Project Report submitted to

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

in partial fulfillment of the requirements for the
Mobile Application Development Laboratory (BCS586) of 5th
semester for the degree

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

Submitted by

Mahalakshmi P S	**1KG19CS064**
Prajwal koushik C	**1KG19CS087**
Raghu kisthannavar	**1KG22CS093**
Rakesh V	**1KG22CS094**

Under the Guidance of

Mrs. Amitha S
Assistant Professor
K S School of Engineering and Management

Department of Computer Science and Engineering
K.S. School of Engineering and Management
No. 15, Mallasandra, off Kanakapura Road, Bengaluru-560109 **2024-2025**

K.S. SCHOOL OF ENGINEERING AND MANAGEMENT
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the mini-project entitled **Billionaire Statistics 2023** is a bonafide work carried out by

Mahalakshmi P S	**1KG19CS064**
Prajwal koushik C	**1KG19CS087**
Raghu kisthannavar	**1KG22CS093**
Rakesh V	**1KG22CS094**

in partial fulfillment for the award of **Bachelor of Engineering** in **Computer Science and Engineering** of **Visvesvaraya Technological University, Belagavi,** during the year 2024-2025. It is certified that all the suggestions indicated during internal assessment have been incorporated in the report and this report satisfies the academic requirement with respect to **Mini-project Laboratory (BCS586) of V semester** prescribed for the degree.

Guide	**Head of the Department**	**Principal**
Mrs Amitha S Assistant Professor Dept. of CSE	**Dr K Venkata Rao** Professor & Head, Dept. of CSE	**Dr. K. Rama Narasimha** Principal/ Director

Examiners

_____ Name and Signature of Examiner-1

_____ Name and Signature of Examiner-2

K.S. SCHOOL OF ENGINEERING AND MANAGEMENT
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We	**Mahalakshmi P S**	**1KG19CS064**
	Prajwal koushik C	**1KG19CS087**
	Raghu kisthannavar	**1KG22CS093**
	Rakesh V	**1KG22CS094**

the students of **BE V Semester (Computer Science and Engineering)** declare that the mini-project entitled **"BILLIONAIRE STATISTICS 2023"** is carried out by us as a partial fulfillment of academic requirement of degree under **Visvesvaraya Technological University.** The content in the report are original and are free from plagiarism and other academic dishonesty and are not submitted to any other University either partially or wholly for the award of any other degree.

NAME	USN	Signature
Mahalakshmi P S	**1KG19CS064**	
Prajwal koushik C	**1KG19CS087**	
Raghu kisthannavar	**1KG22CS093**	
Rakesh V	**1KG22CS094**	

Date:
Place: Bengaluru

ACKNOWLEDGEMENT

The successful completion of this project was made possible with the help of guidance received from our faculty members. We would like to avail this opportunity to express our sincere thanks and gratitude to all of them.

We are grateful to our management for providing the necessary infrastructure and an ambience environment to work. We express our profound gratitude to **Dr. K Rama Narasimha, Principal and Dr. K Venkata Rao, Head, Department of Computer Science and Engineering, KSSEM, Bengaluru** for giving us opportunity and support to complete the mini- project.

We are grateful to my guide **Mrs. Amitha S, Assistant Professor, Department of Computer Science and Engineering, KSSEM, Bengaluru** for her valuable suggestions and advice throughout our work period.

We thank mini project Coordinators, **Mrs.Amitha S, Associate Professor and Mrs.Sougandhika Narayan, Assistant Professor, Department of Computer Science and Engineering, K.S. School of Engineering and Management, Bengaluru**, for their constant support and guidance.

We would also like to thank all the staff members of Department of Computer Science and Engineering for their support and encouragement. Finally, we would like to thank all of our friends without whose help and encouragement this project would have been impossible.

Definitely most, we want to thank our family. Words cannot express our thanks to our family members for all their love and encouragement.

MAHALAKSHMI P S	1KG22CS064
PRAJWAL KOSUHIK C	1KG22CS087
RAGHU KISTHANNAVAR	1KG22CS093
RAKESH V	1KG22CS094

ABSTRACT

In 2023, the global billionaire population reflected significant shifts shaped by dynamic economic conditions and industry innovations. With a total of approximately 2,640 billionaires, their combined net worth reached $13.1 trillion, showing resilience despite ongoing geopolitical tensions and market fluctuations.

Technology remained the leading wealth generator, while sectors like renewable energy, biotechnology, and digital assets gained traction, introducing a new generation of billionaires. Geographically, the United States retained its dominance with the highest number of billionaires, while Asia, led by China and India, saw remarkable growth, solidifying its status as a rising powerhouse in wealth creation.

The year also marked a modest increase in female billionaires, driven by advancements in diverse sectors such as e-commerce, media, and green technologies. Philanthropy played a central role, with many billionaires channeling resources into climate change initiatives, global health, and education, highlighting their growing influence on societal priorities.

This report presents an in-depth analysis of billionaire statistics for 2023, offering valuable insights into their industries, geographic distribution, and evolving contributions to the global economy.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 PROBLEM STATEMENT / AIM

To analyze billionaire statistics for 2023, focusing on their wealth distribution, industry dominance, and regional trends, while highlighting emerging economic and social implications. This study seeks to provide insights into the evolving landscape of global wealth creation.

