**CHAPTER 1**

**ABSTRACT**

* 1. **ABSTRACT**

The aim of this project is to bring the power of Wikipedia on your mobile phone ‘without’ the user accessing the internet via GPRS/EDGE, to provide wireless information, to bridge Knowledge vs. Infrastructure gap, to provide affordable means of information, to take advantage of high mobile penetration, especially in rural areas, and to ensure knowledge is available in every nook and corner. The domain is Web Based/ Text to Speech Conversion/Automated server.

The user will send a SMS to a number assigned to our server, on any topic on which he needs information. The server will accept the SMS information as keyword, search it over Wikipedia, parse the information obtained to only the summary, then store the information in a database and finally convert the text contained in the information to speech and callback the user on his number and relay the message to him voice format. This will help achieve the following. Providing information to the people who do not have access to GPRS but only to a mobile. Relaying information to places where connectivity is not available or the infrastructure has not been set. Information for the visually impaired and blind. Cheap Means of information via SMS.ven low end cell phones without internet facility can be used to mine information from the web.

Also since all the components used i.e. JAVA as programming language and Wikipedia as the source of information, the project is an ode to “open source information sharing”.

**CHAPTER 2**

**PROJECT OVERVIEW**

**2.1 REQUIREMENT OVERVIEW**

It is said that Knowledge is Power. Many a times, when we require information we simply search it over Wikipedia. But what about the people who don’t have access to internet through GPRS or broadband ?

This is where the scarcity of Knowledge arises, especially in rural areas (60% in India) who cannot access GPRS but just the SMS facility on mobile. If we are to empower every corner of our country with information there should be cheap means of doing so until broadband infiltrates.

The project mainly aims at providing quick and accurate information to the masses without them being required to be connected to the internet. All the user needs to have is a mobile phone with SMS sending capabilities and through it he can have the power of Wikipedia on his phone.

**2.2 WORKFLOW OVERVIEW:**

The person needing information on a particular celebrity/place/event/history etc will have to SMS what he requires to a number provided by our server. The steps that take place will be as follows.

1. User types out the person/place etc regarding which the information he needs and sends it to the number assigned to our server via SMS.
2. The server extracts the text in the message through the “SMS pull engine” and feeds it into Wikipedia and searches for it.
3. Once a match is found, the “text to speech engine” will convert the textual information into an audio format.
4. After this step, the server will automatically dial the number of the user who requested for the information and relay it to him via audio, much like the pre-recorded customer care messages. Only this time it will contain “Information that the user requested”.

The text to speech part is handled by the JAVA Text to Speech engine. The voice call and telephony is handled by AT commands. Moreover the whole system will be “automated” which will considerably reduce overhead and need for human intervention.

**CHAPTER 3**

**INTRODUCTION   
 AND  
 MOTIVATION**

**3.1 INTRODUCTION:**

**3.1.1 Internet Scenario in India**

Cities in India show high internet users, but it’s still the villages that are lagging far behind. Although the urban areas constitute only 30 % of India’s population, a survey shows that internet usage in these urban areas is 64 % while the usage in the 70 % Indian villages is only 36 % (Reference-Website 13.2.1.1). This clearly reflects the bad condition of internet usage in Indian villages

DNA survey shows that 80% village folks are unaware of the Internet’s existence, and among the one’s who are aware, Of the ones who did know about it, 85% used the net only to access emails, 13% to know about the latest farming techniques and 2% to look up fertilizers, among other uses. We can acknowledge that internet penetration so far has been weak.”. (Reference-Website 13.2.1.2)

**3.1.2 Cell phone growth in India**

It is mockingly said that India is a country with more mobile phones than public toilets. India added 17 million cell phone subscribers in July 2010, taking total number to 652 million, (TRAI).This figure is only going to increase as cell phone infiltrates every nook and corner.

To go by recent figures India’s latest population is estimated around 1,198,003,000. Of this we are adding close to 15 million mobile users on average in a month, while in case of Internet it is 14 million users a year. Of course, it is not fair to compare the two, but this shows how slowly Internet is gaining traction, especially in the rural sector.

**3.1.3 Mobile telephony in rural areas**

“The quickest way to get out of poverty right now is to have one mobile telephone”- this statement was said by Muhammad Yunus, Nobel Peace Prize winner. In India Mobile telephones has been a “dream come true” for rural areas. Connectivity to the outside world has been made easy. Unnecessary commuting to urban centers has been tremendously reduced.

Today, mobile telephony is being used to provide information to the farmers through SMS. Mobile telephony offers some unique opportunities. These benefits are amplified by the fact that the spread of mobile technology in some rural regions has occurred much faster than with other information and communication technologies (ICTs). In countries such as with high rural population densities, mobile telephony has quickly become much more cost-effective for telecommunication provision. (Reference-Website 7.2.1.3)

Benefits include

• Providing a direct global communication channel to rural communities

• Offers multiple formats for information in one device

• Provides accessibility for illiterate users (i.e. voice and images)

• Offers multiple formats for information in one device

• Making rural services more efficient and cost-effective.

**3.1.4 Knowledge vs. Infrastructure gap**

In order to access knowledge in today’s world, one needs the internet, which in turn needs high infrastructure and connectivity. This cannot be afforded by the poor or people from the villages. However, if a developing country is to grow, it is precisely these people that it needs to empower the information to.

This leads to a catch 22 situation wherein Knowledge cannot be accessed without infrastructure and on the flip side information is needed by those people the most who cannot afford the infrastructure.

This leads to the Knowledge vs Infrastructure gap. We will see further how this gap can be bridged by means of voice telephony.

**3.2 SCOPE OF THE PROJECT:**

**3.2.1 Providing information to the people with access to mobile but not GPRS:**

This point emphasis on providing information to the users who possessed low end cell phones which are not GPRS enabled. These type of cell phones are accompanied with facilities such as call making, call attending, SMS sending, SMS receiving etc.

**3.2.2 Relaying information to places where Connectivity/Infrastructure is not available:**

This point emphasizes on relaying information to users who do not have INTERNET access because of the lack of infrastructure at the place. These users are deprived of a huge knowledge database in form of INTERNET because of the distant placing from main metros.

**3.2.3 Information for the visually impaired and blind:**

Visually impaired users cannot use INTERNET facility to acquire knowledge. But visually impaired User can use the SMS facility provided on the cell phones. Also ,although they cannot read information, they can listen to it over voice telephony.

**3.2.4 Cheap Means of information:**

Charges required to send a SMS are much less compared to charges of INTERNET usage or GPRS usage, this will provide cheap source of information to the people who cannot afford Internet, or are too ignorant to use it.

**3.2.5 To make extracting information a finger touch away:**

Information is made available to user at a single go. User just needs to send a SMS containing a keyword regarding query and he/she will get the information on voice call. Thus the information can be accessed with just a finger touch.

