# Proposal and Implementation of Android Application Smart Prototype-SuFI to Provide Food Delivery

**Sachin Badave**

*Department of Electronics and Communication, KLS Gogte Institute of Technology, Belgaum-590008*

---------------------------------------------------------------------***---------------------------------------------------------------------

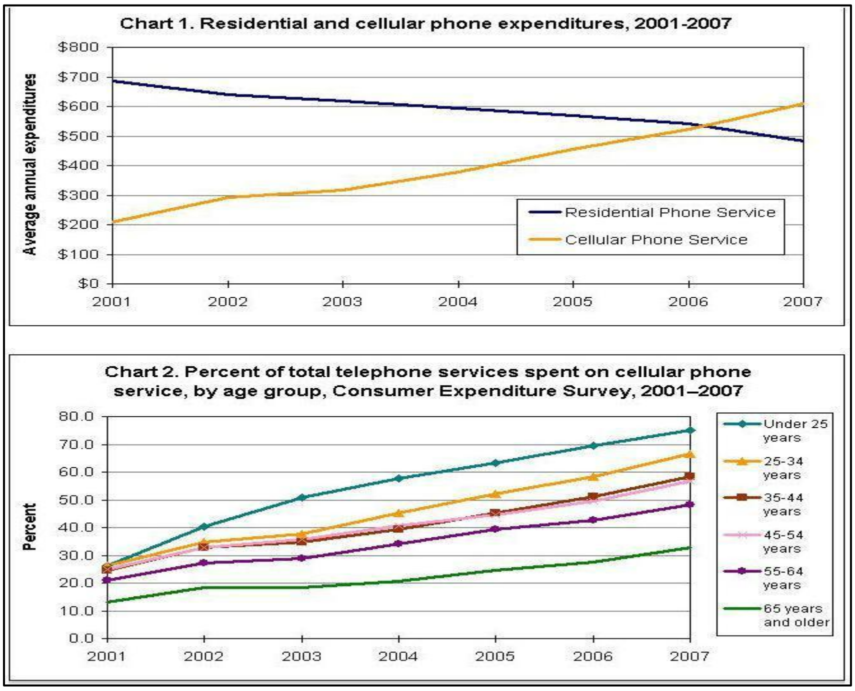
**ABSTRCT:** One recent development in cell phone usage is online restaurant reviews. Before deciding what to eat for dinner or stepping into a restaurant, people will often consult restaurant reviews through their smart phones. In order to help people search and order food with their smart phones, my graduate project is to design and implement a food menu search phone application.

The aim of our project is to develop an android application to provide food delivery by smart prototype. Our application name is Speedup Food India (SuFI). "Presently android market has many applications. In India it has been seen that in many of the functions and parties the remaining food goes to the dustbin, and lakhs of people are dying because of undernourishment. So to overcome this situation another salient feature got added in this project ie Leftover Food Management – Providing leftover food parties and functions to the needy people to avoid undernourishment. As a solution, an android application sufi is developed.

Keywords: Database, Server, Gson, Eclipse, Java Core

## 1. INTRODUCTION

Social networking has become increasingly popular, especially in today's mobile phones. Cell phone technology nowadays allows innovative applications to be implemented in many forms. In the Figure 1, from 2001 to 2007, while the residential phone expenditures are dropping, the cellular phone expenditures are increasing dramatically in all age groups. Chart 1 shows the Residential and cellular phone expenditures from 2001-2007. Chart 2 shows percent of total telephone services spent on cellular service, by age group, consumer expenditure survey.



In order to satisfy the large number of cell phone customers, the mobile phone industry is making improvements in cell phone hardware as well as software. Hence the science fiction devices of yesteryear have become reality. Cellular telephones today are remarkable in their advancement. Whereas cell phones of yesterday were useful simply as communication devices, for many people today, a cell phone is not only just a tool to communicate with others; it serves also as an alarm clock, calculator, internet browser, notepad, camera, calendar, music player, voice recorder, GPS and computer game. This has been made possible with the advancement of cell phones. Today, cell phones are not just telephones; they are small computers with heavy processing power.

