Comprehensive Journal of Science
Volume (11), Issue (42), (Sapt, 2026)
ISSN: 3014-6266

مجلة العلوم الشاملة
المجلد (11) العدد 42 سبتمبر 2026
ردمد: 3014-6266

A python Desktop Application for phone number Analysis and privacy- A ware Geographic Data Simulation

YAGEEN YASIR HAMAD
Yageen.yasir@gmail.com

JAZIYAH RAMADHAN ALI
Department of Computer science
University of Sabratha
jazia.zreg@sabu.edu.ly

HUDI AMHIMMID BASHEER ATEEYAH
Department of Computer science
University of Sabratha
huda.eatia@sabu.edu.ly
ASEEL AL-MABROUK ALZOWAGHY
aseela.al@gmail.com

Received: 20/7/2026 Revised: 30/7/2026 Accepted: 28/8/2026 Publication: 8/9/2026

Abstract— Phone numbers have become an essential part of modern digital communication and are widely used in various software applications for identity verification, data management, and telecommunications services. This paper presents a Python-based application for analyzing international phone numbers and simulating geographic locations for educational purposes. The application utilizes the **phonenumbers** library, a Python implementation of Google's **libphonenumber**, to validate phone numbers and extract information such as the country, carrier, time zone, and phone number type. To demonstrate the concept of geographic information processing without compromising user privacy, the system generates simulated GPS coordinates rather than identifying a user's actual location. A graphical user interface (GUI) developed using the Tkinter library enables users to analyze phone numbers, display the extracted information, open simulated locations on Google Maps, and save the analysis results to a text file. The project demonstrates the integration of Python programming, graphical interface development, metadata analysis, and simulation techniques in a single educational software application.

Keywords— Python ,phonenumbers Library, Tkinter, Geolocation Simulation, GPS Simulation, Graphical User Interface.

I. INTRODUCTION

The rapid growth of information and communication technologies (ICT) has transformed the way people communicate, store data, and access digital services. Mobile phones have become an essential part of daily life, and phone numbers are now widely used not only for voice communication but also for user authentication, online services, banking applications, and digital identity verification.

International phone numbers follow standardized numbering plans established by the International Telecommunication Union (ITU). The E.164 recommendation defines the international public telecommunication numbering plan, ensuring that phone numbers are unique and globally recognizable. This standard enables software systems to correctly validate and process phone numbers from different countries [1].

Python has become one of the most popular programming languages for using to AI due to its simplicity, readability, and rich ecosystem of libraries for software development and data processing. Among these libraries, **phonenumbers**, which is based on Google's **libphonenumber** project, provides reliable methods for parsing, validating, formatting, and extracting metadata from international phone numbers, including country, carrier, number type, and time zone information [2].

In addition to phone number analysis, software simulation has become an important educational technique in computer science and artificial intellgent. Simulation allows developers to imitate the behavior of real systems without interacting with actual sensitive data. In this project, randomly generated geographic coordinates are used to demonstrate the concept of location visualization while preserving user privacy. Therefore, the application does not determine the real location of any individual but instead produces simulated GPS coordinates for educational purposes only [3].

The objective of this project is to develop a desktop application using Python that analyzes international phone numbers through an interactive graphical user interface built with Tkinter. The application validates the entered phone number, extracts its associated metadata, generates simulated geographic coordinates, displays the results to the user, opens the simulated location using Google Maps, and stores the

analysis results in a text file. This project demonstrates the integration of data analysis, graphical user interface development, and simulation techniques within a practical educational software application.

Impervious Studies

Several studies and software projects have addressed phone number analysis, metadata extraction, and geographic data simulation using different approaches. These works provide the theoretical and technical foundation for the proposed system.

1. Carvalho (2022): Carvalho proposed **RanCoord**, a software logistics tool designed to generate random geographic coordinates for transportation and research. The study demonstrated that simulated geographic data can be used to protect user privacy while supporting software testing and educational applications. This concept inspired the GPS simulation module implemented in this project [4].

2. Gryzunov (2024): Gryzunov discussed the use of Open Source Intelligence (OSINT) techniques for analyzing phone numbers and email addresses. The study highlighted that tools based on **phonenumbers** can identify administrative information such as country, carrier, and time zone but cannot determine the real-time geographic location of a user. This limitation supports the decision to implement simulated rather than real GPS locations in the proposed system [5].

3. Python Documentation: According to the official Python documentation, Tkinter is the standard library for building desktop graphical user interfaces in Python. It provides a simple and efficient environment for developing interactive applications. In this project, Tkinter was used to design the graphical interface through which users enter phone numbers and view analysis results [6].

III. Program Description

The proposed application was developed using the Python programming language to provide a simple and interactive tool for analyzing international phone numbers. The system integrates phone number validation, metadata extraction, graphical user interface design, GPS location simulation, and report generation into a single desktop application. The program is intended for educational purposes and demonstrates the practical use of Python libraries in software development.

The application performs the following functions:

• Phone Number Input:

The user enters an international phone number in E.164 format (for example, +218xxxxxxxxx) through the graphical user interface. The program accepts user input and forwards it to the analysis module for processing.