**3.3 MOTIVATION:**

“Knowledge is Power”- This piece of philosophical jewel was bestowed upon us by our fore-fathers and it’s never been truer than it is in this 21’st century. What we live in now is called the information age, and the need for information has indeed never been as much as it is now.

Many a times, when we require information we simply search it over the internet. But what about those people who don’t have access to internet (GPRS/Broadband), or perhaps those people who do not know how to use GPRS facility? Shouldn’t even these people be entitled to information?

This is where the unavailability of Information arises, especially in rural areas (70% in India) where people either cannot access internet or are too ignorant to use it. If we are to empower every corner of our country with easy means of accessing information, then there should be a cheap way of doing so until broadband infiltrates. Until then, we need a stop-gap solution

**CHAPTER 4**

**PROBLEM STATEMENT**

* 1. **PROBLEM DEFINITION:**

In today’s day and information is of the utmost importance. Hence this has also been dubbed as the “Information Age”. Many a times, when we require information we simply search it over Wikipedia. But what about those people who do not have access to the Internet.

This is where the scarcity of Knowledge arises, specially in rural areas (60% in India) who cannot access GPRS but have just the SMS facility on mobile. If we are to empower every corner of our country with information there should be cheap means of doing so until broadband infiltrates.

* 1. **PROBLEM DESCRIPTION:**

The problem can be described in 3 words as “Knowledge vs. Infrastructure gap”. It is a vicious circle wherein the people who most require information i.e. the rural people are most often the ones who do not get it because the infrastructure ahs not yet been developed for internet penetration. And conversely, since these places are rural, they will require quiet a few years until infrastructure is developed there. Meanwhile we need a stop gap arrangement which caters to solve the problem of this knowledge vs. infrastructure gap.

However, another problem arising is that since the people in these rural areas have limited means, the stop gap solution must not include any high end devices. Rather they should include only those devices which are already available to these people .Also another concern is that they shouldn’t have to spend too much money for the information they receive.

Also the problem is not only limited to people who do not have the infrastructure but also to the people who would like to access the information on Wikipedia when they travel to remote locations without GPRS/EDGE coverage.

Thus basically, the problems because which the Internet cannot be accessed are the problems which needs to be solved i.e. infrastructural problems (rural places), remote location problems (faced while travelling), cost problems (faced by poor people) are the main problems which needs to be solved.

**CHAPTER 5**

**REQUIREMENT ANALYSIS**

**5.1** **EXISTING SYSYTEM**

There are many Existing Systems available for information gain. An User can make use of these Systems and acquire knowledge regarding the queries he/she has. Following are some of the existing resources available to User for knowledge retrieval.

**5.1.1 Internet:**

The Internet is a global system of interconnected computer networks that use the standard Internet Protocol Suite (TCP/IP) to serve billions of users worldwide. It is a *network of networks* that consists of millions of private, public, academic, business, and government networks, of local to global scope, that are linked by a broad array of electronic and optical networking technologies. The Internet carries a vast range of information resources and services, such as the inter-linked hypertext documents of the World Wide Web (WWW) and the infrastructure to support electronic mail.

**5.1.2 GPRS:**

General packet radio service (GPRS) is a packet oriented mobile data service on the 2G and 3G cellular communication systems global system for mobile communications (GSM). The service is available to users in over 200 countries worldwide. GPRS was originally standardized by European Telecommunications Standards Institute (ETSI) in response to the earlier CDPD and i-mode packet switched cellular technologies. It is now maintained by the 3rd Generation Partnership Project (3GPP). GPRS usage charging is based on volume of data, either as part of a bundle or on a pay as you use basis.

**5.1.3 EDGE:**

Enhanced Data rates for GSM Evolution (EDGE) (also known as Enhanced GPRS (EGPRS), or IMT Single Carrier (IMT-SC), or Enhanced Data rates for Global Evolution) is a digital mobile phone technology that allows improved data transmission rates as a backward-compatible extension of GSM. EDGE is considered a 3G radio technology and is part of ITU's 3G definition. EDGE was deployed on GSM networks beginning in 2003 — initially by Cingular (now AT&T) in the United States. EDGE is standardized by 3GPP as part of the GSM family. EDGE delivers higher bit-rates per radio channel.

**5.2 PROBLEMS WITH EXISTING SYSTEM**

As mentioned above there are many resources available to User for information gain. But there are some problems associated with those recourses. These problems can be enlisted as follows accordingly each technology.

**5.2.1 Problems with INTERNET:**

INTERENT usage is costly, and it is not affordable to common man. Service providers of INTRENET facilities such as BSNL, HATHWAY charge User according to speed and download limits. In rural areas infrastructure required to deploy this facility is not available. And moreover many of the people from distant areas are not aware of this technology and usage of it.

**5.2.2 Problems with GPRS/EDGE:**

GPRS/EDGE enabled mobile phones are mostly high end cell phones. Hence their cost is on higher side. Also in many faraway places the connection of this 2G service may not be up to the standards required for normal browsing. There is also the problem of infrastructure for GPRS/EDGE, which may not be constructed in rural areas.

Hence we conclude that though there are different resources are available for knowledge gain, a User is deprived of it because of the above mentioned problems.

Thus our proposed system looks forward to make the knowledge available to User irrespective of his/her limitations of making use of existing system.

**5.3 PROPOSED SYSTEM**

This project aims at providing a System which will give solution to the above mentioned problems. In order to make use of the high internet penetration, we will be using mobile phones as agents to send and receive information. The keyword is to be sent via SMS to a number assigned to the server. The server then will extract the keyword, search it over Wikipedia, parse out all the unwanted content and keep only relevant information, then convert the information it gets into audio format and call back the user to relay him/her the information.

**5.4 SYSTEM REQUIREMENTS**

**Operating System**: Windows XP/Vista/7

These are commonly used. Any OS with support for Motorola mobile drivers and JAVA can be used

**CPU**: 1.6Hz

The project is implemented using Net Beans 6.7.1 IDE for which to run smoothly, at least 1 GHz processor speed is required

**RAM**: Minimum 256MB

Minimum RAM required to run Net Beans 6.7.1 IDE.

**Hard Disk:** Minimum 1300MB

Net Beans is 705MB. JDK version 1.6\_19 is 270MB, Text to speech engine is 20MB, WAMP server for database is 115MB, and Motorola USB driver is 3MB. All these application requires almost 1115MB hard disk space. In addition to save all the temporary files we will require total 1300MB hard drive space.

**Drivers**: Support for Motorola USB driver

We will install Motorola Handset\_USB\_Driver\_32\_v3 [1].4.0.

**GSM Enabled Mobile:**

A mobile phone that will receive messages from the client side.

**Internet connection-minimum 64 kbps at server:**

To search the keyword over Wikipedia. To achieve information retrieval at decent speed, minimum 64 kbps connection is required.

**Sound:**

The project requires sound drivers to be installed along with at least 1 speaker with loud volume in order to convey users the message in audio format using mobile callback.

