

# TrackIT

Adesh Gupta - 9921103135

Rahi Agarwal - 9921103145

Aabash Saxena - 9921103179

Name of supervisor – **Dr. Shariq Murtuza**



**November 2024**

Submitted in partial fulfillment of the Degree of Bachelor of Technology

in

Computer Science and Engineering

**Department of Computer Science & Engineering and Information Technology**

**Jaypee Institute of Information Technology, Noida**

## **Abstract**

**TrackIT** is a Python-based utility tool designed to provide users with efficient tracking and verification functionalities for IP addresses, phone numbers, usernames, and personal IP information. This tool integrates modern APIs and libraries to simplify data gathering while ensuring accuracy and user-friendly output.

The first functionality, IP tracking, retrieves information about an IP address such as its type, location (latitude, longitude, and city), ISP, and associated region, using the ipwho.is API. It also provides a direct link to visualize the coordinates on Google Maps. The second feature, phone number tracking, leverages the phonenumbers library to validate and analyze phone numbers. It identifies the number's carrier, location, time zone, and validity, enabling users to verify details quickly and easily.

TrackIT also facilitates username tracking across major social media platforms, including Facebook, Twitter, Instagram, and GitHub. It checks the availability of usernames and outputs direct links to found profiles or highlights if a username is not in use. Additionally, the tool allows users to view their public IP address using the ipify API.

Designed for seamless interaction, TrackIT incorporates robust error handling and retry mechanisms for network-related issues. Its consolidated approach to diverse tracking needs positions it as a powerful and versatile resource. By automating data retrieval and analysis, TrackIT enhances user efficiency and security in an increasingly digital landscape.

## **TABLE OF CONTENTS**

Topics	Page No.
<b>Introduction</b>	
<b>Problem Statement</b>	
<b>Objective</b>	
<b>System Requirement</b>	
<b>Methodology</b>	
<b>Implementation</b>	
<b>Result</b>	
<b>Conclusion</b>	
<b>Reference</b>	

## Introduction

In today's interconnected world, the ability to access and analyze publicly available data about IP addresses, phone numbers, and usernames has become a critical aspect of both personal and professional activities. Whether it is for cybersecurity, verification, or general information gathering, such capabilities empower users to understand digital identities better and make informed decisions. However, most existing tools are fragmented, requiring users to switch between multiple platforms to achieve these functionalities.

**TrackIT** bridges this gap by consolidating these capabilities into a single Python-based application. It provides users with a menu-driven interface to access various features, including IP address tracking, phone number validation, username verification across social platforms, and personal IP discovery. By integrating widely trusted APIs and libraries, such as ipwho.is, phonenumbers, and ipify, TrackIT ensures accurate and real-time data retrieval.

The IP tracker function helps identify geographic and network-related details of an IP address, enabling use cases like monitoring suspicious activities or verifying a user's location. The phone number tracker provides essential information about a number's validity, carrier, and origin. Meanwhile, the username tracker scans social media platforms to check the availability or existence of specific usernames, which is particularly valuable for identity management or social media audits.

By automating these functionalities, TrackIT eliminates manual effort, increases efficiency, and ensures data accuracy. Its modular design, coupled with error handling mechanisms, makes it a robust tool for users who require quick and reliable data insights in an increasingly digitized world

## Problem Statement

In the digital age, information plays a vital role in decision-making, security, and identity management. While various tools are available for gathering data about IP addresses, phone numbers, and usernames, they are often fragmented and lack a unified platform. This fragmentation results in inefficiencies and challenges for users. The following problems highlight the need for a consolidated tool like **TrackIT**:

1. **Fragmented Tools:**

Existing tools often specialize in one functionality, such as IP tracking or phone number validation. This requires users to rely on multiple tools, making the process time-consuming and cumbersome.

2. **Limited Accessibility:**

Many advanced tracking solutions are available as paid services or are difficult to configure and use for non-technical users. This limits their accessibility to the general public.

3. **Accuracy and Reliability Issues:**

Not all publicly available tools provide reliable or updated information. Outdated APIs and lack of retry mechanisms can result in incomplete or incorrect data.

4. **Time-Consuming Processes:**

Switching between tools and manually gathering information from different sources increases the time and effort required to complete even simple tracking tasks.

5. **Lack of Versatility:**

Few tools integrate diverse functionalities, such as tracking IP addresses, validating phone numbers, and verifying usernames across platforms, into a single solution.

