1. INTRODUCTION

In the growing world of technology, cellular devices have quickly emerged as one of the fastest evolving fields. They have increased greatly in both popularity and complexity, requiring more advanced operating systems and applications to meet the demands of the consumer. Android is a software stack for mobile devices that includes an operating system, middleware and key applications. The Android SDK provides the tools and APIs necessary to begin developing applications that run on Android-powered devices. Vocalization is a type of speech synthesis application that is used to convert spoken sound to text and then connects to Wikipedia displays the desired result. And the user can select the textual information and the application converts that text into voice and produces output.

1.1 Purpose of the Project:

The purpose of this application is to make the text readable what the user has selected.

1.2 Scope of the Project:

The scope of the application is to know how the text is converting into voice and voice is converted to text.

1.3. Overall Description

Product Perspective:

Vocalization is a type of speech synthesis application that is used to convert spoken sound to text and then connects to Wikipedia displays the desired result. And the user can select the textual information and the application converts that text into voice and produces output.

Product Functions:

i. **Analysis**

Outputs of all analysis should be in the form of

a. Voice Converted to text.
b. Text converted to voice.

ii. User Classes and Characteristics

All 3 modules uses the user classes and used by the characteristics .The user use modules for only characteristics. The characteristics requirements are completely based on the modules.

iii. Operating Environment

This application will be developed using Eclipse Galileo with software and hardware requirements.

2. LITERATURE SURVEY

2.1 Existing System:

In the existing system generally the user will browse plenty of websites using their mobile, but user has the read the information and at certain point of time the user gets interminable.

The Demerits of the Current System are:

- **a.** Normally if we want to search for particular thing we write in the search input of Wikipedia and then press the search button.
- **b.** Generally we read the results in the webpage displayed manually.
- **c.** The user has to browse plenty of websites for the desired result with their mobile, searching lot of information.
- **d.** Such manual work is prone to errors and the calculation is a tedious task.

2.2 Proposed System:

In the proposed system we are developing an application which will convert Spoken voice to text and search then again textual information into voice i.e., using this app user can give any search request through the voice and can select any info while browsing web and the selected info will be vocalized by the application.

3. SOFTWARE REQUIREMENT SPECIFICATION

3.1 Interfaces

i. User Interfaces:

Interface between the software product and its users:

User friendly interfaces as depicted below will be used.

Screen formats are required to be created with following features:-

a. User Friendly.
b. Indicate the Mandatory fields by asterisk (*).
c. Fill up default values where ever possible
d. Give combo boxes in all input screens.

ii. Hardware Interfaces

Following Hardware Interfaces are required:

a. Processor Pentium-IV
b. Minimum 1GB RAM
c. HDD 40GB
d. Internet Connectivity

iii. Software Interfaces

a. Java 6
b. Android SDK

iv. Communications Interfaces

This application mainly has three modules designed as defined in the Functional Requirements.

3.2 Functional Requirements

Creating a User Interface using Android for demonstrating Vocalization APP

3.2.1 Module-1 Authentication:

a. Description and priority:

In this module, we implement voice to text conversion (i.e., the user's voice is taken as the input and converted to the text).

b. Stimulus/Response Sequences

The response/stimulus for the different classes of users is:

Users: Speaks

c. Functional Requirements

In this we will have the username and password to login to the application.

Input: Voice of the users

Output: Displays the text that user has spoken.

3.2.2 Module-2 Authentication:

a. Description and priority

In this Module converted text is passed as the input to the search engine which connects to Wikipedia website and display the webpage.

b. Stimulus/Response Sequences

i. The response/stimulus for the different classes of users are :

ii. Gets navigated to Wikipedia page representing the search page of the selected text.

c. Functional Requirements

The User can view list of latest vehicles

Input: Text

Output: Wikipedia Search

3.2.3 Module-3 Authentication

a. Description and priority

In this module, text to voice conversion is done i.e., when user selects any contents from web page the application converts that selected text into voice and produces output.

b. Stimulus/Response Sequences

i. The response/stimulus for the different classes of users are:

ii. Listens to the selected text in Wikipedia search page.

c. Functional Requirements

The user can able to see the images of the selected vehicle.

i. **Input:** Selects the vehicle

ii. **Output:** Views the images of the vehicle.

3.3 Non-functional requirements

i. Unit testing, System testing will be done for integrity check and reliability check.

ii. Once the project is deployed it runs around the clock from any remote client using internet.

Maintainability:

All the modules must be clearly separate to allow different user interfaces to be developed in future. Through thoughtful and effective software engineering, all steps of the software development

process will be well documented to ensure maintainability of the product throughout its life time. All development will be provided with good documentation.