1.2 SCOPE

The primary objective of billionaire statistics in 2023 is to analyze the distribution, growth, and trends in global billionaire wealth, focusing on industries, geographic regions, and demographic factors. It aims to identify key drivers of wealth creation, including emerging sectors like renewable energy and technology, while exploring the socioeconomic impacts of concentrated wealth, such as philanthropy and economic inequality. This analysis seeks to provide a comprehensive understanding of how billionaires influence global economies, innovation, and societal priorities, offering insights for policymakers, researchers, and stakeholders.

1.3 PROJECT DESCRIPTION

The "Billionaire Statistics in 2023" project analyzes global trends in billionaire wealth, focusing on their geographic distribution, industries, and demographics. It highlights key sectors like technology and renewable energy, examines gender and age representation, and explores changes in net worth. The project also evaluates the socioeconomic impact of billionaires, including their philanthropic efforts. This study aims to provide clear insights into the role of billionaires in shaping global economies and societal priorities.

The project highlights that over 60% of billionaires are self-made, reflecting entrepreneurship as a key driver of wealth. Women billionaires accounted for less than 15% of the total, signaling a persistent gender gap. The median age of billionaires was 65, with a notable rise in younger billionaires under 40, particularly in tech and startups.

.

Chapter 2

Hardware and Software Requirements

Hardware Requirements:

1. **Processor**:
    - Minimum: Intel Core i5 or AMD Ryzen 5
    - Recommended: Intel Core i7 or AMD Ryzen 7 for better performance in data processing and web development.
2. **RAM**:
    - Minimum: 8 GB
    - Recommended: 16 GB or higher, especially if handling large datasets in Python or SQL databases.
3. **Storage**:
    - Minimum: 500 GB HDD
    - Recommended: 512 GB SSD or more for faster read/write speeds during database operations and project tasks.
4. **Graphics Card**:
    - Minimum: Integrated graphics
    - Recommended: Dedicated GPU (NVIDIA or AMD) if handling data visualizations or complex web-based graphics.
5. **Monitor**:
    - Minimum: 15.6-inch display
    - Recommended: 24-inch or larger for better multitasking, viewing code, and analysis simultaneously.
6. **Internet Connection**:
    - A stable and high-speed internet connection for accessing cloud storage, APIs, and external data sources.

Software Requirements:

1. **Operating System**:
    - Windows 10 or 11, or a Linux-based system for compatibility with development tools.
2. **IDE (Integrated Development Environment)**:
    - **PyCharm** (Community or Professional Edition) for Python development, including data analysis, SQL integration, and web development tasks.
3. **Programming Language**:
    - **Python** (version 3.6 or higher) for data analysis, manipulation, and SQL integration. Libraries such as:
        - **SQLAlchemy** for connecting Python with SQL databases.
4. **SQL Database Management**:
    - **MySQL**, **SQLite**, or **PostgreSQL** for storing and querying billionaire data. Use **MySQL Workbench** or **pgAdmin** for managing databases.
    - **SQLAlchemy** for integrating SQL queries with Python and performing data operations.
5. **Web Development Tools**:
    - **HTML5** for creating the web interface to display the results.
    - **CSS** for styling the webpage and providing a user-friendly layout.
    - **JavaScript** for interactivity and dynamic features, if needed.
    - **Bootstrap** is used for designing responsive, mobile-first websites with prebuilt components and customizable styles.
6. **Version Control**:
    - **Git** for version control, ensuring proper collaboration and code management. Use **GitHub** for repository management.
7. **Data Visualization & Reporting Tools**:.
    - **Excel** or **Google Sheets** for additional data analysis or manual validation if necessary.

8. **Security**:
    - **Antivirus software** and **VPN** for secure online activities.

Tools for Integration and Data Analysis:

1. **Python Libraries**:.
    - **SQLAlchemy** for connecting to and querying SQL databases.
2. **Web Development**:
    - **HTML** and **CSS** for building the website to display the data and insights.
    - **JavaScript** (optional) for enhancing interactivity on the webpage (e.g., data filters or charts).
3. **Database**:
    - **MySQL** for storing billionaire statistics.
    - Use **SQL** queries for extracting, analyzing, and manipulating data from the database.

By using PyCharm for development, Python for data manipulation and analysis, SQL for database management, and HTML for creating a user-friendly web interface.

