

# Advanced IOT Based Blood Bank

[<sup>1</sup>] Aruna R, [<sup>2</sup>] PradyneshPatil, [<sup>3</sup>] Rajeev S, [<sup>4</sup>] Shruti Sanga A, [<sup>5</sup>] Swathi R

[<sup>1</sup>] Assistant Professor, [<sup>2</sup>], [<sup>3</sup>], [<sup>4</sup>], [<sup>5</sup>] UG Scholars

Department Of Electronics and Communication Engineering,  
Sri Sairam College of Engineering, Anekal

**Abstract:** Blood is the saver of all life in the case of emergency and, In an emergency situation, if the stock of blood are insufficient or unavailable, the only source of blood will be the people who donate the blood on a voluntary basis. It is certain that, time is a very crucial component in such situations. For this reason, the health care centre should contact the nearest available donors in order to ensure the service as quickly as possible and hence this is a time-consuming process. This issue can be overcome by automating the entire system along with a portable smartphone application. The blood packets will be monitored and updated periodically. The Application provides a way to synchronize the information between blood banks and Hospitals with the help of Internet. The android application can be accessed by registered hospitals to check the availability of Blood and can send Request for blood to the nearest blood bank or donor matching with blood requirements. Only a registered person, with willingness to donate blood, will be able to access the service. The application uses GPS technology that will provide a way to trace the way to the blood bank or nearby hospitals. The user will get the route to reach the desired location and eliminates the need for manual interventions, thus saving time. The information about the donor, hospitals and blood banks will be stored in the central database and can be accessed globally by authorized user anytime and anywhere.

**Keywords – Blood Bank, Hospitals, Donors, Administrators.**

## I. INTRODUCTION

The population of the world is increasing at a rapid pace every year and so is the possibilities of diseases and health issues. With an increase in the population there is an increase in the need of blood. Due to the lack of communication between the blood donors and the blood recipients, most of the patients in need of blood do not get blood on time and hence lose their lives. There is a need of synchronization between the blood donors and hospitals and the blood banks. This improper management of blood leads to wastage of the available blood inventory. Improper communication and synchronization between the blood banks and hospitals leads to wastage of the blood available. These problems can be dealt with by automating the existing system. A high-end, efficient, highly available and scalable system has to be developed to bridge the gap between the donors and the recipients and to reduce the efforts required to search for blood donors. Today mobile and mobile based applications have become a part of our day to day life. This application is developed to easily search for blood in nearby areas for emergency. In this Android app, one will get clear access to blood in real time and right place. The blood packets will be managed by monitoring it and will be updated automatically.

## II. EXISTING SYSTEM

In existing system, the entire information is communicated manually, in which the requirement for blood is only communicated by using human interfaces and there is no automatic means to detect the presence or absence of blood packets in the tray of the blood packets and hence requires

human intervention and monitoring. And the information regarding the user and hospitals and blood banks is stored in the register. If there are any changes in the information then it must be updated manually in the register. There is no proper coordination between different applications and users. It consumes lot of manpower for better results. Retrieval of data takes lot of time and percentage of accuracy is less. It creates room for errors as the data is entered. It provides less security. Disadvantage

- Delay involved in communication with the various blood banks, hospital and donors.
- Increases the risk posed towards a patient's life.
- It is very difficult to contact the donor during the emergency situations.
- The system is tedious, time consuming and space consuming.
- It includes the risk of the documents and maintenance of the records is difficult.
- 

## III. PROPOSED SYSTEM

The proposed system (Blood Bank Management System) is designed to help the Blood Bank administrator to meet the demand of Blood by sending and/or serving the request for Blood as and when required. The proposed system gives the procedural approach of how to bridge the gap between Recipient, Donor, and Blood Banks. This Application will provide a common ground for all the three parties (i.e. Recipient, Donor, and Blood Banks) and will ensure the fulfilment of demand for Blood requested by Recipient and/or Blood Bank. The entire system is also made automated where

**International Journal of Engineering Research in Electronics and Communication  
Engineering (IJERECE)  
Vol 4, Issue 5, May 2017**

the information about the hospital, donors and blood banks will be updated automatically. The units of blood packets also will be updated automatically.

The proposed system consists of the following goals and has the scope as follows:

**a) Goals:**

- To easy the process of blood donation and reception.
- To improve the existing system.
- To develop a scalable system.
- To make the existing system time saving.

**b) Scope:**

- Ensure that all the functionalities of a manual blood bank are covered
- To include all the blood banks at least within a city.
- Make sure the program is simple and easy to use.