Performance:

The response time, Utilization and Throughput behavior of the system. Care is taken so as to ensure a system with comparatively high performance.

Usability:

The ease of use and training the end users of the system is usability. System should have qualities like- learning ability, efficiency, affect, control. The main aim of the project is to increase the scope of page designer to design a page and to reduce the rework of the programmer.

Modifiability:

The ease with which a software system can accommodate changes to its software is modifiability. Our project is easily adaptable for changes that is useful for the application to withstand the needs of the users.

Portability:

The ability of the system to run under different computing environments. The environment types can be either hardware or software, but is usually a combination of two.

Reusability:

The extent to which an existing application can be reused again in new applications. Our application can be reused a number of times without any technical difficulties.

Security:

The factors that protect the software from accidental or malicious access, modification, destruction, or disclosure. Security can be ensured as the project involves authenticating the users.

4. System Analysis and Design

4.1Introduction:

Analysis involves a detailed study of the current system, leading to specifications of the new system. Analysis is a detailed study of various operations performed by a system and their relationships within and outside the system. During analysis, data are collected on the available files, decision points and transactions handled by the present system. Interviews, on-site observation and questionnaire are the tools used for system analysis. Using the following steps it becomes easy to draw the exact boundary of the new system under consideration:

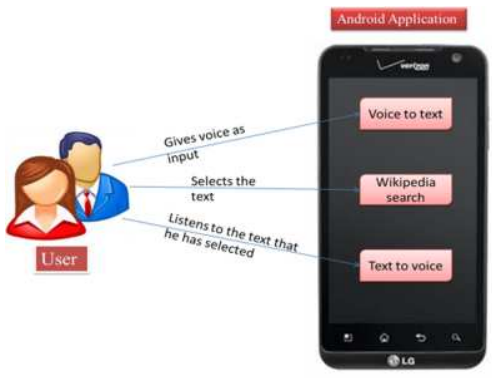
Keeping in view the problems and new requirements

Workout the pros and cons including new areas of the system.

The main points to be discussed in system analysis are:

a. Specification of what the new system is to accomplish based on the user requirements.
b. Functional hierarchy showing the functions to be performed by the new system and their relationship with each other.
c. Function network which are similar to function hierarchy but they highlight those functions which are common to more than one procedure.
d. List of attributes of the entities - these are the data items which need to be held about each entity (record)

4.2 Architecture:



4.3 Development Cycle:

A software development process, also known as a software development life cycle (SDLC), is a structure imposed on the development of a software product. Some people consider a lifecycle model a more general term and a software development process a more specific term. It aims to be the standard that defines all the tasks required for developing and maintaining software.

It consists of the following activities:

i. **Requirements:** In this phase of the SDLC, the requirements for the project are gathered.

ii. **Design:** In this phase, the system is designed using various design tools and is given for approval.

iii. **Implementation:** Once the design is approved, the implementation begins. Implementation is the part of the process where software engineers actually program the code for the project.

iv. **Testing:** Software testing is an integral and important phase of the software development process. This part of the process ensures that defects are recognized as soon as possible.

v. **Deployment:** Deployment starts after the code is appropriately tested, is approved for release and sold or otherwise distributed into a production environment.

vi. **Maintenance:** Maintaining and enhancing software to cope with newly discovered problems or new requirements can take far more time than the initial development of the software. It may be necessary to add code that does not fit the original design to correct an unforeseen problem or it may be that a customer is requesting more functionality and code can be added to accommodate their requests.

4.4 Use Case Diagrams:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.

The various functions (as shown in the Use Case Diagram) provided by the system have been described below:

i. Gives Voice as input: The User's voice is taken as the input and converted to the text.

ii. Views Wikipedia Search: Converted text is passed as the input to the search engine which connects to Wikipedia website

iii. Selects the text: User selects any contents from web page.

iv. Listens: The application converts that selected text into voice and produces output and the User listens to the required selected text.

