Smart Stick for Blind

Submitted in fulfillment of the requirements of the degree of

Bachelor of Engineering

By:

Sanveg Rane (1311036)

Hardik Shah (1311049)

Dhiraj Uchil (1311061)

Supervisor:

Prof. Prasanna Shette

Department of Computer Engineering

K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

2016-2017

Project Report Approval for B.E.

Project Report entitled Smart Stick for blind

Submitted by:
Sanveg S Rane (1311036)
Hardik R Shah (1311049)
Dhiraj B Uchil (1311061)

is approved for the degree of Bachelor of Engineering.

Examiners

1.--

2.--

Supervisors

1.--

Date:

Place:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Name of student and Roll No.)	-- (Signature)
(Name of student and Roll No.)	-- (Signature)
(Name of student and Roll No.)	-- (Signature)

Date:

Abstract

The term blindness is used for complete or nearly complete vision loss. Blind people need some help while travelling to feel safe along with proper guided navigation. In this project, we have developed a smart stick which increases the accessibility of blind person to move around by providing alerts about the potholes or obstacle that might occur on his path while travelling. The stick developed in this project, called 'Smart Stick', uses pothole detection algorithm and obstacle detection algorithm to detect potholes and obstacles within a fixed range close to the user. This smart stick is cheap, lightweight and fast. The stick consists of 3 ultrasonic sensors for obstacle detection and 1 ultrasonic sensor for pothole detection. The android application is linked with ultrasonic sensors for notification of obstacle and pothole. The application also consists of real time navigation system with speech recognized destination, automatic launch, Gesture detection with speed dial, emergency alarm etc. The project makes an effort to help blind and visually challenged people travel safely without help of others.

Acknowledgement

We would like to express our gratitude and appreciate all those who gave us the opportunity and help to complete this project successfully. A special thanks to our project mentor, Prof. Prasanna Shette whose help, suggestions and encouragements helped us co-ordinate our project along with the reports and technical papers efficiently.

We would also like to thank all the faculty members of the Computer Department who guided us along with important timely suggestions for improvements and new features in the project. Their help proved pivotal for necessary implementations and improvements during the entire course of our project.

The project was a great experience and provided many opportunities to learn new concepts in computer science along with their applications. We are indebted of those who provided assistance and reviews to our task and apologize to anyone if may have failed to mention.

Index

Chapter 1: Introduction

1.1 Introduction and Motivation

As of 2012, there were nearly 285 million people who were visually impaired of which 246 million had low vision and nearly 39 million were blind. Blind people often have problems while moving around in an environment where they are not familiar. They use the traditional white cane. This cane detects obstacle only when they touch it and hence prior detection of obstacle is problem. The cane also cannot properly detect obstacle which are at certain height. Thus, blind people will feel confident to move around only when obstacles are known from far distance. This can be done with help of Ultrasonic sensors. There are various types of technologies and sensors available but ultrasonic sensors are chosen because they are cheap and light weight and can detect obstacle up to 400cm. There is also a possibility that blind person might not know the route or might have some emergency. These problems can be taken care with real time navigation and gesture recognition. Smart stick will help the blind person in easy mobility just like normal person.

1.2 Existing system

An Electronic Travel Aid is form of Assistive technology for enhancing mobility of blind person. The research problem of designing a better ETA is tough one. Blind people find travelling difficult because they cannot determine where obstacles are. This process is also called as 'Spatial Sensing'. There are various problems with existing system. First, the rangefinder technology is unreliable in detecting steps. Second, Blind people find various sound pitches and vibrations difficult to understand. Third, these systems are quite expensive and since blind person have to depend financially on someone, they don't feel worth to invest so much. Problems and various existing systems:

1. Walkmate – Developed in 1993. It had detection only up to 1.83m.
2. Miniguide US- Developed in 2004. The price was $545 which is quite expensive.
3. Laser cane- The latest one is named as 'N2000'. It provides 3 beams straight,

overhead and for downward drops. It is still available and is priced at $2650.

The developed smart stick has range of 400 cm and it costs less than 35$.

1.3 Problem Definition

Though many steps are taken to help the visually disabled commute, still they are not completely reliable and as helpful for the blind. The stick is only used as a support and it is difficult for the user to detect obstacles and potholes. Thus, a smart stick is developed with aim of detecting obstacles and potholes along with accessibility features to help the blind person. The obstacle and pothole are detected by the sensors and output is given to user in the form of voice. Blind people also have problems to navigate around if they are not familiar with the environment. Hence, the app also has feature to provide navigation to the user with destination taken in the form of voice input. There is also a possibility that in case of emergency the blind person needs to make a call but has difficulty in dialing the number. Thus, the accessibility features include gesture recognition i.e. to make emergency calls just by drawing a particular gesture on screen.

1.4 Scope

The application will provide the following functions:

- Detect potholes and obstacle by using the pothole detection and obstacle detection algorithms respectively.
- Alert users about potholes through vibrations and voice output.
- Guide users for proper directions through detection of obstacle and associated voice outputs.
- Navigation using location service for user’s start location and voice based input for destination.
- Gesture based emergency features.

This application is developed using android operating system since it is most easily available operating system in today’s world. Android application can be developed easily and made available through online Android PlayStore.

The hardware was developed using Arduino UNO, which is most widely used microcontroller for controlling multiple sensors simultaneously. SR-04 ultrasonic sensors used for distance measurement. The SR-04 sensors are cheap and widely used sensors and their multiple concurrent use is simple. The low costs of Arduino and SR-04 reduce the overall price of the hardware in the system.

This project includes development of smart stick for blind. It contains three ultrasonic sensors on the top of the stick and one ultrasonic sensor on the bottom. The sensors on the top are used for obstacle detection while the one at the bottom is used for pothole detection. These sensors are connected to Arduino which is connected to the Android App. The app receives the data from Arduino and gives voice feedback to the user. The app also has accessibility features like gesture recognition for emergency calling and voice navigation to help person move around.

1.5 Organization of Report

The following chapters have been included in the report:

1. Introduction: The chapter justifies and highlights the problem posed, defines the topic and explains the aim along with the scope of the work presented.

2. Literature Review: A critical appraisal of previous works published in the topic of interest have been presented. Various features required in the project studied from various range of papers previously published.

3. Project Management Plan: The section portrays how the project development process was planned and managed.

3.1 Feasibility analysis: The analysis of feasible of the project in terms of cost, technicality, software and hardware aspects.

3.2 Life cycle model: The Iterative Lifecycle model decided to be most suitable.

3.3 Project cost and time estimation: The cost of the hardware required and other utilities decided and an estimation of time allocation done for implementing features.

3.4 Resource plan: A general review of resources required and a plan for their usage presented in the topic.

3.5 Task and Responsibility Assignment: A the topic consists of a table that depicts how tasks and responsibilities were assigned to each member of the group.

4. Project Analysis and Design: The section gives an overview of the designing phase of the application.

4.1 Software Architecture diagram: The diagram of software architecture discussed here.

4.2 Architectural style and justification: The topic presents a justification of the architectural style used and modules included in the architecture diagram.

4.3 Software Requirements Specification Document: The document containing functional and non-functional requirements, resource requirements, hardware and software requirements, etc. has been attached here. Various use case diagrams like Class diagram, use case diagram, State diagram, DFDs have been explained here.

4.4 Software design document: Contains the User Interface design and component diagram explaining the software design of the project.

5. Project Implementation: This section gives an idea of how the application was developed and executed.

5.1 Approach / Main Algorithm: A description of the 'Pothole detection algorithm' and 'Obstacle detection algorithm' given in this topic.

5.2 Programming Languages used for Implementation: A list of programming languages used for various purposes has been presented in this topic.

5.3 Tools used: A list of various tools and hardware components used during the implementation of the project presented here.

5.4 Deployment diagram: The deployment diagram used by the project team has been portrayed here.

6. Integration & Testing: Once the application has been developed, it needs to be tested for errors and bugs. This section describes the testing approached followed during the testing process.

6.1 Testing Approach: The methodology used for testing each module discussed under this section.

6.2 Test Plan: This gives an idea about the testing procedure carried out and tasks involved in the project.

6.3 Unit Test Cases: The outputs of various modules on testing individually have been discussed.

6.4 Integrated System Test Cases: The output of the entire system as a whole with all modules functioning have been discussed.

7. Conclusion and Future work: Provide the possible future features and works that can be implemented on the existing system.

8. References: This section provides references to all the websites, books, journals, papers referred during the analysis, planning and implementation phases of the project.

Chapter 2: Literature review

Distance Measurement of an Object or Obstacle by Ultrasound Sensors using P89C51RD2:

Distance measurement has various applications like detecting an obstacle or checking presence of pothole. This paper discusses some distance measurement techniques and suggests the use of Ultrasonic Sensor which is efficient and cheap as well. This paper gives the technical details and characteristics of sensors. Ultrasonic transducer uses the physical characteristics and various other effects of ultrasound of a specific frequency. The Ultrasonic wave propagation velocity in the air is approximately 340 m/s at 15°C of air or atmospheric temperature, the same as sonic velocity. Based on these details, distance calculation is suggested as follows:

The time from signal transmission till the signals received are measured.

The measured distance is calculated on the basis of travel time.

The formula to calculate the distance:

Distance (cm) = (Travel Time*(10^-6) * 34300) / 2

Some errors may occur while calculating distance due time period between the starting of the timer1 and actual time of the transmission of the first pulse of the burst pulse train by the ultrasonic transmitter and the reflected signal received by the ultrasonic receiver sensor is passed on to the receiver amplifier, which amplifies it and generates the interrupt signal INT2.

Advanced guide cane for the visually disabled people:

The paper focuses on developing a new type of cane that can be used by blind and visually challenged people for navigating indoors and outdoors as well. The cane consists obstacle detection system and a gps navigation assistance. The setup implemented by using rechargeable power sources and with low power consumption.

For GPS navigation, the data is read from GPS, the coordinates are compared and the user is informed. In case of Obstacle detection, the echo waves are received using ultrasonic sensors and sent to Raspberry pi to calculate the distance.

Android text messaging application for visually impaired:

The paper discusses a speech-to-text methodology that can be implemented to enable visually impaired people to write SMS using speech. The technology can be used to convert user speech to text when the user enters the data he needs to send. An android application is developed which uses speech recognition and conversion to implement speech-to-text.

Application mentioned in the paper will always be in running state at the background once it is started. It is built on top of SMS, so that once application is installed on mobile, all SMS related activities are by default performed by application. With respect to user perspective, application working is divided in two ways – One application is used for sending messages and other when application is used to read received messages. For sending message, the voice command is provided to open application to send message. Once application is opened, it will ask for contact of receiver, then it will ask for the message to be sent, then it will speak that message to check, after conforming the message it will send it to corresponding receiver. Every time the application asks anything, through voice and user also provides response with voice commands that are told by guide.

Chapter 3: Project Management Plan

3.1 Feasibility Analysis

The proposed system consists of an android phone, a stick and four ultrasonic sensors connected to the Arduino. The stick is made is made of polycarbonate plastic which makes it reliable and feasible for any kind of roads and weather. The Arduino and sensors are easily available and can be easily connected making the project technically feasible.

Every individual is assumed to own an android phone which is used to alert and navigate the user. The cost of project is around thousand rupees thereby making it feasible by cost.

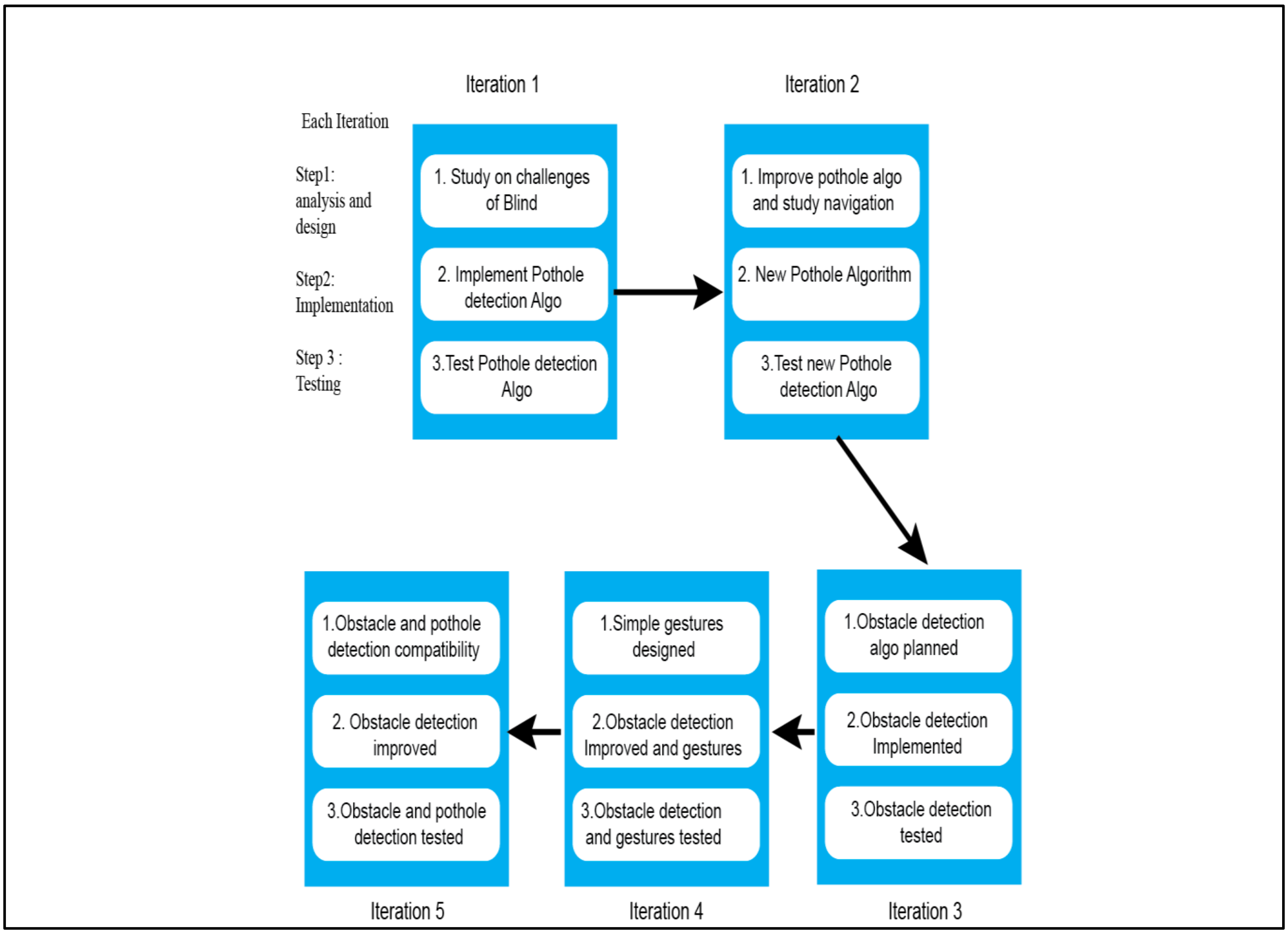
The entire project is implemented in Arduino Uno and Android. Both are open source and thus a lot of updates and improvements can be provided thereby making it optimal and much more feasible in terms of software.

