



**NEW HORIZON
COLLEGE OF ENGINEERING**

Autonomous College, Affiliated to VTU | Approved by AICTE New Delhi & UGC
Accredited by NAAC with 'A' Grade & Accredited by NBA

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

A MINI PROJECT REPORT ON

“PhoneLOC”

Submitted in the partial fulfillment of the requirements in the 3rd semester of

**BACHELOR OF ENGINEERING
IN**

INFORMATION SCIENCE AND ENGINEERING

FOR

**COURSE NAME: MINI PROJECT
COURSE CODE: 20ISE49A**

By

**NAMRATHA S – 1NH21IS097
NIKHILA SUVEDA – 1NH21IS101**



NEW HORIZON COLLEGE OF ENGINEERING

Autonomous College, Affiliated to VTU | Approved by AICTE New Delhi & UGC
Accredited by NAAC with 'A' Grade & Accredited by NBA

CERTIFICATE

Certified that the project work entitled “PhoneLoc” carried out by Mr. , bearing both bonafide students of 3rd semester in partial fulfillment for the award of Bachelor of Engineering in Information Science & Engineering of New Horizon College of Engineering, an autonomous institute affiliated to the Visvesvaraya Technological University, Belagavi during the year 2023-2024 It is certified that all corrections / suggestions indicated for Internal Assessment have been incorporated. The project report has been approved as it satisfies the academic requirements in respect of Mini Project work prescribed for the said Degree.

Name & Signature of Guide

Ms. KRISHNAVENI A

Name & Signature of HOD

DR. MOHAN H S

Examiners :

Name

Signature

1.

.....

2.

.....

ACKNOWLEDGEMENT

Any project is a task of great enormity and it cannot be accomplished by an individual without support and guidance. I am grateful to a number of individuals whose professional guidance and encouragement has made this project completion a reality.

We have a great pleasure in expressing our deep sense of gratitude to the beloved Chairman **Dr. Mohan Manghnani** for having provided us with a great infrastructure and well-furnished labs.

We take this opportunity to express our profound gratitude to the Principal **Dr. Manjunatha** for his constant support and management.

We are grateful to Head to **Dr. Mohan H S** of Department of ISE, New Horizon College of Engineering, Bengaluru for her strong enforcement on perfection and quality during the course of our mini project work.

We would like to express our thanks to the guide Ms. Krishnaverni A Senior Assist, Department of ISE, New Horizon College of Engineering, Bengaluru who has always guided us in detailed technical aspects throughout our mini project.

We would like to mention special thanks to all the Teaching and Non-Teaching staff members of Information Science and Engineering Department, New Horizon College of Engineering, Bengaluru for their invaluable support and guidance.

NAMRATHA S 1NH21IS097

NIKHILA SUVEDA 1NH21IA101

TABLE OF CONTENTS

CHAPTER 1	1
Introduction	1
1.1 Motivation of the Project	
1.2 Problem Statement	
 CHAPTER 2	 4
Literature Survey	
2.1 Existing System	
2.2 Proposed System	
2.3 Objectives of the Proposed System	
 CHAPTER 3	 8
System Requirement Specifications	
3.1 Hardware Requirements	
3.2 Software Requirements	
 CHAPTER 4	 12
System Design	
4.1 Architectural Design.....	12
4.2 Algorithm/Flowchart	
 CHAPTER 5	 18
Implementation	
5.1 System Modules	
 CHAPTER 6	 38
Results	
 CHAPTER 7	 39
Conclusion & Future Enhancement	39
 REFERENCES	

LIST OF FIGURES

Figure No.	Figure Name	Page No.
4.1	System design	
5.1.1	giveloc.py	
5.1.2	Phoneloc.py	
6.1.1	GUI Entry Panel	
6.1.2	Output	
6.1.3	Map	

Abstract

We've all experienced the panic when our phones go missing, even if it's only for a moment. Apart from the high costs of the mobile phones, it's packed with private information, from bank details to social media accounts. The importance of mobile phones goes beyond just making phone calls or sending data.

Sometimes it just so happens that we lose our phone or we may have to track someone else's phone. The catch is that we don't have that specific phone's IMEI number. Rather we just have the mobile number.

This is when PhoneLOC comes into play. The interface PhoneLOC developed can

accurately track down the location of the phone. With the help of just the phone number and Python packages, it gives us details of the service provider and the location can be viewed in maps.