Chapter 3

DESIGN

3.1 ER – DIAGRAM

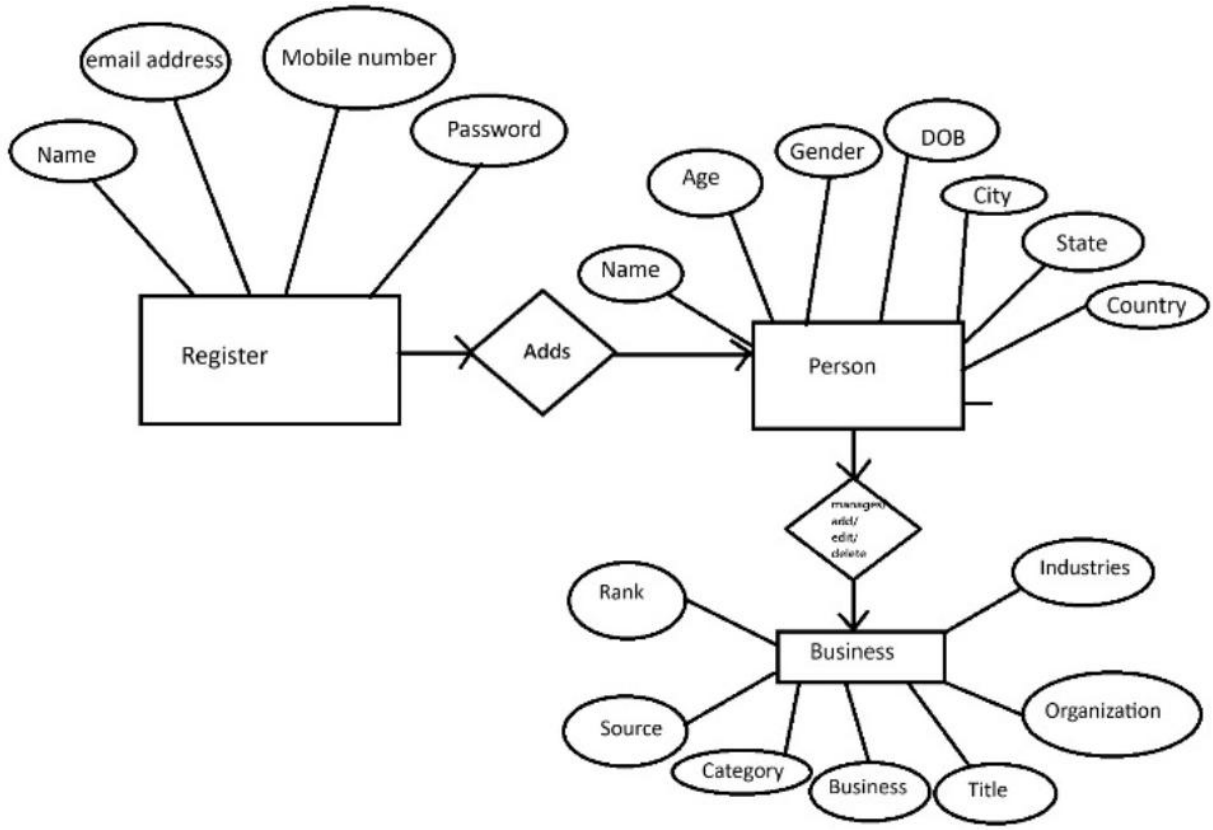


Fig 3.1 Flow of billionaires statistics 2023

3.2 Files Used

1. HTML Files (Frontend)

- **base.html**: base template that includes common layout elements (like headers, footers, navigation) used by other HTML templates.
- **home.html**: A page that displays the billionaires names and networth and contains search option.
- **Add_data.html**: In web development, 'Add Data' refers to the action of inserting new information or records into a system, database, or application
- **help.html**: A page that displays the FAQ's, contact us.
- **statistics.html**: a dashboard is a user interface that provides an overview of key information, metrics, and controls, often displayed in a visually organized and interactive format.

2. Python Files (Backend)

- **app.py**: The main Flask application file, containing routes, logic for rendering templates, and handling requests from users (adding, updating, deleting informations).
- **data.py**: Contains the billionaires names and networth,industries.
- **db.py**: Handles the database setup and connection logic (for example, connecting to SQLite or another database).

3. CSS Files (Styling)

- **styles.css**: Defines the styles for your pages, such as colors, layouts, fonts, etc.

4. Database Files

- **database.db**: The actual SQLite database file where billionaire statistics data and other information are stored.

3.3 Description of Function

Table 3.3 Description of Functions

Sl.No	Functions	Description
1.	Register	In web development, a register refers to a user interface element or system function that allows users to create an account by providing required information like username, email, and password.
2.	Update	In web development, an update refers to modifying or improving existing content, data, or software to ensure accuracy, functionality, and performance
3.	Button	In Android, Button represents a push button. A Push buttons can be clicked, or pressed by the user to perform an action.
4.	Delete	In web development, delete refers to the action of removing data, content, or records from a system, database, or interface permanently or temporarily
5	Submit	In web development, submit refers to the action of sending data or a form's input to a server or system for processing
6	See more	In web development, 'see more' is typically a link or button that allows users to view additional content or details beyond what is initially displayed