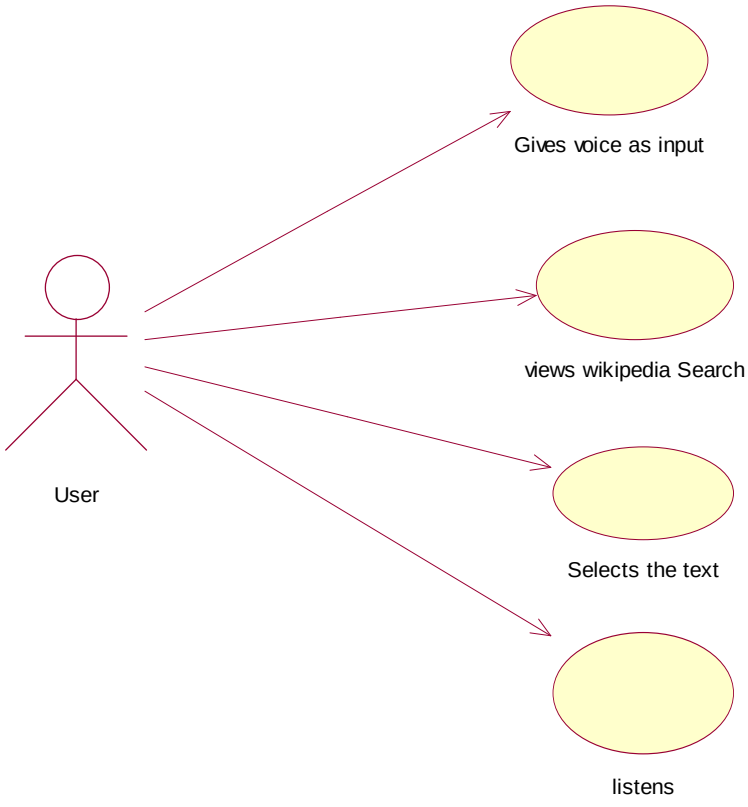


Fig1.1 Use case Diagram for User

4.5 Sequence Diagram:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order.

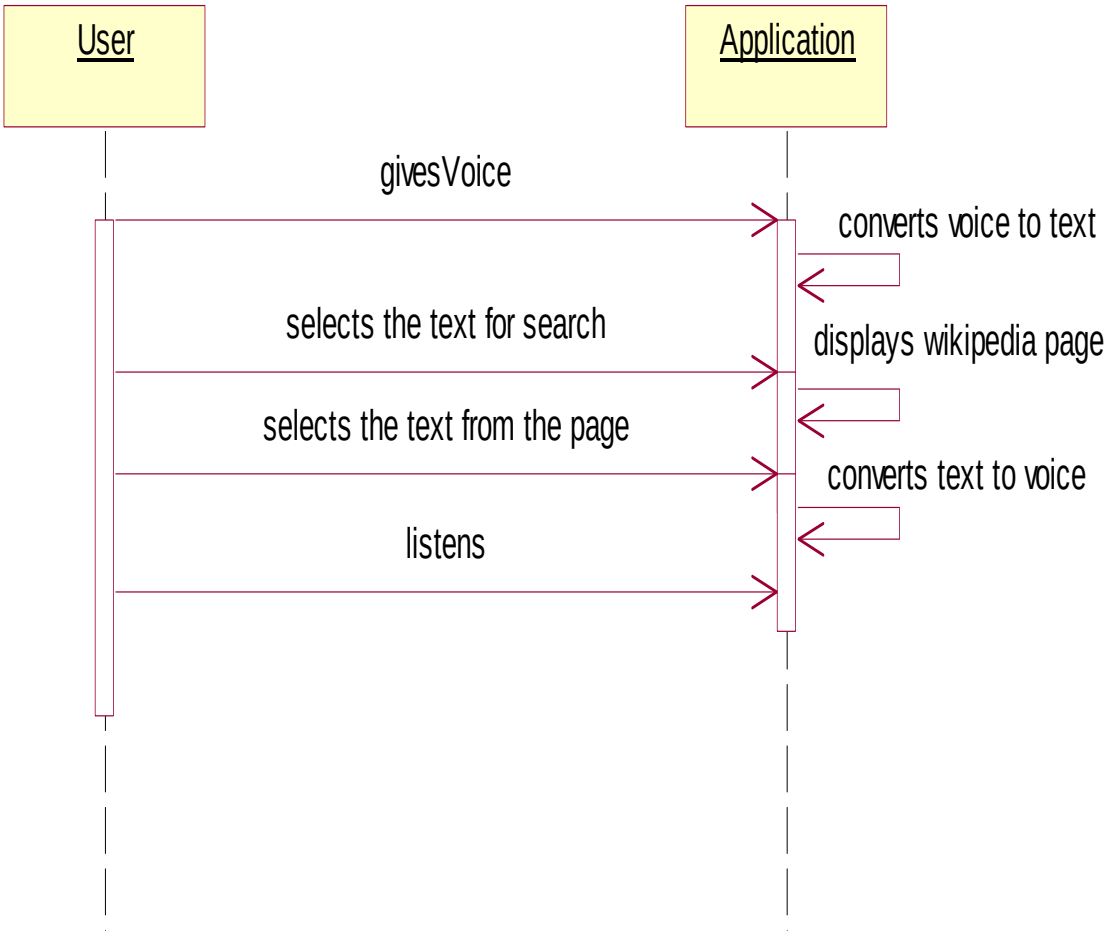


Fig1.2 Sequence Diagram between User and Application

4.6 Class Diagram:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes.

The following figures depict the class diagrams of "Vocalization on Android" categorized into its modules.