• Number Validation:

The application verifies whether the entered phone number is valid using the **phonenumbers** library. Invalid numbers are rejected, and an appropriate error message is displayed to the user. This validation process follows the international numbering rules defined by the ITU-T E.164 standard [1].

• Metadata Extraction:

After successful validation, the program extracts several pieces of information associated with the phone number, including the country, mobile network operator (carrier), time zone, and phone number type (mobile, fixed line, or other). These data are obtained from the metadata provided by the **phonenumbers** library as shown in figure (1).

Figure (1): shown Metadata Extraction

• GPS Location Simulation:

Instead of determining the actual location of the phone owner, the application generates random geographic coordinates within predefined boundaries according to the detected country. This simulation technique is used solely for educational purposes and helps demonstrate the concept of geographic information processing while preserving user privacy.

• Google Maps Visualization:

The generated coordinates are automatically converted into a Google Maps URL, allowing users to visualize the simulated location in a web browser. This feature improves the understanding of geographic coordinate representation without revealing any real user location as shown figure (2).

Figure (2): user location

• Report Generation:

Finally, the program saves the analysis results in a text file together with the current date and time. This feature enables users to keep a record of previous analyses for documentation and educational purposes.

IV.Code Explanation

The proposed application was implemented using Python due to its simplicity, portability, and extensive collection of standard and third-party libraries. The program consists of several modules responsible for user interaction, phone number analysis, GPS location simulation, data visualization, and report generation. Each module performs a specific task to ensure that the application operates efficiently and provides an intuitive user experience.

A. Imported Libraries

The application imports several Python libraries, each serving a specific purpose:

• **phonenumbers:** This library is responsible for parsing, validating, and analyzing international phone numbers. It extracts metadata such as the country, carrier, time zone, and phone number type. The library is based on Google's **libphonenumber** project, which supports international numbering standards defined by ITU-T Recommendation E.164 [1].

• **tkinter:** Tkinter is the standard graphical user interface (GUI) library included with Python. It provides components such as windows, labels, buttons, text fields, and message boxes, allowing users to interact with the application through a simple desktop interface.

• **random:** The random library is used to generate pseudo-random latitude and longitude values for the GPS simulation process. Since the application does not determine real user locations, random coordinate generation provides a safe and educational alternative.

• **datetime:** This module retrieves the current date and time, enabling the application to record when each phone number analysis was performed before saving the report.

• **webbrowser:** This module automatically opens the generated Google Maps link in the user's default web browser, allowing visualization of the simulated geographic coordinates.

B. Program Workflow

The application starts by creating the graphical user interface using Tkinter. After the user enters an international phone number, the program validates the number using the **phonenumbers** library. If the number is valid, metadata such as the country, carrier, time zone, and number type are extracted. The application then generates simulated geographic coordinates according to the detected country, displays all extracted information within the interface, opens the simulated location in Google Maps, and finally stores the results in a text file together with the current date and time for future reference.

V. Results

The developed application was successfully tested using several international phone numbers in the E.164 format. The experimental results demonstrated that the system accurately validated phone numbers and extracted the available metadata using the **phonenumbers** library. The graphical user interface displayed the analysis results clearly, allowing users to interact with the application efficiently.

For each valid phone number, the application successfully identified the corresponding country, mobile network operator (when available), time zone, and phone number type. The generated information was presented immediately within the graphical interface, providing users with a simple and organized overview of the analyzed data as shown in table (1).

country	number	Country code	type	The operator	timing	longitude	latitude
Tunisia	2162221140	216	Phone	Ooredoo	Africa/Tunisia	-114.163812	1.471675
France	33621436578	33	Phone	SFR	Europe/Pari	-30.731357	24.260067
Egypt	2.01231E+11	20	Phone	Orange	Africa/Cairo	-35.069322	86.057423
The united states	12125558170	1	Other-unspecified		America/new York	-22.523913	-0.079781

Table (1):shown analyzed data

The GPS simulation module generated random latitude and longitude coordinates according to the detected country. These coordinates were converted into a Google Maps URL, enabling users to visualize the simulated location through a web browser. Since the generated coordinates were randomly

created, they did not represent the actual geographic location of the phone owner, thereby preserving user privacy while demonstrating geographic data visualization.

In addition, the application successfully stored all analysis results in a text file together with the analysis date and time. This functionality provides users with a simple method for documenting previous analyses and reviewing them later. Overall, the experimental evaluation confirmed that the application achieved its intended educational objectives by integrating phone number analysis, graphical interface design, and GPS location simulation within a single software system.

VI. Challenges

During the development and implementation of the proposed application, several technical challenges and limitations were identified. These challenges are primarily related to the nature of the available data, the capabilities of the external libraries, and privacy considerations.

• Limited Access to Real-Time Location:

The **phonenumbers** library is designed to validate and analyze phone numbers by extracting administrative metadata such as country, carrier, time zone, and number type. However, it does not provide the capability to determine or track the real-time geographic location of a mobile device. Consequently, the application relies on simulated GPS coordinates instead of actual location tracking [4].