One recent development in cell phone usage is online restaurant reviews. Before deciding what to eat for dinner or stepping into a restaurant, people will often consult restaurant reviews through their smart phones. In order to help people search and order food with their smart phones, my graduate project is to design and implement a food menu search phone application. The aim of our project is to develop an android application to provide food delivery by smart prototype. Our application name is Speedup Food India (SuFI). "Presently android market have many applications like swiggy, areo food panda etc. But they only provide food from restaurants, which does not include the features and services as our applications have." The number of web applications have grown dramatically as the Internet has developed. As long as a user has a computer with internet access and a browser installed, he can make a request to a static URL web application server. All the technical detailed implementations are hidden from the users, it is quite easy for anybody to use anytime. According to the survey made in India, 14.5% of people undernourished last year because of hunger. According to the survey in 2017, 190.7 million people go hungry everyday in India. So our application is developed to fulfil customer needs by home makers, restaurants and stalls and also to feed poor people by providing them leftover food. The main objective of our project is to provide food delivery ranging from five star restaurants to roadside food stalls. Our application also includes food delivery from home makers, which is beneficial to the people who stay away from home as they get home taste food and where it creates employment. SuFI also has an easy authentication to the delivery person who can be any auto driver or any candidate with valid driving licence. SuFI has got another feature - Leftover food management, which feeds hunger of the needy people (orphanage, poor people).

## METHODOLGOY

### Phone Application

We have already discussed some characteristic of a web based application in the previous section. Most of those benefits carry over when used on a smart phone. In fact, there are lots of web server can handle communication with both phone and computer clients. Figure 2.1 show general phone application architecture. We can see how both phone and computer can communicate with the server via the Internet.

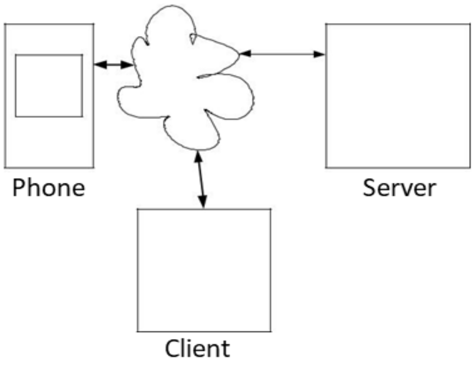


Figure 2 Phone Application Architecture

One successful example is Google Maps, one of the most popular map applications. Similar to GPS devices, some applications provides point of interest for users to retrieve their desired information. By using Google Maps, similar to GPS' POI feature, users can search for points of interest in different categories. Unlike a GPS, as long as internet access is provided, users can get the most updated information at no charge.

### Architectural Design:

The below figure 3 shows an architecture of android application.

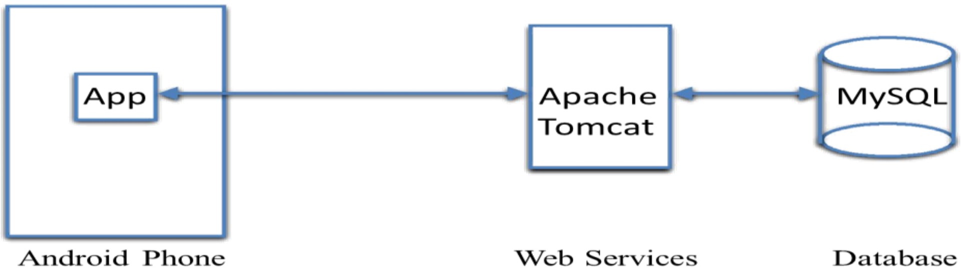


Figure 3 Architecture of Android App

SuFI food will be installed on Android Phone with Android OS 5.0 and above. App will make an asynchronous HTTP request to the RESTFul Web-Service installed on Apache Tomcat Web Server. Web-services are built using JersyRESTFul web services API.

Web-service layer (middleware) will connect to MySQL database using the Hibernate Framework to perform the CURD operations.