Chapter 4

IMPLEMENTATION 4.1 Code Snippets

4.1.1 MainActivity.app.py

```
from flask import Flask, jsonify, request, render_template, redirect, url_for
from utils.db import db
from models.data import *
from flask_sqlalchemy import SQLAlchemy

flask_app = Flask(__name__)
flask_app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///data.db'

db.init_app(flask_app)

# Create tables
with flask_app.app_context():
    db.create_all()


@flask_app.route('/')
def index():
    data = Business.query.all()
    return render_template('index.html', content=data)


@flask_app.route('/help')
def help_page():
    return render_template('help.html')


@flask_app.route('/modify')
def modify():
    data = Business.query.all()
    return render_template('modify.html', content2=data)

@flask_app.route('/elon_mask')
def elon_mask():
    return render_template('elon_mask.html')

@flask_app.route('/Bernard')
def Bernard():
    return render_template('Bernard.html')
```

```
@flask_app.route('/jeffbezos')
def jeffbezos():
    return render_template('jeffbezos.html')

@flask_app.route('/larryellison')
def larryellison():
    return render_template('larryellison.html')


@flask_app.route('/warrenbuffet')
def  warrenbuffet():
    return render_template('warrenbuffet.html')

@flask_app.route('/dashboard')
def dashboard():
    return render_template('dashboard.html')


@flask_app.route('/about-us')
def about():
    return render_template('about-us.html')


@flask_app.route('/add_data')
def add_data():
    return render_template('add_data.html')


@flask_app.route('/register')
def register_page():
    return render_template('register.html')


@flask_app.route('/submit', methods=['POST'])
def submit():
    try:
        form_data = request.form.to_dict()
        print(f"form_data: {form_data}")

        # Person data
        personName = form_data.get('personName')
        age = form_data.get('age')
        gender = form_data.get('gender')
        birthdate = form_data.get('birthdate')
        city = form_data.get('city')
        state = form_data.get('state')
        country = form_data.get('country')
```

```
        # Business data
        rank = form_data.get('rank')
        source = form_data.get('source')
        finalWorth = form_data.get('finalWorth')
        category = form_data.get('category')
        organization = form_data.get('organization')
        industries = form_data.get('industries')
        title = form_data.get('title')

        # Check if person exists
        person = Person.query.filter_by(personName=personName).first()
        if not person:
            person = Person(
                personName=personName,
                age=age,
                gender=gender,
                birthdate=birthdate,
                city=city,
                state=state,
                country=country,
            )
            db.session.add(person)
            db.session.commit()

        # Add Business
        business = Business(
            rank=rank,
            source=source,
            finalWorth=finalWorth,
            category=category,
            organization=organization,
            industries=industries,
            title=title,
            person_id=person.id,
        )
        db.session.add(business)
        db.session.commit()
        print("Submitted successfully")
        return redirect(url_for('index'))
    except Exception as e:
        db.session.rollback()
        return jsonify({"error": str(e)}), 500


@flask_app.route('/register-user', methods=['POST'])
def register_user():
    try:
```

```
form_data = request.form.to_dict()
        print(f"form_data: {form_data}")

        name = form_data.get('name')
        mobile_no = form_data.get('mobile_no')
        email_id = form_data.get('email_id')
        password = form_data.get('password')

        # Check if user already exists
        if Register.query.filter_by(email_id=email_id).first():
            return jsonify({"message": "User already exists"}), 400

        # Register new user
        user = Register(name=name, mobile_no=mobile_no, email_id=email_id, password=password)
        db.session.add(user)
        db.session.commit()
        print("Registered successfully")
        return redirect(url_for('add_data'))
    except Exception as e:
        db.session.rollback()
        return jsonify({"error": str(e)}), 500


@flask_app.route('/update/<int:id>', methods=['GET', 'POST'])
def update(id):
    try:
        data = Business.query.get_or_404(id)
        print(data.person.id)

        if request.method == 'POST':
            # Update person data
            data.person.personName = request.form['personName']
            data.person.age = request.form['age']
            data.person.gender = request.form['gender']
            data.person.birthdate = request.form['birthdate']
            data.person.city = request.form['city']
            data.person.state = request.form['state']
            data.person.country = request.form['country']

            # Update business data
            data.rank = request.form['rank']
            data.source = request.form['source']
            data.finalWorth = request.form['finalWorth']
            data.category = request.form['category']
            data.organization = request.form['organization']
            data.industries = request.form['industries']
            data.title = request.form['title']
```

```
            db.session.commit()
            print("Updated successfully")
            return redirect(url_for('modify'))

        return render_template('update.html', data=data)
    except Exception as e:
        db.session.rollback()
        return jsonify({"error": str(e)}), 500


@flask_app.route('/delete/<int:id>', methods=['POST'])
def delete(id):
    try:
        data = Business.query.get(id)
        if not data:
            return jsonify({'message': 'Data not found'}), 404

        db.session.delete(data)
        db.session.commit()
        print("Deleted successfully")
        return jsonify({'message': 'Data deleted successfully'}), 200
    except Exception as e:
        db.session.rollback()
        return jsonify({'error': str(e)}), 500


if __name__ == '__main__':
    flask_app.run(host='127.0.0.1', port=8005, debug=True)
```

4.1.2 BASE.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Billionaire Statistics 2023</title>
    <style>
        body {


            background-image: url(https://img.freepik.com/premium-photo/black-dollar-signs-black-studio-background_241146-2551.jpg); /* Add your image URL */
```

```
background-size: cover;
        color: #333;
        font-family: Arial, sans-serif;
        margin: 0;
        padding: 0;
        text-align: center;
        background-color: #f9f9f9;
    }
    .title {
        color: white;
    }

    /* Header */
    header {

        text: white

        text-align: center;

        background-color: #2c3e50;
        color: white;
        padding: 20px;
    }
    p {
    color: white;
    }

    /* Navigation */
    nav {
        display: flex;
        justify-content: center;
        background-color: #34495e;
        padding: 10px;
    }

    nav a {
        color: white;
        text-decoration: none;
        padding: 10px 20px;
        margin: 0 10px;
        border-radius: 5px;
        transition: background-color 0.3s ease;
    }

    nav a:hover {
        background-color: #575757;
```

```
}

    /* Container */
    .container {
       padding: 20px;
       text-align: center;
    }

