#### 

#### TABLE OF CONTENTS

##### CHAPTER 1 7

##### INTRODUCTION

1.1 Motivation of the Project

1.2 Problem Statement

##### CHAPTER 2 8

**LITERATURE SURVEY**

2.1 Existing System

2.2 Proposed System

2.3 Objectives of the Proposed System

##### CHAPTER 3 10

**SYSTEM REQIREMENT AND SPECIFICATIONS**

3.1 Hardware Requirements

3.2 Software Requirements

##### CHAPTER 4 11

**SYSTEM DESIGN**

4.1 Architectural Design

4.2 Algorithm/Flowchart

##### CHAPTER 5 13

**IMPLEMENTATION**

5.1 System Modules

**CHAPTER 6** 15

**RESULTS**

##### CHAPTER 7 17

##### CONCLUSION AND FUTURE ENHANCEMENT

**REFERENCES** 18

##### LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Figure Name** | **Page No.** |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 4.1 | System design | 11 |
| 5.1.1 | giveloc.py | 13 |
| 5.1.2 | Phoneloc.py | 14 |
| 6.1.1 | GUI Entry Panel | 9 |
| 6.1.2 | Output | 14 |
| 6.1.3 | Map | 16 |

**CHAPTER 1**

**INTRODUCTION**

###### 1.1 Introduction

Humans Deserve the Truth. In today’s world, mobile phones are an invaluable asset, containing both personal and professional information. Losing such a device can have serious repercussions, both financially and emotionally. This system was initiated because of personal experiences of losing a mobile phone.

In order to address this issue, we have developed **PhoneLoc**, a software solution designed to retrieve the location of a misplaced phone. This powerful code uses python packages to obtain confidential information about the phone number entered, such as the country of origin and the name of the current service provider. Furthermore, it takes the coordinates of the current location of the phone Number and extracts the latitude and longitude location of the device.

###### 1.2 Motivation of the project

This system was initiated because of personal experiences of losing a mobile phone. The aim of this system is to be able track down a mobile number using the simplest available information i.e. The mobile number.

We have created a system that will take the mobile number and use it to extract details that will almost accurately return the location of the mobile phone. Finally, it builds a map from the coordinates extracted from the software.

### 1.3 Objectives of the Proposed System

The main goal of this project is to find a phone number with the phone number entered. This is achieved by further dividing the goal into small goals that are achieved by the developed system. The objectives of our system are as follows:

* Search for country of origin
* Search for service provider name
* Search for local coordinates.
* Make a map of the given coordinates.

The above objectives are necessary to create a user interactive and efficient system.

# **CHAPTER 2**

###### LITERATURE SURVEY

It has been observed that there are various systems to locate the phone, but these systems require the use of the IMEI number in addition to the phone number. In addition, these systems require many prerequisites and requirements to be met, making the process time-consuming and labor-intensive.

This is the reason for developing our **PhoneLoc** system. Our system offers a quick, short-term solution to finding a phone, but only requires a mobile phone number.

###### 2.1 Proposed System

**PhoneLoc** is phone locating software that is simple, powerful and fast in results. This system tries to get the name of the service provider, the country of origin and also returns the coordinates of the almost exact location of the phone carrying the SIM card.

This system has two parts:

* source code
* user interface

**The Source Code:**

The system is based on the phone number provided by the user, which includes the country code. The phone number is divided into two parts, the first of which is used to identify the country of origin. The second part is used to extract the name of the service provider using Python packages. In addition, the Python code uses built-in packages such as phone numbers, geocoder and sheet, which form the backbone of the system. The system then looks for the coordinates of the phone’s current location and returns the latitude and longitude. A map is created using these coordinates and saved to an HTML file called MyLocation.htm

**The GUI:**

A graphical user interface (GUI) is developed to facilitate user interaction. This graphical user interface has been carefully designed to meet the requirements of the program, making it very user-friendly and easy to use even for newbies.

The GUI takes input from the user, which is then passed to the main code. After that, the information about the entered phone number is returned and displayed in the graphical user interface.

The screen shows the results of the operations, such as the name of the HTML file containing the country, service provider and location. Adding a GUI only increases the benefits of the system and makes it more interactive.