**Server Client Model**

The project is based on a typical server-client model as shown in Figure 4 below:

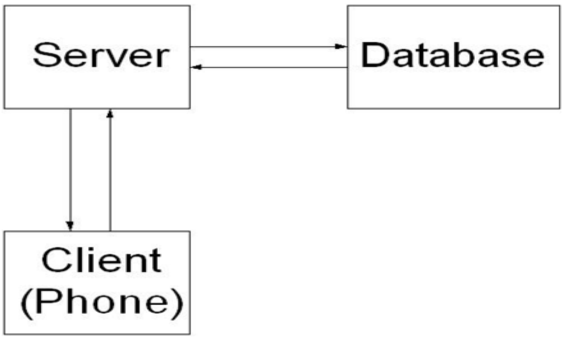


Figure 4 Server - Client Architecture

The developing environment includes a remote computer, acting as a web server, with a static domain name set up, and a G1 (Android phone), acting as a client. Because this project is based on a relatively small scale database, in order to reduce the number of hardware components, the database is also set up on the server side. We are using pre-setup server which contains Apache tomcat (version 2.0), database MySQL (version 4.1.20). For the client, we also installed Eclipse and Android emulator plug-in for development and testing purposes. The client is written in (Android) Java language. The project is tested on both emulator and a G1 (Android) phone.

**Design Consideration**

**Platform**

In many markets, Android is one of the earliest significant software platforms for open source mobile applications. Its operating system is Linux kernel-based and user applications are built in Java by using the Android SDK. Due to the simple set up, open sources and the code being based on regular Java syntax, developing Android has big potential market in the near future. One of the big comparators is the iPhone. In order to develop applications on the iPhone, a Mac OS is required. In order to develop Android application, programmers can develop code on both Mac and Windows systems. This makes Android applications more flexible.

By installing the Android add-on Android SDK, we can develop and run the appropriate version of Android phone applications on both of the emulator and the real Android Google phone. The Android emulator is pretty easy to set up and use. For most of the features for our food application, we can use the Android emulator. For some of the features such as instant file storage, usage of camera and location detection, using the real Android phone would be more reliable. Besides, we can also see all the debug messages from Eclipse when we are testing both on the emulator and on the real phone.

**Application Functionality Design**

When using this application, users are not allowed to use their phone to create a new food or restaurant. This is because we want to avoid people entering incorrect information to our system. For example, we do not want somebody to input the wrong price/food item to affect a restaurant's reputation. Besides, incorrect information could cause our users to lose their trust in our system, discouraging them from using it. On another hand, users are allowed to input information such as comments, ratings, and pictures. This is a nice feature to encourage users to upload more pictures, give some feedback, and rate the food items they just ate. With real users' review and ratings, our phone application's information will be more trustable.

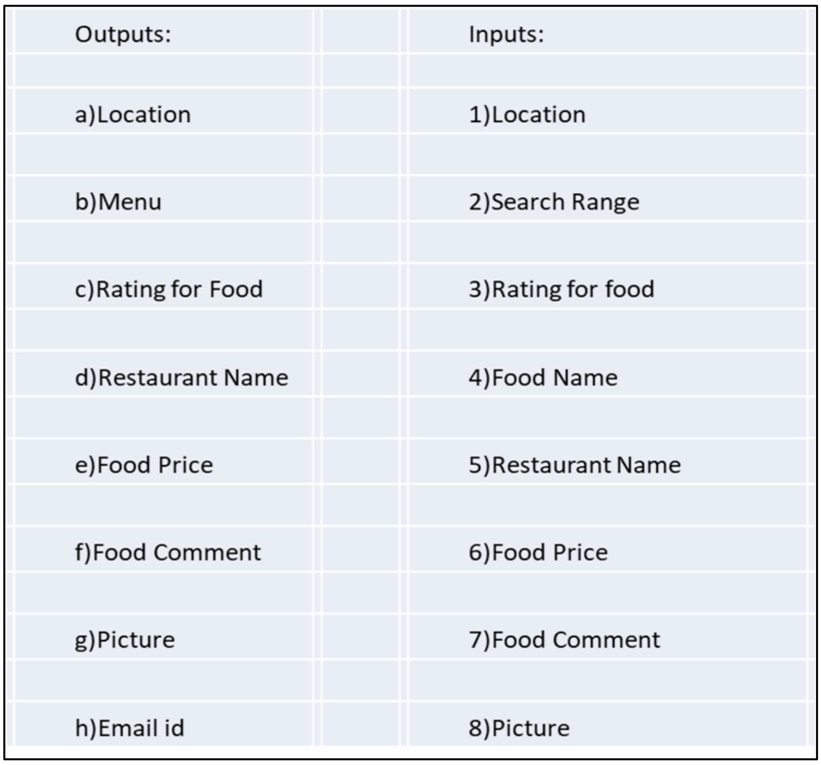
Another thing to encourage users using our application is having a user friendly phone application UI design. Unlike web applications and GPS devices, which have bigger screen, Android phone's screen size (resolution 340X240) is limited. When we want to develop a phone application, the user interface is one of the most important factors and must be handled delicately. Some phone applications, such as Urbanspoon and iCrave, just launch the web browser on the phone to display food information. Within the small screen, it is not easy for user to see or click any buttons and hyper links on the screen. Our application, however, gave users a comfortable UI to work on. For example, all of our buttons and text message are big and easy to click. For multipleitems (restaurants/food), we tried to use a lot of list view with a scrollable bar on the side so that the users can have more active space to view each item.

