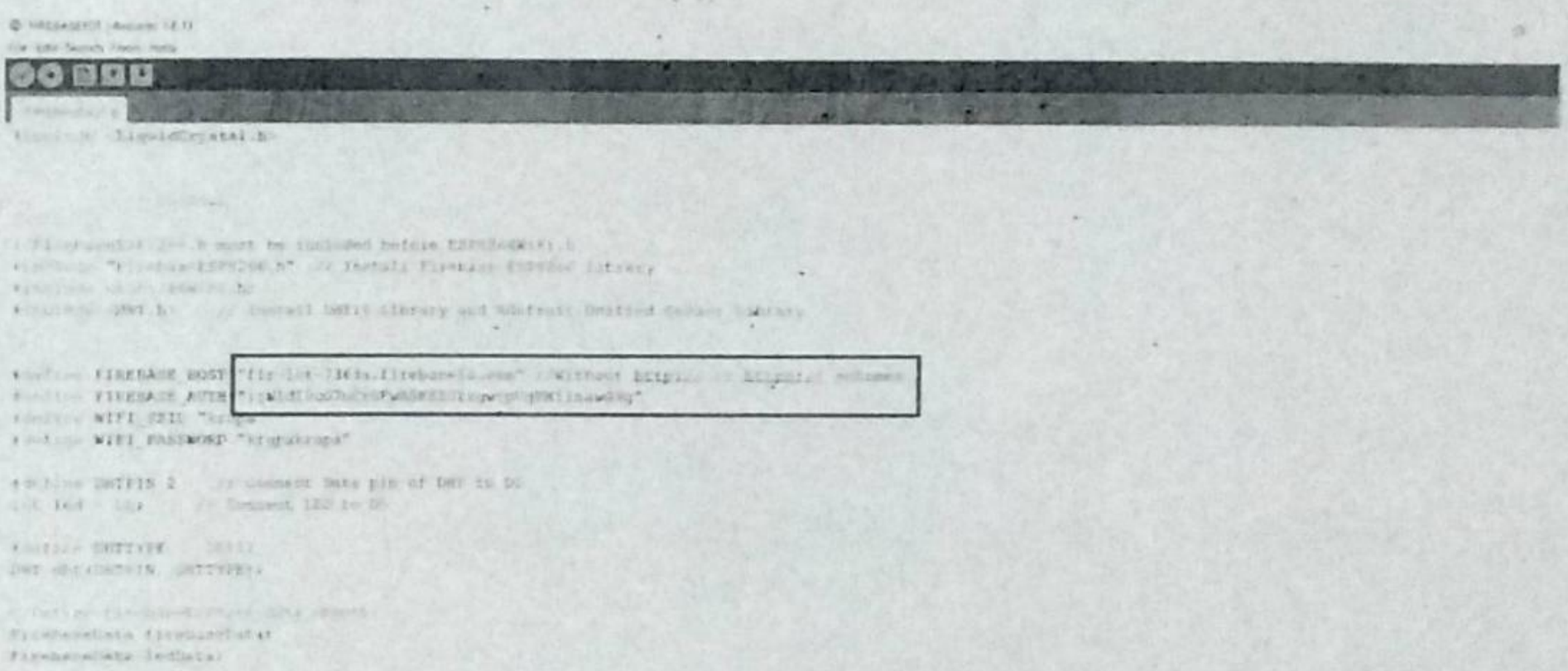

Fig 3.5 Arduino: pasting URL and token in code.

Step 2: paste the URL and token in the code, where URL to firebase_host and token to firebase_AUTH as shown in the above fig. this completes the connection between firebase and Arduino.

Step 3: type your wifi_ssid and password in code, where ssid to wifi_ssid and password to wifi_password.