The app is tested on a lot of phones and is made sure that it works on all kinds of phones. The relation between Android and Arduino and between Arduino and ultrasonic sensors is also reliable. This makes the project feasible in terms of hardware.

Thus, the project is feasible in terms of cost, technicality, software and hardware. Hence, this project provides a safe, reliable and cost effective alternative to the traditional canes or sticks used by visually challenged people.

3.2 Life Cycle Model

The Iterative life cycle model decided to be used for implementation of project. Initially basic features were added and more were added in the later iterations.



First Iteration:

Analysis & Design: Various challenges faced by blind people were studied and the basic design was created.

Pothole detection algorithm was designed by studying already available distance sensing algorithms.

Implementation: The Arduino was connected to android using USB. pothole detection algorithm was implemented.

Testing: Pothole algorithm was tested for various depth.

Second Iteration:

Analysis & Design: The pothole algorithm design was improved to calibrate according to the height and habit of user. Also challenges of Navigation was studied.

Implementation: The new pothole algorithm was implemented. Turn based navigation was implemented where user can enter his destination by voice. 3 more sensors were interfaced with Arduino for obstacle detection.

Testing: The new pothole algorithm was tested by holding stick at different heights.

Third Iteration:

Analysis & Design: Obstacle detection algorithm was designed using the 3 sensors. Study of simple gestures was started.

Implementation: Obstacle detection algorithm was implemented.

Testing: Obstacle detection algorithm tested for different angles.

Fourth Iteration:

Analysis & Design: Simple gestures were designed to perform quick action like calling a person. Obstacle detection design was improved.

Implementation: Obstacle detection algorithm was improved and gestures were implemented.

Testing: Obstacle detection algorithm tested for different angles and different distance of obstacles,

Gestures were tested for proper functioning.

Fifth Iteration:

Analysis & Design: Obstacle detection modified to work well with pothole detection.

Implementation: New Obstacle detection algorithm was implemented.

Testing: Obstacle detection algorithm tested for different angles and different distance of obstacles.

3.3 Project Cost and Time estimation

The following table shows the components used and the price of each component.

Sr.no	Components	Cost (Rs.)
1	Arduino	400
2	HC-SR04 Ultrasound Sensor X 4	125*4
3	Jumper Wires	100
4	9V battery	20
5	Battery holder	20
6	Battery Connector	10
7	Mini Breadboard	90
	Total	1140

The following table shows the time required to complete the project:

Phase	Description	Time
Phase 1	Analysis of challenges faced by blind people. Components required and designing of project	July (1 month)
Phase 2	Distance from sensors and implementation of pothole detection	August (1 month)
Phase 3	Building App to show distance between pothole and sensor and implementation of GPS navigation	September-October mid (1.5 months)
Phase 4	Interfacing of 4 sensors and implementation of obstacle detection algorithm	December - January (2 months)
Phase 5	Implementation of Gesture Recognition and mounting on stick	February (1 month)
Phase 6	Testing and final design of user interface	March (1 month)

Total time required to complete this project is seven and half months.

3.4 Resource plans

The following resources are required and used:

Resources	Detail	Used
Financial Resources	Budget of 1500	used in July 2016 to buy components
Inventory Resources	Components of project	Used as per requirement
Human Resources	Developers	from start to end
Sales and Production Resources	Mass production and selling of project	NA

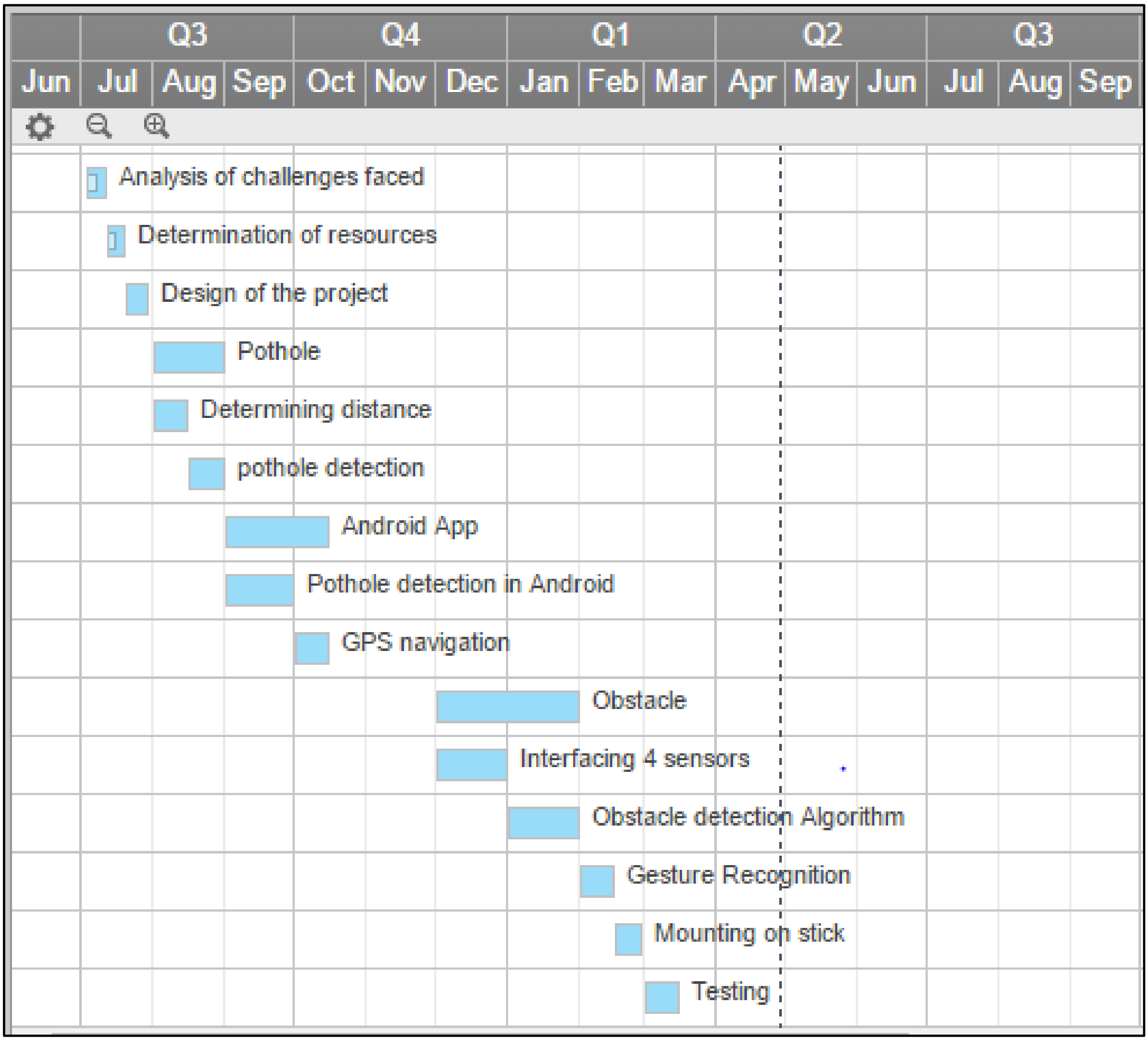
3.5 Task & Responsibility Assignment Matrix

The following table shows the task and the responsibility of that task:

Task Name	Responsibility
Analysis of challenges faced	Dhiraj, Hardik
Determination of resources	Sanveg, Dhiraj
Design of the project	Sanveg, Dhiraj, Hardik
Determining distance	Hardik, Sanveg
Pothole detection	Dhiraj, Sanveg
Pothole detection in Android	Hardik, Dhiraj, Sanveg
GPS navigation	Dhiraj, Sanveg
Interfacing 4 sensors	Sanveg, Hardik, Dhiraj
Obstacle detection Algorithm	Sanveg, Hardik, Dhiraj

Gesture Recognition	Sanveg, Hardik, Dhiraj
Mounting on stick	Sanveg, Hardik, Dhiraj
Testing	Sanveg, Hardik, Dhiraj

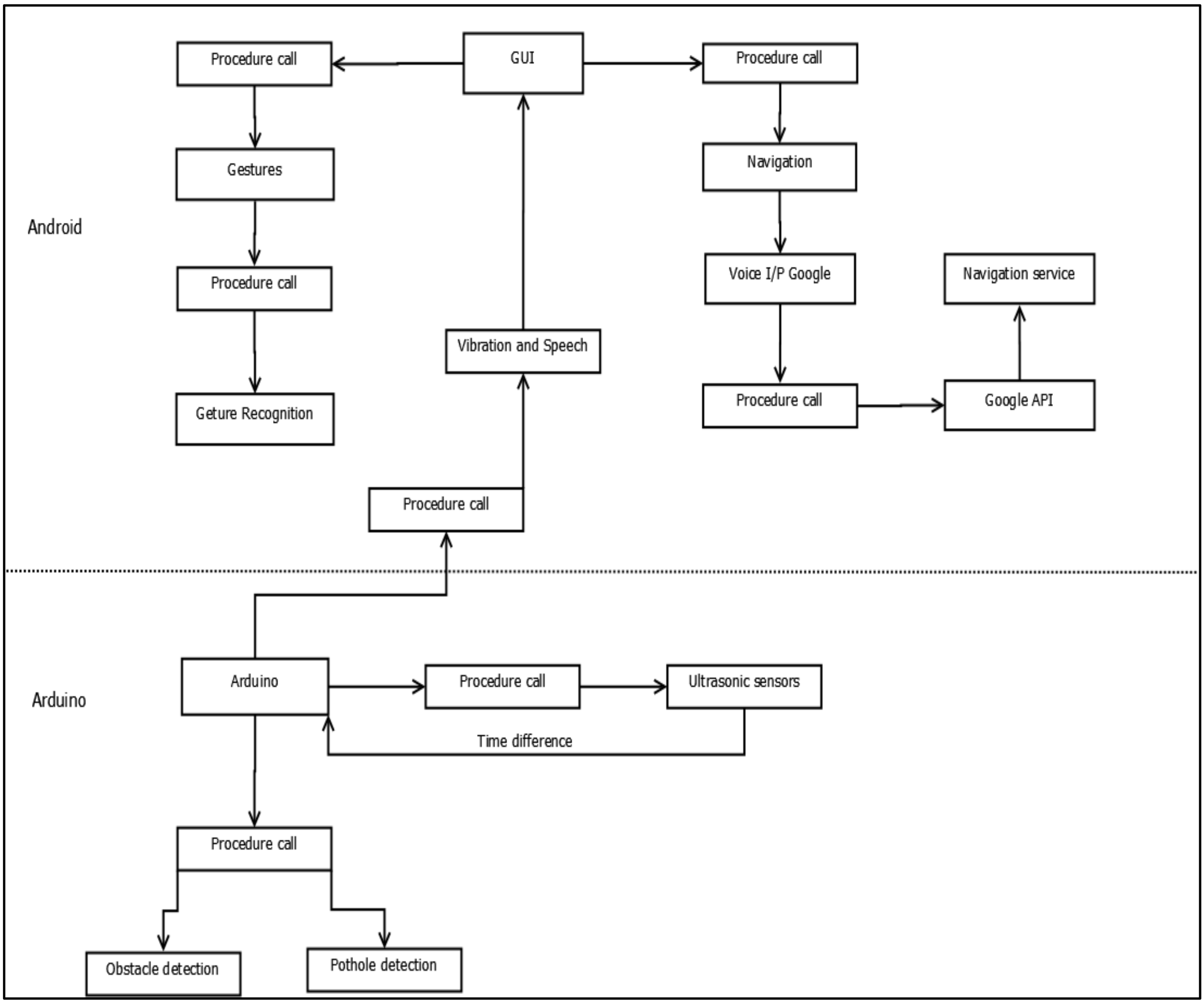
3.6 Project Timeline Chart



Task Name	**Start Date**	**End Date**	**Status**
Analysis of challenges faced	07/03/16	07/11/16	Completed
Determination of resources	07/12/16	07/19/16	Completed
Design of the project	07/20/16	07/29/16	Completed
Determining distance	08/01/16	08/15/16	Completed
Pothole detection	08/16/16	08/31/16	Completed
Pothole detection in Android	09/01/16	09/30/16	Completed
GPS navigation	10/01/16	10/15/16	Completed
Interfacing 4 sensors	12/01/16	12/31/16	Completed
Obstacle detection Algorithm	01/01/17	01/31/17	Completed
Gesture Recognition	02/01/17	02/15/17	Completed
Mounting on stick	02/16/17	02/27/17	Completed
Testing	03/01/17	03/15/17	Completed

Chapter 4: Project Analysis and Design

4.1 Software Architecture diagram



4.2 Architectural style and justification

Layered Architecture:

The architecture used is layered style. Two layered are used, Android and the Arduino.