As it is not a program with high maintenance, there is no need to upgrade the program once it is developed and hence can be available around the clock.

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

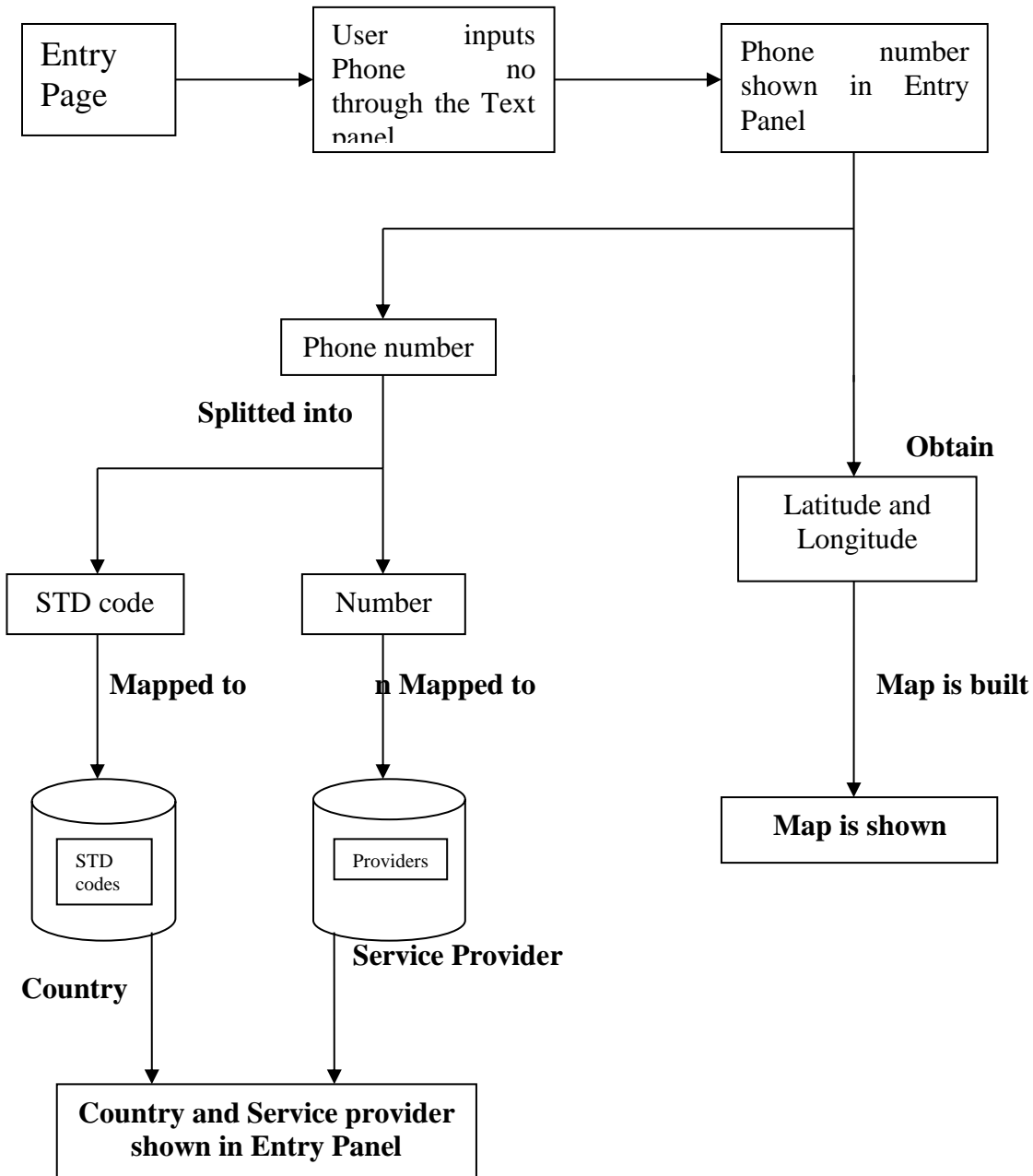


Fig 4.1 System Design

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

CHAPTER 5

IMPLEMENTATION

5.1 Code Implementation

giveloc.py :

```
import phonenumbers
import opencage
import folium
import time
from phonenumbers import geocoder

def fcountry(number):
    pepnumber = phonenumbers.parse(number)
    location = geocoder.description_for_number(pepnumber, "en")
    return(location)

from phonenumbers import carrier
def fserprov(number):
    service_pro= phonenumbers.parse(number)
    return(carrier.name_for_number(service_pro, "en"))

from opencage.geocoder import OpenCageGeocode
def giveloc(number):
    key = 'd2c03150d1d244c393dc090c87e9d4d1'

    geocoder = OpenCageGeocode(key)

    query = str(fcountry(number))

    results = geocoder.geocode(query)

    lat = results[0]['geometry']['lat']
    lngpip = results[0]['geometry']['lng']
    print(lat,lngpip)

    myMap = folium.Map(location=[lat, lngpip], zoom_start= 9)
    folium.Marker([lat, lngpip], popup=fcountry(number)).add_to(myMap)

    myMap.save("myLocation.html")
```

Fig 5.1.1 giveloc.py

PhoneLoc.py :

```

from tkinter import *
from findloc import fcountry,fserprov,giveloc

def find():
    TextPanel.config(state=DISABLED)
    TextPanel.yview(END)
    phno=EntryPanel.get("1.0",'end-1c').strip()