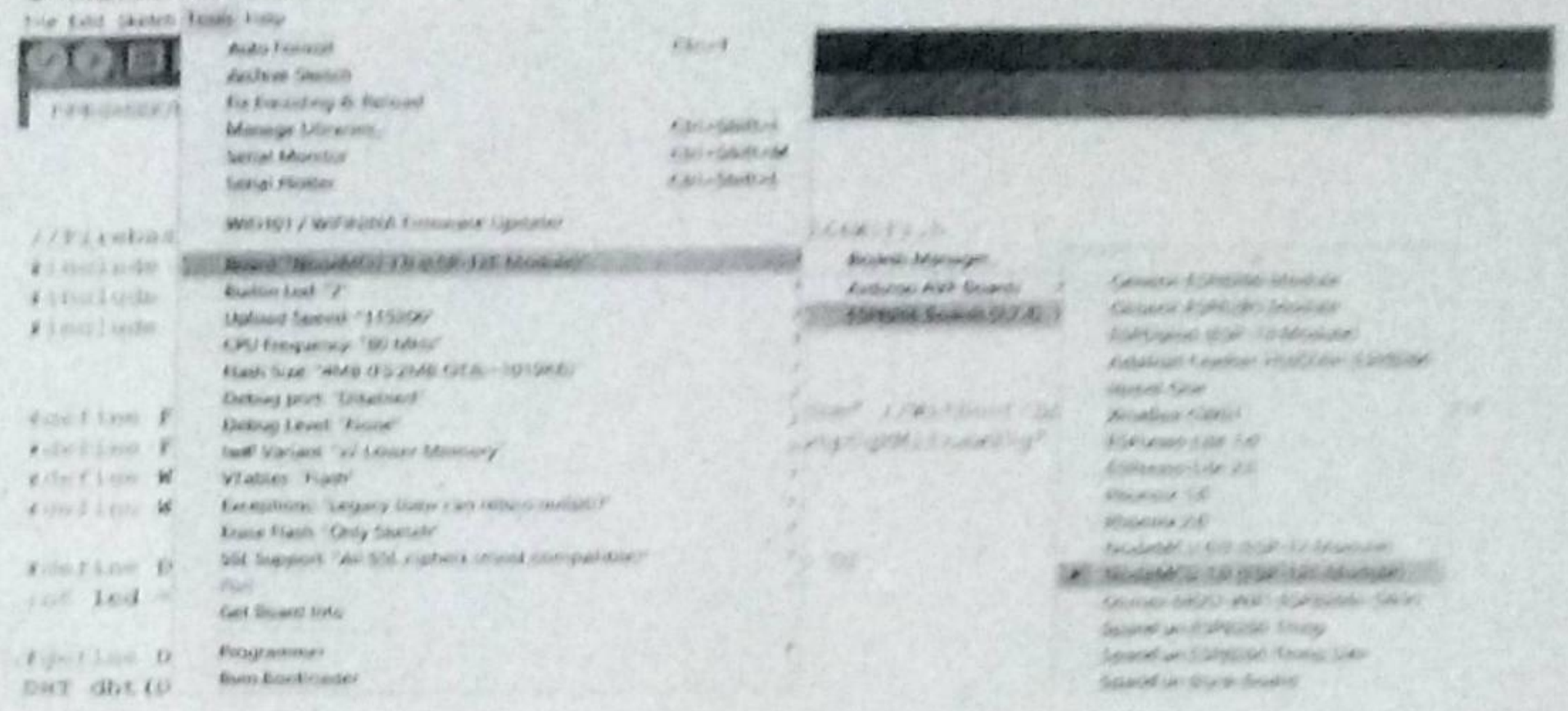

Fig 3.6 Selection of board

Step 4: next step is to select board from tools, once we finish typing the code save it and then select tools where we have to select board, then select ESP8266 Board (2.7.4) wherein we find NodeMCU(ESP-12E Module) select it.