Another example the project uses is the menu buttons in order to save the screen space. If a user does not push the menu button, the bigger size of the screen could be displayed to the user. Additionally, when we slide the keyboard out, the screen will automatically switch from landscape to portrait. As a result of defining both landscape and portrait views for each activity, we can have both keyboards, virtual and physical, facilitating user inputs, which help users to have an easier time typing in their input.

**Application Design**

Firstly, we need to design a protocol between the client and the server. It takes two steps to do so, the first step, called Application Design, is the highest level to define what kind of tasks the food server and client can handle.

Table 1 shows all the possible inputs and outputs for this phone application:



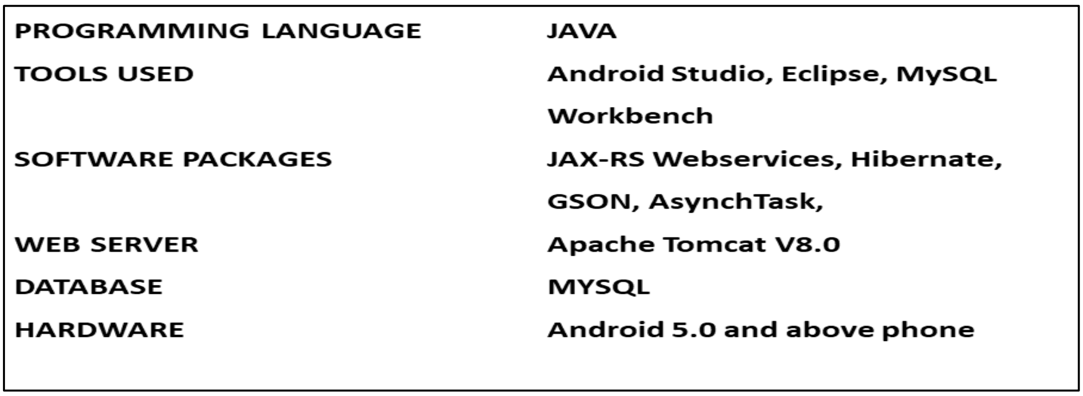
| Outputs: | | Inputs: |
|---|---|---|
| a)Location | | 1)Location |
| b)Menu | | 2)Search Range |
| c)Rating for Food | | 3)Rating for food |
| d)Restaurant Name | | 4)Food Name |
| e)Food Price | | 5)Restaurant Name |
| f)Food Comment | | 6)Food Price |
| g)Picture | | 7)Food Comment |
| h)Email id | | 8)Picture |

Table 1 Mapping of the Inputs and Outputs

This step is very important because it helps a developer understand what kind of information the user can get from and send to the server. After that, we can map each of the output to the some of the inputs to define all getter and setter functions:

**IMPLEMENTATION**

There are lots of web base applications which utilize LAMP (Linux as operating system, Apache HTTP Server, MySQL PHP) software stack as it is an open source. As it is easy to use and also reliable and reasonable in performance, we are using LAMP set up on our server side. The below table 2 shows all the software tools and packages.



| | |
|---|---|
| PROGRAMMING LANGUAGE | JAVA |
| TOOLS USED | Android Studio, Eclipse, MySQL Workbench |
| SOFTWARE PACKAGES | JAX-RS Webservices, Hibernate, GSON, AsynchTask, |
| WEB SERVER | Apache Tomcat V8.0 |
| DATABASE | MYSQL |
| HARDWARE | Android 5.0 and above phone |

Table 2 Software tools and packages

**Database**
Similar to many web applications, for restaurants and their menu's storage and user logging information, it is necessary to have a database set up for the system. Since this system's database is simple, MySQL database is used for this phone application.