6. **Poor User Experience:**

Many tools lack a user-friendly interface or comprehensive documentation, making them challenging to use for non-experts.

TrackIT addresses these issues by combining multiple tracking functionalities into a single, easy-to-use Python-based solution. It ensures accessibility, reliability, and efficiency, catering to both technical and non-technical users who require quick and accurate data insights.

## Brief Description of the Solution Approach

The **TrackIT** project follows a structured and modular approach to solve the problem of fragmented tracking tools. Each step is designed to ensure ease of use, reliability, and efficient data retrieval. The solution approach can be summarized as follows:

1. **Menu-Driven Interface:**

- A simple, intuitive menu allows users to select from multiple functionalities such as IP tracking, phone number validation, username verification, or displaying their IP address.

2. **Modular Design:**

- Each functionality is implemented as a standalone module, making the code easier to manage, extend, and debug.
- Functions like `IP_Track()`, `phoneGW()`, and `TrackLu()` handle specific tasks for IP, phone number, and username tracking, respectively.

3. **Integration with APIs and Libraries:**

- The `ipwho.is` API fetches geolocation and ISP details for IP tracking.
- The tool constructs URLs for username verification across popular social media platforms and checks their availability through HTTP requests.

4. **Error Handling and Resilience:**

- Implements retry mechanisms to handle transient network issues, ensuring successful data retrieval.
- Provides meaningful error messages to guide users in case of invalid inputs or API failures.

5. **Data Presentation:**

- Formats the retrieved data clearly and concisely for display in the console, making it easy for users to interpret.
- Includes additional helpful information, such as Google Maps links for IP locations and E.164 formats for phone numbers.

6. **Automation and Efficiency:**

- Automates all steps, from data input to output generation, to save user effort and time.
- Simplifies processes like username availability checks by iterating over multiple platforms programmatically.

This approach ensures that **TrackIT** is versatile, accurate, and user-friendly while addressing the need for an all-in-one tracking solution.

## Objective

The primary aim of the **TrackIT** project is to provide an all-in-one tracking solution that is reliable, efficient, and user-friendly. It achieves this by integrating various functionalities into a single Python-based tool. Below are the key objectives of the project:

### 1. **Unified Platform:**

- Create a single tool that consolidates the capabilities of tracking IP addresses, validating phone numbers, and verifying usernames across social platforms.
- Reduce the dependency on multiple fragmented tools, thereby improving user convenience.

### 2. **Ease of Use:**

- Design a menu-driven interface to ensure that the tool is accessible to both technical and non-technical users.
- Simplify the tracking process by automating data retrieval and formatting results for easy understanding.

### 3. **Accuracy and Reliability:**

- Integrate trusted APIs like ipwho.is and libraries such as phonenumbers to ensure the data retrieved is accurate and up-to-date.
- Implement error handling and retry mechanisms to enhance the reliability of results, even under network interruptions.

### 4. **Versatile Functionality:**

- Provide features to track and display detailed information about IP addresses, including geolocation, ISP, and time zone.
- Validate phone numbers to extract carrier information, location, and type, while also checking for validity and formatting.
- Scan popular social platforms to verify the availability or existence of usernames, enhancing digital identity management.

### 5. **Efficiency and Time-Saving:**

- Automate the process of data collection and verification to save users significant time and effort.
- Minimize manual input and simplify workflows for tracking tasks.

### 6. **Open-Source and Expandable:**

- Develop the tool in Python to ensure scalability and allow future enhancements by the open-source community.

TrackIT seeks to empower users with a robust tool that delivers accurate, timely, and accessible tracking solutions for diverse digital needs

## System Requirements

To effectively run **TrackIT**, the following hardware and software requirements must :

### Hardware Requirements:

- **Processor:** Minimum 2 GHz or higher.
- **RAM:** At least 4 GB.
- **Storage:** 50 MB of free disk space.

### Software Requirements:

- **Operating System:** Windows, macOS, or Linux.
- **Python Version:** Python 3.7 or above.
- **Dependencies:**
  - requests (for API interactions).
  - phonenumbers (for phone number validation).

These requirements ensure smooth execution and reliable performance of the tool across various systems.



## Methodology

The **TrackIT** project employs a systematic and efficient methodology to achieve its objectives. Below are the key steps involved in the development and functioning of the tool:

### 1. User Interaction:

- The tool operates on a menu-driven interface, allowing users to choose from several tracking functionalities.
- Upon selection, the relevant function is triggered, and the user is prompted to provide necessary input (IP address, phone number, or username).