There are various functions in Android. Based on functions selection, appropriate procedure call is made. If Navigation is selected, then first the procedure call of voice input is called, to enter the destination, then the procedure for Google Api is called to start the navigation.

If Gesture Activity is called, the procedure call for gesture recognition is done. Based on gesture drawn, the appropriate number is called.

Considering the second layer, as soon as the app is launched in android, the Arduino starts reading the data through all four sensors. For each reading in ultrasonic sensors, proper procedure call is made. The procedure call for pothole distance sensing and obstacle distance sensing are different. These sensors sense the distance and send the data whether a pothole is detected or obstacle is detected. If obstacle, then the obstacle detection algorithm checks left and right sensors. Calculating the values from all sensors it decides the appropriate direction and suggests it to the user.

The data transfer between two layers takes place via universal serial bus. The data is then received in above layer and thus appropriate decision is displayed as well as voice output is provided.

Thus, the Graphic user interface, Gesture detection, Navigation and display of direction becomes the part of first layer and Arduino and ultrasonic sensors become part of second layer.

4.3 Software Requirements Specification Document

Introduction:

4.3.1 Product Overview:

Smart stick for blind is a stick having a Arduino sensor connected to ultrasonic sensor and this module is connected to Android phone which detects the potholes, provides GPS and emergency features. This end user in this product is blind person and the aim is to improve the accessibility.

External Interface Requirements:

4.3.2 User Interface Requirements

The user interface is quite simple as person using is blind. The user interface is android app having braille language implementation which is operated using vibrations, Voice to text for GPS and emergency alarm.

4.3.3 Hardware Interface requirements

Hardware consists of Arduino, Ultrasonic sensor and Mobile phone. Arduino is interfaced with ultrasonic sensor which fetches distance and sends to Arduino. Arduino does processing of data and sends to phone for pothole algorithm implementation.

Software Interface Requirements:

App is developed on Android which accepts the data from Arduino process it to decide threshold and determine presence of pothole. Along with pothole detection, there is also a proposal for implementation for Braille language which gives vibration feedback to person and GPS navigation which has voice to text feature. Software used is Android Lollipop (or above), Arduino burner, Android Studio and Google API's.

4.3.4 Software Product Features

1. Pothole detection: takes input as distance received from Arduino and based on threshold value it determines whether pothole is present or not.
2. Distance measurement: Ultrasonic sensor senses the obstacles, measures the distance and sends to Arduino.
3. Gesture Detection: Swiping right at the bottom opens the gesture detection where user can draw simple predetermined gestures to call a contact.
4. Navigation: Takes input as voice, converts to text and enters the destination to provide navigation.

Software System Attributes

4.3.5 Reliability

The System is reliable as phone and sensors are synchronized properly. The mean time to failure is the time till the phone battery survives.

4.3.6 Availability

The software is available as required and provides correct information to person. There is no database required.

4.3.7 Maintainability

The System is very easy to maintain, updates are easily available as android is open source and Arduino has large community.

4.3.8 Portability

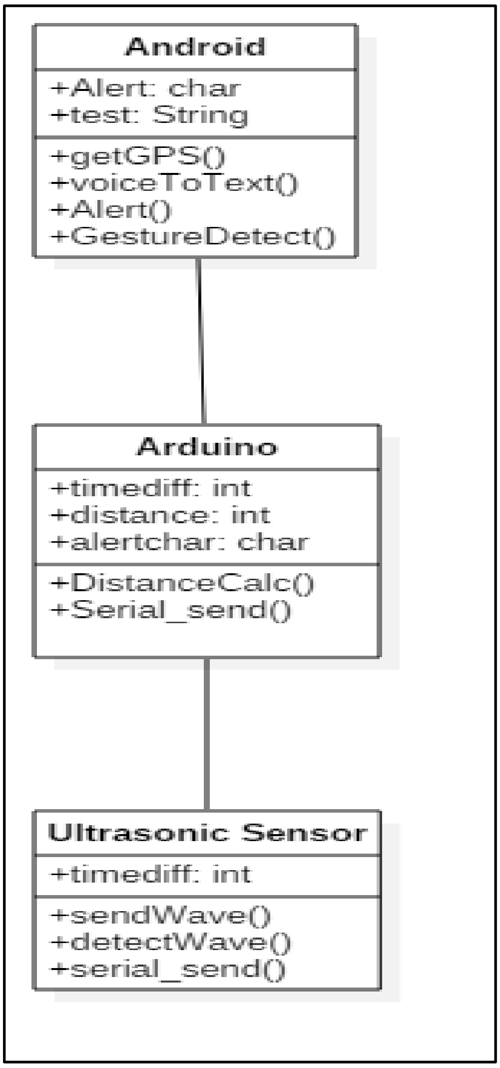
The System is quite portable as host needs it to travel, Used language is Android and is backwards compatible. So any new update will be able to run software. Software used is android studio as it the most efficient with respect to SDK's required.

4.3.9 Performance

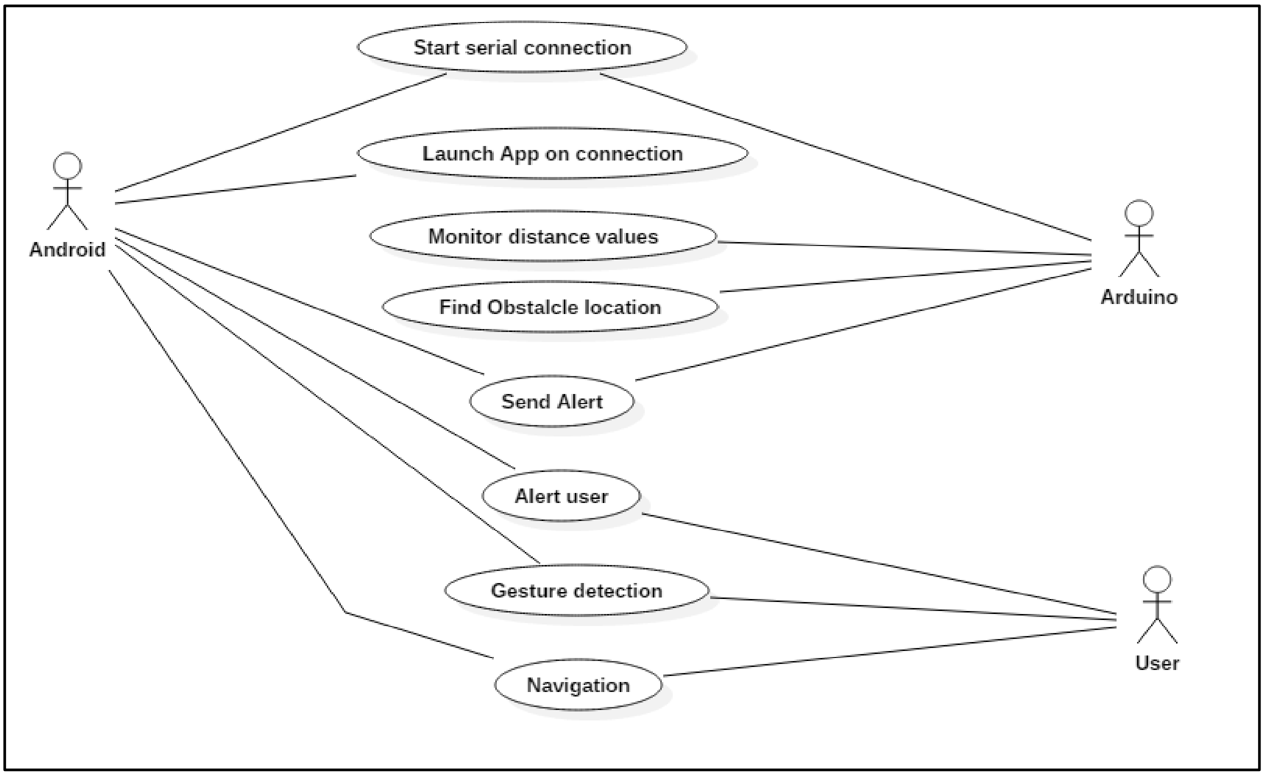
The overall performance has passed most test cases. Only one user can operate at a time. There is no pressure of peak workload condition unless RAM is too less and minimum data storage is size of app.