**CHAPTER 6**

**PROJECT DESIGN**

**6.1 ITERATIVE DIAGRAM:**-

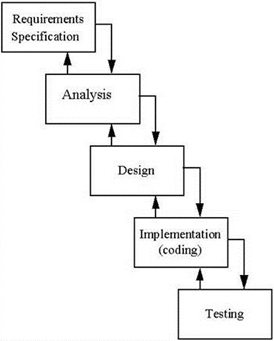


Figure 1:- Development model

**6.1.1. Requirement Specification**

We went through a lot of surveys over the internet which made us come to the conclusion that

* Internet penetration is little in rural areas
* Internet Awareness is rural areas is less too
* Mobile penetration is large

This led us to the conclusion that information from the rich internet source needed to be provided to the users via the comparatively cheaper mobile source.

**6.1.2 Analysis**

In this phase, we surveyed the existing source of information namely the internet and gave its drawbacks while proposing a solution of our own for the problems encountered. We also listed our hardware and software specifications in this stage

**6.1.3 Design**

In the design phase we gave the and the functional view of the software. A number of diagrams including the use case diagram, activity diagram, deployment diagram and sequence diagram were used to model the software.

**6.1.4 Coding**

Coding has been completed in 4 modules which are

* Receiving message from user, and parsing to obtain only keywords.
* Searching the keyword over Wikipedia and obtaining the parsed summary.
* Store this summary in a database and check the database for each iteration.
* Call back user using AT commands to deliver the information .

**6.1.5 Testing**

We have be followed a testing program that involves unit testing, integration testing, and validation testing. Also test cases for all the modules have been prepared to check the modules for error and optimality.

**6.2 BLOCK DIAGRAM:-**

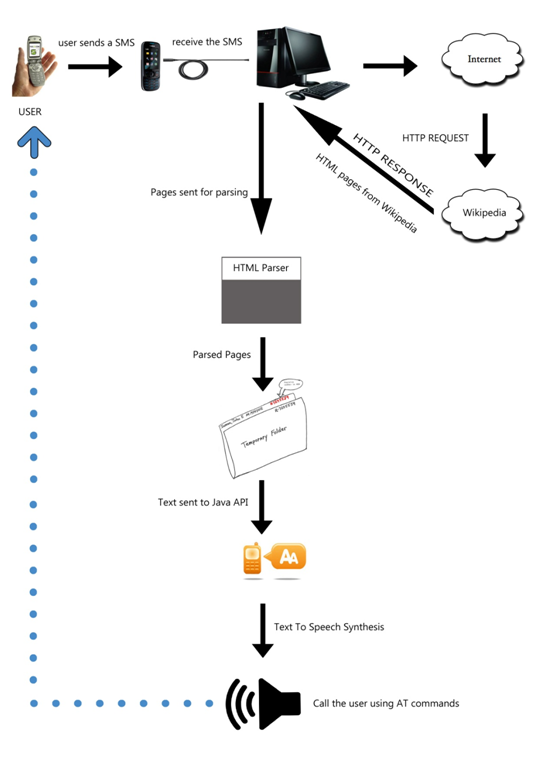
****

Figure 2: Block Diagram

**6.3 DATA DESIGN:-**

**6.3.1 USE-CASE DIAGRAM:-**

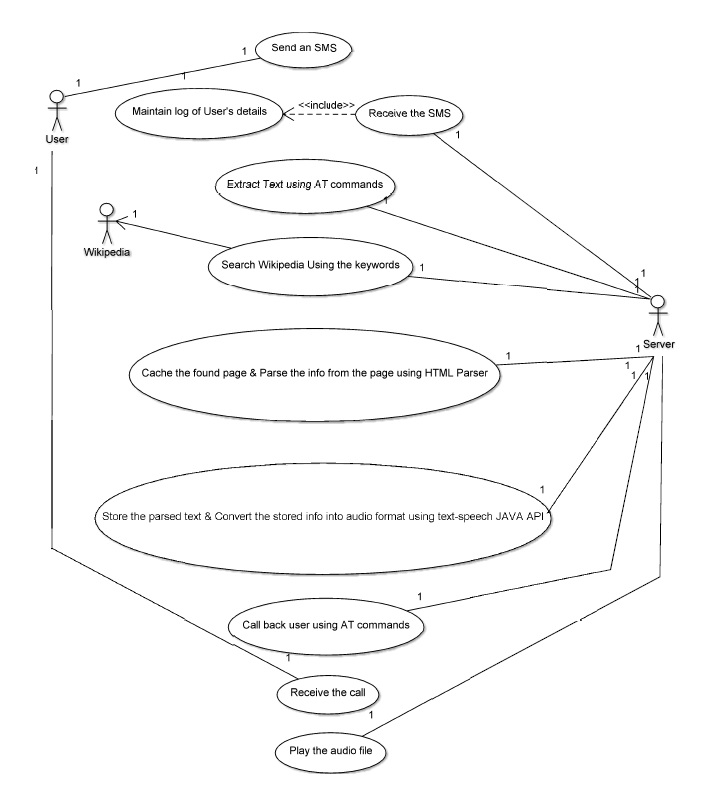
In this diagram user, Wikipedia server and Voice Wiki server are the actors. 

Figure 3: Use Case Diagram

**6.3.2 ACTIVITY DIAGRAM:-**

Activity diagram gives the graphical representation for the flow of the system. It includes all the activities that describe the system flow. This diagram describes the overall flow of controls from the sending of the SMS to callback to user.

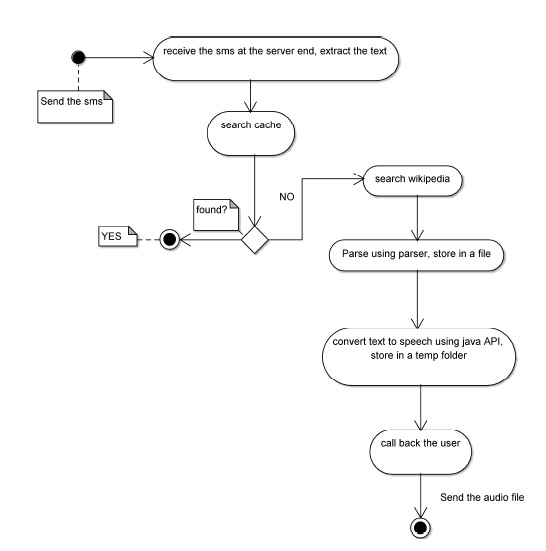


Figure 4: Activity Diagram

**6.3.3 SEQUENCE FLOW DIAGRAM:-**

A Sequence diagram gives the sequential flow of the workflow as well as the life of each component. In this diagram we can see the interaction between the 3 main components, namely the user, system server and Wikipedia server as well as the life of each of these components.