Class Diagram for User Module:

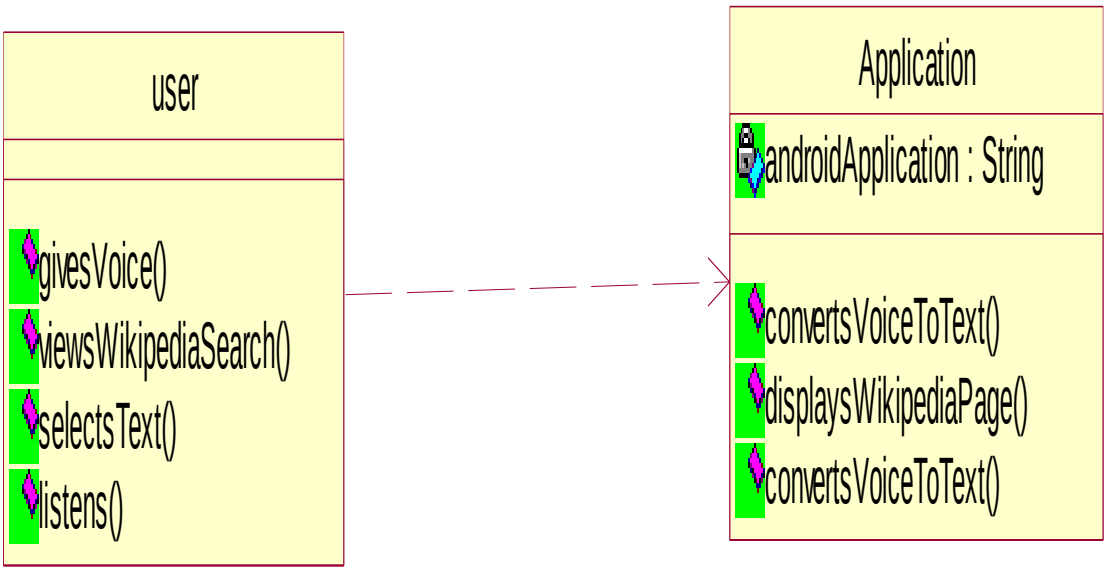


Fig1.3 Class Diagram between User and Application

4.7 Collaboration Diagram:

UML Collaboration diagrams (interaction diagrams) illustrate the relationship and interaction between software objects. They require use cases, system operation contracts, and domain model to already exist. The collaboration diagram illustrates messages being sent between classes and objects (instances). A diagram is created for each system operation that relates to the current development cycle (iteration).

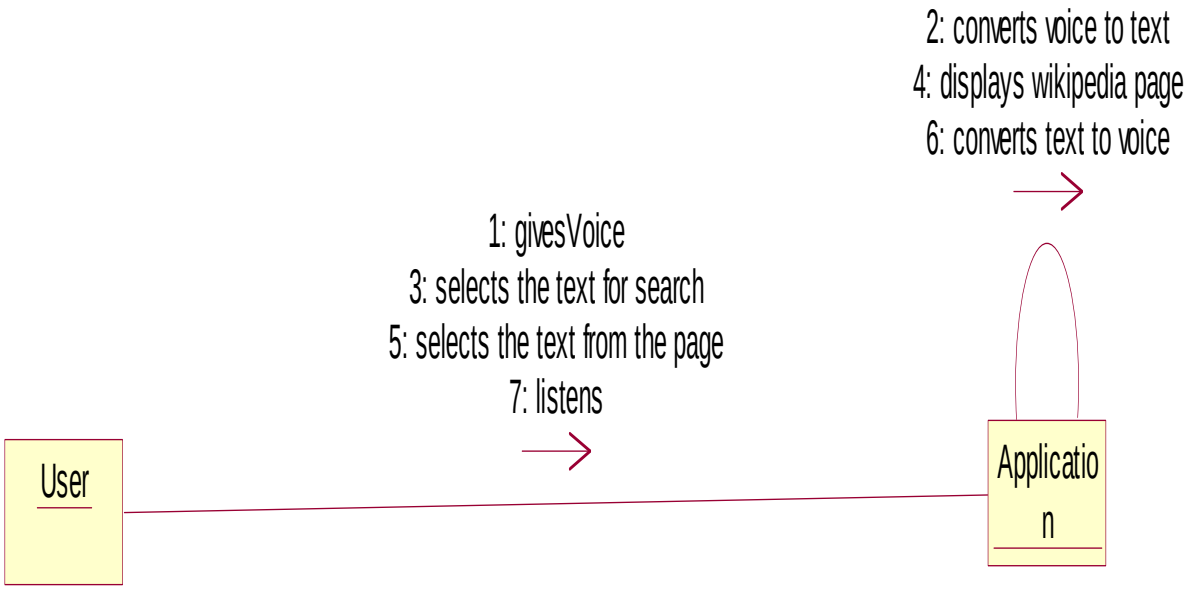


Fig1.4 Collaboration Diagram between User and Application

5. Implementation

5.1 Modules

The project has been categorized into the following three modules:

I. Voice to Text Conversion:

In this module, we implement voice to text conversion (i.e., the user's voice is taken as the input and converted to the text)

II. Displaying Wikipedia Page:

In this Module converted text is passed as the input to the search engine which connects to Wikipedia website and display the webpage.

III. Text to Voice Conversion:

In this module, text to voice conversion is done i.e., when user selects any contents from web page the application converts that selected text into voice and produces output.

5.2 Algorithms:

Speech to Text:

i. A speech recognizer is a speech engine that converts speech to text. *Recognizer* is embedded in the extension
ii. The *Device* will handle the *audio capture* and *streaming* responsibilities.
iii. *Audio* is collected from the *microphone,* and then sent to a Google server (a Gwebservice) using *HTTPSPOST,* which returns a *JSON* object with results.
iv. ***Google*** servers will do the search functionality of speech recognition.

Text To Speech:

a. The quality of a speech synthesizer is judged by its similarity to the human voice and by its ability to be understood. An intelligible text-to-speech program allows people with visual impairments or reading disabilities to listen to written works on a home computer. Many computer operating systems have included speech synthesizers since the early 1990s.

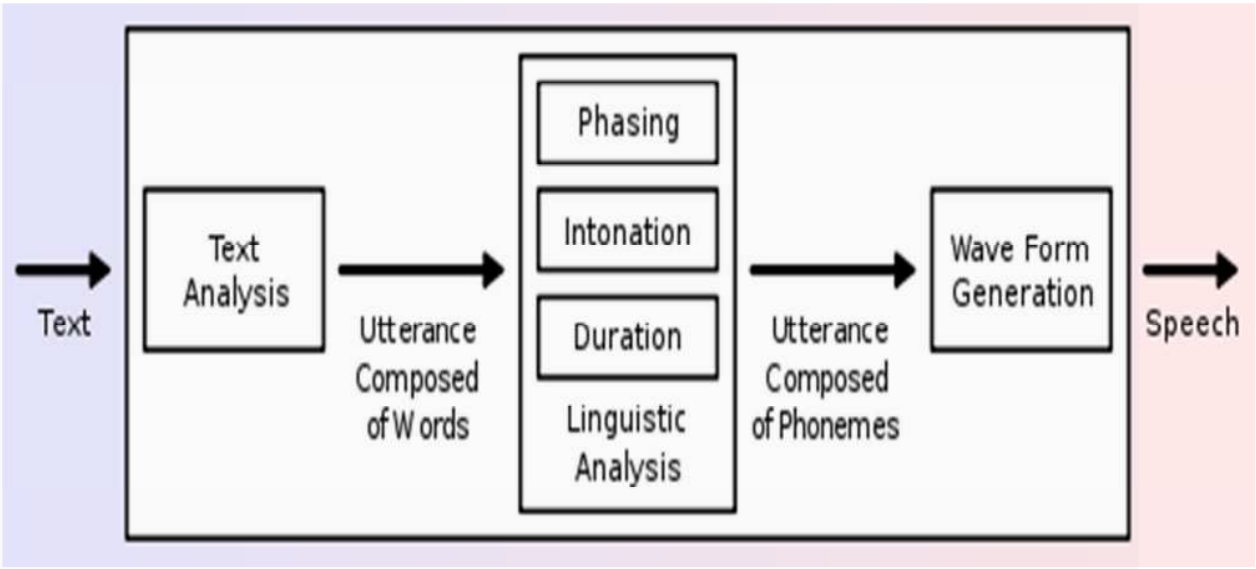


Fig1.5 Text to Speech Processing

b. A text-to-speech system (or "engine") is composed of two parts: a front-end and a back-end. The front-end has two major tasks. First, it converts raw text containing symbols like numbers and abbreviations into the equivalent of written-out words.

c. This process is often called *text normalization*, *pre-processing*, or *tokenization*. The front-end then assigns phonetic transcriptions to each word, and divides and marks the text into prosodic units, like phrases, clauses, and sentences.

d. The process of assigning phonetic transcriptions to words is called *text-to-phoneme* or *grapheme-to-phoneme* conversion. Phonetic transcriptions and prosody information together make up the symbolic linguistic representation that is output by the front-end. The back-end—often referred to as the *synthesizer*—then converts the symbolic linguistic representation into sound. In certain systems, this part includes the computation of the *target prosody* (pitch contour, phoneme durations), which is then imposed on the output speech.