4.3.10 UML Diagrams:

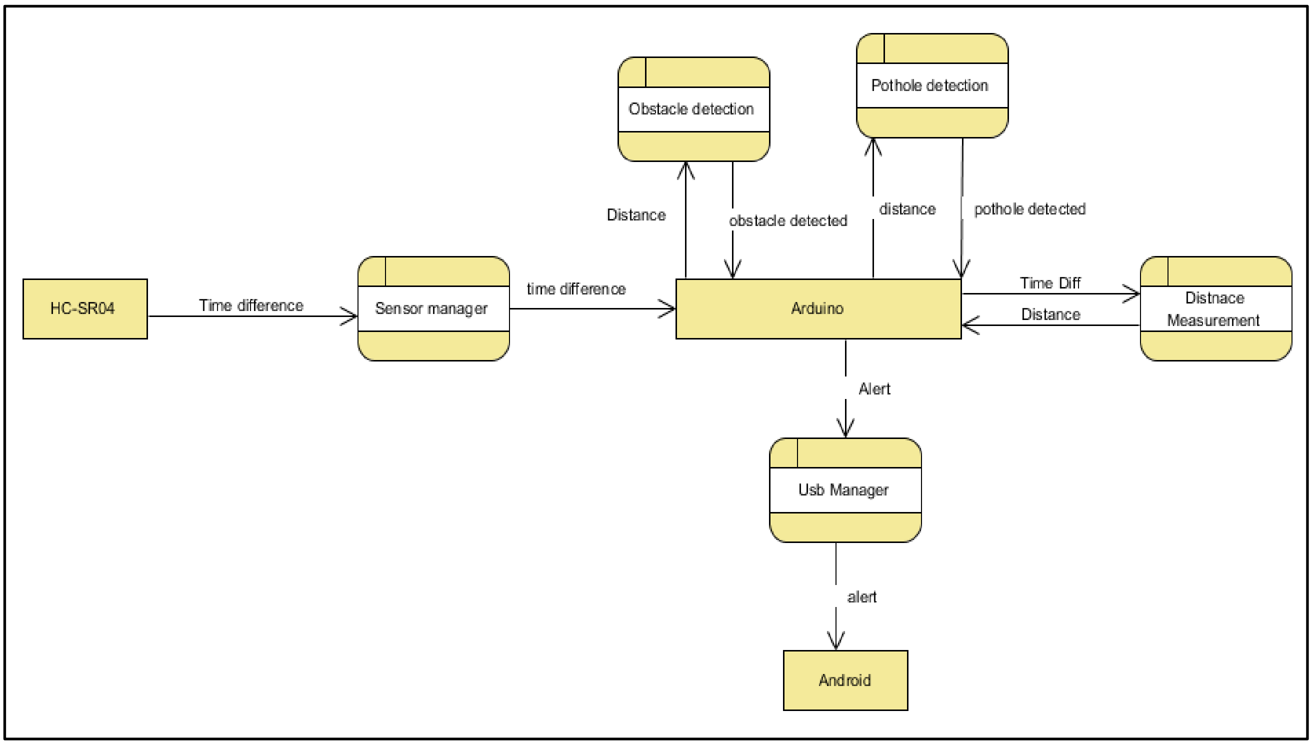
Class Diagram



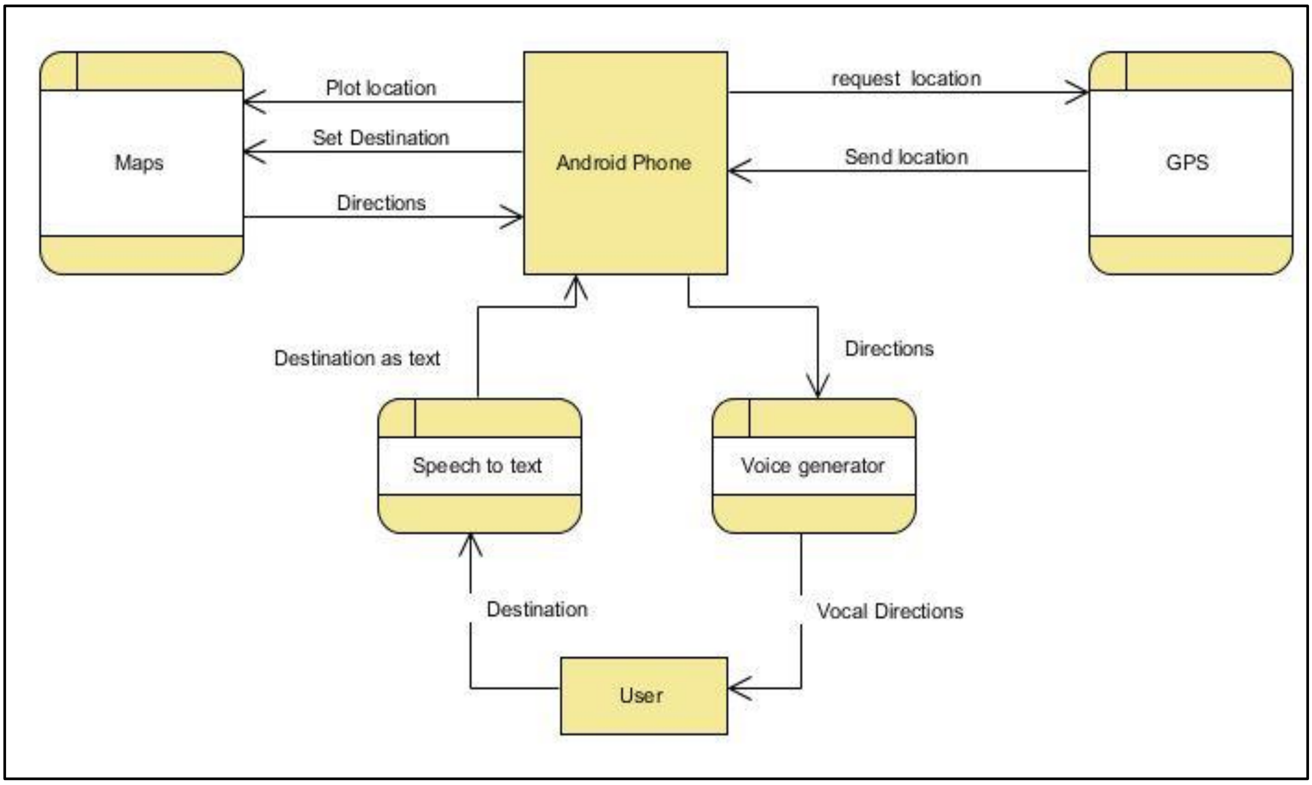
Use Case Diagram



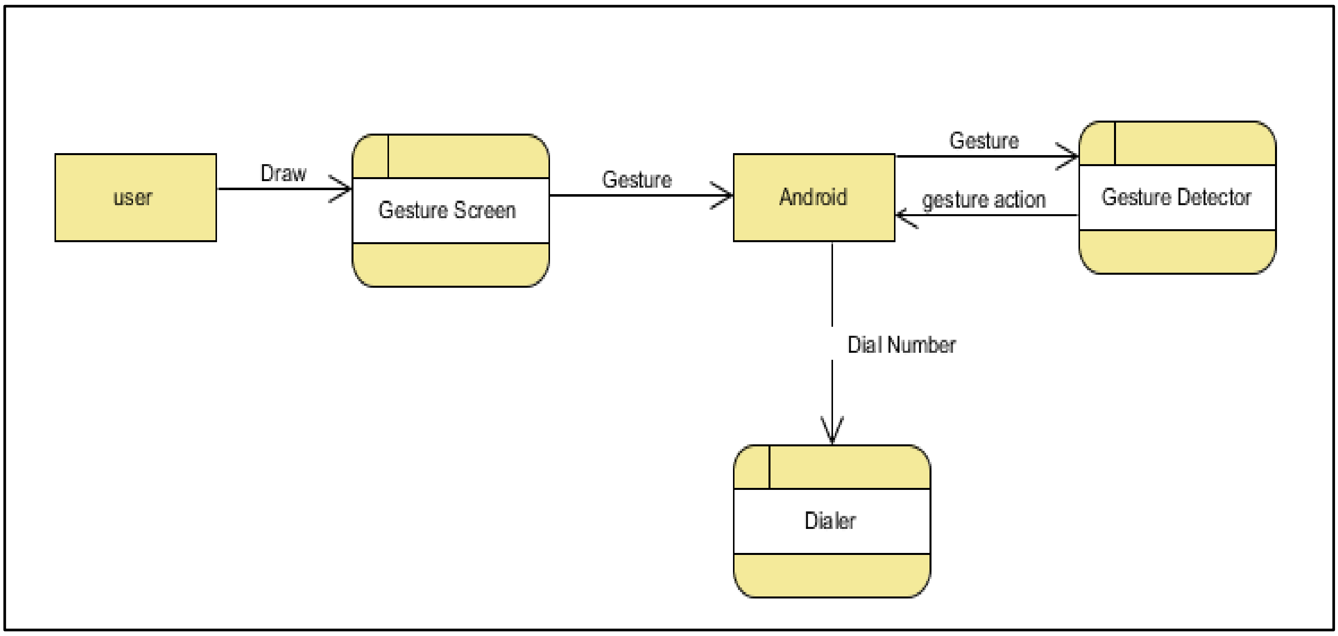
Data Flow Diagram: For distance and obstacle detection



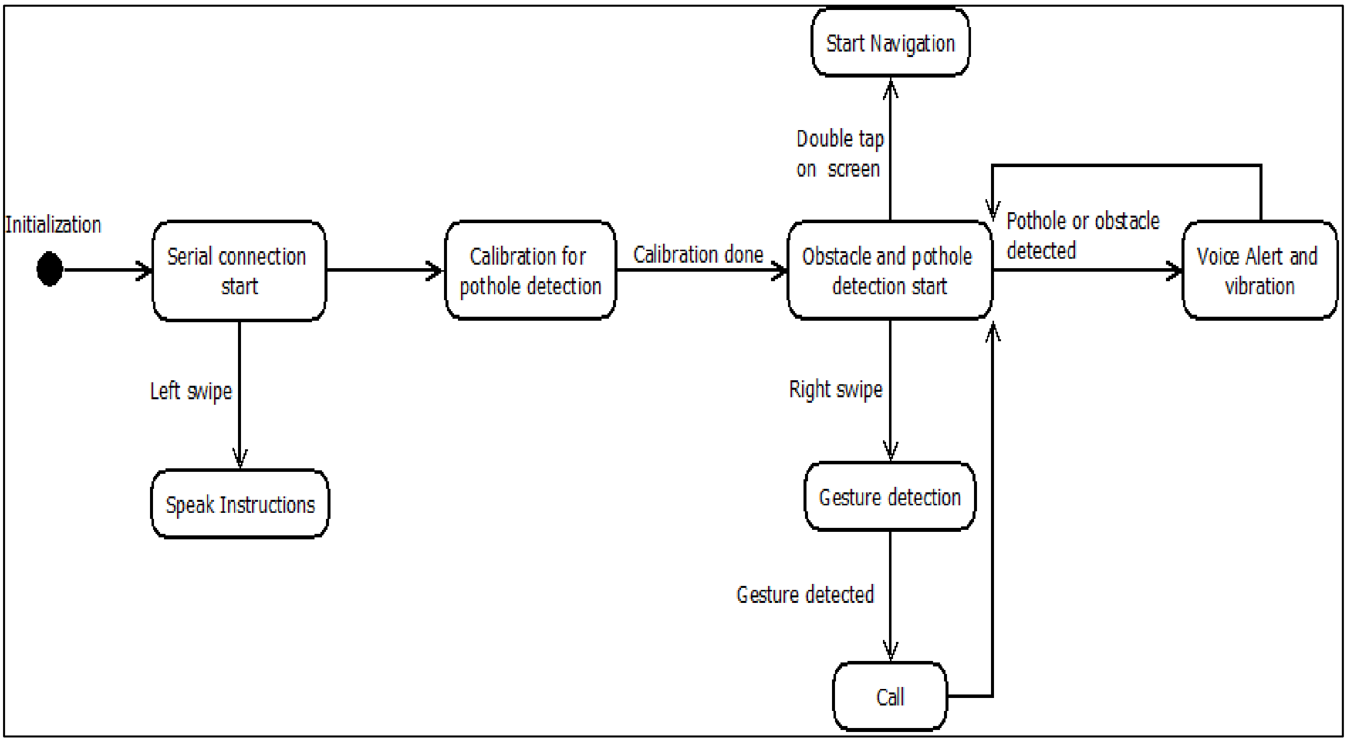
Data Flow Diagram: For navigation



Data Flow Diagram: For gesture detection



State diagram



4.4 Software Design Document

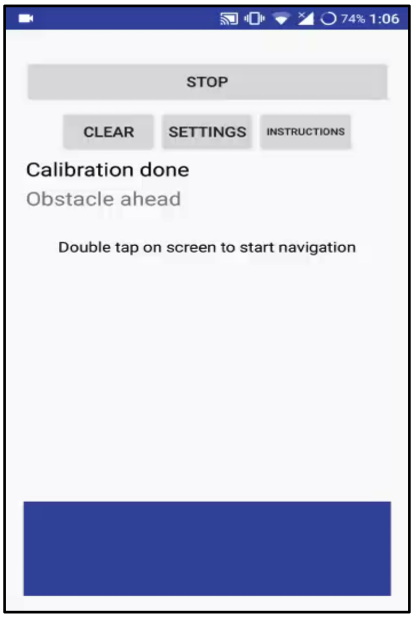
4.4.1 UI design

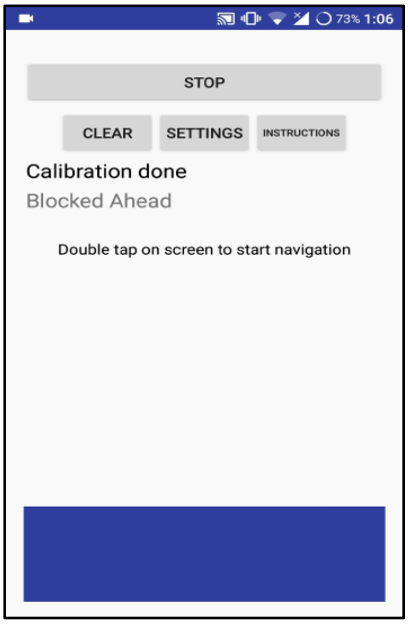
MainActivity:

Purpose: It the activity that opens on app launch. This alerts the user about various obstacles and potholes.

Inputs: Double tap to launch navigation, swipe right at bottom to launch gesture activity

Outputs: Speaks out the alerts like “pothole detected”, “obstacle ahead”.



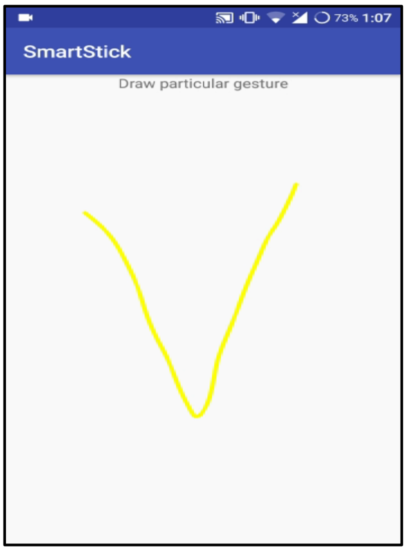


GestureActivity:

Purpose: Used to detect gestures made by the user and fire an intent to call a contact associated with the gesture.

Inputs: Draw simple gestures like ‘o’.

Outputs: Calls a particular contact.

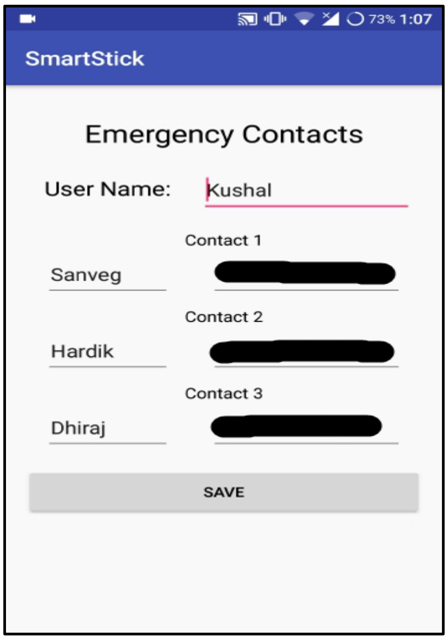


SettingsActivity:

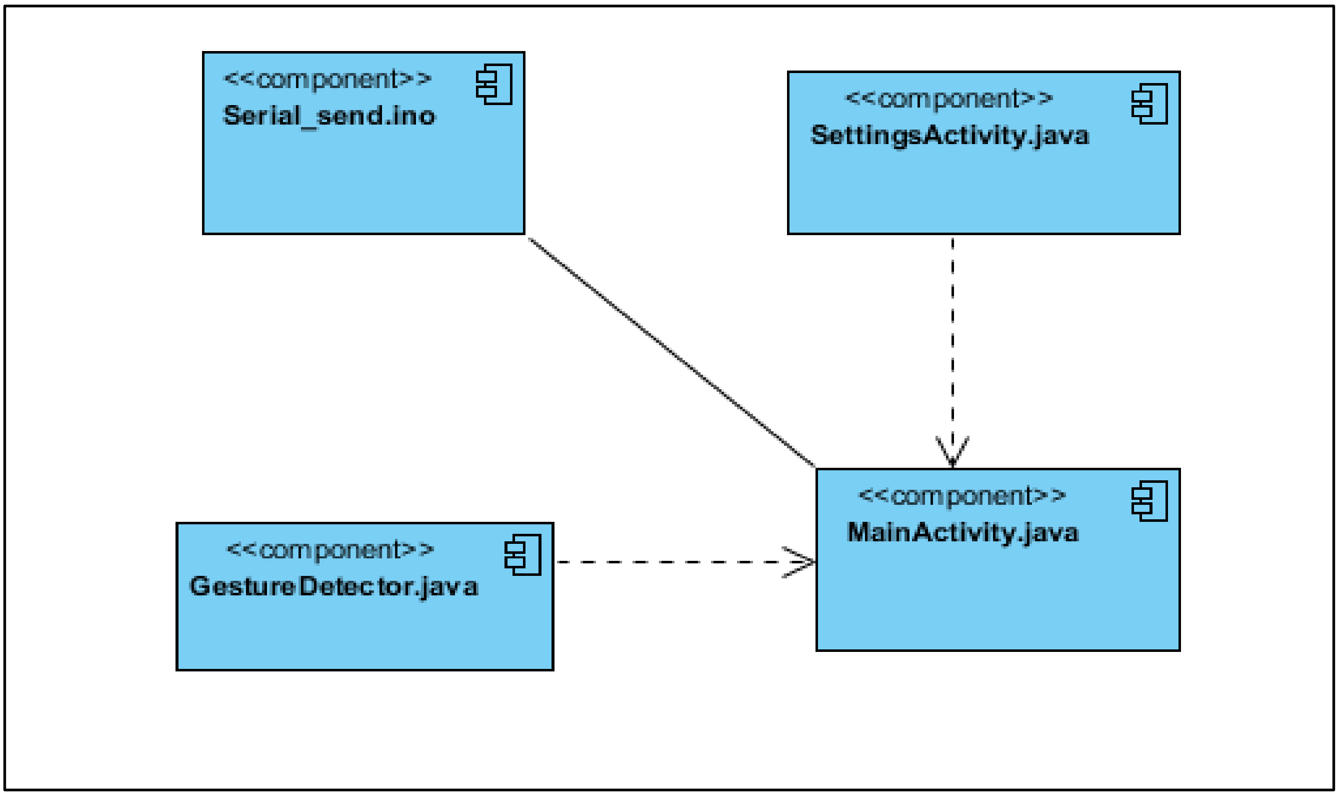
Purpose: Used to save contacts for dial using gestures.