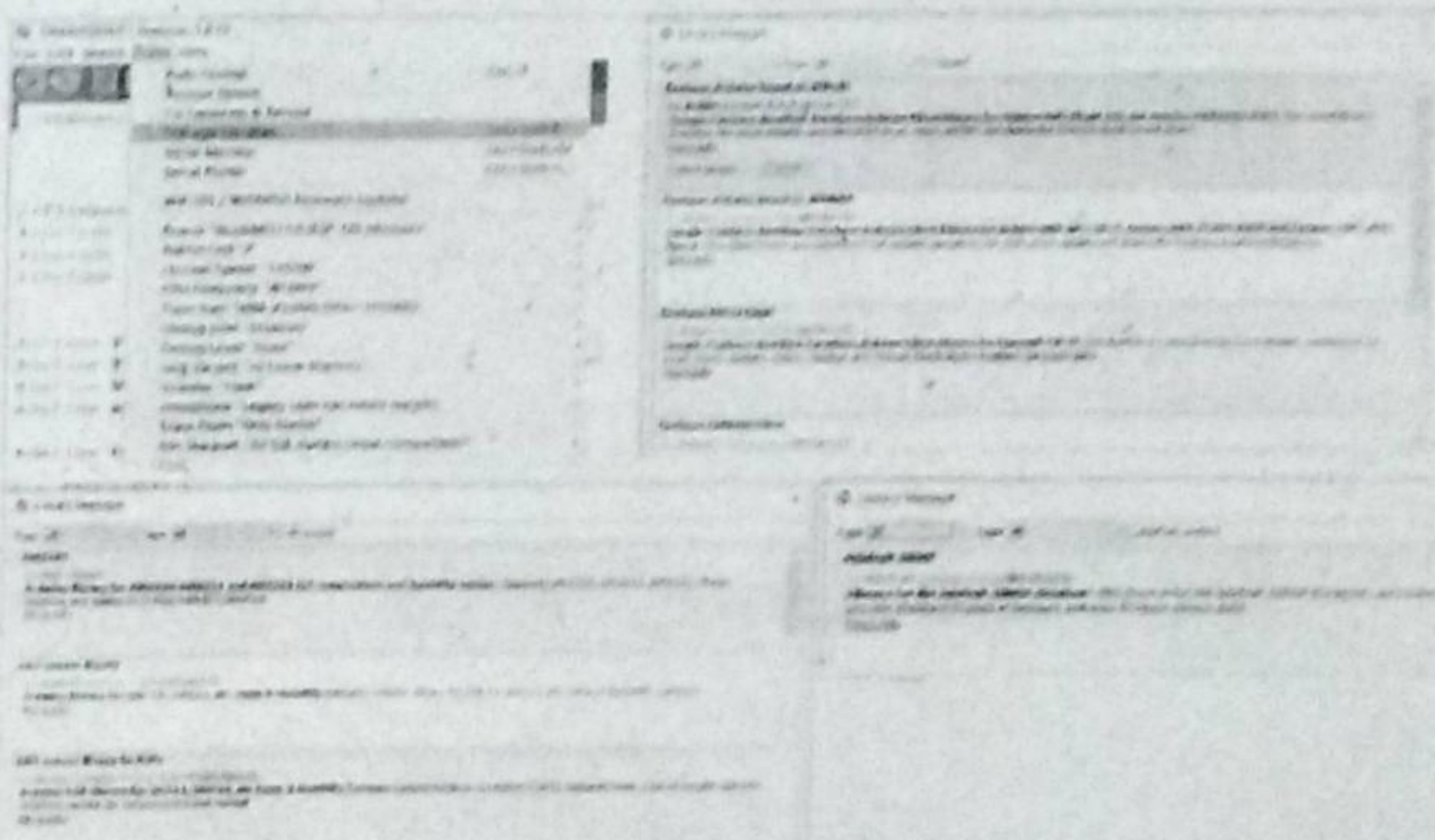

Fig 3.7 Adding libraires.

Step 5: after selection of board, lets add libraries, *again go to tools where we can find* manage libraries select it, after which u get a page where we *have to install the following* libraries:

- Firebase ESP8266 Library
- DHT Sensor Library
- Adafruit Unified Sensor Library