5.3 Integration:

Integration is all about combining the individual parts of the system and making the system into a single unit. Here in this “Vocalization” we are combining the modules i.e., voice-to-text, Wikipedia search and voice-to-text.

6. TESTING:

Compilation Test

It was a good idea to do our stress testing early on, because it gave us time to fix some of the unexpected deadlocks and stability problems that only occurred when components were exposed to very high transaction volumes.

Execution Test

This program was successfully loaded and executed. Because of good programming there were no execution errors. The complete performance of the project “Vocalization” was good.

Output Test

The successful output screens are placed in the output screens section above with brief explanation about each screen.

6.1 Test Cases:

Test Case-1: (Voice Recognition)

i. The user’s voice is taken as input and the voice is converted to text.
ii. If the voice is not recognized correctly then it displays an alert “Speak again..!”
iii. If the internet is not available in the mobile then it gives the alert “Server Problem..!”

Test Case-2 :(Displaying Wikipedia Page)

i. After converting the voice to text, the application navigates to the Wikipedia page displaying the information related to words spoken.
ii. If the internet is not available in the mobile then it displays “Webpage cannot be loaded…!”
iii. Here the user selects the text.

Test Case-3: (Text to Voice)

Selected text is processed and converted to the voice

7. RESULTS:

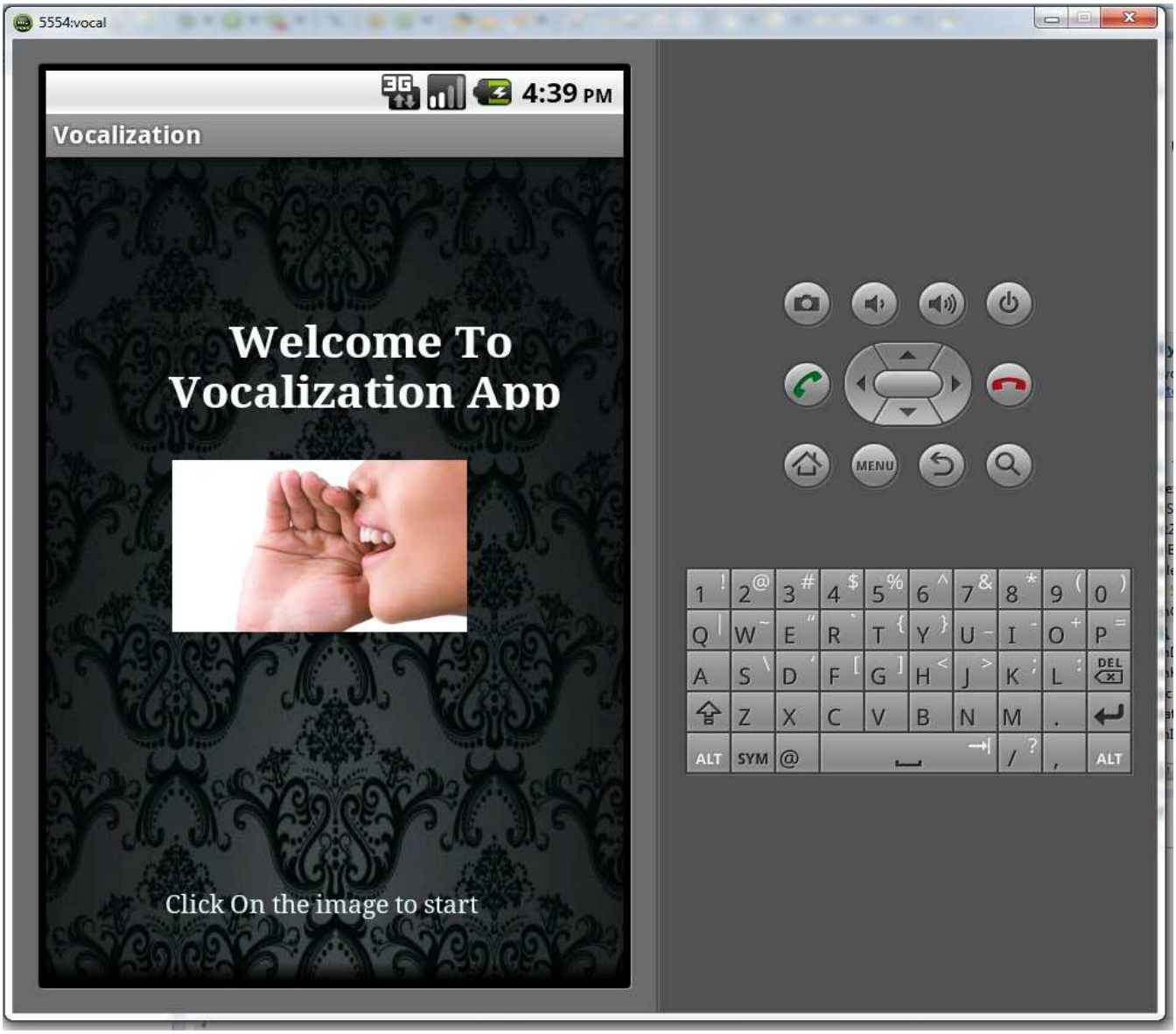


Fig1.6 Screen Shot of the User Welcome Screen.

Gui Design of Voice to Text:

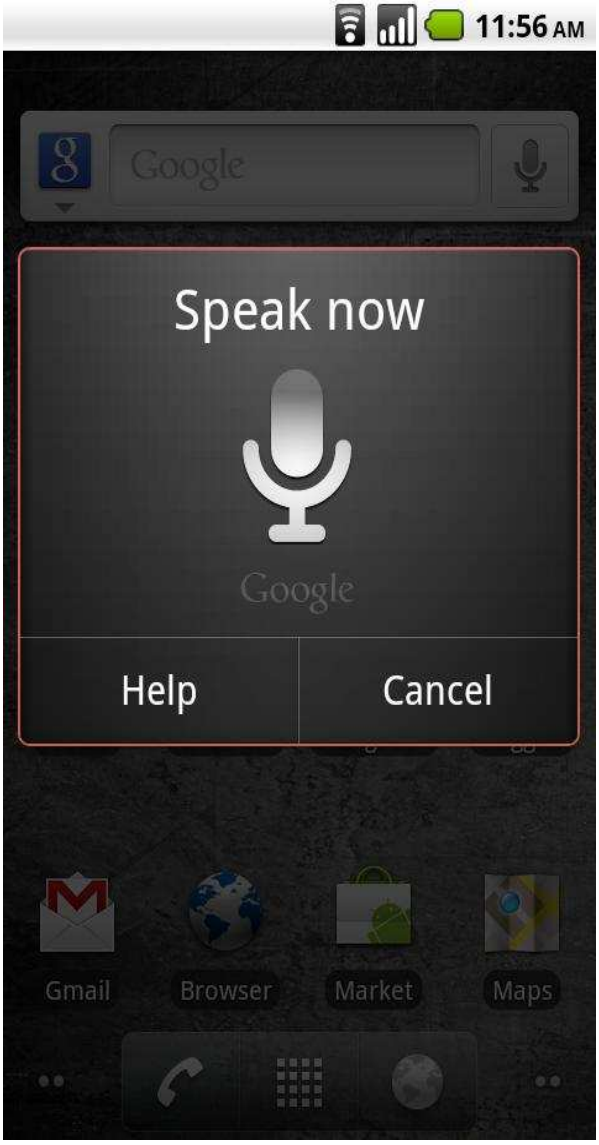


Fig1.7 Asking the User to Speak

Fig1.8 Speech Processing

Result Display:

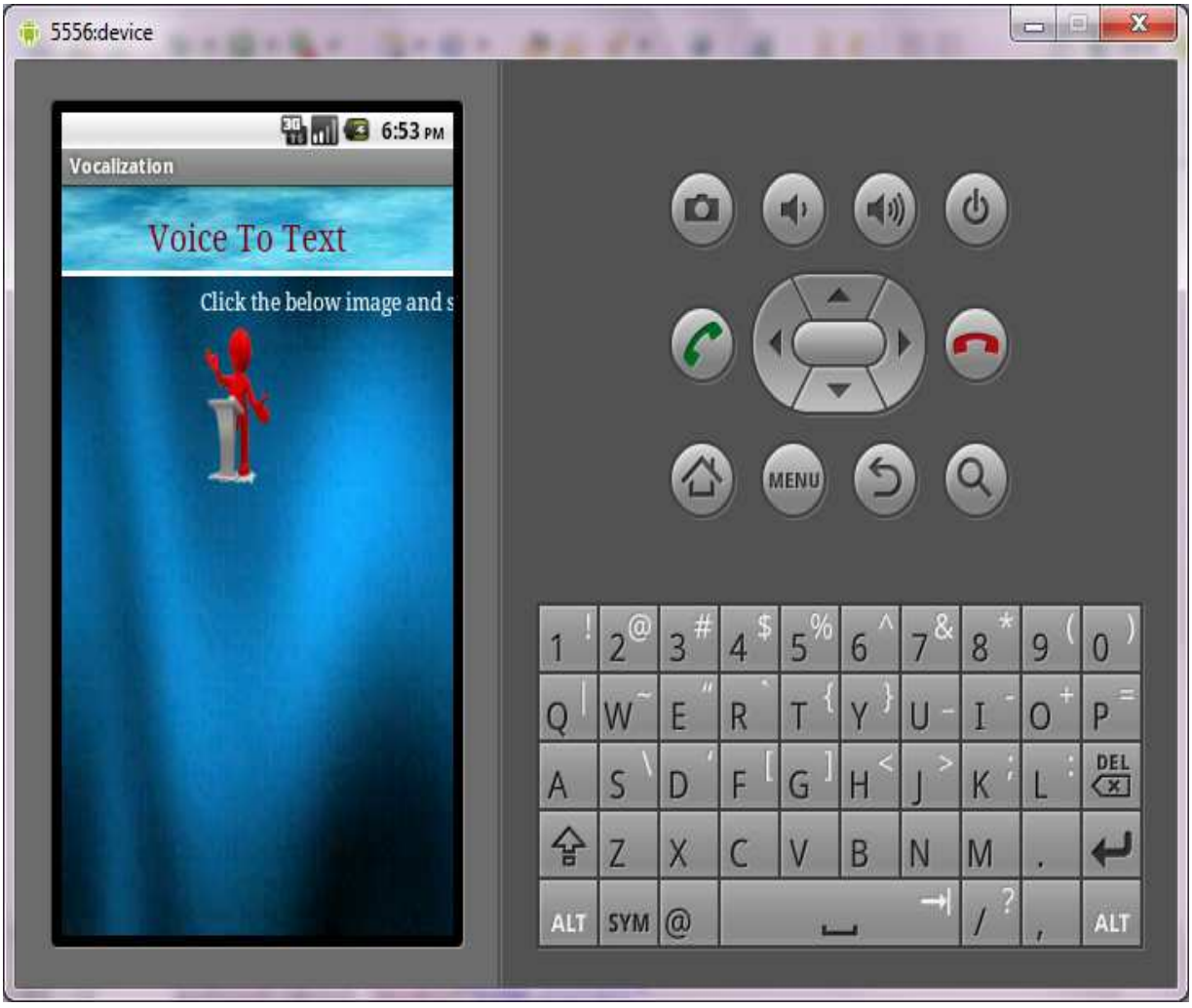


Fig1.9 Result Display

7. CONCLUSION

7.1 Conclusion

Vocalization is an Android Mobile Application. In this application the user's voice is converted into text and displays the words spoken by the user. Then it is navigated to the Wikipedia page which displays the information regarding to the word spoken by the user. If the user doesn't wants to read the page he can select the text and that selected text is converted to voice.

7.2 Future Enhancements:

Interesting Developments

Augmented reality will make possible textual and rich-media conversations in real world too, not just in virtual space. Computers will process speech you hear and generate visual information for you, such as visual maps of arguments (Argument Graphs).

Sub vocal recording (NASA project) is a technology that can record thoughts "spoken" in your mind. It is easier and closer than one might think (2015-2025). Here an advantage of speech recognition is that it's a more direct link between the mind and a computer than keyboard or mouse. The only better technology is a direct brain-computer interface.

Current state

Limited-vocabulary speech recognition is *very good,* and presently expanding into corporate phone trees (implementing voice applications with existing phone system).

Large vocabulary (general) speech recognition still isn't perfect. You still have to speak a little slower, and corrections are necessary. But the computer is pretty good at recognizing context, and letting you correct it and can even learn your language use patterns using your e-mail and document archive. (Flash demonstration of ASR as of Nov 2004, article).

Forecasts

IBM intends to have better-than-human ASR by 2010. Bill Gates predicted that by 2011 the quality of ASR will catch up to humans. Justin Rattner from Intel said in 2005 that by 2015, computers will have "strong capabilities" in speech-to-text.

REFERENCES

Books:

i. The Unified Modeling Language User's Guide - Grady Booch.

ii. Software Engineering, A practitioner's approach
-Roger S Pressman.

iii. Software Project Management
-Walker Royce.

iv. The applicable IEEE standards as published in 'IEEE standards collection, for the preparation of SRS'.

v. Backup policy, Naming Conventions as per Teleparadigm Conventions.

Websites:

- http://developer.android.com/guide/components/index.html
- http://docs.oracle.com/javaee/6/tutorial/doc/
- http://android-developers.blogspot.in
- http://www.codeproject.com
- http://www.wikipedia.org
- http://future.wikia.com/wiki/Speech_Recognition