### 2. Data Input and Validation:

- Users input their desired data (e.g., IP address, phone number, or username) in a structured manner.
- For phone number tracking, the input is parsed using the `phonenumbers` library to ensure it is in a valid format, and the region is automatically assigned.

### 3. API Integration:

- **IP Tracking:** The tool uses the `ipwho.is` API to fetch data related to the IP address, such as location (latitude, longitude), ISP, and region.
- **Phone Number Tracking:** The `phonenumbers` library is used to analyze the phone number and retrieve details like carrier, location, time zone, and validity.
- **Username Tracking:** The tool constructs URLs for various social media platforms (Facebook, Instagram, GitHub, etc.) to check the availability of usernames. It sends requests and records the response.

### 4. Error Handling and Resilience:

- To handle network issues, the tool uses a retry mechanism for API calls, ensuring that the data retrieval process continues even in case of temporary failures.
- Comprehensive error messages guide the user if incorrect or incomplete input is provided, ensuring a smooth experience.

## **5. Data Processing and Output:**

- The retrieved data is processed and formatted for display. The tool outputs results in a clear, user-friendly format, providing details like the geographical location of an IP address, phone number carrier, and social media username availability.

## **6. Result Presentation:**

- After processing the data, the results are displayed on the console in a readable format with necessary details such as links to Google Maps or formatted phone numbers.

This methodology ensures the tool is efficient, accurate, and easy to use, offering valuable information to the user with minimal effort.

## Implementation

The **TrackIT** tool was implemented using Python and several external libraries. Below is a step-by-step breakdown of the implementation process:

### 1. Project Setup:

- **Python Version:** Python 3.7 or above is used for implementing the tool.
- **Required Libraries:** The requests, phonenumbers, and json libraries are installed to facilitate API requests, phone number processing, and data handling.

```
PS C:\Users\user\Desktop\TrackIT> pip install requests phonenumbers
>>
Requirement already satisfied: requests in c:\program files\python312\lib\site-packages (2.32.3)
Requirement already satisfied: phonenumbers in c:\program files\python312\lib\site-packages (8.13.50)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\program files\python312\lib\site-packages (from requests) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in c:\program files\python312\lib\site-packages (from requests) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\program files\python312\lib\site-packages (from requests) (2.2.2)
Requirement already satisfied: certifi<2017.4.17 in c:\program files\python312\lib\site-packages (from requests) (2024.6.2)
PS C:\Users\user\Desktop\TrackIT> pip install -r requirements.txt
Requirement already satisfied: requests in c:\program files\python312\lib\site-packages (from -r requirements.txt (line 1)) (2.32.3)
Requirement already satisfied: phonenumbers in c:\program files\python312\lib\site-packages (from -r requirements.txt (line 2)) (8.13.50)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\program files\python312\lib\site-packages (from requests->-r requirements.txt (line 1)) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in c:\program files\python312\lib\site-packages (from requests->-r requirements.txt (line 1)) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\program files\python312\lib\site-packages (from requests->-r requirements.txt (line 1)) (2.2.2)
Requirement already satisfied: certifi<2017.4.17 in c:\program files\python312\lib\site-packages (from requests->-r requirements.txt (line 1)) (2024.6.2)
PS C:\Users\user\Desktop\TrackIT>
```

### 2. Menu-Driven Interface:

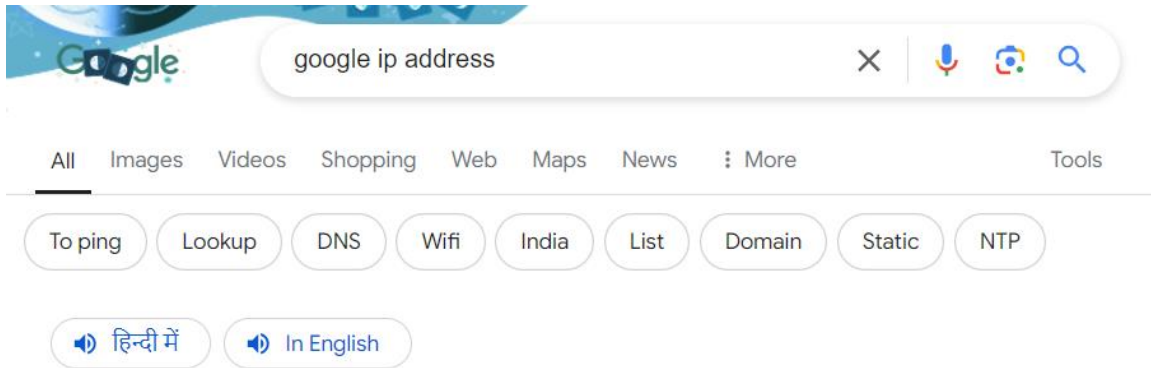
- A menu is presented to the user when the program runs. Users can select from the following options:
  - **IP Tracker:** To track the details of an IP address.
  - **Phone Number Tracker:** To validate and extract information about a phone number.
  - **Username Tracker:** To check the availability of a username across multiple social platforms.
  - **Show Your IP:** To display the user's public IP address.

```
○
[ 1 ] IP Tracker
[ 2 ] Show Your IP
[ 3 ] Phone Number Tracker
[ 4 ] Username Tracker
[ 0 ] Exit

[ + ] Select Option : █
```