Inputs: Enter the phone number of the contact and press save.

Outputs: Numbers will be saved.

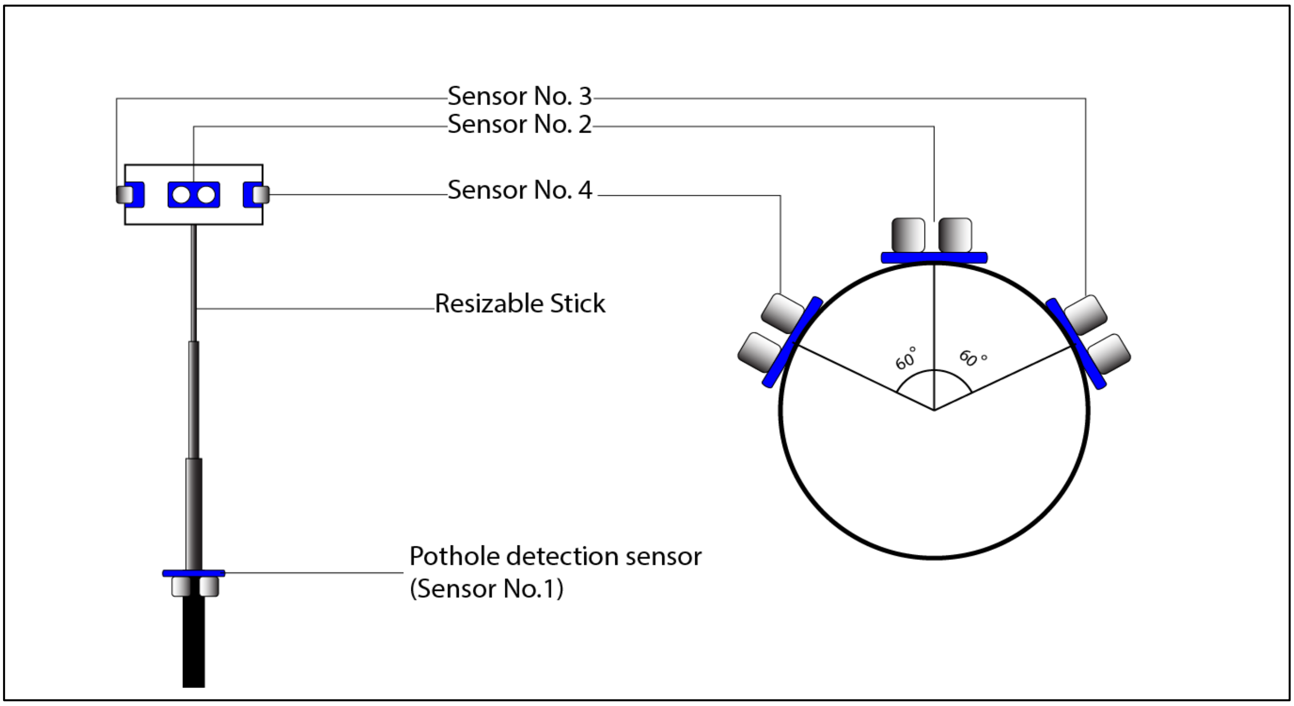


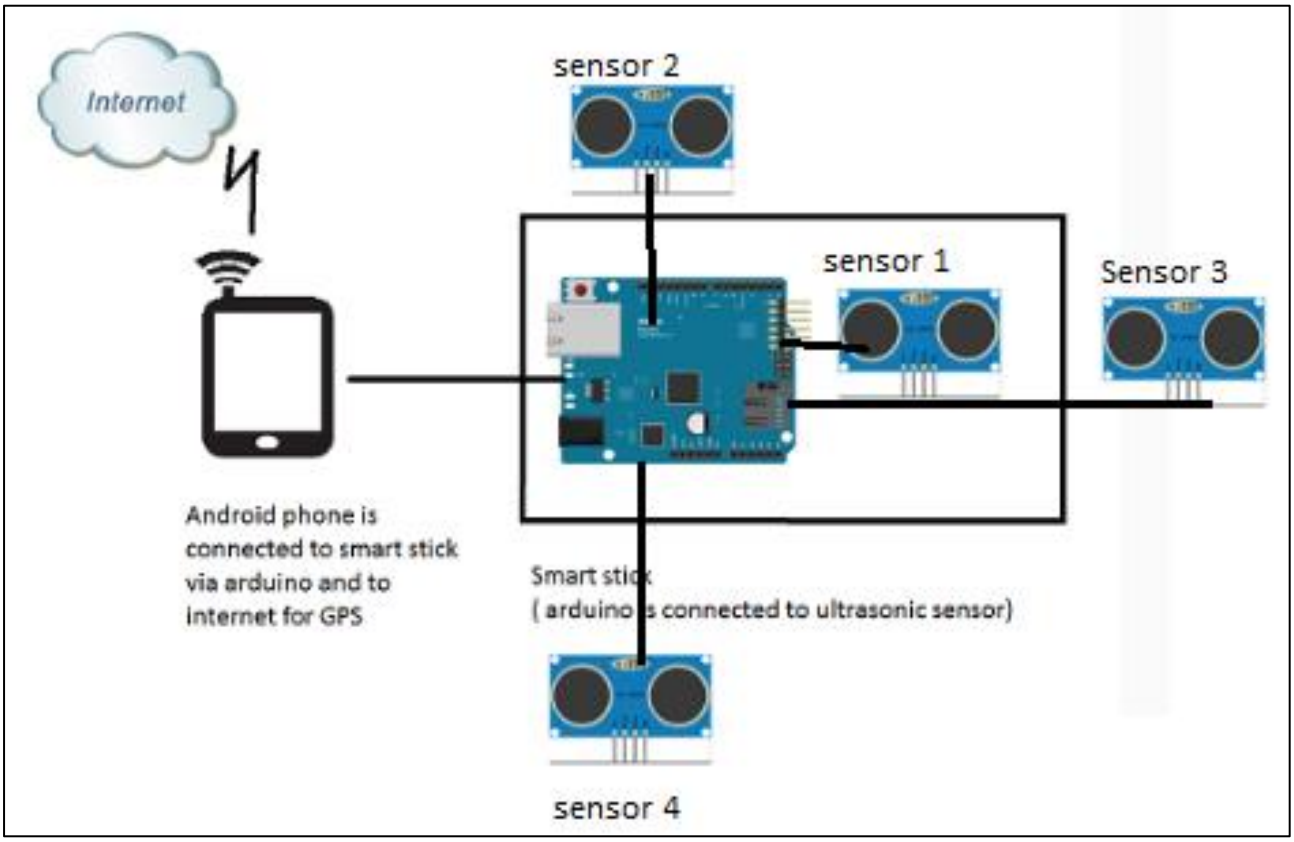
4.4.2 Component Diagram



Chapter 5: Project Implementation

5.1 Approach/System Architecture / Main Algorithm / Methodology





The diagram explains the System Architecture, consisting of android phone connected to Arduino setup attached to the stick. The Android to Arduino connection done using USB connections and android phone is connected to internet for navigation

Ultrasonic sensors connected to Arduino are used to calculate distances of objects from the sensors. The following formula implemented in Arduino code calculates the distance of any object within the range of sensors.

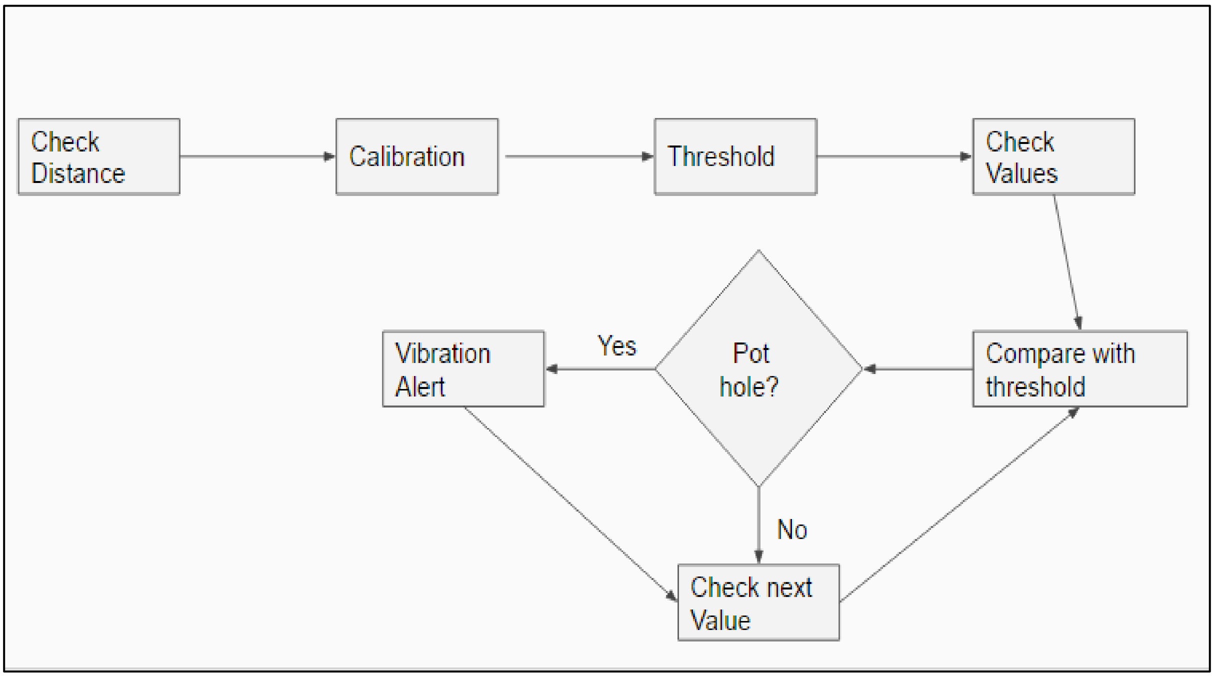
Speed of sound in air =340 m/s

Distance= (speed*time)/2

Distance is divided by 2 because initial distance received is for sending signal plus receiving signal.

<u>Pothole detection algorithm:</u>

The distance is calculated by the Arduino in each cycle. Initially values are used to calculate the threshold value and later each distance value calculated is compared with the threshold to check for pothole. Pothole detection detects a pothole on user's path by comparing each new value with the calculated threshold value.



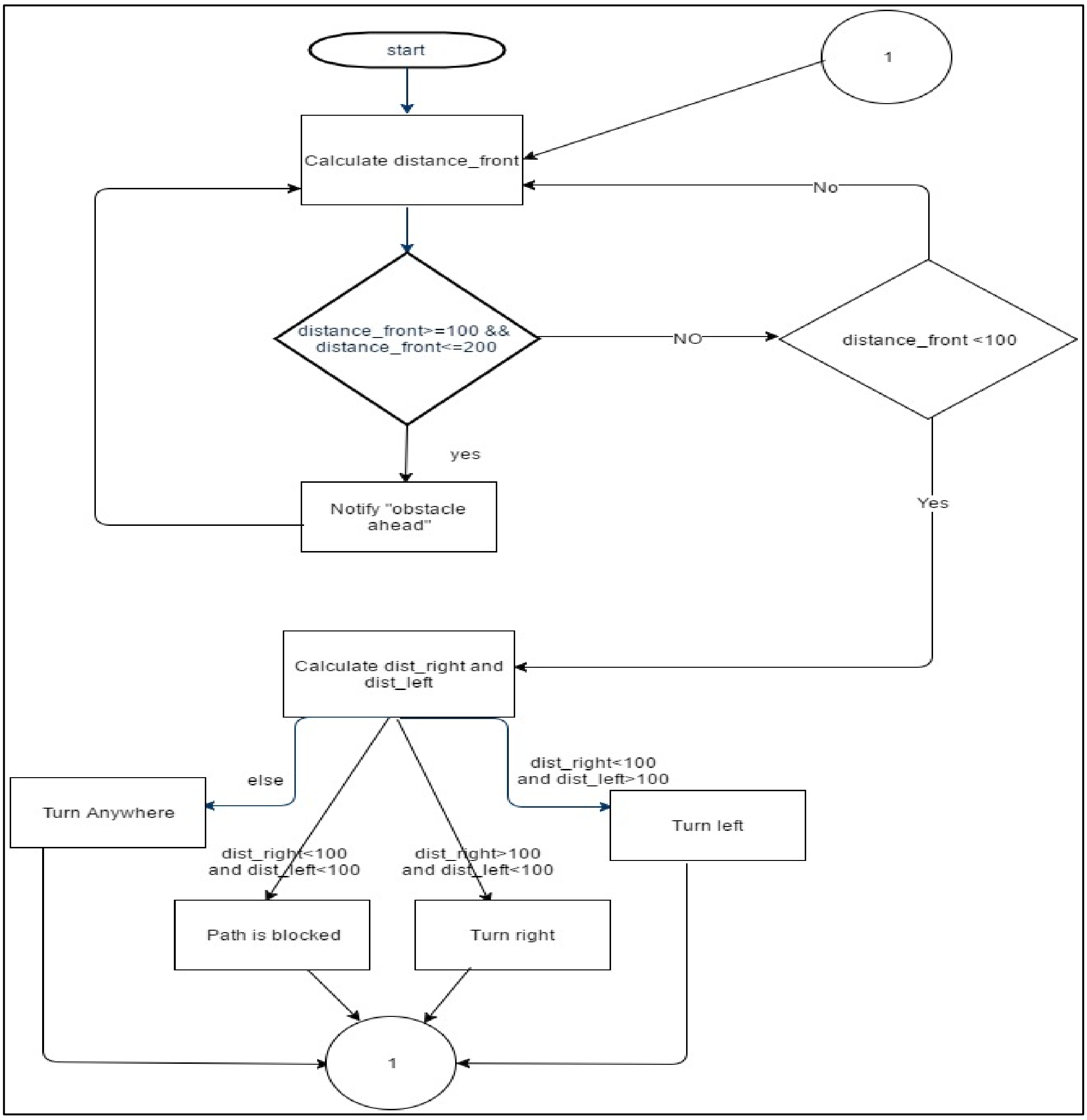
Pothole detection calculates a threshold value by measuring the stick user's patterns of using the stick (since each person's height is different and stick's holding point is also different), which is then used for detecting a pothole on the user's path. Initial ten values are used for calibration of the threshold value for pothole detection. The value within certain limit are considered for calibration as user's misjudgments might lead to ambiguous results. The valid results are then summed to calculate their average value and the largest value in the set is also recorded. The average value is an indicator of the height to which the user generally lifts the stick while commuting, whereas the highest value is the maximum deviation from the average during the calibration. The difference between average and the maximum value gives a maximum fluctuation value, which then is doubled and added to the average to calculate threshold i.e. error is also considered. This approach increases the accuracy of the threshold and provides the Arduino with a value indicating the maximum possible distance from the ground.