**c) E- Blood Bank:**

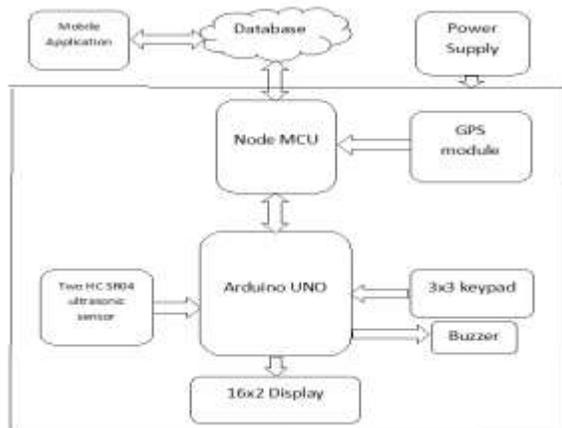
This application helps you find people donating blood in your area. You can contact them through phone number or email address. You can see the location of user in map and if register yourself with App, you can get push notification in case your blood group matches with the need of blood. You can find nearby hospitals and access them.

**Features:**

- Find donor with your specific blood group and with your respective states and cities.
- Send notification: This will help you know who all are having the same blood group in your local area.
- Find nearby hospitals in maps.

Provided helpline numbers in case of emergency.

**IV. BLOCK DIAGRAM**



**V. WORKINGPRINCIPLE**

**A. Arduino Uno Microcontroller:**

Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

ATmega328 is The Atmel 8-bit AVR RISC-based microcontroller combines 32 kBISP flash memory with read-while-write capabilities. Keypad, 16x2 LCD Display, Buzzer and HC-SR04 connected with microcontroller. The Microcontroller is used for controlling the operation of entire system. It will be in hospital which takes the input from keypad and gives it to the Node MCU as an input. Address of every hospital will be stored in the Node MCU. Whenever the request comes from main microcontroller it will attach the stored address along with the input of keypad and upload it to the server with the latitude and longitude values of the hospital.

**B. Node MCU:**

NodeMCU is an open source IoT platform. It includes firmware which runs on the [ESP8266 Wi-Fi SoC](#) from [Espressif Systems](#), and hardware which is based on the ESP-12 module. The term "NodeMCU" by default refers to the firmware rather than the dev kits. The firmware uses the [Lua](#) scripting language. It is based on the [eLua](#) project, and built on the [Espressif Non-OS SDK for ESP8266](#). It uses many open source projects, such as lua-cjson and spiffs.

This Node MCU will be connected to the main microcontroller. The main function of the Node MCU is to upload or download the data from the main server. This helps us to make system faster and real time. It will work as a transmitter and receiver between microcontroller and main server.

**C. HC-SR04 Ultrasonic sensor:**

The ultrasonic sensor HCSR-04 is one of the most commonly used for distance measuring ultrasonic sensors and it works extremely with the Arduino.

We need HC-SR04 to check the availability of blood packets in blood bank. We will define specific distance for each blood packet. The output of the sensor will be given to

**International Journal of Engineering Research in Electronics and Communication  
Engineering (IJERECE)  
Vol 4, Issue 5, May 2017**

the microcontroller. Based on the reading of the sensor, the microcontroller sends data to Node MCU.

**D. 16X2 LCD Display:**

This display is used to show the information which comes from main server like hospital name, hospital address, required entities.

**E. Main server:**

This server is used to make out the system real time. In which all database will be stored like Donor information (Donor profile, Donor ID, Password), hospital names, hospital address and blood availability etc. This will be updated after any changes. All hospitals, blood banks and donor are connected to each other through this server. In case of any emergency it will take data from needful hospital and provide it to every hospital, blood bank and donor.

**F. Android Application:**

This will be installed in every authorised blood donor mobile. In case of any emergency the donor will be notified through this application. If the donor is nearby the needful hospital and if he/she is ready to give blood. Then the hospital which is in need can directly contact the donor or donor can directly contact the hospital.

**G. 3x3 Keypad:**

In case of any emergency the needful hospital will give the information about required things to microcontroller using keypad. Different keys will be assigned for different blood groups. Each blood group will have separate key which help the system to be fast.