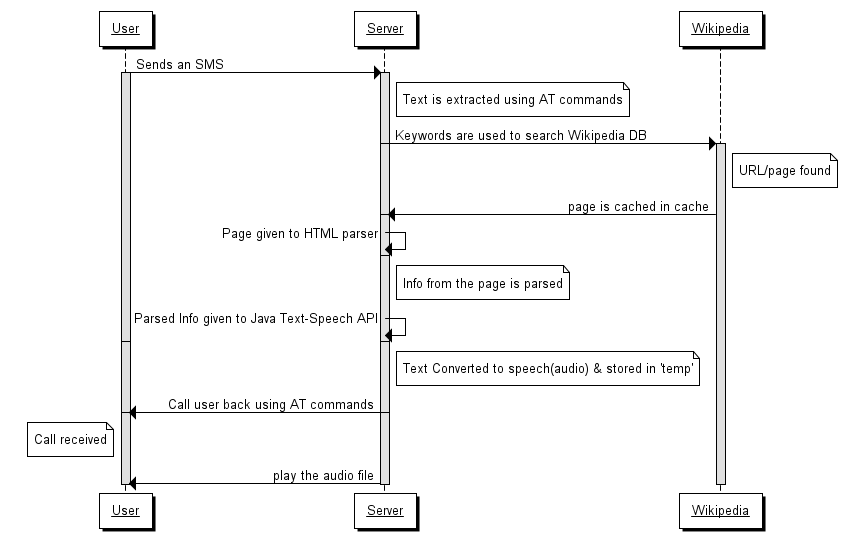


Figure 5: Sequence Diagram

**CHAPTER 7**

**IMPLEMATION DETAILS**

**7.1 PREQUISITES**

* System should be connected to GSM enabled mobile phone which supports AT commands (preferably Motorola phone).
* System should be connected to the Internet with a minimum speed of 64 kbps.

**7.2 INSTALLATION**

WAMP server must be installed on the system. Next the database must be bought online by running the newly installed WAMP server. Connection must be established with the phpMyAdmin. Finally a new database must be created for storing the Wikipedia information.

**7.3 IMPLEMENTATION**

The implementation consists of one main file namely,

***VoiceWiki.exe***

This is the main file which starts the Voice Wikipedia server. It will give an error if the mobile modem is not attached or if it is not connected to the internet. Also the database must be online before this file is executed. Once the file is executed, it will continuously check for messages on the mobile, and once a new message is encountered, it will launch into the code for information retrieval and will resume from the beginning only when the user gets the information through callback

***JAVA Imports***

Package statement defines the name space where all the classes are stored. Any file in this package has its first line for including the package name. Any class declared within that file will belong to that specific package. Packages are good mechanism for compartmentalizing diverse classes from each other.

The packages that are included are

* mysql-connector-java-5.0.8-bin.jar
* VoiceWikiCall.vshost.exe
* Player.jar
* FreeTTSEmacspeakServer.jar
* EmacspeakServer.jar
* MixedVoices.jar

***JAVA Classes***

1. MainClass.java:-

This class runs continuously. It first checks if the there is any new unread SMS by calling the class ReadSMS.java class. After this it stores the keyword and compares it with the entries in the database, and if it finds a match it skips the next part of searching the keyword over the internet and directly calls the user with the information converted to audio format. If a match is not found, it calls VoiceWiki.java after which it stores the summary obtained in the database, converts text to speech by calling MySpeakable.java, and finally calls UserCall.java. The code for main class is as follows.

import java.io.BufferedReader;

import java.io.File;

import java.io.FileNotFoundException;

import java.io.FileReader;

public class MainClass implements Runnable

{

VoiceWiki objWiki;

String[] WikiCOntent = new String[100];

String[] PhnoArray = new String[100];

int i =0;

String Wikitext = null;

VoiceTest objVoiceTest;

public MainClass()

{

ReadSMS obj = new ReadSMS();

string[] args = new String[10] ;

try

{

obj.main(args);

}

catch (IOException ex)

{ ex.pRINTStackTrace(); }

System.out.println("DONE READING");

File file = new File("C:\\tmp\\SMS.txt");

StringBuffer contents = new StringBuffer();

BufferedReader reader = null;

try

{

reader = new BufferedReader(new FileReader(file)); String smstext = null;

String[] arraySMS = null;

while ((smstext = reader.readLine()) != null)

{

arraySMS = smstext.split(",");

WikiCOntent[i] = arraySMS[1];

PhnoArray[i] = arraySMS[0];

i++;

}

}

catch (IOException e)

{ e.printStackTrace();}

finally

{

try

{

if (reader != null)

{ reader.close();}

}

catch (IOException e)

{e.printStackTrace(); }

}

for(int j =0 ; j<i ;j++)

{

System.out.println("in loop" + PhnoArray[j] + ";;" );

objWiki = new VoiceWiki("C:\\tmp\\" + PhnoArray[j] + ".txt", WikiCOntent[j]);

objVoiceTest = new VoiceTest();

objVoiceTest.SpeakNOW(WikiCOntent[1]);

}

}

File Wikifile = new

File("C:\\tmp\\" + PhnoArray[0] + ".txt");

StringBuffer contentsWiki = new StringBuffer();

BufferedReader readerWiki = null;

try

{

readerWiki=newBufferedReader(newFileReader(Wikifile));

while ((Wikitext = readerWiki.readLine()) != null)

{

System.out.println("Wiki text is " +Wikitext );

objVoiceTest = new VoiceTest();

Thread myThread = new Thread(this);

myThread.start();

try

{Thread.sleep(10000);}

catch (InterruptedException ex)

{ ex.PrintStackTrace();}

objVoiceTest.SpeakNOW(Wikitext.trim().

substring(1, 400)); }

}

catch (IOException e)

{e.printStackTrace();}

finally

{

try

{

if (reader != null)

{ reader.close(); }

}

catch (IOException e)

{e.printStackTrace(); }

}

}

public static void main(String[] args)

{

new MainClass();

}

public void run()

{

try

{

Runtime.getRuntime().exec("VoiceWikiCall COM5 wikicontente[2]");

}

}

catch (IOException ex)

{ex.printstacktrace();}

}

2. ReadSMS.java:-

This class checks if there is a new unread message with the help of AT commands and if there is a new message, it reads the message contents and passes it on to the class MainClass.java.

3. VoiceWiki.java:-

This class takes as input the user keyword, parses it for anomalies such as spacing and casing. Then it searches the keyword over Wikipedia, extracts the summary, parses the summary, and stores it in a flat file and gives the control back to MainClass.java

4. MySpeakable.java:-

This class takes as input the summary information from the database and converts into audio format using the free-tts text to speech engine..

**7.4 OUTPUT OF IMPLEMENTATION:-**

The output of executing the file VoiceWiki.exe will be that when a new SMS is delivered to the mobile, it will be read, parsed, checked in the database. Then if a match is not found, the database will make a new entry for the keyword and keyword will be searched over Wikipedia and contents of the summary found over Wikipedia be stored in the database. Then the information will be converted into audio format on the go simultaneously as the user is called and the summary relayed to him as a voiceover from the speakers via the mobile phone.