avg_value = (sum of 10 values) / 10.
max_value = maximum of initial 10 values.
fluct_value = max_value-avg_value.
Threshold = 2 * fluct_value+avg

The Arduino calculates distance from ground with each loop, using the readings from the ultrasonic sensor, and compares the new value with threshold value calculated previously to check for potholes. A value greater than the threshold value indicates the possibility of presence of a pothole ahead of the user and the Arduino sends a value to the android application to warn the user.

Obstacle detection algorithm:

The three sensors placed on the top of the stick are used for obstacle detection. Each sensor senses the obstacle towards the front using time required to receive the signal and then distance is calculated in the arduino using distance formula.



The algorithm used for obstacle detection is:

Step 1: dist_front = distance received from sensor 2 or front sensor.

Step 2: if (dist_front>=100 && dist_front<=200)

alert that obstacle is ahead.

go to step 5

Step 3: calculate dist_right and dist_left i.e. distance from right and left sensor respectively.

Step 4: if dist_right<100 && dist_left>100

alert Turn left

```
        else if dist_left<100 && dist_right>100
        alert Turn right
          else if dist_left<100 && dist_right<100
       alert Path is blocked
        else alert Turn anywhere
      go to step 1
Step 5: check dist_front again for closeness
step 6: if (dist_front<100)
       alert obstacle is very close in front
      go to step 3
```

This output is send to android application using specific characters assigned to each direction.

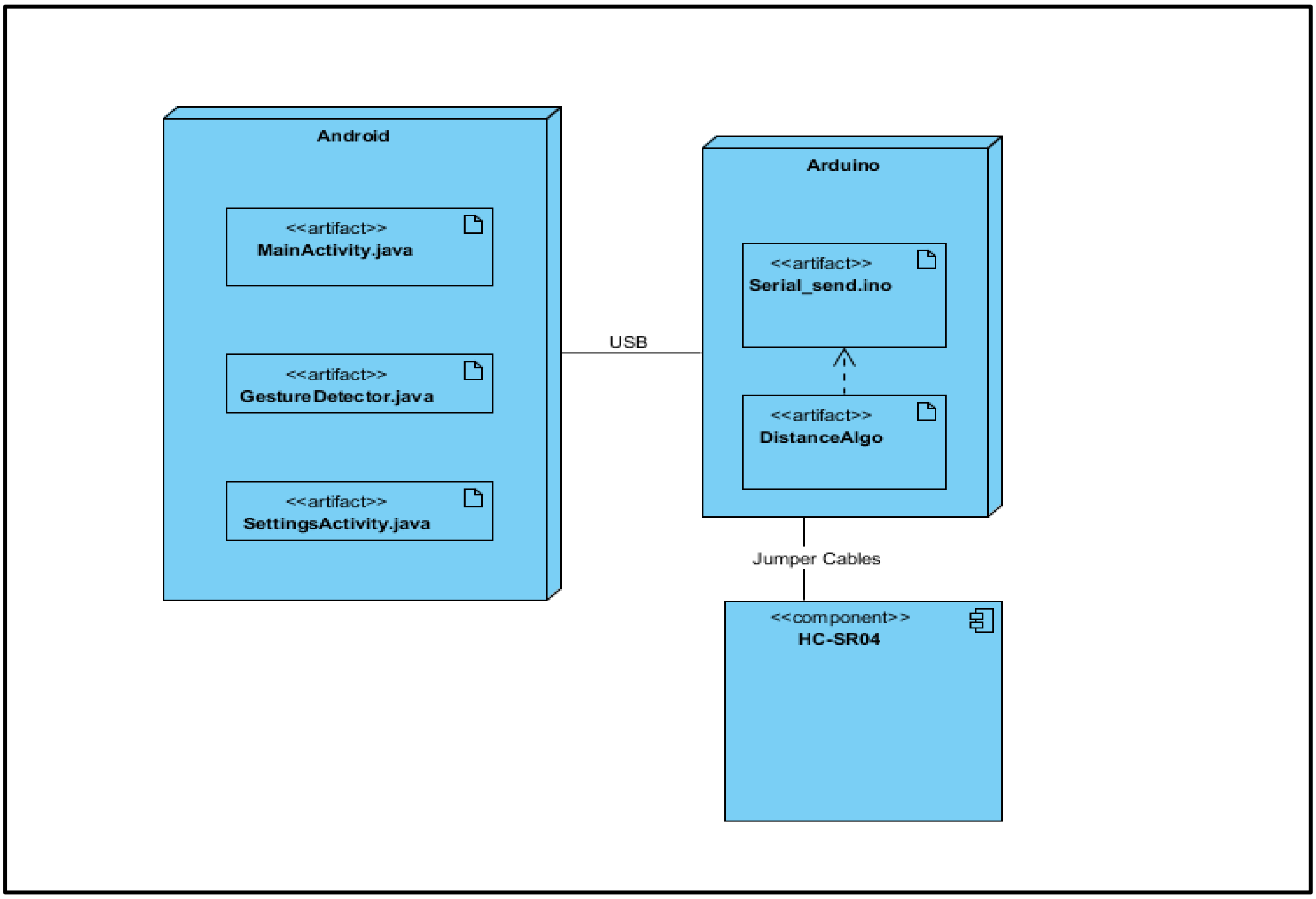
5.2 Programming Language used for Implementation

- Pothole detection and obstacle detection algorithms implemented in C language using Arduino SDK.
- Android application implemented in Java using the Android SDK.

5.3 Tools used for implementation

- Arduino IDE used for programming Arduino.
- Android application developed using Android Studio.
- Google maps API used for implementing the navigation feature in Android along with speech to text for destination.
- Arduino interfaced with android using USB.
- Object oriented methodology used for coding every module. (Java)
- Visual paradigm software used for UML diagrams.

5.4 Deployment diagram



5.5 Deliverable

Chapter 6: Integration & Testing

6.1 Testing Approach

Reactive testing technique used as the modules were tested after they were coded and compiled successfully. As in Reactive testing technique, the testing is not started until after the designing and coding of modules is completed.

Various modules implemented :

1. Pothole detection
2. Obstacle detection
3. Navigation
4. Gestures

Each module was tested in a reactive testing approach wherein the module was first designed, coded and then tested for various test cases. Every module was tested independently after its completion after which bugs were corrected and updates were made until the test results were as desired. After successful completion of testing, next module was designed and coded.

6.2 Testing Plan

Introduction: Every important module of the project was individually coded and tested following the reactive approach. Test conditions and input values for desired outputs were detailed and the actual outputs were compared with the estimated outputs. Every module was tested and updated till the actual outputs were almost close to desired values.

Test Items:

- Android phone to check appropriate warnings and alerts.
- Microphone for voice input.
- Android log to verify values calculated by algorithms.
- USB connection from Arduino to Android.

Features to be tested:

- Pothole detection and warning.
- Obstacle detection and appropriate alerts with guidance.
- Voice recognition with accurate interpretation of spoken destination.
- Gestures detection.

Approach:

1. Pothole detection testing:
    - The sensor on the base of stick to detect potholes.
    - Initial values recorded for threshold calculation.
    - Hovering the stick over various road conditions.
    - Walking the stick to various locations.
2. Obstacle detection testing:
    - Sensors on the top end of the stick detect obstacles within a range of 180 degrees.
    - Using the stick along with sensors for commuting.
    - Walking the stick in various traffic and crowd conditions.
    - Recording the response suggested by the obstacle detection algorithm.
3. Voice recognition:
    - Various voice inputs to test the conversion to text.
    - Numerous locations tested to test the interpretation capability.
4. Gestures detection:
    - Launching the gesture screen using accurate method.
    - Using applied gestures along with random patterns.
    - Checking for proper gesture recognition.

Test Environment: Environment used for testing purpose was simulated close to real world conditions in which visually challenged people might use the setup. Different routes and road conditions were selected for pothole detection. Various crowded locations and busy streets for obstacle detection simulation.

Various locations tested for simultaneous pothole and obstacle detection. Noisy as well as silent locations used to test working of voice recognition module for navigation.

Risks: Various risks identified during testing period:

- Proper connections and handling of equipment necessary for accurate results.
- Net connectivity required for navigation purposes.
- Cautious power supply mechanism.
- Mobile coverage availability for emergency calling feature.

6.3 Unit Test Cases

- The calculation of threshold from the starting ten values.

Initial Values (cm)	Threshold Values (cm)
13,15,17,19,20,14,16,15,14,12	24.5
12,19,21,24,13,19,15,18,15,16	30.8

- The result of voice navigation. First column shows the destination given by user. Second column shows whether it was converted properly from voice to text. Third column shows whether navigation was started or not.

Destination	Voice to Text destination	Navigation started?
Vidyavihar station	“Vidya vihar station”	Yes
Thane station	“Thane station”	Yes

- The output of gesture recognition. Column one shows what gesture is drawn by user. Column two shows what gesture is detected and column three shows whether function specific to that gesture is called or not

Gesture drawn	Gesture Detected	Function called
“0”	“0”	call emergency contact 1
“V”	“V”	call emergency contact 2
“^”	“^”	call emergency contact 3

6.4 Integrated System Test Cases

The result of obstacle detection. First Column shows in which direction the obstacle is present. Second column shows the actual output of the direction of obstacle as detected by sensors. The third column shows the range at which obstacle is detected and fourth column shows how accurate the result is.

Obstacle from	**Output**	**Range (cm)**	**Result**
front	"obstacle ahead"	145	accurate
front	"obstacle ahead turn anywhere"	78	accurate
front and left	"turn right"	60	accurate
front and right	"turn left"	82	accurate
front, left and right	"turn right"	59	Error
front, left and right	"path blocked, turning suggested"	72	accurate

Pothole	**Output**	**Distance from ground**	**Result**
pothole present	"pothole detected"	30	accurate
pothole not present	"pothole detected"	28	error
pothole not present	no output	19	accurate

Chapter 7: Conclusion & Future work

7.1 Conclusion

In this project, solution is provided to help blind people so that they can walk with confidence by detecting obstacles and potholes in their path. Solution consisted of arrangement of sensors. The horizontal sensors were able to detect obstacle whereas bottom most inclined sensor was able to detect pothole and output was provided in form of voice. Navigation and gesture recognition are also accurate enough, thereby increasing the accessibility for the blind and increasing their confidence to walk in non-familiar environment.

7.2 Future Work

Although this project covers almost every aspect of various possibilities but still there are scopes of improvement. Better expensive sensors can be used instead of ultrasonic sensors. Ultrasonic sensors were used because of inexpensiveness and reliability. Cameras can be used to detect obstacle and pothole along with edge detection algorithms. APIs of other applications can be used for better collaboration. Some of the above features however might be implemented before but it either makes system too expensive or it is inefficient. Hence, cost and efficiency along with above features can be implemented in future.

Chapter 8: References

[1] M. Varghese, S. S. Manohar, K. Rodrigues, V. Kodkani and S. Pendse, "The smart guide cane: An enhanced walking cane for assisting the visually challenged," 2015 International Conference on Technologies for Sustainable Development (ICTSD), Mumbai, 2015, pp. 1-5. doi: 10.1109/ICTSD.2015.7095907

[2] D. S. Raghuvanshi, I. Dutta and R. J. Vaidya, "Design and analysis of a novel sonar-based obstacle-avoidance system for the visually impaired and unmanned systems," *2014 International Conference on Embedded Systems (ICES)*, Coimbatore, 2014, pp. 238-243. doi: 10.1109/EmbeddedSys.2014.6953165

[3] S. Gupta, I. Sharma, A. Tiwari and G. Chitranshi, "Advanced guide cane for the visually impaired people," *2015 1st International Conference on Next Generation Computing Technologies (NGCT)*, Dehradun, 2015, pp. 452-455. doi: 10.1109/NGCT.2015.7375159

[4] A.K.Shrivastava, A.Varma and S.P.Singh "Distance measurement of an object or an obstacle by ultrasound sensors using P89C51RD2" International Journal of Computer Theory and Engineering, Vol.2, No.1, February 2010.