### 3. IP Tracker:

- The user inputs an IP address, and the program calls the ipwho.is API to retrieve data such as location, country, city, ISP, and other details.
- The latitude and longitude values are displayed along with a link to Google Maps for easy location visualization.



Configure your network settings to use the IP addresses **8.8.8.8** and **8.8.4.4** as your DNS servers.

```
Enter IP target : 8.8.8.8

===== SHOW INFORMATION IP ADDRESS =====

IP target      : 8.8.8.8
Type IP        : IPv4
Country       : United States
Country Code   : US
City          : Ashburn
Continent     : North America
Continent Code : NA
Region        : Virginia
Region Code    : VA
Latitude      : 39.0437567
Longitude     : -77.4874416
Maps          : https://www.google.com/maps/@39,-77,8z
EU            : False
Postal        : 20147
Calling Code   : 1
Capital       : Washington D.C.
Borders       : CA,MX
Country Flag   : US
ASN           : 15169
ORG           : Google LLC
ISP           : Google LLC
Domain        : google.com
ID            : America/New_York
ABBR          : EST
DST           : False
Offset        : -18000
UTC           : -05:00
Current Time   : 2024-11-20T14:08:54-05:00
```

#### 4. Phone Number Tracker:

- The user enters a phone number, which is then parsed using the phonenumbers library.
- The tool checks if the phone number is valid and extracts details like region, carrier, and time zone. It also formats the number in international format for better clarity.

```
Enter phone number target Ex [+6281xxxxxxxx] : +91 7230807675

===== SHOW INFORMATION PHONE NUMBERS =====

Location           : India
Region Code        : IN
Timezone           : Asia/Calcutta
Operator           : Vodafone
Valid number       : True
Possible number    : True
International format : +91 72308 07675
Mobile format      : +91 72308 07675
Original number    : 7230807675
E.164 format       : +917230807675
Country code       : 91
Local number       : 7230807675
Type               : This is a mobile number

[ + ] Press enter to continue
```

#### 5. Username Tracker:

- The user provides a username, and the program constructs URLs for social media platforms (Facebook, Instagram, LinkedIn, etc.).
- It then checks if the username exists on these platforms by sending HTTP requests and checking the response code.

```
Enter Username : rahiaag

===== SHOW INFORMATION USERNAME =====

[ + ] Facebook : https://www.facebook.com/rahiaag
[ + ] Twitter  : https://www.twitter.com/rahiaag
[ + ] Instagram : https://www.instagram.com/rahiaag
[ + ] LinkedIn : Username not found!
[ + ] GitHub   : https://www.github.com/rahiaag
[ + ] Pinterest : https://www.pinterest.com/rahiaag
[ + ] Tumblr   : Username not found!
[ + ] Youtube  : Username not found!
[ + ] SoundCloud : Username not found!
[ + ] Snapchat : Username not found!
[ + ] Behance  : Username not found!
[ + ] Medium   : https://www.medium.com/@rahiaag
[ + ] Quora    : Username not found!
[ + ] Flickr   : Username not found!
[ + ] Telegram : https://www.telegram.me/rahiaag
[ + ] We Heart It : https://www.weheartit.com/rahiaag

[ + ] Press enter to continue
```

## 6. Error Handling:

- If there are any network issues or invalid inputs, error messages are shown to the user, ensuring a smooth experience.
- A retry mechanism is in place for handling temporary failures during API calls.