**7.5 INPUT CHOICES TO USER:-**

The user should SMS whatever he wants to the number assigned to the server. The keyword he sends is not case sensitive nor is it space sensitive because provisions have been made at the server side to take care of these.

The user however has two options that he can make use of

Figure 5: User Input Diagram

1. User can send only the keyword. The output he will get will be the first paragraph of the summary found for his key word over Wikipedia.
2. Alternatively the user can send his keyword and enter the word ”fully” at the end separated by a space. This will return the user the entire summary found for his keyword over Wikipedia.

**7.6 DEPLOYMENT DIAGRAM**

Deployment Diagram shows how the application is deployed and what main components are required at each module for deployment.

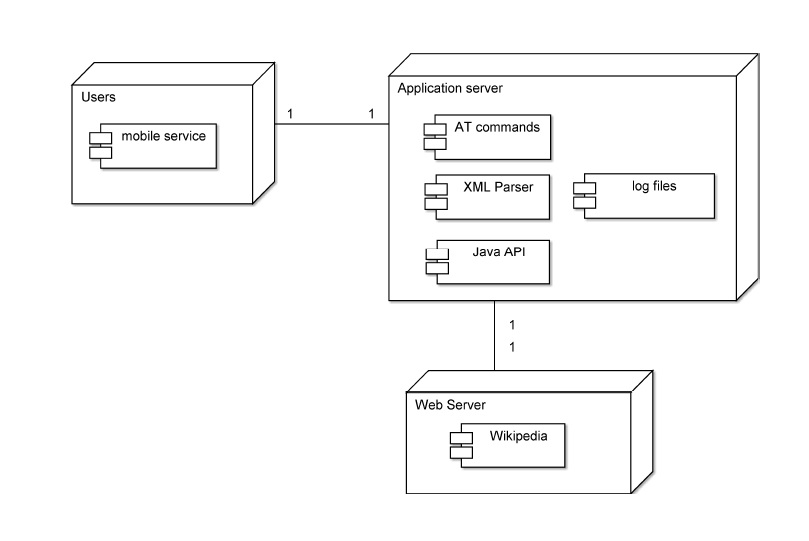
****

Figure 6: Deployment Diagram

**CHAPTER 8**

**TECHNOLOGIES USED**

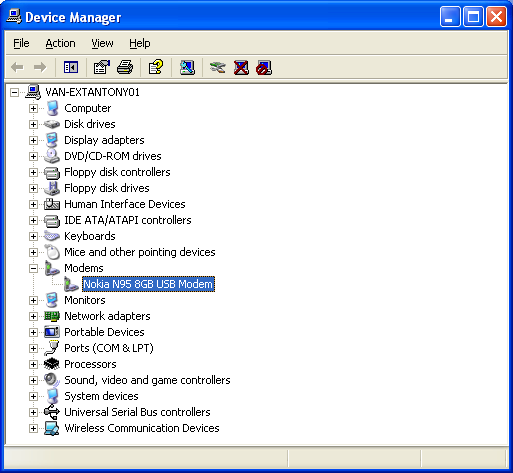
**8.1 AT COMMANDS**

User will send the SMS having keyword regarding the information needed. This SMS will be received by the modem enabled cell phone connected to dedicated server. Then this SMS will be extracted by the server using AT commands. They are called “Attention Terminal” command.

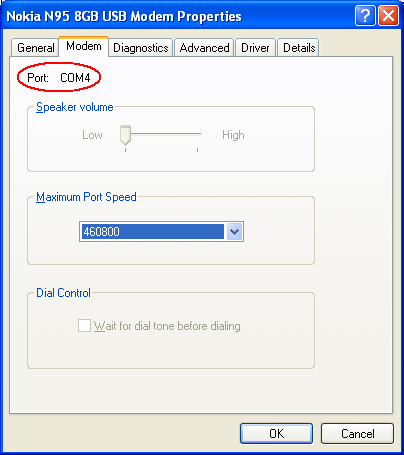
AT commands are used to control modems to do their specified functions. Cellular phones are not much different from the old dial-up modems that are still found in many computers. The commands are sent to the phone's modem, which can be a GSM modem or PC modem. AT commands can be used for operations that are usually done from the keypad.

Steps to use AT Commands:

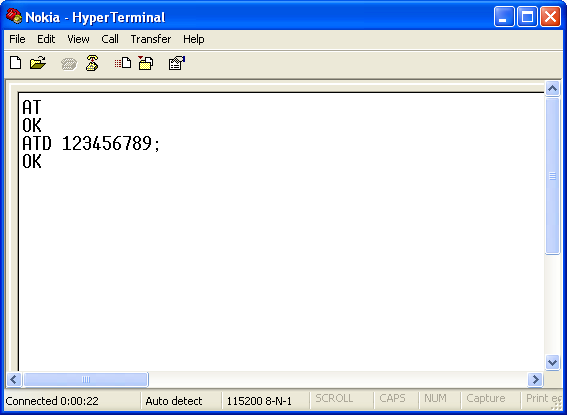
1. Connect the mobile phone to your PC in PC Suite mode using any available connection (Bluetooth, USB, or IR).
2. Make sure that you have installed the correct GSM modem driver on your PC. You can check it from Control Panel | System | Hardware | Device Manager. Check the Modems section. If you see something like "Motorola USB Modem" or "Motorola XY Bluetooth Modem", the device has a built-in GSM modem.

[](http://wiki.forum.nokia.com/index.php/File:Controlpanel_gsmmodem.png)

1. Open the HyperTerminal, which is a communication utility on Windows OS.
   1. HyperTerminal is located in Start | Programs | Accessories | Comm.
2. Create a new connection set on HyperTerminal. You may need to set some parameters, such as baud rate (for example, 9600).
3. Note that you also need to select the communication port of your mobile
   1. The port number is found by right-clicking the modem in Control Panel.

[](http://wiki.forum.nokia.com/index.php/File:Controlpanel_gsmmodem_port.png)

* 1. After this, you can give the basic AT command. Simply type AT in the window and you will get the "OK" response.

[](http://wiki.forum.nokia.com/index.php/File:Hyperterminal_example.png)

1. The above picture also shows how to dial a number using the ATD command. Note that there is a semi-colon (;) at the end of the phone number indicating that this is a voice call. If there is no semi-colon, data call will be performed.