    /* Billionaire Section */
    .billionaire {
       margin: 20px auto;
       text-align: center;
    }

    .billionaire img {
       width: 150px;
       height: 150px;
       border-radius: 50%;
       box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
    }
    button, a {
       display: inline-block;
       padding: 10px;
       border: none;
       border-radius: 5px;
       font-size: 20px;
       color: white;
       cursor: pointer;
       text-align: center;
       text-decoration: none;
       transition: background-color 0.3s ease;
    }

    /* Card */
    .card-container {
       display: flex;
       flex-wrap: wrap;
       justify-content: center;
       gap: 20px;
       margin: 20px 0;
    }

    .card {
       background-color: white;
       border: 1px solid #ddd;
       border-radius: 8px;
```

```
box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
        width: 300px;
        text-align: center;
        padding: 20px;
    }

    .card-title {
        font-size: 18px;
        font-weight: bold;
        margin-bottom: 10px;
    }

    .card-text {
        font-size: 16px;
        color: white;
        margin: 5px 0;
    }

    /* Footer */
    footer {
        background-color: #2c3e50;
        color: white;
        text-align: center;
        padding: 1px 0;
        position: unset;
        width: 100%;
        bottom: 0;
    }

    /* FAQ Section */
    .faq-section {
        max-width: 600px;
        margin: 20px auto;
        text-align: left;
    }

    .faq-button {
        background-color: #2c3e50;
        color: white;
        padding: 15px;
        border: none;
        width: 100%;
        text-align: left;
        font-size: 18px;
        cursor: pointer;
        margin: 5px 0;
```

```
border-radius: 5px;
        transition: background-color 0.3s ease;
    }

    .faq-button:hover {
        background-color: #34495e;
    }

    .faq-content {
        display: none;
        background-color: #ecf0f1;
        padding: 15px;
        border-radius: 5px;
        margin-top: 10px;
    }

    /* Login Form */
    .login-container {
        background: white;
        padding: 40px;
        margin: 20px auto;
        width: 300px;
        border-radius: 10px;
        box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
    }

    .form-group {
        margin-bottom: 20px;
    }

    label {
        display: block;
        font-weight: bold;
        margin-bottom: 5px;
    }

    input[type="text"],
    input[type="password"] {
        width: 100%;
        padding: 10px;
        border: 1px solid #ccc;
        border-radius: 5px;
    }

    .login-btn {
        width: 100%;
```

```
padding: 10px;
            background-color: #007bff;
            color: white;
            border: none;
            border-radius: 5px;
            font-size: 16px;
            cursor: pointer;
            margin-top: 10px;
        }

        .login-btn:hover {
            background-color: #0056b3;
        }

        .forgot-password {
            display: block;
            text-align: center;
            margin-top: 10px;
            color: #007bff;
            text-decoration: none;
        }

        .forgot-password:hover {
            text-decoration: underline;
        }
        h1 {
            color: white;
        }
    </style>
</head>
<body>
    <!-- Header -->
    <header>
        <h1 style="text-align: center; margin-top: 20px; color: white;">
            Billionaire Statistics 2023

    </h1>
        <p>Top Billionaires of the World</p>
    </header>

    <!-- Navigation -->
    <nav>
        <a href="/">Home</a>
        <a href="/help">Help</a>
```

```
<a href="/register">Add-Data</a>
        <a href="/modify">Edit-Data</a>
        <a href="/dashboard">Statistics</a>

    </nav>

    <!-- Main Content -->
    <div class="container">
        {% block content %}
        <!-- Dynamic content goes here -->
        {% endblock %}
    </div>

    <!-- Footer -->
    <footer>
        <p>&copy; 2023 Billionaire Statistics | All Rights Reserved</p>
    </footer>
</body>
</html>
```

4.1.3 add_data.html

```
{% extends 'base.html' %}

{% block title %}Add Data Page{% endblock %}

{% block content %}
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Add Data Form</title>
    <style>
        body {
            font-family: 'Arial', sans-serif;
            background-color: #f7f9fc;
            margin: 0;
            padding: 0;
        }

        header {
            text-align: center;
            <h1 style="text-align: center; margin-top: 20px; color: white;">

    </h1>
```

```
background-color: #343a40;
            color: #fff;
            padding: 20px;
            font-size: 1.5rem;
        }

        .form-container {
            display: flex;
            justify-content: center;
            align-items: center;
            min-height: 100vh;
            padding: 20px;
        }

        form {
            background-color: #ffffff;
            border-radius: 10px;
            box-shadow: 0px 4px 8px rgba(0, 0, 0, 0.2);
            padding: 30px;
            width: 100%;
            max-width: 800px;
        }

        h1 {
            text-align: center;
            color: #333;
            margin-bottom: 20px;
            font-size: 1.8rem;
        }

       table {
    width: 100%;
    border-collapse: separate; /* Use 'separate' to apply spacing */
    border-spacing: 20px; /* Add space between columns and rows */
}

th,
td {
    padding: 10px;
    text-align: left;
}

td {
    vertical-align: top;
}
```

```
form {
    background-color: #ffffff;
    border-radius: 10px;
    box-shadow: 0px 4px 8px rgba(0, 0, 0, 0.2);
    padding: 30px;
    width: 100%;
    max-width: 800px;
    margin: 0 auto;
}

input[type="text"],
input[type="date"] {
    width: calc(100% - 20px); /* Account for padding within table cells */
    padding: 10px;
    border: 1px solid #ddd;
    border-radius: 4px;
    font-size: 1rem;
}

        label {
            font-weight: 600;
            color: #555;
            display: block;
            margin-bottom: 8px;
        }

        input[type="text"],
        input[type="date"] {
            width: 100%;
            padding: 10px;
            border: 1px solid #ddd;
            border-radius: 4px;
            font-size: 1rem;
        }