[5] Elec Freaks Ultrasonic Ranging Module HC - SR04 [Online] Available: http://www.micropik.com/PDF/HCSR04.pdf

[6] Jack Loomis, Reginald Golledge and Roberta Klatzky, "Navigation System for Blind – Auditory Display Modes and Guidance", Presence: Vol. 7, No. 2, April 1998, pp. 193–195.

[7] Loomis JM, Klatzky RL, Golledge RG, Cincinelli JG, Pellegrino JW, Fry PA, "Nonvisual navigation by blind and sighted: Assessment of path integration ability", J Exper Psychol General 122(1):73–91 (1993).

[8] H. He, and J. Liu, "The design of ultrasonic distance measurement system based on S3C2410," Proceedings of the IEEE International conference on Intelligent Computation Technology and Automation, Oct. 2008, pp. 44-47.

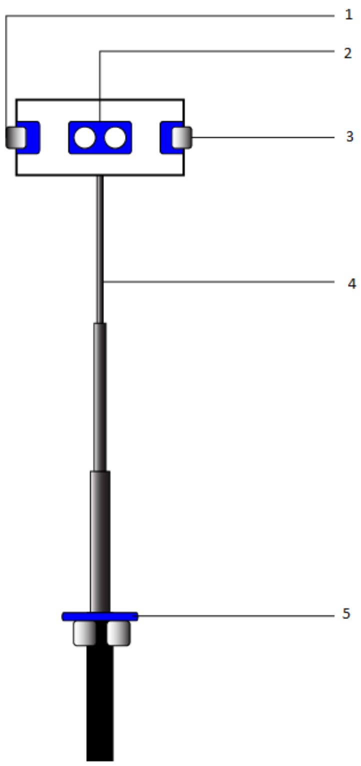
[9] Y. Jang, S. Shin, J. W. Lee, and S. Kim, "A preliminary study for portable walking distance measurement system using ultrasonic sensors," Proceedings of the 29th Annual IEEE International Conference of the EMBS, France, Aug. 2007.

Chapter 9: Appendix

9.1 Minimum System Requirement

- Java Virtual machine
  Java Runtime Environment 7
  Java Development Kit 7
- Android SDK (API Level 21 or up)
- Dual Core processor
- Memory
  2 GB RAM
  3 GB memory space for installing Android Studio

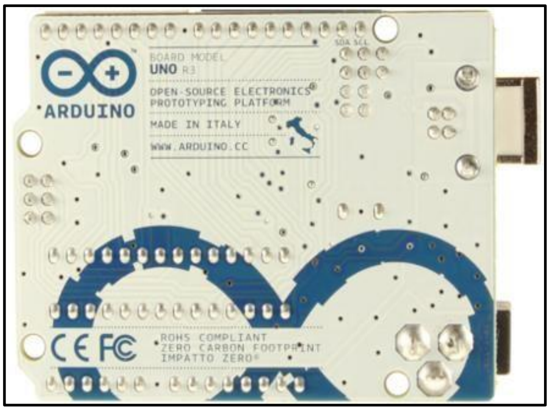
9.2User's Manual



1: Ultrasonic sensor (Right)
2: Ultrasonic sensor (Front)
3: Ultrasonic sensor (Left)
4: Resizable Stick
5: Ultrasonic sensor (Pothole detection)

9.3 Data Sheets of chips used

Arduino Uno

Overview

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, 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.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the index of Arduino boards.

Summary

1	Microcontroller	ATmega328
2	Operating Voltage	5V
3	Input Voltage (recommended)	7-12V
4	Input Voltage (limits)	6-20V
5	Digital I/O Pins	14 (of which 6 provide PWM output)
6	Analog Input Pins	6
7	DC Current per I/O Pin	40 mA
8	DC Current for 3.3V Pin	50 mA
9	Flash Memory	32 KB (ATmega328)
10	SRAM	2 KB (ATmega328)
11	EEPROM	1 KB (ATmega328)
12	Clock Speed	16 MHz

Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts. The power pins are as follows:

- VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- 5V: This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.

- 3V3. A 3.3volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- GND. Ground pins.

Memory

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.
- PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analogWrite() function.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference() function. Additionally, some pins have specialized functionality:

- TWI: A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library.

There are a couple of other pins on the board:

- AREF. Reference voltage for the analog inputs. Used with analogReference().
- Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the mapping between Arduino pins and ATmega328 ports. The mapping for the Atmega8, 168, and 328 is identical.

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A SoftwareSerial library allows for serial communication on any of the Uno's digital pins. The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation for details. For SPI communication, use the SPI library.

Programming

The Arduino Uno can be programmed with the Arduino software (download). Select "Arduino Uno from the Tools > Board menu (according to the microcontroller on your board). For details, see the reference and tutorials.

The ATmega328 on the Arduino Uno comes pre burned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see these instructions for details.

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available. The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

USB Overcurrent Protection

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Four screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

9.4 Papers Published

ISSN 2321 3361 © 2017 IJESC

IJESC

Research Article Volume 7 Issue No.4

Smart Stick for Blind using Arduino, Ultrasonic Sensor and Android

Hardik R. Shah[1], Dhiraj B. Uchil[2], Sanveg S. Rane[3], Prasanna Shete[4]
B.E. Student[1, 2, 3], Assistant Professor[4]
Department of Computer Engineering
K.J. Somaiya College of Engineering, Mumbai, India

Abstract:
The term blindness is used for complete or nearly complete vision loss. Blind people need some help while travelling to feel safe. In this paper, we have developed a smart stick which increases the accessibility of blind person to move around and voice output is given when obstacle is or pothole is detected. This smart stick is cheap, lightweight and fast. The stick consists of 3 ultrasonic sensors for obstacle detection and 1 ultrasonic sensor for pothole detection. The android application is linked with ultrasonic sensors for notification of obstacle and pothole. The application also consists of real time navigation system with speech recognized destination, automatic launch and Gesture detection with speed dial, emergency alarm etc.

Keywords: Blind people, GPS (Global Positioning System) navigation, Ultrasonic Sensors, Arduino, Gestures.

I. INTRODUCTION

As of 2012, there were nearly 285 million people who were visually impaired of which 246 million had low vision and nearly 39 million were blind. Blind people often have problems while moving around in an environment where they are not familiar. They use the traditional white cane. This cane detects obstacle only when they touch it and hence prior detection of obstacle is problem.

The cane also cannot properly detect obstacle which are at certain height. Thus, blind people will feel confident to move around only when obstacles are known from far distance. This can be done with help of Ultrasonic sensors. There are various types of technologies and sensors available but ultrasonic sensors are chosen because they are cheap and light weight and can detect obstacle up to 400cm. There is also a possibility that blind person might not know the route or might have some emergency. These problems can be taken care with real time navigation and gesture recognition. Smart stick will help the blind person in easy mobility just like normal person.

A. Abbreviations and Acronyms

ETA–Electronic Travel Aid, GPS- Global Positioning System.

II. PROBLEMS WITH EXISTING SYSTEM

An Electronic Travel Aid is form of Assistive technology for enhancing mobility of blind person. The research problem of designing a better ETA is tough one. Blind people find travelling difficult because they cannot determine where obstacles are. This process is also called as 'Spatial Sensing'. There are various problems with existing system. First, the rangefinder technology is unreliable in detecting steps. Second, Blind people find various sound pitches and vibrations difficult to understand. Third, these systems are quite expensive and since blind person have to depend financially on someone, they don't feel worth to invest so much. Problems and various existing systems:

1. Walkmate – Developed in 1993. It had detection only upto 1.83m.

2. Miniguide US- Developed in 2004. The price was $545 which is quite expensive.

3. Laser cane- The latest one is named as 'N2000'. It provides 3 beams straight, overhead and for downward drops. It is still available and is priced at $2650. The developed smart stick has range of 400 cm and it costs less than 35$.

III. PROPOSED SYSTEM

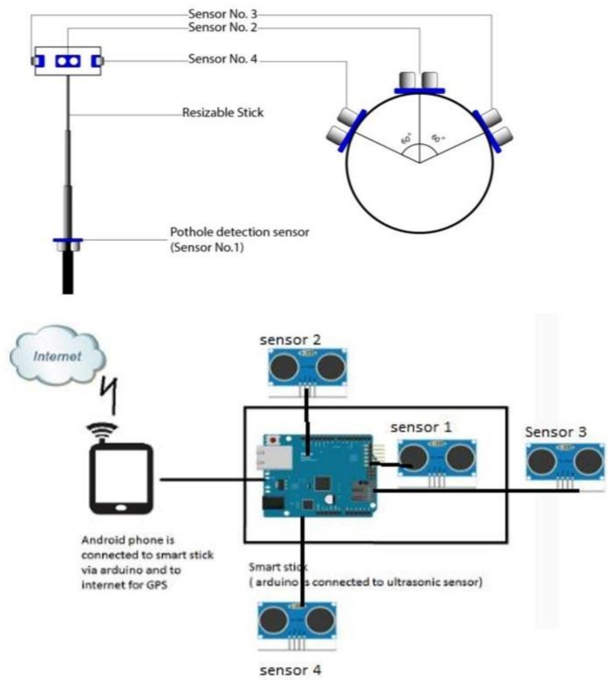


Figure.1.Proposed System

The above two Fig. shows the architectural diagram of the system. The system consists of stick on which 3 ultrasonic sensors are placed (sensor 2, 3 and 4). These sensors are used for obstacle detection. Range of each sensor is 400 cm front and 60 degrees wide. Hence these, sensors are placed in such manner that covers most of the obstacles.

At the bottom of stick it consists of another ultrasonic sensor (sensor 1) which is used for pothole detection. The entire stick is variable in length according to height of the user. All the ultrasonic sensors are connected to the arduino.

The arduino is connected to the mobile which consists of android application. The application is used for obstacle and pothole notification through voice, real time GPS navigation system and gesture detection.

The entire project is divided into several parts:

a) Obstacle detection

b) Pothole detection

c) Android application

IV. METHODOLOGY

A. Obstacle Detection

The three sensors placed on the top of the stick are used for obstacle detection. Each sensor senses the obstacle towards the front using time required to receive the signal and then distance is calculated in the arduino using distance formula.
The algorithm used for obstacle detection is:

Step 1: dist_front = distance received from sensor 2 or front sensor.

Step 2: if (dist_front>=100 && dist_front<=200)
alert that obstacle is ahead.
Go to step 5

Step 3: calculate dist_right and dist_left i.e. distance from right and left sensor respectively.

Step 4: if dist_right<100 && dist_left>100

alert Turn left

else if dist_left<100 && dist_right>100

alert Turn right

else if dist_left<100 && dist_right<100

alert Path is blocked

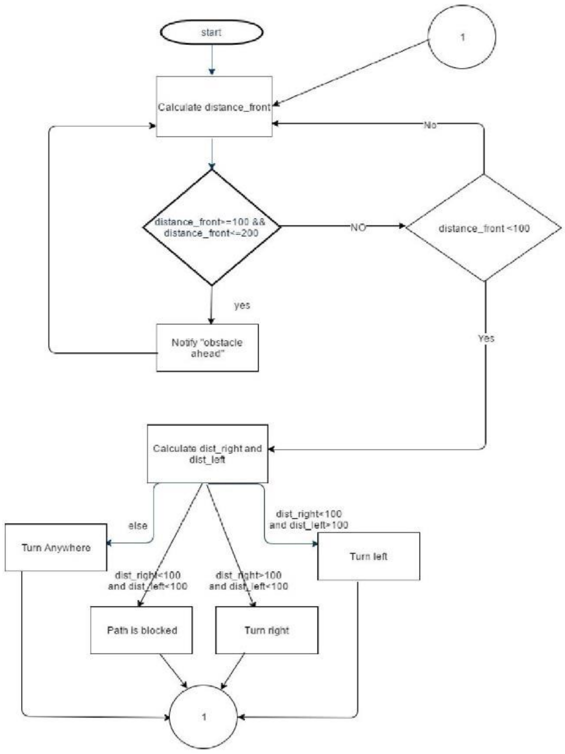
else alert Turn anywhere

go to step 1

Step 5: check dist_front again for closeness

step 6: if (dist_front<100)
alert obstacle is very close in front
go to step 3

This output is send to android application using specific characters assigned to each direction. Using text to speech in android, the output is given in form of voice.



B. Pothole Detection

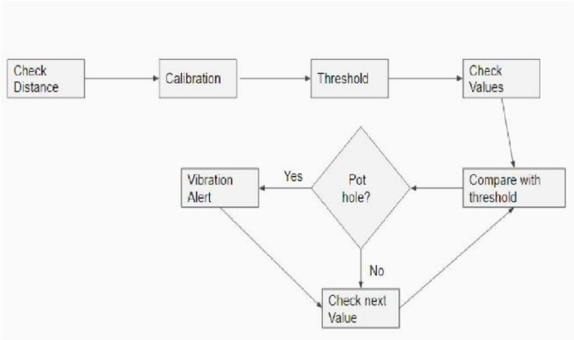
The pothole sensor attached at the bottom of the stick, facing towards ground, sends reading of the time required for the ultrasonic waves to reflect back from the ground; which is then converted to distance using the distance formula.

Speed of sound in air =340 m/s
Distance= (speed*time)/2
Distance is divided by 2 is done because initial distance received is for sending signal plus receiving signal.

Initially values are used to calculate the threshold value and later each distance value calculated is compared with the threshold to check for pothole. Pothole detection detects a pothole on user's path by comparing each new value with the calculated threshold value. Pothole detection calculates a threshold value by measuring the stick user's patterns of using the stick (since each person's height is different and stick's holding point is also different), which is then used for detecting a pothole on the user's path. Initial ten values are used for calibration of the threshold value for pothole detection. The value within certain limit is considered for calibration as user's misjudgments might lead to ambiguous results. The valid results are then summed to calculate their average value and the largest value in the set is also recorded. The average value is an indicator of the height to which the user generally lifts the stick while commuting, whereas the highest value is the maximum deviation from the average during the calibration. The difference between average and the maximum value gives a maximum fluctuation value, which then is doubled and added to the average to calculate threshold i.e. error is also considered. This approach increases the accuracy of the threshold and provides the arduino with a value indicating the maximum possible distance from the ground.

avg_value= (sum of 10 values)/10.
max_value= maximum of initial 10 values.
fluct_value= max_value-avg_value.
Threshold= 2*fluct_value+avg

The arduino calculates distance from ground with each loop, using the readings from the ultrasonic sensor, and compares the new value with threshold value calculated previously to check for potholes. A value greater than the threshold value indicates the possibility of presence of a pothole ahead of the user and the arduino sends a value to the android application to warn the user.