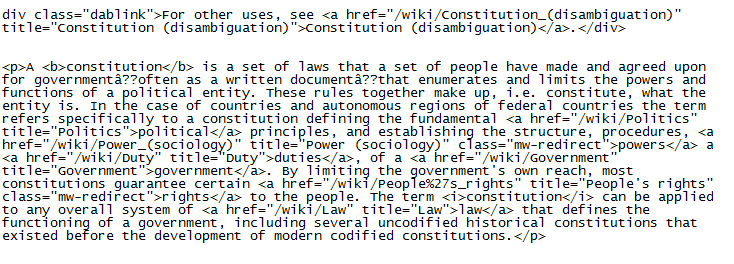
**8.2 EXTRACTION AND PARSING:**

Information is obtained in form of HTML output. This information cannot be relayed to user as it is because of the following problems

* It will contain a lot of information which is cumbersome for the user to listen
* Conversion of such a lot of text into speech will take a lot of time
* Most of the times the main information is contained only in the summary

Testing Wikipedia with keywords

After testing 30 words we saw that summary contains 240-280 words, hence it is ideal to convert while having the necessary information. Also it was observed that the information is most of the time in constant format which makes it easier to parse it as the HTML keywords have a constant format. Information from Wikipedia is in following format:



The summary we have obtained in HTML format will now be parsed, this is done to eliminate the tags and to keep only the relevant summary. This parsing will be done by manipulating strings through java. The final parse will contain only that information which is to be converted in speech

**8.3 TEXT TO SPEECH API**

**8.3.1 What is API**

An Application Programming Interface (API) is an interface that a software program that implements in order to allow the software to interact with it; much in the same way that software might implement a user interface in order to allow humans to interact with it.APIs are implemented by applications, libraries and operating system to define how other software can make calls to or request services from them

**8.3.2 Java Speech API**

After parsing of the web page, information which needs to be delivered to user should be in voice format. This Text to Speech conversion is done using Java Speech API. The Java Speech API Mark-up Language (JSML) and the Java Speech API Grammar Format (JSGF) are companion specifications to the Java Speech API. JSML defines a standard text format for marking up text for input to a speech synthesizer.

The Java Speech API allows you to incorporate speech technology into user interfaces for your applets based on Java technology. Two core speech technologies are supported through the Java Speech API: speech synthesis and speech recognition. In our project’s scope we will be using only Speech synthesis technology supported by Java Speech API. We have used the free tts API in the implementation of our project.

**8.3.3 Speech synthesis using java speech API**

Speech synthesis provides the reverse process of producing synthetic speech from text generated by an application, an applet, or a user

The major steps in producing speech from text are as follows:

* Structure analysis: Processes the input text to determine where paragraphs, sentences, and other structures start and end.
* Text pre-processing: Analyzes the input text for special constructs of the language. In English, special treatment is required for abbreviations, acronyms, dates, times, numbers, currency amounts, e-mail addresses, and many other forms.

The remaining steps convert the spoken text to speech:

* Text-to-phoneme conversion: Converts each word to phonemes(unit of sound).
* Prosody analysis: Processes the sentence structure, words, and phonemes to determine the appropriate prosody for the sentence.
* Waveform production: Uses the phonemes and prosody information to produce the audio waveform for each sentence.

Speech synthesizers can make errors in any of the processing steps described above. Human ears are well-tuned to detecting these errors, but careful work by developers can minimize errors and improve the speech output quality. In this way information related to the query given by User will be converted to speech. This information will now be relayed back to the user using AT commands.

**8.4 WIKIPEDIA**

**8.4.1 Why Wikipedia**

Wikipedia the free encyclopedia that anyone can edit. Wikipedia's articles provide links to guide the user to related pages with additional information. Wikipedia is written collaboratively by Internet volunteers who write without pay (except where editing is restricted). Wikipedia is open to a large contributor base, drawing a large number of editors from diverse backgrounds. This allows Wikipedia to significantly reduce regional/cultural bias. The features why we chose Wikipedia are:

* Assigning Boundaries for information search over the vast internet pool
* It is open source (free)
* It has Regularly updated Database
* It has a constant format, which makes it easier to parse information
* It is Reliable, since it is a moderated website

Errors are generated in Wikipedia if

* + *User enters wrong spelling*

The server cannot do anything in this case except send back a prompt to the user saying that he/she has entered an incorrect spelling

* + *No word exists in the database*

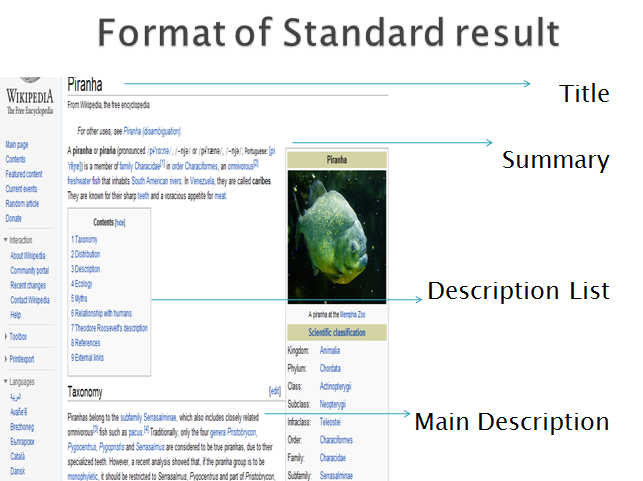
Again the server is not at fault if the word the user asked for is not found in the database. In this case the server will prompt the user that the word he requested was not found in the database.

* + *The word is ambiguous*

Some words have multiple meanings when entered for search.E.g.:- mercury is ambiguous in the sense that it can either mean mercury as a planet or mercury as an element. In this case the user will have to enter the additional information within parenthesis e.g. he can enter either Mercury (planet) or Mercury (Element)

**8.4.2 Format of standard Wikipedia page**

The standard Wikipedia page will contain title, summary, description list, and main description. This will help when parsing the page for information from a particular section

****

Thus, when firing any keyword the output received will be in the same format every time and this will enable the parsing process since the HTML structure of the Wikipedia search page essentially remains the same. In this project, contents of only the summary will be extracted as the entire contents of the page will be too tedious for the user to listen to.

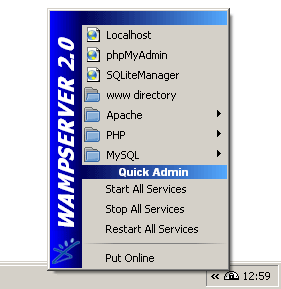
**8.5 Database**

The tool used for database is WAMP server. The version is WampServer2.0i.Wamp Server is a Windows web development environment. It allows you to create web applications with Apache, PHP and the MySQL database. It also comes with PHPMyAdmin and SQLiteManager to easily manage your databases.