        .gender-labels {
            display: flex;
            gap: 10px;
            align-items: center;
        }

        input[type="radio"] {
            margin-right: 8px;
        }

        input[type="submit"] {
```

```
background-color: #007bff;
            color: #fff;
            padding: 12px;
            border: none;
            border-radius: 5px;
            font-size: 1rem;
            cursor: pointer;
            transition: background-color 0.3s ease;
            width: 100%;
        }

        input[type="submit"]:hover {
            background-color: #0056b3;
        }

        .form-footer {
            text-align: center;
            margin-top: 20px;
            font-size: 0.9rem;
            color: #777;
        }
    </style>
</head>

<body>

    <div class="form-container">
        <form method="POST" action="/submit">
            <h1>Add New Data</h1>
            <table>
                <tr>
                    <td>
                        <label for="personName">Name:</label>
                        <input type="text" id="personName" name="personName" required>
                    </td>
                    <td>
                        <label for="age">Age:</label>
                        <input type="text" id="age" name="age" required>
                    </td>
                </tr>
                <tr>
                    <td colspan="2">
                        <label>Gender:</label>
                        <div class="gender-labels">
                            <input type="radio" id="male" name="gender" value="male" required>
                            <label for="male">Male</label>
```

```
<input type="radio" id="female" name="gender" value="female" required>
                    <label for="female">Female</label>
                    <input type="radio" id="other" name="gender" value="other" required>
                    <label for="other">Other</label>
                </div>
            </td>
        </tr>
        <tr>
            <td>
                <label for="birthdate">Date of Birth:</label>
                <input type="date" id="birthdate" name="birthdate" required>
            </td>
            <td>
                <label for="city">City:</label>
                <input type="text" id="city" name="city" required>
            </td>
        </tr>
        <tr>
            <td>
                <label for="state">State:</label>
                <input type="text" id="state" name="state" required>
            </td>
            <td>
                <label for="country">Country:</label>
                <input type="text" id="country" name="country" required>
            </td>
        </tr>
        <tr>
            <td>
                <label for="rank">Rank:</label>
                <input type="text" id="rank" name="rank" required>
            </td>
            <td>
                <label for="source">Source:</label>
                <input type="text" id="source" name="source" required>
            </td>
        </tr>
        <tr>
            <td>
                <label for="finalWorth">Final Worth:</label>
                <input type="text" id="finalWorth" name="finalWorth" required>
            </td>
            <td>
                <label for="category">Category:</label>
                <input type="text" id="category" name="category" required>
            </td>
```

```
</tr>
            <tr>
              <td>
                <label for="organization">Organization:</label>
                <input type="text" id="organization" name="organization" required>
              </td>
              <td>
                <label for="industries">Industries:</label>
                <input type="text" id="industries" name="industries" required>
              </td>
            </tr>
            <tr>
              <td colspan="2">
                <label for="title">Title:</label>
                <input type="text" id="title" name="title" required>
              </td>
            </tr>
            <tr>
              <td colspan="2">
                <input type="submit" value="Submit">
              </td>
            </tr>
          </table>
          <div class="form-footer">
            <small>By submitting, you agree to our terms and conditions.</small>
          </div>
        </form>
    </div>
</body>

</html>

{% endblock %}
```

4.1.4 aboutus.html

```
{% extends 'base.html' %}

{% block title %}about-us page{% endblock %}
{% block content %}
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>About Us</title>
    <style>
```

```
body {
        font-family: Arial, sans-serif;

        margin: 0;
        padding: 0;
        line-height: 1.6;
    }

    section {
        padding: 2rem;
        max-width: 800px;
        margin: 0 auto;
    }
    h1, h2{
        color: white;
    }
    h3{
    color: black;
    }
     p {
        color: white;
    }
    .team {
        display: flex;
        flex-wrap: wrap;
        gap: 3rem;
        margin-top: 3rem;
    }
    .team-member {
        flex: 1 1 calc(33.333% - 1rem);
        background: #f4f4f4;
        padding: 1rem;
        text-align: center;
        border-radius: 5px;
    }
    .team-member img {
        width: 200px;
        height: 200px;
        border-radius: 100%;
        margin-bottom: 0.5rem;
    }

  </style>
</head>
<body>
```

```
<section>
      <h1>Vision</h1>
      <h2>

To be the leading platform for insights, inspiration, and in-depth information on the world's most successful individuals, empowering users to learn, grow, and achieve extraordinary success.
</h2>
      <h1>Mission</h1>
<h2>