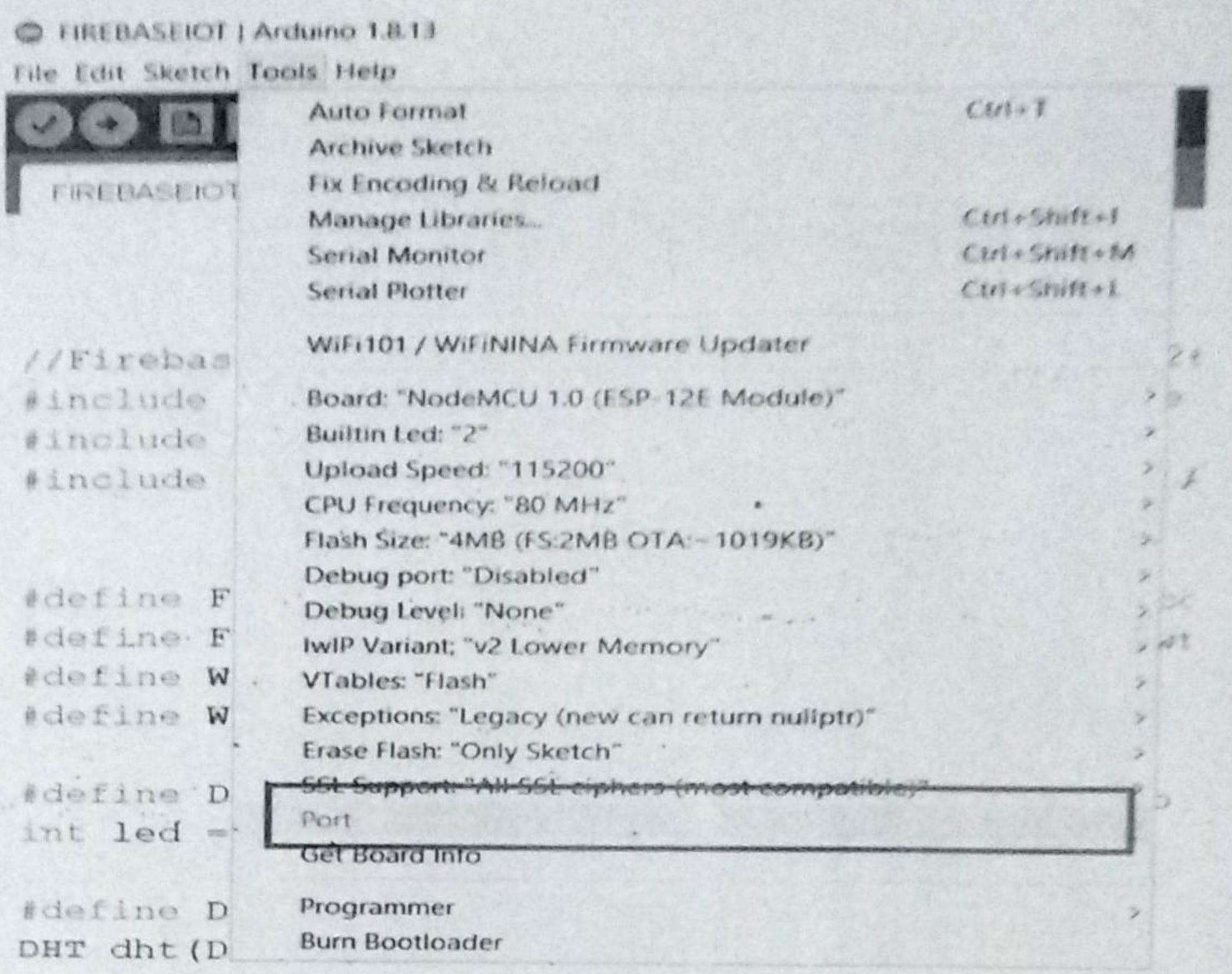

Fig 3.8 Selection of port.

Step 6: after installation of required libraires select port, now again go to tools select the port and we will find options over there as comb3, comb5... etc. select one of those.