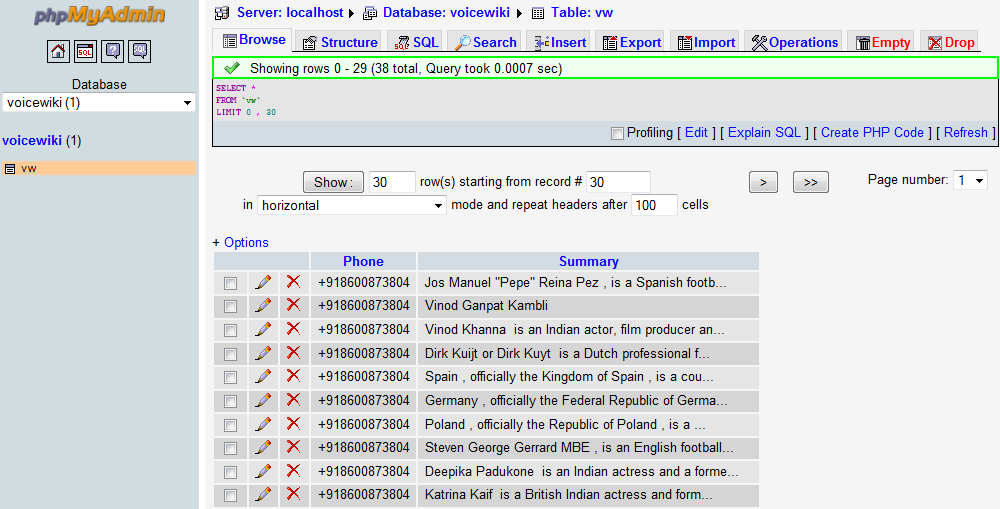
Steps to install WAMP Server

a. Click on Wampserver.exe file.

b. Right click the icon on the taskbar and select the “put online” option.

[](http://tng.lythgoes.net/wiki/index.php?title=Image:WampServer_menu.gif)

c. Click on phpMyAdmin, create a new table in the database. Click browse to view table.



**CHAPTER 9**

**TEST CASES**

# TEST CASES:-

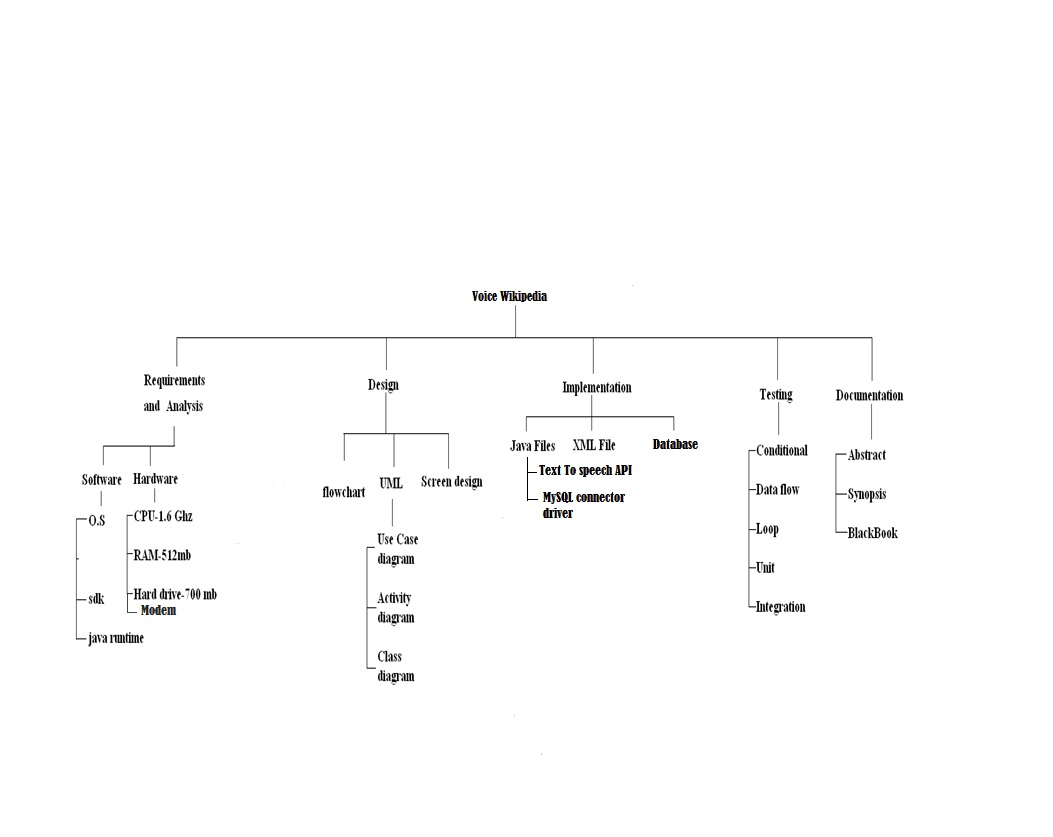
# Testing is the one of the most important phases of the project. It is the process of executing the program with the program with intent of finding an error.

# Test cases:-

|  |  |  |
| --- | --- | --- |
| **Test Cases** | **Expected Result** | **Actual Result** |
| Tested AT commands for retrieving message. | It should read the UNREAD message. | As expected result. |
| Tested the code to separate caller’s no. and message content. | It should extract the keyword and search on Wikipedia. | As expected result. |
| Tested the code for case sensitivity. | It should search Wikipedia irrespective of the cases (upper/lower) used. | Initially it could not search but with some modifications in the code the problem was solved. |
| Tested code for removing the spaces between two words from the keyword. | It should remove the spaces and replace the spaces between 2 words with ‘\_’ . | As the expected result. |
| Tested the code to extract summary (full or less). | It should extract the summary from Wikipedia as per user’s demand | As expected result. |
| Tested Database connectivity. | Database should be updated with the caller’s no. and keyword. | Initially only one entry could be made but with minor changes in the code the database was getting updated. |
| Tested the Text To Speech API. | It should convert the extracted text from text to audio. | As expected result. |
| Tested AT commands to call the user. | It should call back the user and play the audio. | As expected result. |