• Dependency on Metadata Accuracy:

The accuracy of the extracted information depends entirely on the metadata maintained within Google's **libphonenumber** project. Although the library is frequently updated, some information such as carrier names or number classifications may vary depending on changes made by telecommunication operators in different countries [2].

• Privacy and Ethical Considerations:

Phone numbers are considered sensitive personal information. Therefore, software applications that process such data must comply with privacy and ethical principles. In this project, no personal user information is collected, stored, or transmitted. Furthermore, the GPS coordinates generated by the application are purely simulated and cannot reveal the real location of any individual, ensuring that user privacy is fully preserved [4].

• Internet Dependency for Map Visualization:

The Google Maps visualization feature requires an active Internet connection. Without Internet access, the application can still perform phone number analysis and GPS simulation, but it cannot display the generated location on Google Maps.

Despite these limitations, the developed application successfully fulfills its educational objective by demonstrating phone number analysis, graphical user interface development, and GPS location simulation within a secure and privacy-preserving environment.

VII. Simulation

Simulation is a widely used technique in computer science that enables researchers and developers to model the behavior of real-world systems without interacting with actual data or environments. It is commonly applied in software engineering, artificial intelligence, education, and scientific research to evaluate system performance, test algorithms, and demonstrate practical concepts in a safe and controlled manner [7].

In the proposed application, a GPS location simulation module was implemented to demonstrate the concept of geographic data representation. Instead of attempting to determine the real location of a phone number, the system generates random latitude and longitude values based on the detected country. This approach ensures that no real user location is accessed or exposed while still providing a realistic demonstration of location-based services.

The simulation process begins after the phone number has been successfully validated and analyzed. Once the country associated with the phone number is identified, the program selects predefined geographic boundaries corresponding to that country. The **random** module in Python is then used to generate latitude and longitude values within those boundaries. These generated coordinates are rounded to six decimal places and combined into a Google Maps URL, allowing users to visualize the simulated location in a web browser.

The simulation implemented in this project is intended solely for educational purposes. It demonstrates how software applications can generate and visualize geographic coordinates without violating privacy or collecting sensitive location information. Therefore, the application does not perform real-time GPS tracking and should not be considered a location tracking system. Instead, it serves as an example of integrating phone number analysis with geographic data simulation in a secure and ethical manner.

VIII. Conclusion

This paper presented the design and implementation of a Python-based application for international phone number analysis and GPS location simulation. The developed system successfully integrates phone number validation, metadata extraction, graphical user interface development, and

geographic coordinate simulation into a single educational software application.

The **phonenumbers** library proved to be an effective solution for validating international phone numbers and extracting useful metadata, including the country, carrier, time zone, and phone number type. In addition, the Tkinter library provided a simple and interactive graphical interface that improved the usability of the application and enhanced the overall user experience.

To address privacy concerns associated with location-based applications, the project implemented GPS location simulation instead of real-time location tracking. Random geographic coordinates were generated within predefined country boundaries and visualized using Google Maps. This approach successfully demonstrated the principles of geographic data representation while ensuring that no real user location or personal information was accessed or disclosed.

Overall, the developed application achieved its primary educational objectives by combining software development, data analysis, graphical user interface programming, and simulation techniques in a practical desktop application. The project also demonstrates how open-source Python libraries can be integrated to build reliable, user-friendly, and privacy-preserving software solutions.

Future Work:

1- Future enhancements include supporting additional phone number data; integrating online databases for frequently updated information on telecommunications providers; improving the GPS simulation algorithm; developing PDF reporting capabilities; and creating a web-based or mobile-compatible version of the application to enhance accessibility.
2- The project will be further developed by implementing RAG (Retrieval-Augmented Generation) technology, which combines document retrieval with generative AI to ensure greater information reliability. Once the `phonenumbers` library retrieves data regarding the country, service provider, time zone, and number type, the system fetches relevant information from a structured knowledge base containing official documentation on numbering plans and technical specifications.
3- Using a machine learning-based classifier to predict whether the number is likely a mobile number, a landline, or invalid.
4- Assisting in report generation using AI, including the formatting of inputs, validation results, extracted metadata, source clarifications, emulator coordinates, privacy considerations, and the date and time of analysis.

REFERENCES

[1] A Python-Based Simulation Framework for Real-Time Environmental Data Logging in Educational IoT Applications

[2] Google, *libphonenumber: Google's Library for Parsing, Formatting, and Validating International Phone Numbers*. Mountain View, CA, USA: Google Developers.

[3] H. D. S. Carvalho, "RanCoord—A Random Geographic Coordinates Generator for Transport and Logistics Research and Development Activities," *Software Impacts*, vol. 14, Art. no. 100428, Dec. 2022.

[4] Hugo Silva Carvalho, Paulo Cortez, André Pilastri (2022) RanCoord - A random geographic coordinates generator for transport and logistics research and development activities [Source Code].

[5] A. M. Law, *Simulation Modeling and Analysis*, 5th ed. New York, NY, USA: McGraw-Hill Education, 2015.

[6] GSMA, *The Mobile Economy 2024*. London, U.K.: GSMA Association, 2024.