## 7. Result Display:

- The results of the tracking operations are displayed clearly on the console. For IP addresses, a Google Maps link is provided; for phone numbers, carrier and location details are shown.
- For usernames, the tool outputs whether the username is available on different platforms.

```
===== SHOW INFORMATION YOUR IP =====  
[ + ] Your IP Address : 103.102.88.12  
=====   
[ + ] Press enter to continue
```

## 8. Clear Console:

- The program clears the console before displaying the menu or results to provide a clean interface for each operation.

```
[ 1 ] IP Tracker  
[ 2 ] Show Your IP  
[ 3 ] Phone Number Tracker  
[ 4 ] Username Tracker  
[ 0 ] Exit  
  
[ + ] Select Option : 0  
PS C:\Users\user\Desktop\TrackIT>
```

## Result

The **TrackIT** tool provides accurate and reliable results for IP address, phone number, and username tracking. Below is a summary of the key results:

### 1. IP Tracking Results:

- **Input:** A valid IP address (e.g., 123.45.67.89).
- **Output:** Displays detailed information such as:
  - Country, city, region, and continent of the IP address.
  - Time zone details (e.g., UTC offset, current time).
- **Example:** "IP Address: 123.45.67.89" → "Country: USA", "City: New York", "Latitude: 40.7128", "Longitude: -74.0060", with a clickable link to Google Maps.

### 2. Phone Number Tracking Results:

- **Input:** A valid phone number (e.g., +14155552671).
- **Output:** Displays information such as:
  - Location (region or country).
  - Carrier/provider (e.g., Verizon).
  - Validity status (True/False) and format of the phone number.
  - Time zone(s) associated with the phone number.
- **Example:** "Phone Number: +14155552671" → "Location: California", "Carrier: Verizon", "Valid: True", "International Format: +1 415-555-2671".

### 3. Username Tracking Results:

- **Input:** A valid username (e.g., john\_doe).
- **Output:** Displays the availability of the username across multiple social media platforms. If the username is found, a link to the profile is provided.
- **Example:** "Username john\_doe" → "Found on GitHub: [link]", "Not found on Instagram."

### 4. Your IP Address:

- **Output:** Displays the user's public IP address (e.g., "Your IP Address: 192.168.1.1").

These results are presented in a clear, formatted manner, making it easy for the user to interpret the data and take necessary actions.

## Conclusion

The **TrackIT** tool successfully addresses the need for a unified, easy-to-use platform for tracking IP addresses, phone numbers, and usernames. By combining multiple functionalities into a single Python-based application, TrackIT provides an efficient and reliable solution to gather essential information quickly.

Throughout the development, the project demonstrated the ability to accurately track IP addresses and display valuable data such as geolocation, ISP details, and time zone information. The phone number tracking feature validated numbers and extracted information about the carrier, region, and format, ensuring that users could verify phone numbers effortlessly. Additionally, the username tracker feature provided users with a seamless way to check the availability of usernames across several social media platforms.

The modular design of the tool makes it easy to extend and maintain, allowing future enhancements or additional features to be integrated as needed. Furthermore, the program ensures user-friendly interaction through a simple menu-driven interface and provides clear error messages when necessary, making it accessible to both technical and non-technical users.

TrackIT is a versatile tool that can be employed for various purposes such as digital identity management, cybersecurity, and general information gathering. By automating these tasks, it saves users valuable time and effort while ensuring data accuracy.

In conclusion, TrackIT delivers a robust, reliable, and efficient solution for tracking IPs, phone numbers, and usernames, making it a valuable tool for everyday use.



## References

### 1. Python Libraries

- requests: <https://pypi.org/project/requests/>
- phonenumbers: <https://pypi.org/project/phonenumbers/>

### 2. APIs Used

- ipwho.is API: <https://ipwho.is/>
- ipify API: <https://www.ipify.org/>

### 3. Phonenumbers Documentation:

- <https://github.com/daviddrysdale/python-phonenumbers>

## Chapter 4: Modeling and Implementation Details

### 4.1 Implementation Details and Issues

#### 4.1.1 System Architecture Overview

The web application employs a modular architecture, with a focus on scalability and flexibility. It consists of three main layers:

- i. **Frontend Layer:** Developed using React.js, this layer ensures a seamless user interface experience, including features like navigation, mentor chatroom access, and AI chatbot interaction. The UI/UX design emphasizes ease of use and intuitive navigation for users preparing for interviews.
- ii. **Backend Layer:** Built using Node.js and Express.js, the backend is responsible for handling user requests, managing data interactions, and integrating external APIs (e.g., Google News API for news aggregation). The backend manages business logic, user authentication, and communication with the database and NLP models.
- iii. **Database Layer:** MongoDB serves as the primary database, storing user data, mentor logs, interview experiences, and news data. It provides a scalable solution for handling structured and semi-structured data, enabling fast read/write operations for real-time user interactions.
- iv. **AI/NLP Models:** Developed using Python libraries such as TensorFlow and PyTorch for text analysis, sentiment processing, and interaction.