**CHAPTER 10**

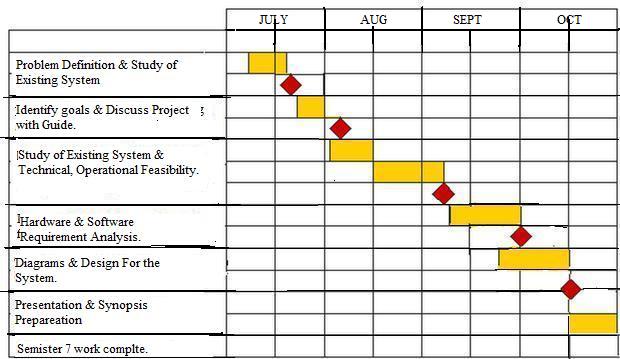
**TASK DISTRIBUTION**



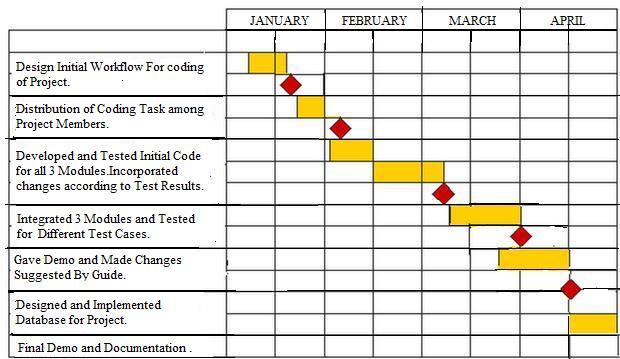
**CHAPTER 11**

**PROJECT TIME LINE**

**11.1 First Half:**

****

**11.2 Second half:-**

****

**CHAPTER 12**

**CONCLUSION AND FUTURE WORK**

**12.1 PROPOSED VS. IMPLEMENTED SYSTEM & FUTURE WORK**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***No.*** | ***Description of module*** | ***Initially Proposed*** | ***Actually Implemented*** | ***Additional Implementation*** | ***Future Prospect*** |
| 1 | Module 1- Receiving message at server end | Continuously check modem end for new messages | C:\Users\Nelson\Desktop\Voice  Wiki\Refernce material\Websites & Images\120px-Symbol_Check.svg.png |  | The server can be an android enabled device instead of a desktop PC |
| Extract user SMS through AT commands and send it to server | C:\Users\Nelson\Desktop\Voice  Wiki\Refernce material\Websites & Images\120px-Symbol_Check.svg.png |  |
| 2 | Module 2- Searching the keyword over Wikipedia | Take the SMS as keyword which HAS to be in a particular format and search it over the internet | C:\Users\Nelson\Desktop\Voice  Wiki\Refernce material\Websites & Images\120px-Symbol_Check.svg.png | Option to receive 1 paragraph wiki information or complete summary | To provide support for abbreviations |
| Removing Case sensitivity of keyword |
| Removing the need to send SMS with words separated by ‘\_’ |
| Removal of un- warranted space |
| 3 | Module 3-  Retrieving the information from Wikipedia | To parse the HTML contents of Wikipedia page to eliminate tags from summary | C:\Users\Nelson\Desktop\Voice  Wiki\Refernce material\Websites & Images\120px-Symbol_Check.svg.png | Naming the corresponding text file with phone number of the requestor. | To inform user if error is encountered in searching his keyword via SMS |
| To store the parsed summary contents in a flat file | C:\Users\Nelson\Desktop\Voice  Wiki\Refernce material\Websites & Images\120px-Symbol_Check.svg.png | To retrieve multiple unread messages |
| Preventing bottleneck at server by employing additional resources |
| 4 | Module 4-  Storing the Wikipedia file in database | To store summary of each requested keyword in database and check for matches before searching for the information over internet | C:\Users\Nelson\Desktop\Voice  Wiki\Refernce material\Websites & Images\120px-Symbol_Check.svg.png |  | To maintain record of keywords a particular user requests |
| To periodically check for changes in wiki page of the entry and update it |
| 5 | Module 5-  Converting text information to speech | To convert the summary contents into audio format | C:\Users\Nelson\Desktop\Voice  Wiki\Refernce material\Websites & Images\120px-Symbol_Check.svg.png |  | Support for languages other than English |
| 6 | Module 6-  To call back the user | To do this using AT commands and relay the information in audio format | C:\Users\Nelson\Desktop\Voice  Wiki\Refernce material\Websites & Images\120px-Symbol_Check.svg.png | Giving the user a short goodbye and thank you message after relaying summary | To end the call from the server side after summary is relayed |

**12.2CONCLUSION**

We undertook the project titled Voice Wikipedia. We analyzed the requirements for the implementation of the project. The project was intended to be one which uses free tools and it has been achieved through the use of JAVA as the programming language for developing the application, WAMP server as the Database tool and free-tts as the voice synthesis tool.

Due to lack of funds, hardware could not be brought to eliminate the bottleneck when calling back the user and hence the application is not highly scalable. However the server has been coded such that it is totally automated and does not require any human intervention, and performs all tasks uninhibited. The basic requirement of the application has been successfully achieved.

**CHAPTER 13**

**REFERENCES**

**REFERENCES**

**13.1 Books:**

7.1.1 Herbert Schildt: “The Complete Reference Java 2” Fifth Edition, TATA McGraw Hill.

.

**13.2 Websites:**

**13.2.1 Requirement Elicitation**

13.2.1.1 <http://trak.in/tags/business/2010/04/07/internet-usage-india-report-2010/>

13.2.1.2 <http://www.dnaindia.com/mobile/report.php?n=1437531>

13.2.1.3 <http://www.e-agriculture.org/uploads/media>

13.2.1.4 <http://www.engineeringproject.org>

**13.2.2 Analysis & Design**

13.2.2.1 <http://www.developershome.com/sms/howToReceiveSMSUsingPC.asp>

13.2.2.2 <http://www.developershome.com/sms/sendSmsByAtCommands.asp>

13.2.2.3 <http://www.control.com.sg/at_commands_sms.aspx>

13.2.2.4 <http://nds1.nokia.com/phones/files/guides/Nokia_AThelp.pdf>

13.2.2.5 [www.java.sun.com/products/java-media/speech/](http://www.java.sun.com/products/java-media/speech/)

13.2.2.6 <http://msdn.microsoft.com/en-us/library/ms720163(VS.85).aspx>

13.2.2.7 [http://www.wave-report.com/tutorials/](http://www.wave-report.com/tutorials/TTS.htm)

13.2.2.8 <http://onjava.com/pub/a/onjava/2004/09/08/IRCinJava.html>

13.2.2.9<http://www.wikipedia.org>

**CHAPTER 14**

**APPENDIX**

**APPENDIX**

**GSM: GLOBAL SYSTEM FOR MOBILE COMMUNICATION**

**EDGE: ENHANCED DATA-RATE FOR GPRS EVOLUTION**

**GPRS: GENERAL PACKET RADIO SERVICE**

**JDK: JAVA DEVELOPMENT KIT**

**TRAI: TELECOM REGIULATORY AUTHORITY OF INDIA**

**JSML: JAVA SPEECH API MARK-UP LANGUAGE**

**JSGS: JAVA SPEECH API GRAMMER FORMAT**

**ACKNOWLEDGEMENT**

We would like to thank our principal, Dr. M. B. Lele for giving us the opportunity to work on this project.

We would also like to thank our H.O.D. Prof. S. M. Patil of Computer Engineering Department, K.G.C.E, Karjat, for his constant encouragement and guidance.

We would like to express our wholehearted gratitude towards our college project guide Prof. Mr.Anup Kunte who helped us a lot by giving constructive criticism and also helped thoroughly in the analysis of the system.

We would also like to thank our co-guide Mrs.Ashwini Borle. A final word of thanks to our classmates, jyoti jain and pooja surnis for the hardware they lent us.

We are very thankful to all of the above, that in spite of their busy schedule, they could take time off to guide us. It is through their initiative and support that we have come so far in our endeavors.