To provide accurate, engaging, and inspiring content about billionaires, their journeys, and their impact on the world, fostering a community of aspirational thinkers and future changemakers.
</h2>
   </section>
   <section>
      <h2>Contact US</h2>
      <div class="team">
         <div class="team-member">
            <img src="{{ url_for('static', filename='images/Rakesh.jpg')}}" alt="Team Leader">
            <h3>Name : Rakesh V</h3>
            <h3>Team Leader</h3>
            <h3>Mobile-no : <underline>+91 9353975261</underline> </h3>
          <h3>Email : ry352004@gmail.com</h3>
         </div>
         <div class="team-member">
            <img src="{{ url_for('static', filename='images/Raghu.jpg')}}" alt="Team Leader">
            <h3>Name : Raghu K</h3>
            <h3>Team Member</h3>
            <h3>Mobile-no : <underline>+91 8073090366</underline> </h3>
          <h3>Email : raghurk090@gmail.com</h3>
         </div>
         <div class="team-member">
            <img src="{{ url_for('static', filename='images/Prajwal.jpg')}}" alt="Team Leader">

            <h3>Name : Prajwal Koushik C </h3>
            <h3>Team Member</h3>
            <h3>Mobile-no : <underline>+91 8864531129</underline> </h3>
          <h3>Email : prajwalkoushik27@gmail.com</h3>
         </div>
         <div class="team-member">
            <img src="{{ url_for('static', filename='images/Mahalakshmi.jpg')}}" alt="Team
Leader">
            <h3>Name : Mahalakshmi PS</h3>
            <h3>Team Member</h3>
            <h3>Mobile-no : <underline>+91 9901654281</underline> </h3>
          <h3>Email : mahareddy.544@gmail.com</h3>
```

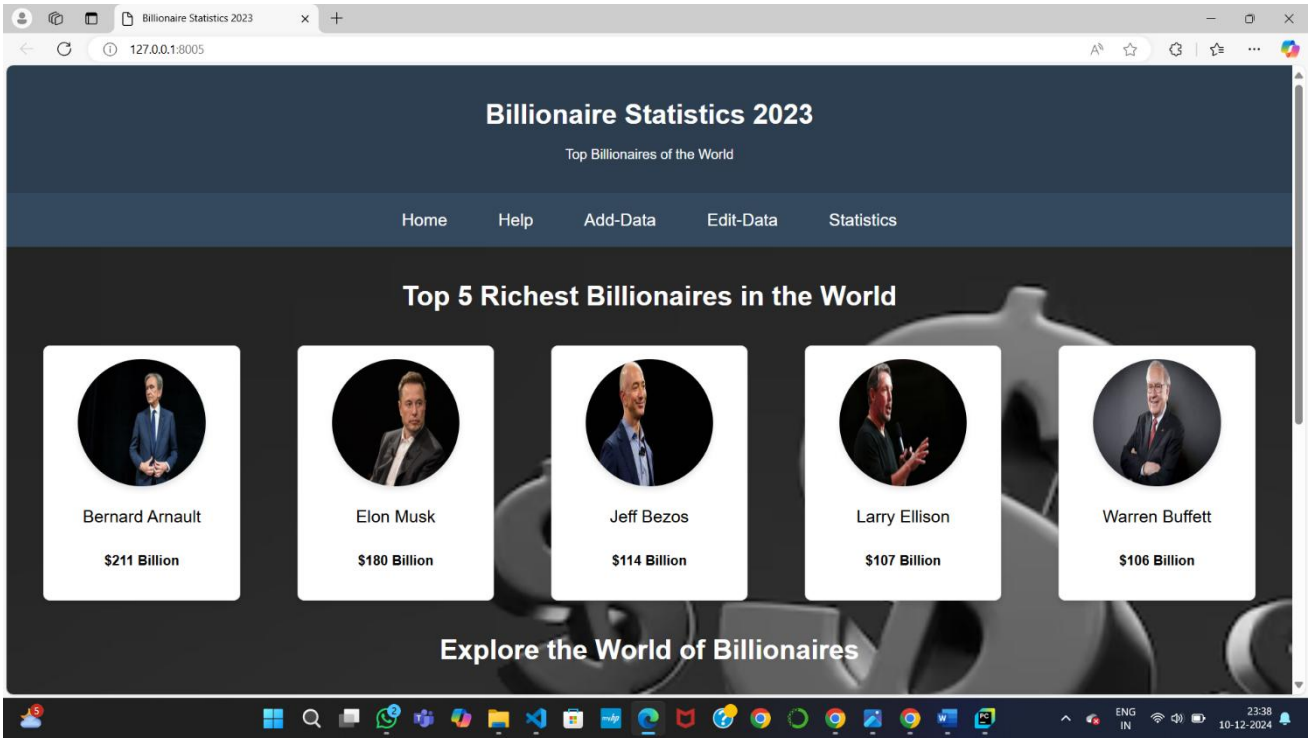
```
</div>
        </div>
    </section>