**H. Buzzer:**

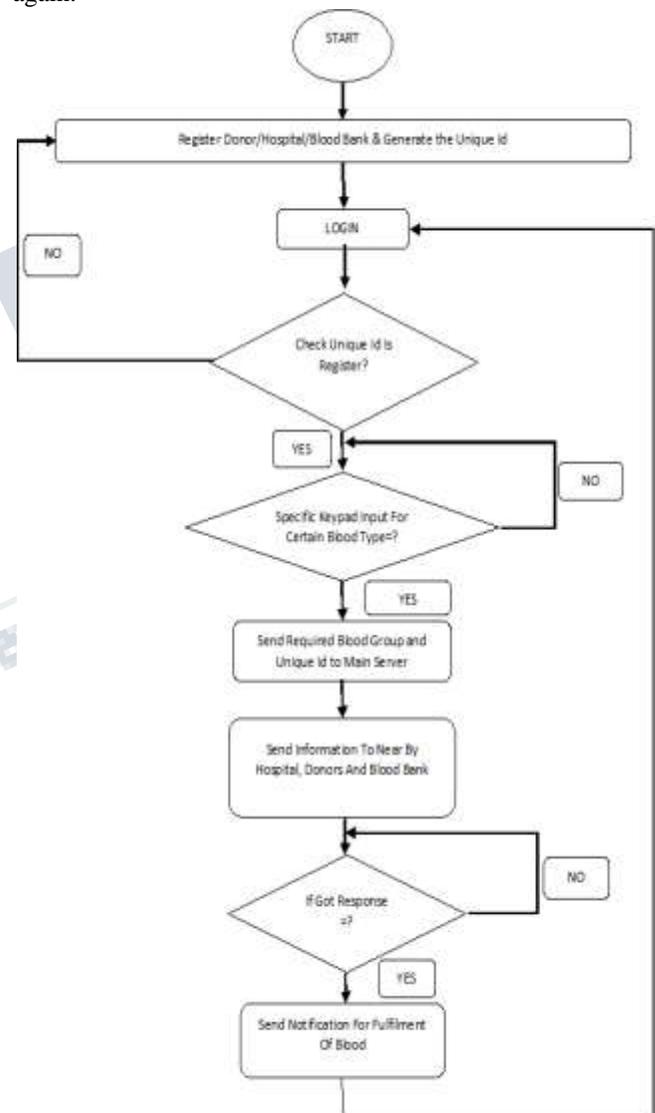
Whenever the node MCU gets request from server it will send the data to main microcontroller. After getting request from the server we use the buzzer to indicate that a new request has been arrived.

Step 6 – The required blood group and unique ID will be sent to the main server.

Step 7 – The information from the main server is sent to the near by hospitals, blood banks, donors.

Step 8 – It will check for the response. If yes, goes to Step 9 or else it repeats Step 8 until it gets a response.

Step 9 – If the response is received then it sends a notification of the fulfilment of blood or else it goes to Step 3 again.



**VI. FLOWCHART**

- Step 1 – Start
- Step 2- The hospital, donors and blood banks can register and after the registration a unique ID will be generated for the authorised user.
- Step 3 – The user will be logged in.
- Step 4 – It will check if the unique ID has been registered. If yes it goes to Step-5 or else it repeats the Step-2 for registration process.
- Step 5 –A specified keypad input which is assigned to a certain blood group. If yes, goes to Step 6 or else repeats Step 5 again.

**International Journal of Engineering Research in Electronics and Communication  
Engineering (IJERECE)  
Vol 4, Issue 5, May 2017**

**VII. RESULTS OF ANDROID APPLICATION  
LOGIN PAGE**



The user has to login or register to get started with the Android application. After clicking LOGIN button, the entered password and the entered username will be sent to the back end database and the user will be granted access if he is registered within the system. On clicking the 'REGISTER NOW' button, the user will be directed to the registration form.

**REGISTRATION PAGE**



Here a new user can register about his/her information regarding name, username, mobile number, address and password. After entering all these details the user will be registered.

**VIII. ADVANTAGES**

- System is automatic.
- Time saving.
- Provides security.
- Can be used in ambulance.
- Proper and managed details of each and every donor will be maintained.
- Proper database will be generated and the use of registers and books can be avoided.

**IX. CONCLUSION**

Technology is introducing new innovations day by day, thus reducing the time required to do things. The proposed system can be used to reduce the time required to deliver required blood to the needy in cases of emergency. The Android application can be used by the people interested in donating their blood by locating their nearest blood bank. The android application provides a way of communication and synchronization between the hospitals and the blood banks. It also provides them with the facility of communicating with the nearby donors in emergency. The database is a vital aspect of the system. The database of the hospitals and the blood banks must be checked for consistency on regular basis for smooth working of the system. The system uses Google Maps which provides the user with an efficient way of locating the nearby donors/blood banks. The Android application is developed using Android Studio which is an open source software, while the android application for the hospitals and the blood banks is also developed using open source tools, hence the system developed is quite feasible, provides security and is time saving.

**REFERENCE**

1. <http://www.bloodbankindia.net/>
2. <http://www.bharatbloodbank.com/>
3. <http://www.lionsbloodbank.net/>
4. <http://www.eblooddonors.com/Index.aspx>
5. [http://en.wikipedia.org/wiki/Blood\\_bank](http://en.wikipedia.org/wiki/Blood_bank)

**International Journal of Engineering Research in Electronics and Communication  
Engineering (IJERECE)  
Vol 4, Issue 5, May 2017**

---

6. <http://www.webbloodbank.org/index.php>
7. Android Location API using Google Play Services  
<<http://www.androidhive.info/2015/02/android-locationapi-using-google-play-services>>
8. Android Login and Registration with PHP, MySQL and SQLite  
<<http://www.androidhive.info/2012/01/androidlogin-and-registration-with-php-mysql-and-sqlite/>>
9. Markers | Google Maps Android API  
<[https://developers.google.com/maps/documentation/android-api/marker#code\\_samples](https://developers.google.com/maps/documentation/android-api/marker#code_samples)>
10. 

|         |        |              |
|---------|--------|--------------|
| Android | Studio | Installation |
|---------|--------|--------------|

  
<<https://developer.android.com/studio/install.html>>