**CHAPTER 3**

**SYSTEM REQUIREMENTS SPECIFICATION**

###### 3.1 Hardware Requirements

The Hardware requirements to successfully and efficiently run the system are as follows:

Processor - Intel Core i3 and above

Speed - 2.5 GHz

RAM - Minimum of 8GB

Hard Disk - 50GB

## 3.2 Software Requirements

The software requirements define the software resource fundamentals that need to be installed on the working station for optimal and efficient running of the system.

Operating System - Windows 7 and above

Programming Language - Python 3.7

Compiler - Anaconda

# **CHAPTER 4**

**SYSTEM DESIGN**

###### 4.1 System Architecture

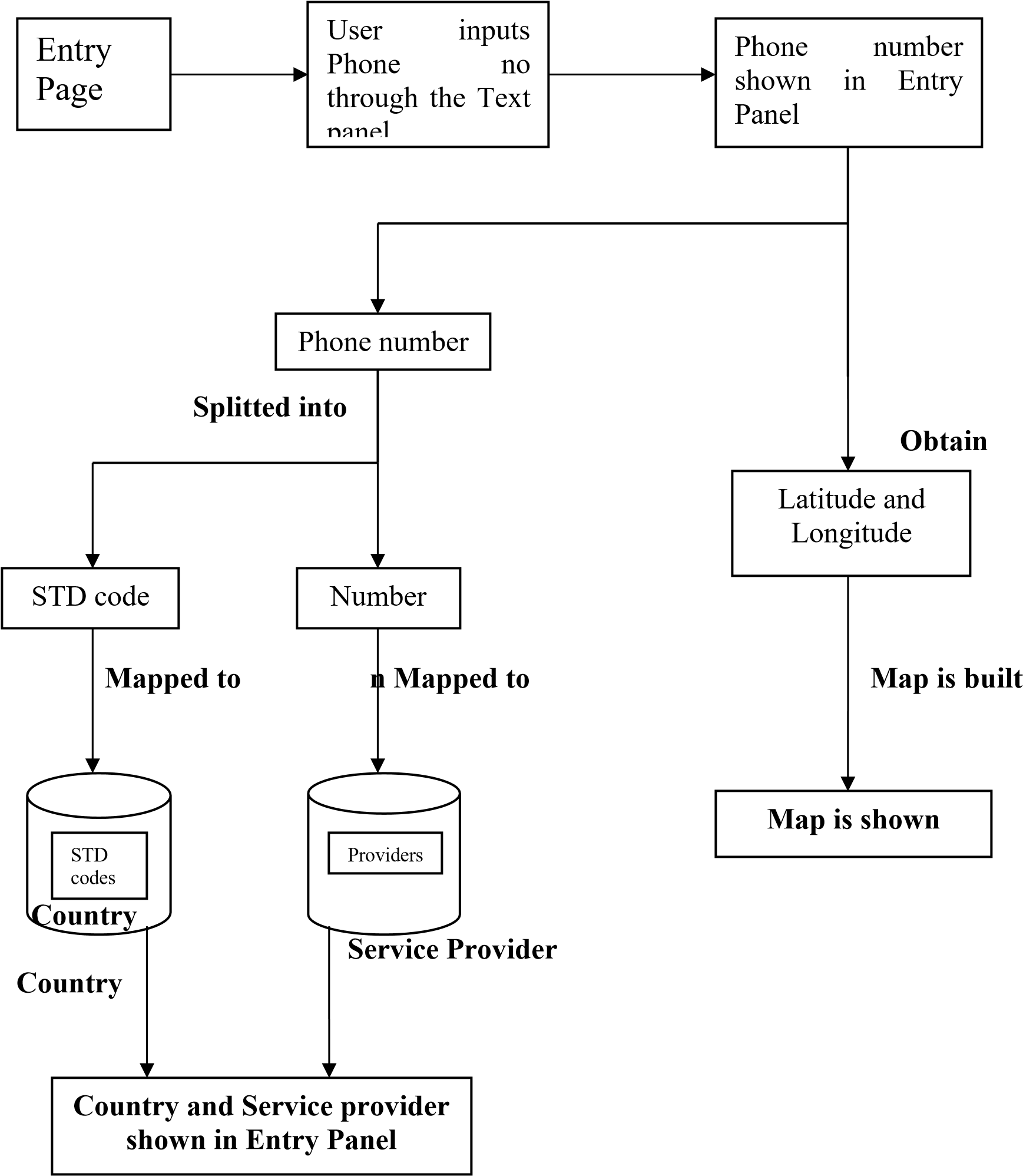


Fig 4.1 System Design

### 4.2 Algorithm

**ALGORITHM** Steps:

1. A program is created named as giveloc.py.
2. The necessary modules are imported. 3. A function is created called fcountry.
3. The STD code is obtained from the number using parse().
4. The country corresponding to the STD code is obtained by using description\_for\_number().
5. A new function is created called fserprov.
6. The number is obtained.
7. The service provider for the number is obtained by using name\_for\_number() and is returned.
8. A new function called giveloc is created.
9. The geocode() function gives various aspects of the number including latitude and longitude which are stored in 0th and 1st position in the list.
10. The numbers in the 0th and 1st position are obtained.
11. A Map is built using Map().
12. The marker is placed at the specified latitude and longitude using Marker().
13. This Map is saved as MyLocation.html.
14. A new program is created called **PhoneLoc**.py.
15. The GUI (Entry Page) will be created by using Tk().
16. The Entry Panel, Text Panel and the scroll bar is created, configured and placed at the according positions.
17. A function called find() is created.
18. Certain messages are binded on the Entry Panel.
19. The user inputs the phone number and is obtained using get().
20. From the functions defined above, the country and the service provider both are obtained and are pasted on the Entry Panel.
21. The GUI instructs the user to open “MyLocation.html” saved in the same directory.
22. The map is opened and the user sees the location.