C. Android Application

The android application is used to alert the user about the various signals received from the sensors and also provides other functionalities. When a pothole or an obstacle is detected the app uses text to speech to speak out the alert to the user. It tells the user the position of the obstacle (front, left or right). The user can also start navigation from his current location by double tapping the screen.

Double tapping launches the speech to text service, the user has to speak out the name of his destination and the navigation will be started. Swiping left or right on bottom of screen launches gesture recognition. Since the person is blind, just sliding finger in particular pattern will recognize number using gesture recognition. Each number is assigned to call specific number.

V. RESULTS

Table 1
Table 1 shows the calculation of threshold from the starting ten values.

Initial Values (cm)	Threshold Values (cm)
13,15,17,19,20,14,16,15,14,12	24.5
12,19,21,24,13,19,15,18,15,16	30.8

Table 2
Table 2 shows the result of obstacle detection. First Column shows in which direction the obstacle is present. Second column shows the actual output of the direction of obstacle as detected by sensors. The third column shows the range at which obstacle is detected and fourth column shows how accurate the result is.

Obstacle from	Output	Range (cm)	Result
front	"obstacle ahead"	145	accurate
front	"obstacle ahead turn anywhere"	78	accurate
front and left	"turn right"	60	accurate
front and right	"turn left"	82	accurate
front,left and right	"turn right"	59	Error
front,left and right	"path blocked,turning suggested"	72	accurate

Table 3
Table 3 shows the output of pothole detection. First Column shows that pothole is present or not. Second column shows the actual output of the pothole as detected by sensor. The third column shows the distance from the ground at which pothole is detected and fourth column shows how accurate the result is.

Pothole	Output	Distance from ground	Result
pothole present	"pothole detected"	30	accurate
pothole not present	"pothole detected"	28	error
pothole not present	no output	19	accurate

Table 4
Table 4 shows result of voice navigation. First column shows the destination given by user. Second column shows whether it was converted properly from voice to text. Third column shows whether navigation was started or not.

Destination	Voice to Text destination	Navigation started?
Vidyavihar station	"vidya vihar station"	Yes
Thane station	"Thane station"	Yes

Table 5
Table 5 shows the output of gesture recognition. Column one shows what gesture is drawn by user. Column two shows what gesture is detected and column three shows whether function specific to that gesture is called or not

Gesture drawn	Gesture Detected	Function called
"0"	"0"	call emergency contact 1
"V"	"V"	call emergency contact 2
"^"	"^"	call emergency contact 3

.

Figure.3. shows the actual implementation of stick

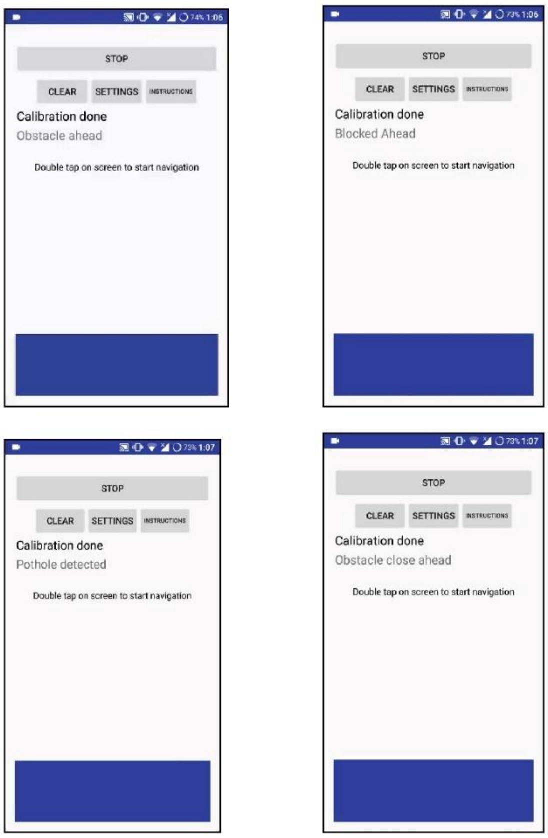


Figure.4. shows the screenshots of application for obstacle and pothole detection.

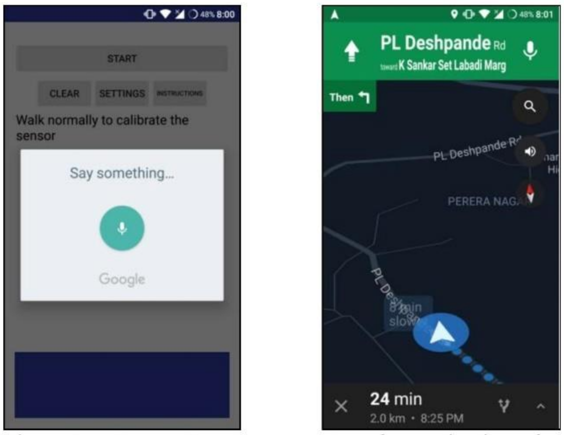


Figure.5. shows the screenshots of application for navigation.

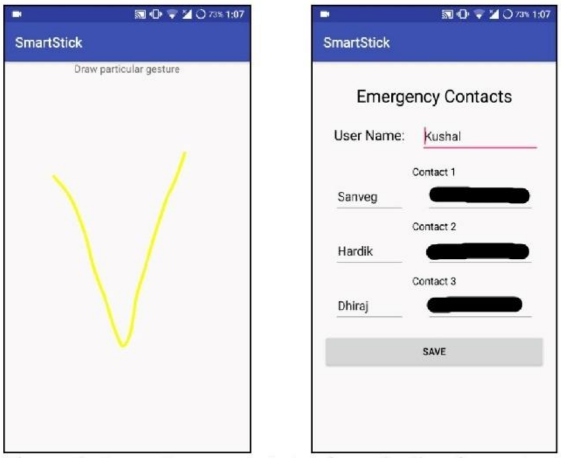


Figure.6. shows the screenshots of application for gesture detection.

VI. CONCLUSION

A. In this paper, solution was proposed to help blind people so that they can walk with confidence by detecting obstacles and potholes in their path. Solution consisted of arrangement of sensors. The horizontal sensors were able to detect obstacle whereas bottom most inclined sensor was able to detect pothole and output was provided in form of voice. Navigation and gesture recognition are also accurate enough, thereby increasing the accessibility for the blind and increasing their confidence to walk in non familiar environment

VII. REFERENCES

[1]. M. Varghese, S. S. Manohar, K. Rodrigues, V. Kodkani and S. Pendse, "The smart guide cane: An enhanced walking cane for assisting the visually challenged," 2015 International Conference on Technologies for Sustainable Development (ICTSD), Mumbai, 2015, pp. 1-5. doi: 10.1109/ ICTSD. 2015.7095907

[2]. D. S. Raghuvanshi, I. Dutta and R. J. Vaidya, "Design and analysis of a novel sonar-based obstacle-avoidance system for the visually impaired and unmanned systems," *2014 International Conference on Embedded Systems (ICES),*

Coimbatore, 2014, pp. 238-243. doi: 10.1109/ Embedded Sys.2014.6953165

[3]. S. Gupta, I. Sharma, A. Tiwari and G. Chitranshi, "Advanced guide cane for the visually impaired people," *2015 1st International Conference on Next Generation Computing Technologies (NGCT)*, Dehradun, 2015, pp. 452-455. doi: 10.1109/NGCT.2015.7375159

[4]. A.K.Shrivastava, A.Varma and S.P.Singh "Distance measurement of an object or an obstacle by ultrasound sensors using P89C51RD2" International Journal of Computer Theory and Engineering, Vol.2, No.1, February 2010.

[5] Elec Freaks Ultrasonic Ranging Module HC - SR04 [Online] Available: .http://www.micropik. com/PDF/ HCSR 04. pdf

[6].Jack Loomis, Reginald Golledge and Roberta Klatzky, "Navigation System for Blind – Auditory Display Modes and Guidance", Presence: Vol. 7, No. 2, April 1998, pp. 193–195.

[7].Loomis JM, Klatzky RL, Golledge RG, Cincinelli JG, Pellegrino JW, Fry PA, "Nonvisual navigation by blind and sighted: Assessment of path integration ability", J Exper Psychol General 122(1):73–91 (1993).

[8]. H. He, and J. Liu, "The design of ultrasonic distance measurement system based on S3C2410," Proceedings of the IEEE International conference on Intelligent Computation Technology and Automation, Oct. 2008, pp. 44-47.

[9]. Y. Jang, S. Shin, J. W. Lee, and S. Kim, "A preliminary study for portable walking distance measurement system using ultrasonic sensors," Proceedings of the 29th Annual IEEE International Conference of the EMBS, France, Aug. 2007.

k. j. somaiya college of engineering

An autonomous college affiliated to Mumbai University

Presents

PRAKALPA'17

State level Conference and Working Model Exhibition

Organised by

ISTE Students' Chapter (MH-60)

Certificate

This is to certify that Bro/Sis. Sanveg Rane

has participated in Working Model/ Project Competition based on theme "Artificial Intelligence: Innovation towards future" in Prakalpa'17 held on 1st April, 2017

Principal

Faculty Advisor

k. j. somaiya college of engineering

An autonomous college affiliated to Mumbai University

Presents

PRAKALPA'17

State level Conference and Working Model Exhibition

Organised by

ISTE Students' Chapter (MH-60)

Certificate

This is to certify that Bro/Sis. Hardik R. Shah

has participated in Working Model/ Project Competition based on theme "Artificial Intelligence: Innovation towards future" in Prakalpa'17 held on 1st April, 2017

Principal

Faculty Advisor

k. j. somaiya college of engineering

An autonomous college affiliated to Mumbai University

Presents

PRAKALPA'17

State level Conference and Working Model Exhibition

Organised by

ISTE Students' Chapter (MH-60)

Certificate

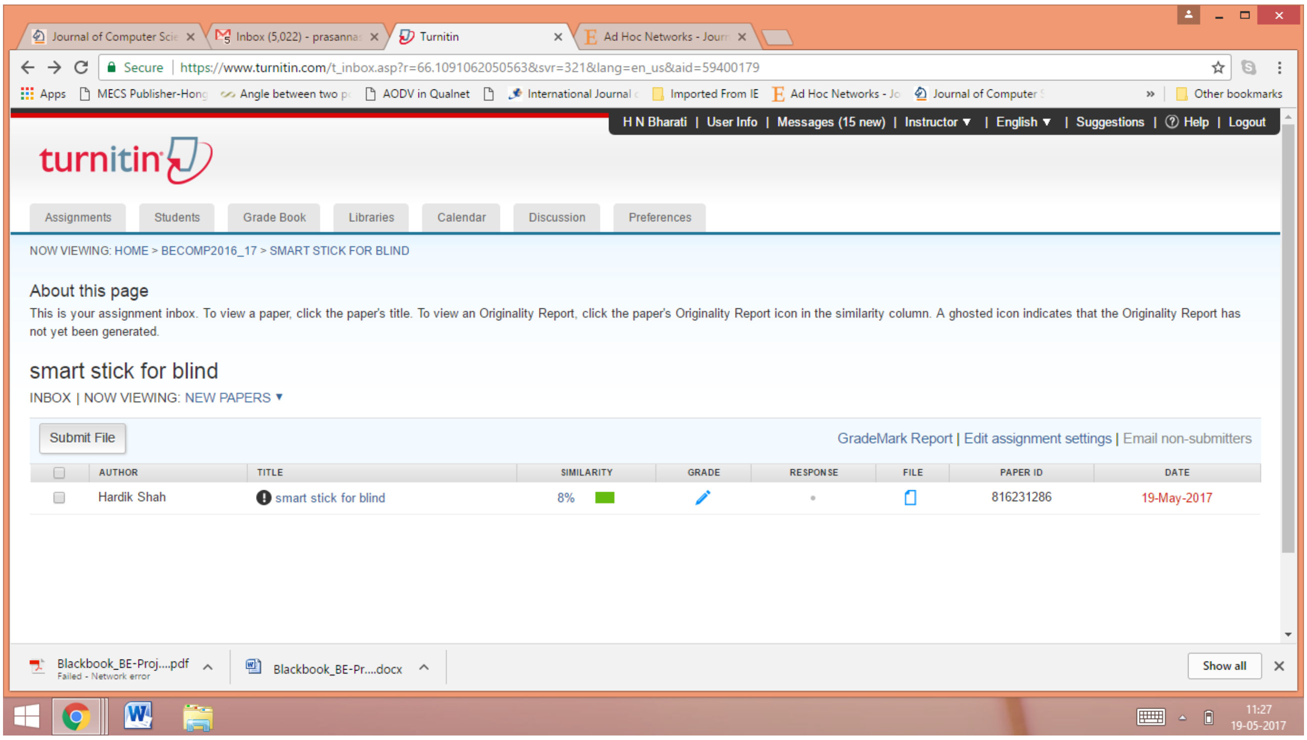
This is to certify that Bro/Sis. Dhiraj Uchil

has participated in Working Model/ Project Competition based on theme "Artificial Intelligence: Innovation towards future" in Prakalpa'17 held on 1st April, 2017

Principal

Faculty Advisor

9.5 Plagiarism report for final BE project report.



Only 8% similarity detected in plagiarism report.

Head,

Department of Computer Engineering