**Server**
Apache Tomcat is a webcontainer which allows to run servlet and JavaServer Pages (JSP) based web applications. Most of the modern Java web frameworks are based on servlets, e.g. JavaServer Faces, Struts, Spring. Apache Tomcat also provides by default a HTTP connector on port 8080, i.e., Tomcat can also be used as HTTP server. But the performance of Tomcat is not as good as the performance of a designated web server, like the Apache HTTP server.

**Gson**
Gson is a Java library that can be used to convert Java Objects into their JSON representation. It can also be used to convert a JSON string to an equivalent Java object.Gson is an Open Source project, which can be found under https://github.com/google/gson . The webpage provides a convenient download link. Gson can work with arbitrary Java objects including pre-existing objects that you do not have source-code of. For this purpose Gson provides several built in serializers and deserializers. A serializer allows converting a Json string to corresponding Java type and a deserializers allows to convert from Java to a JSON representation. You can also configure Gson to use custom representations of your objects. Gson allows to serialize Collections of objects of the same type. As the generics information is persisted in the JSON you need to specify the type to which you want to serialize too. For this you have different options.

**Android Studio**
Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, mac OS and Linux based operating systems.

**Eclipse (software)**
Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications, but it may also be used to develop applications in other programming languages via plug-ins, including Ada, ABAP, C, C++, C#, COBOL, D, Fortran, Haskell, JavaScript, Julia, Lasso, Lua, NATURAL, Perl, PHP, Prolog, Python, R, Ruby (including Ruby on Rails framework), Rust, Scala, Clojure, Groovy, Scheme and Erlang. It can also be used to develop documents with LaTeX (via a TeXlipse plug-in) and packages for the software Mathematica. Development environments include the Eclipse Java development tools (JDT) for Java and Scala, Eclipse CDT for C/C++, and Eclipse PDT for PHP, among others.

**Database Access**
One of the most important roles for the eclipse module is helping users to communicate with our database indirectly. From the previous section, we defined all public function names precisely in the eclipse file, which means, the server will return correct result only if users put the correct URL with correct function names and parameter names with appropriate values. Otherwise, user will receive an error.

### Upload File

Another responsibility is accepting files (picture files) from clients. As it is built on a Linux system, since the given default setting for upload directory is unknown, within the hibernet, it is necessary to change the change the permission code in order to allow users to have write permission. Secondly, since the uploaded pictures will only display on the phone screen, for a better performance, the picture resolution does not really need to be high. This can save lots of space for the server and downloading and uploading time from the server to transfer files.

### Java Core

After we made sure the server is working correctly for all valid URL, we can start to work on the Java core module. The main responsibility for Java core module is to assembly all URL for all functions with correct parameters and appropriate value , make sure they can get exact output as we use a browser and send request to the server. Theoretically, this module could be written in other language as well as it could generate URL and parse the result into appropriate form to the next module (Android UI module). Java is obviously the best solution in here because using Java to build the URL and parsing the result are very simple and most importantly, Android is already include the JDK library.

Same as doing basic java networking code, we need to set up an URL object with an appropriate URL, then use a Http URL Connection to open a new connection and read in data by using the Input stream. When we get the http response from our tomcat server, as it is not the result from a web services, Java core module needs to handle the parsing so that the next module can retrieve this result easily.

### Java Android

Since this is a phone application, the Java Android is the most crucial part for the project. It directly interface with users, handles all user input events to trigger internal functionalities for the phone and display user understandable output to the screen. Besides, this module tried to take the advantages of Android smart phone's advance features which will be described in the following sections.

### Fancy List View

By default, we can only create a simple list view on the phone. The list is only text message which are stored in an array variable. When we need to create a fancy list view, it is necessary to extend the ArrayAdapter in order to add customized style of into the list view. In this way, we can add different components dynamically to the list and display them on the phone at runtime.

### Working of android application-SuFI

The working model consists of ordering process, hotel registration, uploading of menu and delivery to the ordered customer and creation of leftover food and its management. The following figure 5 shows the block diagram of android application.