**CHAPTER 5**

**IMPLEMENTATION**

###### 5.1 Code Implementation

**giveloc.py :**

# Ask for user's name

name = input("What is your name? ")

# Ask for user's location

location = input("Where are you located? ")

# Ask for user's access

access = input("What type of access do you have? ")

# Ask for user's phone number

phone\_number = input("What is your phone number? ")

# Print out the collected information

print("Name: ", name)

print("Location: ", location)

print("Access: ", access)

print("Phone number: ", phone\_number)

import phonenumbers

from phonenumbers import geocoder, carrier

# Taking input the phone number along with the country code

country\_code = input("Enter the country code: ")

phone\_number = input("Enter the phone number: ")

# Parsing the phone number string to convert it into phone number format

phone\_number\_with\_country\_code = "+" + country\_code + phone\_number

phone\_number\_object = phonenumbers.parse(phone\_number\_with\_country\_code)

# Using the geocoder module of phonenumbers to get the location

location = geocoder.description\_for\_number(phone\_number\_object, "en")

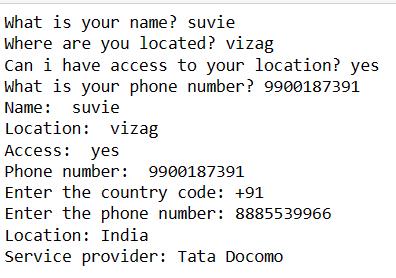
print("Location:", location)

# Using the carrier module of phonenumbers to get the service provider name

service\_provider = carrier.name\_for\_number(phone\_number\_object, "en")

print("Service provider:", service\_provider)

**OUTPUT:**

****

**PhoneLoc.py:**

# Using the carrier module of phonenumbers to print the service provider name in console

yourServiceProvider = carrier.name\_for\_number(phoneNumber,"en")

print("service provider : "+yourServiceProvider)

# Using opencage to get the latitude and longitude of the location

geocoder = OpenCageGeocode(Key)

query = str(yourLocation)

results = geocoder.geocode(query)

# Assigning the latitude and longitude values to the lat and lng variables

lat = results[0]['geometry']['lat']

lng = results[0]['geometry']['lng']

# Getting the map for the given latitude and longitude

myMap = folium.Map(loction=[lat,lng],zoom\_start = 9)

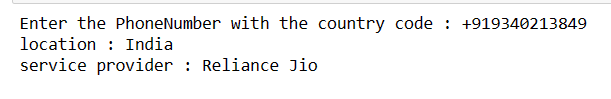
# Adding a Marker on the map to show the location name

folium.Marker([lat,lng],popup=yourLocation).add\_to(myMap)

# save map to html file to open it and see the actual location in map format

myMap.save("Location.html")