3.1.3 CONNECTION OF ARDUINO IDE AND HARDWARE

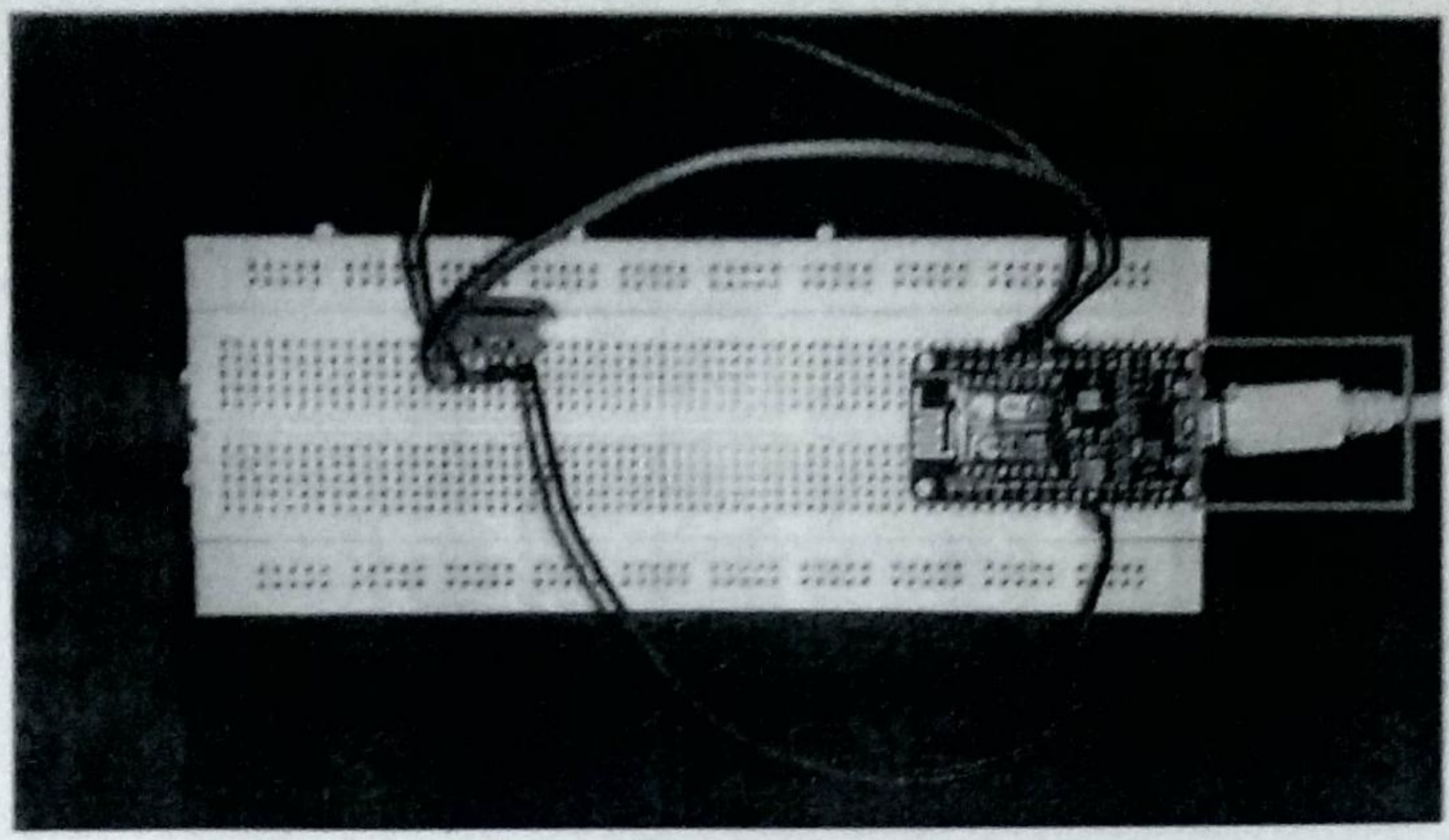

Fig 3.9 Hardware circuit.

As we can see USB cable in the above figure, this cable connects the Arduino IDE to hardware, where one end of USB cable is connected to PC and the other end to NodeMCU.

Chapter 4

LEARNING OUTCOMES

- Developing a technical artifact requiring new technical skills.
- Using profession specific terminology appropriately.
- Effectively utilizing a new software tool to complete a task.
- Maintaining and troubleshooting Technology.
- Analyzing or visualizing data to create information.
- Writing requirements documentations.
- Selecting appropriate technologies.
- Acquiring and evaluating information.
- Performing effective and informative user testing.
- Identifying and creating appropriate test cases for systems.
- Demonstrating understanding of professional customs and practices.
- Organizing and maintaining information.
- Applying knowledge to the task.
- Improving problem-solving and critical thinking skills.
- Monitoring and correcting performance.
- Exercising leadership.
- Behaving professionally.