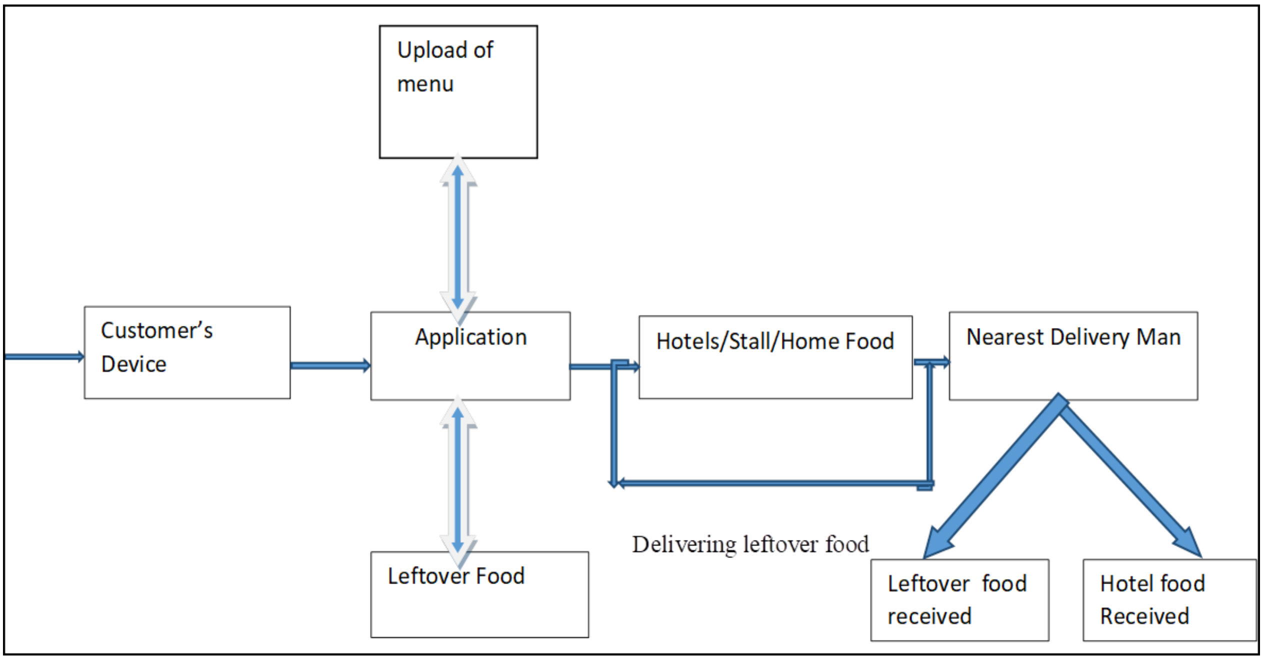


Figure 5 Block Diagram

The working model consists of ordering process, hotel registration, uploading of menu and delivery to the ordered customer and creation of leftover food and its management. The following figure 5 shows the block diagram of android application.

**Ordering process of the food from android application-**

- Initiation of application.
- Selection of hotel/stall/home maker by user.
- Search for nearest delivery source by application.
- Collection of food from source from deliver person.
- Delivery to customer.

**Requesting Leftover food-**

- Initiation of application.
- Selection of leftover food.
- Search for nearest delivery source by application.
- Collection of food from source from deliver person.
- Delivery to the needy person.

**Upload of food by restaurants/stalls/homemakers/leftover food-**

- Run the application.
- Upload the menu with price and pictorial description.
- Update of location using gps and maps.

**Advantages**

- Increase in employment rate.
- 24/7 service for delivery of food.
- Serving hungry people.
- Real time tracking of food and fast delivery of food.
- Providing service from all kinds hotels including food stalls.
- Prevention of wasting of leftover food.

**Disadvantages**

- No guarantee of Quality of food delivered (AS the restaurant will be responsible for the quality of food).
- Delivery charges can be extra upon food cost.

**Applications**

- Smart prototype for delivering food.
- Smart advertisement for home makers and food stalls.

## CONCLUSION

Hereby, The food is delivered to the customers from all the food sources such as restaurants, mess, food stalls etc by means fast delivery application SuFi. Its has also provided the employment to the home food makers, delivery candidates. SuFI has also done a great job of serving hungry people and also a needy people such as NGOs, Orphanages by delivering them a Leftover food from functions and hotels as well. So the application has successfully served its purposes.

## REFERENCES

The sites which were used while doing this project:

1. www.wikipedia.com
2. www.allaboutcircuits.com
3. www.microchip.com
4. www.howstuffworks.com