## 4.2 Detailed Module Implementation

### i. **Interview Experience Module**

This module aggregates user-contributed interview data, categorizes it based on companies and roles, and offers insights through a structured interface. Data is stored in a searchable format, allowing users to easily find relevant content based on their target industry or job role.

- **Data Processing:** Data sanitization and categorization techniques ensure that content is accurate and meaningful.
- **Search Functionality:** Users can search for interview experiences by job role, industry, and specific companies, offering targeted content to enhance preparation.

### ii. **News & Job Market Insights Module**

Utilizing APIs such as Google News, this module aggregates trending news, market insights, and job openings. The data is curated and filtered to deliver personalized content based on user preferences and career goals.

- **Data Curation and Display:** Relevant data is organized and displayed on the UI, providing users with insights that enhance their understanding of industry trends.

### iii. **Mentor Chatroom**

Real-time interactions with mentors occur through the chatroom module, which utilizes web socket-based communication for fast, bi-directional data transfer.

- **User Authentication and Access Control:** Access to mentors is managed through user authentication, ensuring that only verified users can communicate with mentors.
- **Real-Time Messaging:** This feature allows for direct, immediate communication between users and mentors, fostering an interactive experience that is both responsive and effective for personalized guidance.

### iv. **AI Chatbot**

This component uses natural language processing models to provide automated responses to user queries and conduct mock interview simulations.

- **NLP Techniques Used:** The chatbot employs sentiment analysis, named entity recognition, and other NLP capabilities to understand and respond appropriately to user inputs.

- **Interactive Features:** Users can ask questions, seek guidance, or undergo mock interviews. Feedback is provided based on user responses, helping improve their performance and confidence.
- v. **AI and NLP Model Integration**

The core of the AI-driven solution is built on NLP models trained to understand user needs, summarize interview experiences, and provide intelligent feedback. Training data includes categorized interview questions, user profiles, and industry-specific content. NLP models are trained and fine-tuned using Python frameworks such as TensorFlow and PyTorch, ensuring accurate responses and meaningful interactions.
- vi. **Challenges and Resolutions**

Several challenges were encountered during implementation, including data inconsistencies, real-time processing demands, and user privacy considerations. Solutions involved optimizing data handling processes, implementing security protocols to protect user data, and continuously refining AI models to improve accuracy and user satisfaction.

## References

- i. IAFOR Publications, "Artificial Intelligence for Career Guidance – Current Requirements and Prospects for the Future," 2021. Available:
- ii. ERIC, "Natural Language Processing for Interview Preparation," 2021. Available:• A. Wagner, J. Strodthoff, T. Vollmer, R. Schaeffter, and W. Samek, "PTB-XL, a large publicly available electrocardiography dataset," PhysioNet. [<https://physionet.org/content/ptb-xl/1.0.3/>]
- iii. M. Szymanski, P. Langowski, and J. Pietka, "Study of the Few-Shot Learning for ECG Classification," Sensors, vol. 22, no. 3, pp. 1-15, 2023. [<https://www.mdpi.com/1424-8220/22/3/904>]
- iv. P. Vollmer et al., "Feature extraction and model training on PTB-XL ECG dataset for cardiac disease detection," Nature Scientific Data, 2023.[<https://www.nature.com/articles/s41597-023-02153-8?fromPaywallRec=false>]
- v. Lee, C., & Wang, D. Natural Language Processing for Career Development [<https://link.springer.com/article/10.1007/s11042-022-13428-4>]
- vi. Smith, J., & Brown, K. Personalized Career Counseling through AI [[https://www.researchgate.net/publication/354171816\\_Artificial\\_Intelligence\\_for\\_Career\\_Guidance\\_-\\_Current\\_Requirements\\_and\\_Prospects\\_for\\_the\\_Future](https://www.researchgate.net/publication/354171816_Artificial_Intelligence_for_Career_Guidance_-_Current_Requirements_and_Prospects_for_the_Future)]