    EntryPanel.delete("0.0",END)
    TextPanel.config(state=NORMAL)
    TextPanel.insert(END,"\nYour Inserted Phone number is "+ phno+'\n')

    country=fcountry(phno)
    service=fserprov(phno)
    TextPanel.config(state=NORMAL)
    TextPanel.insert(END,"\nDetails of the phone number:\n")
    TextPanel.insert(END,"\nCountry = " + country + '\n')
    TextPanel.insert(END,"\nService Provider = "+service+"\n")
    TextPanel.insert(END,"\nPlease open the HTML file\n")
    TextPanel.insert(END,"\n\"Mylocation.html\" to view the results\n")
    giveloc(phno)
    TextPanel.insert(END,"\n\nPlease Enter the phone number\n")
    TextPanel.config(state=DISABLED)
    TextPanel.yview(END)

```

```

EntryPage=Tk()
EntryPage.title("PhoneLOC")
EntryPage.geometry("550x700")
EntryPage.resizable(width=FALSE,height=FALSE)

TextPanel=Text(EntryPage,bd=0,bg="ivory",height="8",width="50",font="Centaur")
TextPanel.config(foreground="#442265",font=("Verdana",12))
TextPanel.insert(END,"Please Enter the phone number\n")

scroller=Scrollbar(EntryPage,command=TextPanel.yview,cursor="heart")
TextPanel['yscrollcommand']=scroller.set

sendicon=Button(EntryPage,font=("Garamond",16,'bold'),text="Enter",width="12",
                height=5,bd=0,bg="red",activebackground="white",fg='white',command= find )

EntryPanel=Text(EntryPage,bd=0,bg="#23e8ad",width="29",height="5",font="Arial")

scroller.place(x=520,y=6,height=556)
TextPanel.place(x=6,y=6,height=556,width=509)
EntryPanel.place(x=150,y=568,height=127,width=366)
sendicon.place(x=6,y=568,height=127)

EntryPanel.mainloop()