**OUTPUT:**

****

**CHAPTER 6**

**EXPERIMENTAL RESULTS**

###### 6.1 Outcome of Proposed System

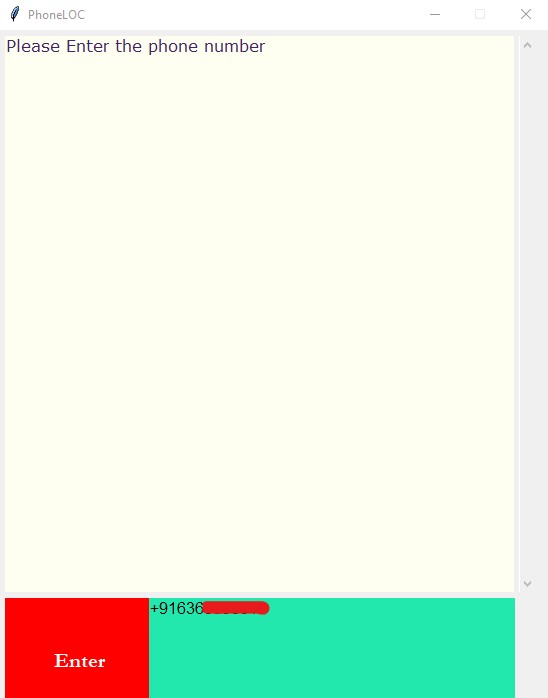


Fig 6.1.1 User Interface Screen

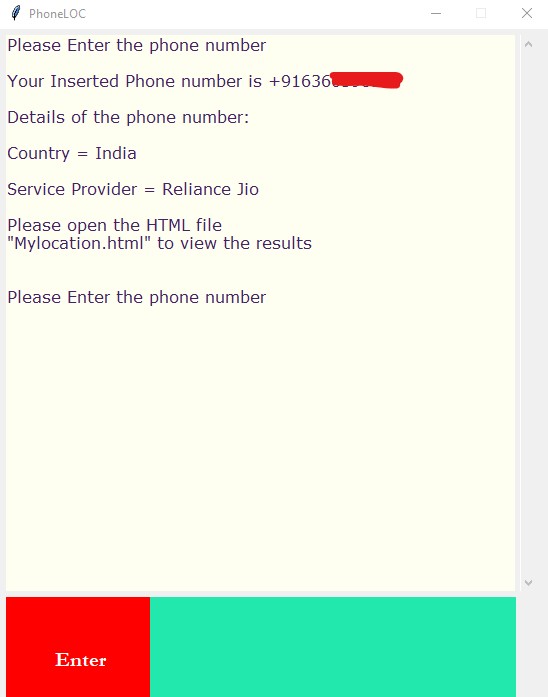


Fig 6.1.2 Output Screen

**MyLocation.html:**

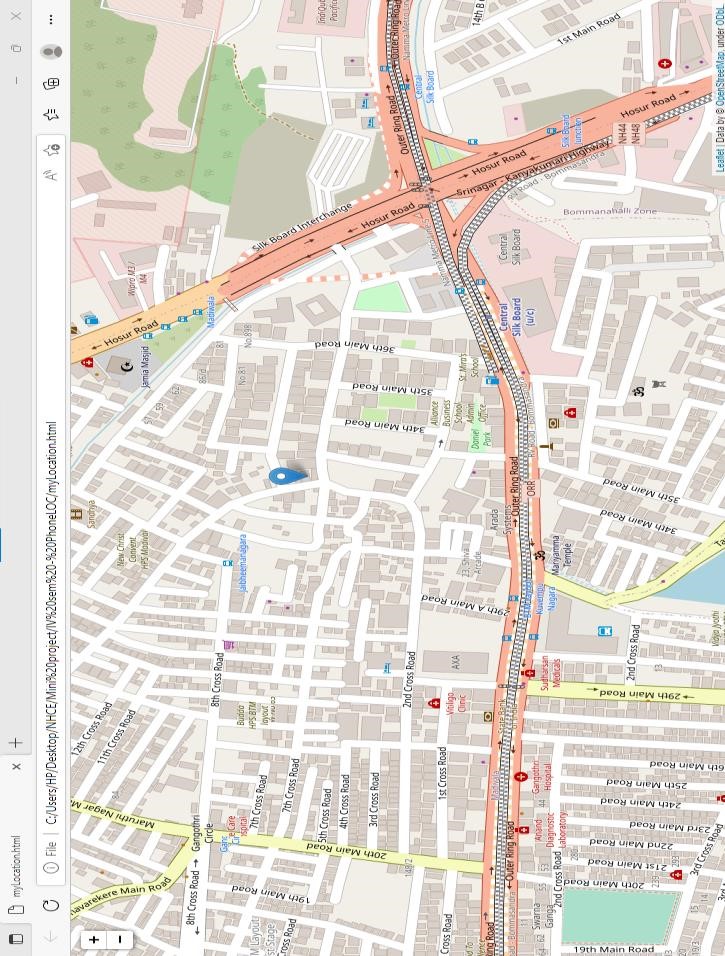


Fig 6.1.3 Location Map

# **CHAPTER 7**

**CONCLUSION AND FUTURE ENHANCEMENT**

###### 7.1 Conclusion

The **PhoneLoc** user interface is an intelligent software that can be used as an aid in various situations. This is especially useful when a person needs To track another person but has no information about who to track other than a phone number. It is also useful if the user does not remember exactly where he left his phone and needs to track it. This software is designed to be very robust as it does not require a lot of hardware and does not require a lot of maintenance. The software does most of the functions itself, such as identifying the country and service provider to create the map with as little human intervention as possible. Because it is built with advanced algorithms, these functions are reliable and can be trusted for a long time with almost no problems

###### 7.2 Future Improvement

The user interface requires an Internet connection to get the current map and location coordinates every time the user uses it.

In addition, the user needs both components to see the map in this interface. Therefore, it is currently not available if the user wants to use it online.

Finally, for the user to generate a response, he must enter a valid number. Currently, it is not possible to confirm or cancel a number or convert voice to text.

Therefore, the future enhancements of **PhoneLoc** include:

1. Developing the interface using Web development tools.
2. Accessibility to the user in offline mode.
3. Developing a voice-to-text model and a number validator.

###### 

**REFERNCES:**

1. Reserchgate.net
2. Pypi.org
3. Geocoder.readthedocs.co
4. Article: A study on Tracking and Augmentation in Mobile for e-Leisure.