Chapter-5

SNAPSHOTS

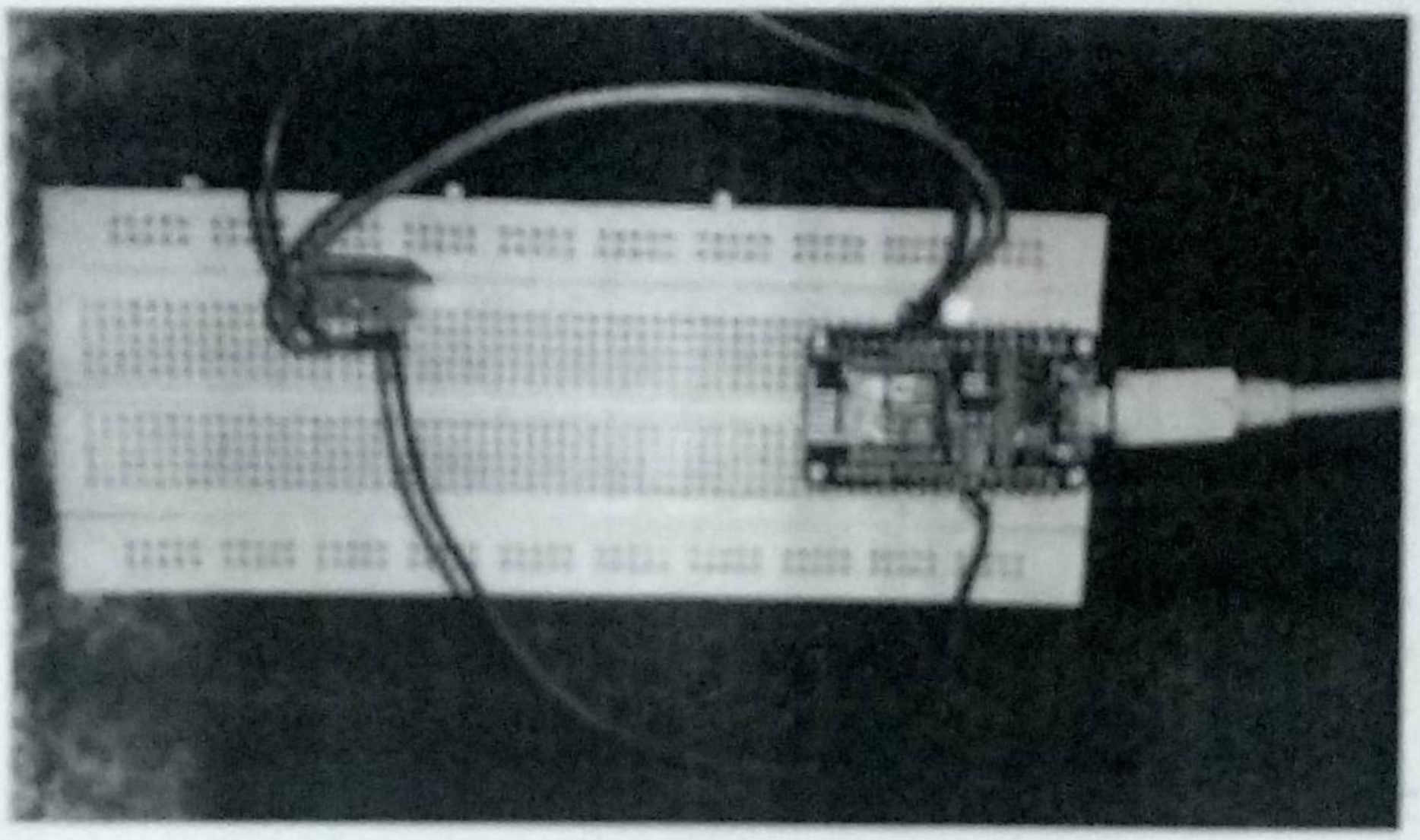

Fig 6.1 : FINAL OUTPUT WITH LED ON

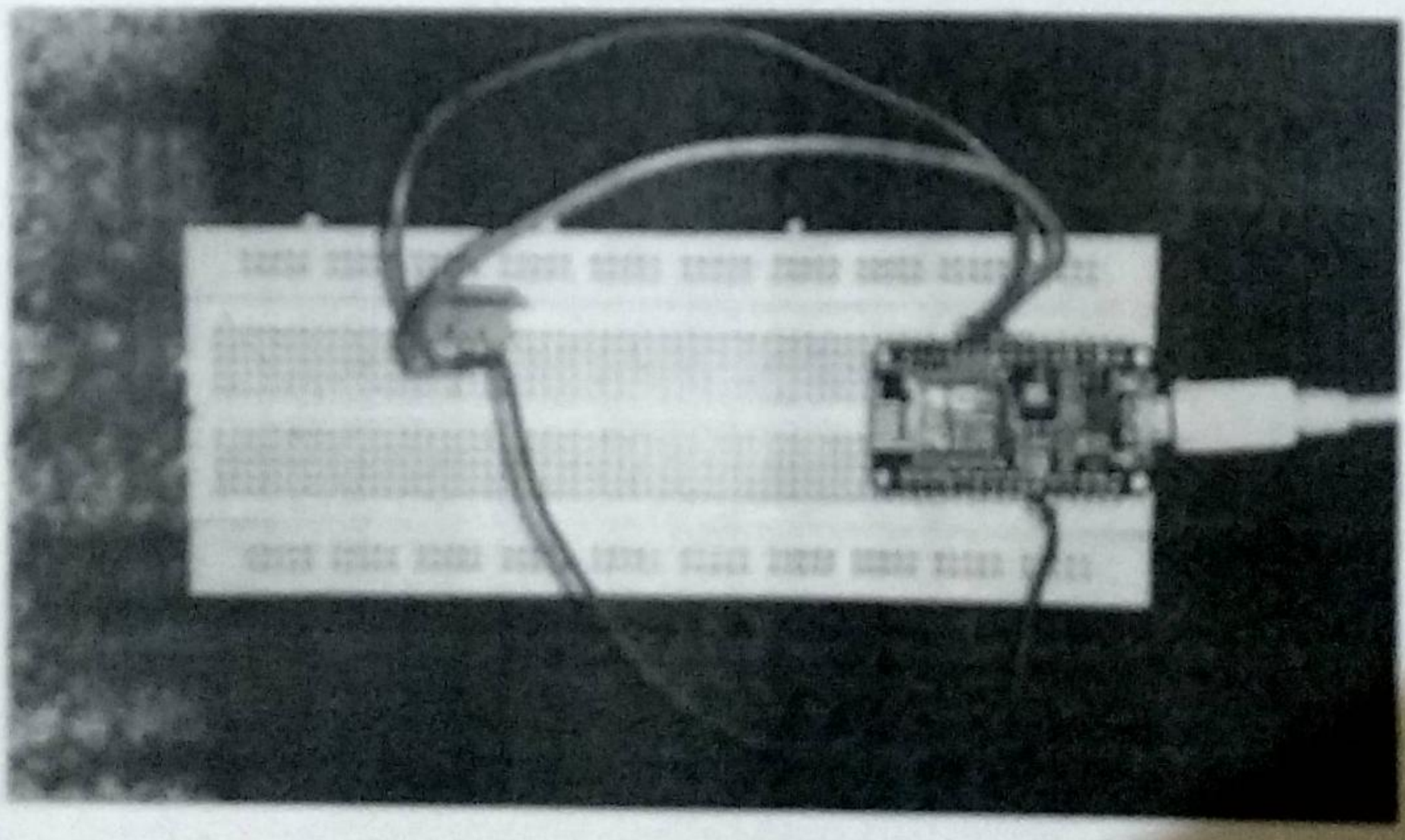

Fig 6.2 : FINAL OUTPUT WITH LED OFF

Fig 6.3 : OUTPUT ON ANDROID MOBILE APPLICATION

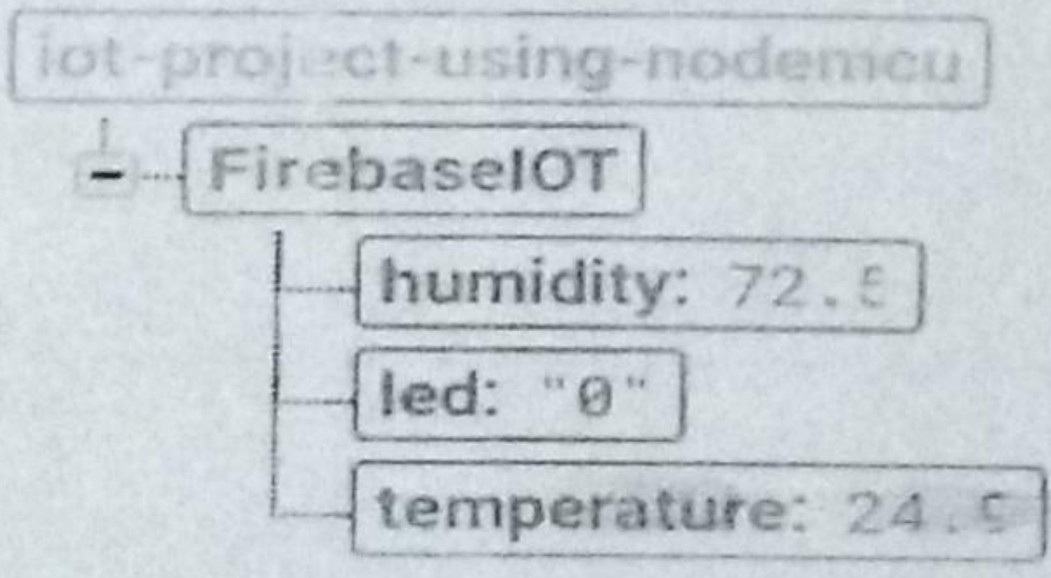

Fig 6.4 : OUTPUT ON REALTIME DATABASE - FIREBASE

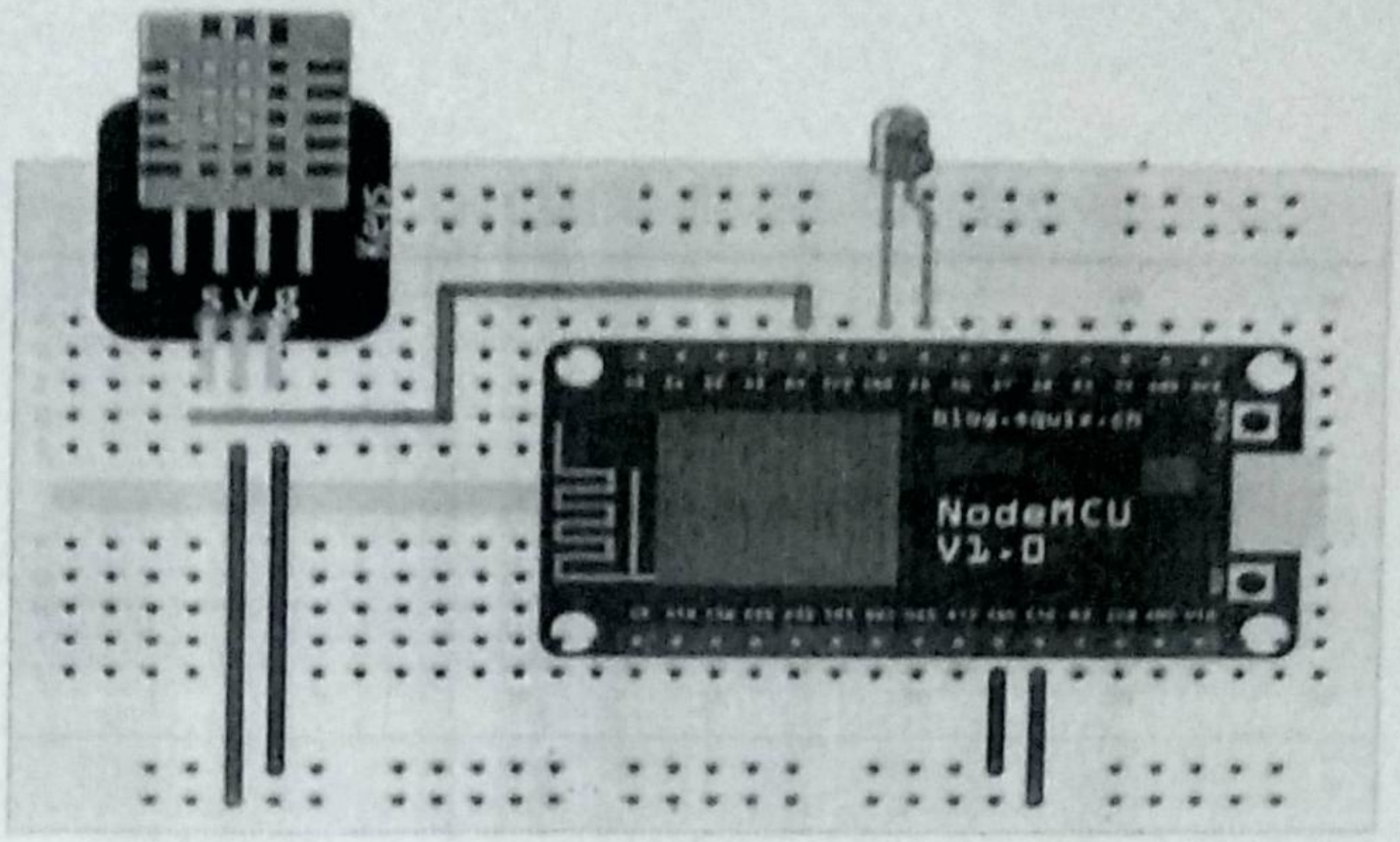

Fig 6.5 : CIRCUIT DIAGRAM

Fig 6.6 : SNAPSHOT OF WORKING AT INTERNSHIP

BIBLIOGRAPHY — Reference

- https://firebase.google.com/ - For Learning Basics Of Firebase And Creating Realtime Database
- https://appinventor.mit.edu/ - To Create Android Mobile Application For The System
- https://www.javatpoint.com/computer-packet-tracer - To Learn The Basics Of Cisco Packet Tracer
- http://www.nptel.ac.in/ - To Learn Iot Basics And Start Building Working Models Based On Iot.
- https://www.omnisci.com/technical-glossary/embedded-systems - To Learn Embedded Systems And Its Basics Workings

DECLARATION

I do here by solemnly affirm to declare that, the contents of the *Internship Report based on the* Software Part of work entitled "**TEMPERATURE MONITORING ANDROID APP WITH GLOBAL REAL TIME DATA ACCESSIBILITY USING IoT**" *submitted to the* department of **CSE, B.E** whichever is of **City Engineering College** *Bangalore, affiliated to* **VTU Belgaum,** Approved AICTE – New Delhi, is in majority the courtesy by the referred "National/International journals, Text-books, papers presented and published" by the learned Authors in the areas of their Research and also available in the public domain. I sincerely acknowledge all of them. This Internship Report has been submitted by me to satisfy the academic requirement affiliating university for the award of **Bachelor Degree** in the discipline of my study.

Date:

Place:

KRUPA D

[1CE17CS052]