```

Fig 5.1.2 PhoneLoc.py

CHAPTER 6

EXPERIMENTAL RESULTS

6.1 Outcome of Proposed System



Fig 6.1.1 User Interface Screen

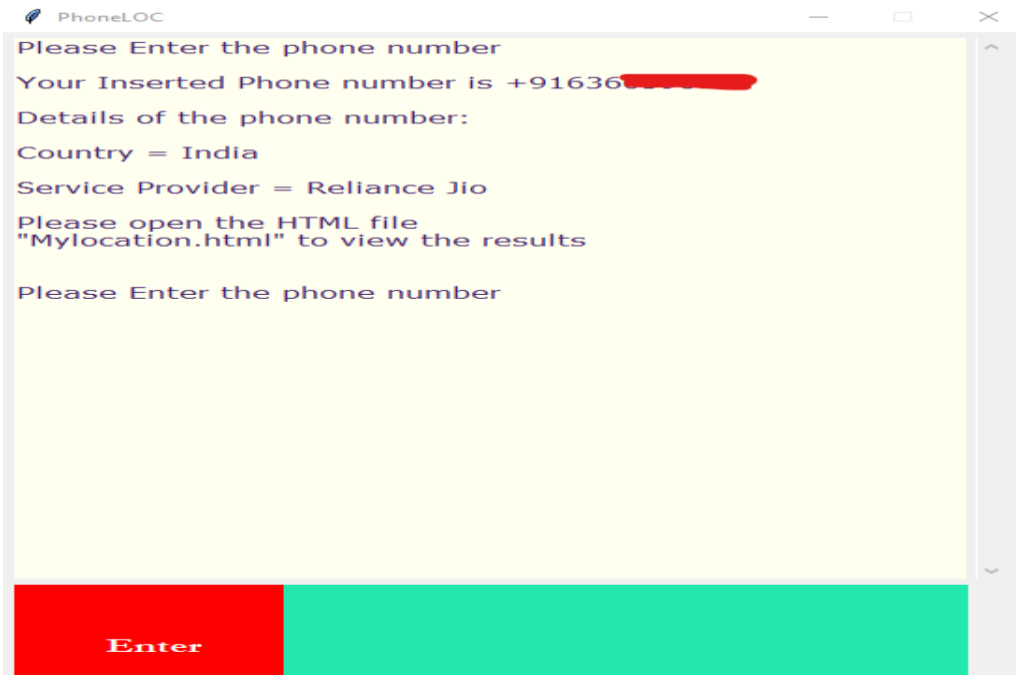


Fig 6.1.2 Output Screen

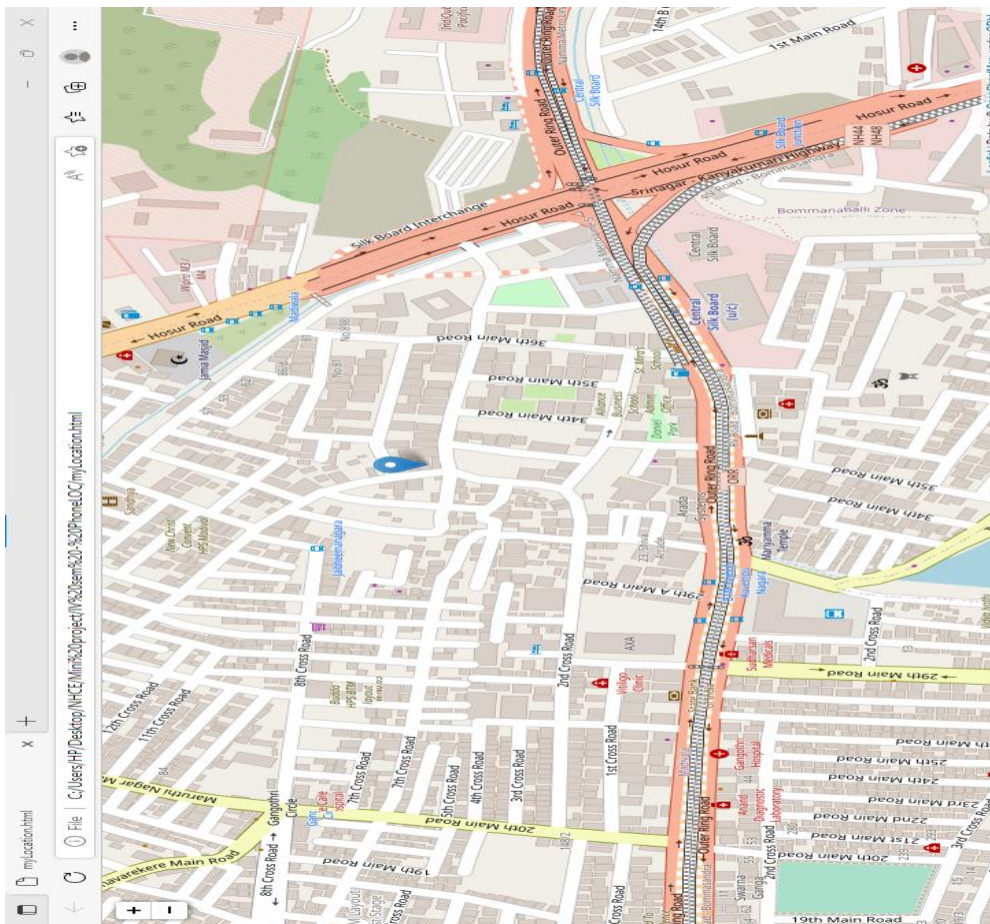
MyLocation.html:

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.

-

REFERENCES

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