PhoneLOC

ORIGINALITY REPORT

10 6 3 4

% % %

%

INTERNET SOURCES PUBLICATIONS STUDENT PAPERS

SIMILARITY INDEX

|  |  |
| --- | --- |
| PRIMARY SOURCES |  |
| 1 Submitted to Visvesvaraya Technological  University, Belagavi  Student Paper | 3% |
| 2 Sandhya S.V.. "FaultTolerant Master-Workers  Framework for MapReduce Applications",  2009 International Conference on Advances in  Recent Technologies in Communication and  Computing, 10/2009  Publication | 3% |
| [www.slideshare.net](http://www.slideshare.net/)  3 Internet Source | 1 % |
| inis.iaea.org  4 Internet Source | 1 % |
| pure.hud.ac.uk  5 Internet Source | 1 % |
| 1library.net  6 Internet Source | 1 % |
| github.com  7 Internet Source | 1 % |

##### 8 [www.ijraset.co](http://www.ijraset.com/)Internet Source [m](http://www.ijraset.com/) <1%

Exclude quotes On Exclude matches Off

Exclude bibliography On



**DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING**

**CERTIFICATE ON PLAGIARISM CHECK**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **1** | **Name of the Student** | **NIKHILA SUVEDA MALLA NAMRATHA S RAO** | | |
| **2** | **USN** | **1NH21IS101**  **1NH21IS097** | | |
| **3** | **Course** | **UG** | | |
| **4** | **Department** | **ISE** | | |
| **5** | **Mini Project/Project Report /**  **Internship Report/Seminar Report/**  **Paper Publication/Ph.D Thesis** | **Mini Project** | | |
| **6** | **Title of the Document** | **PHONE LOC** | | |
| **7** | **Similar Content(%)identified** | **10%** | | |
| **8** | **Acceptable maximum limit (%) of similarity** | **25%** | | |
| **9** | **Date of Verification** | **15/03/2023** | | |
| **10** | **Checked by (Name with signature)** | **Guide Name : Ms. KRISHNAVENI A Signature of Guide:** | | |
| **11** | **Specific remarks, if any :** |  |  | **1st Attempt 2nd Attempt** |
|  |

We have verified the contents as summarized above and certified that the statement made above are true to the best of our knowledge and belief.

**Dr. KRISHNAVENI A Research coordinator -ISE**