</body>
</html>
{% endblock %}
```

Chapter 5

RESULT AND DISCUSSION

Screenshots



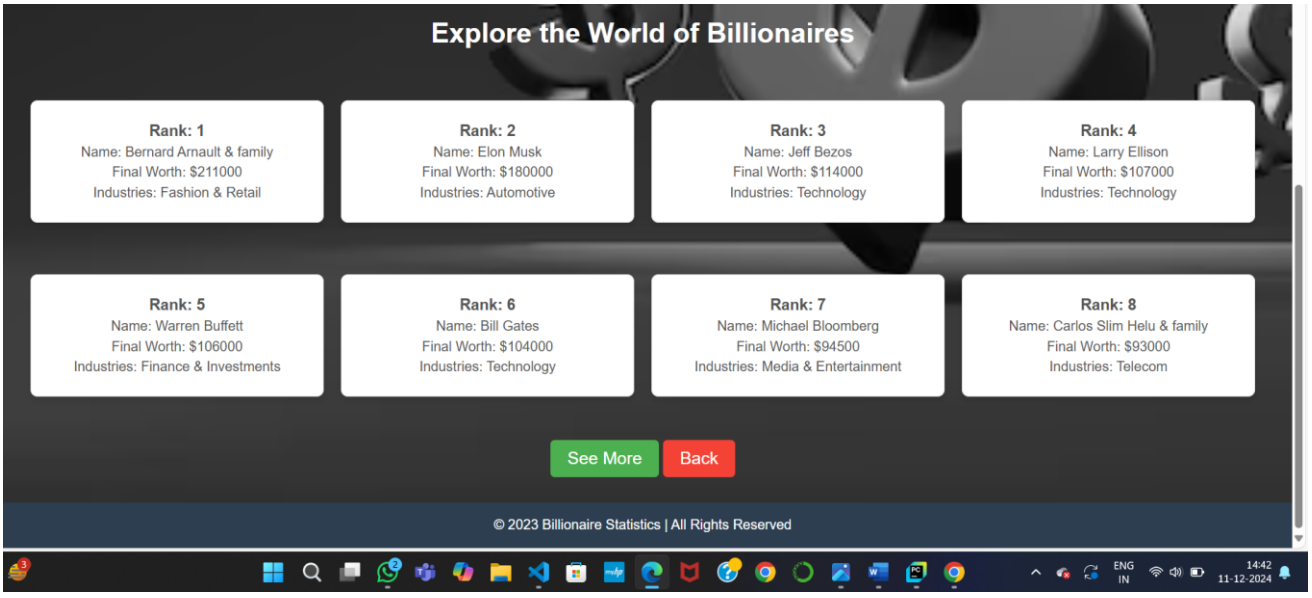


Fig 5.1 Home page

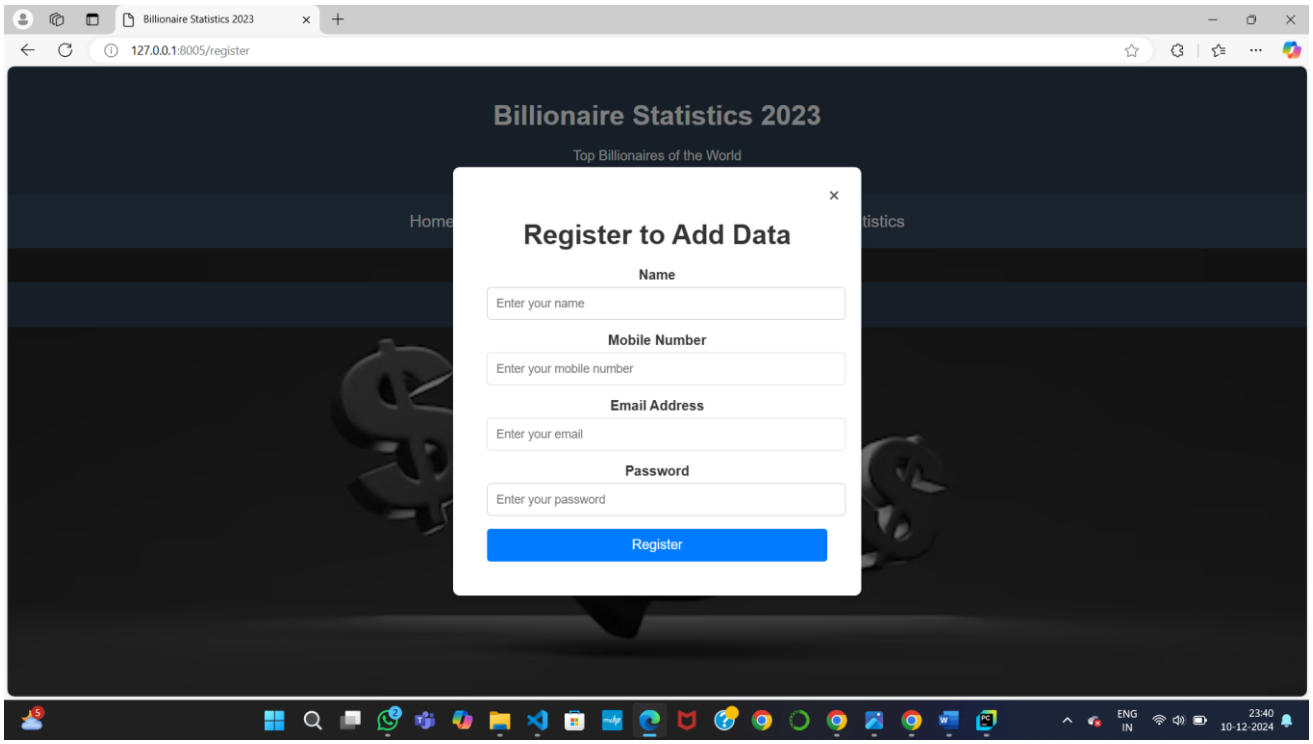


Fig 5.2 add data Page

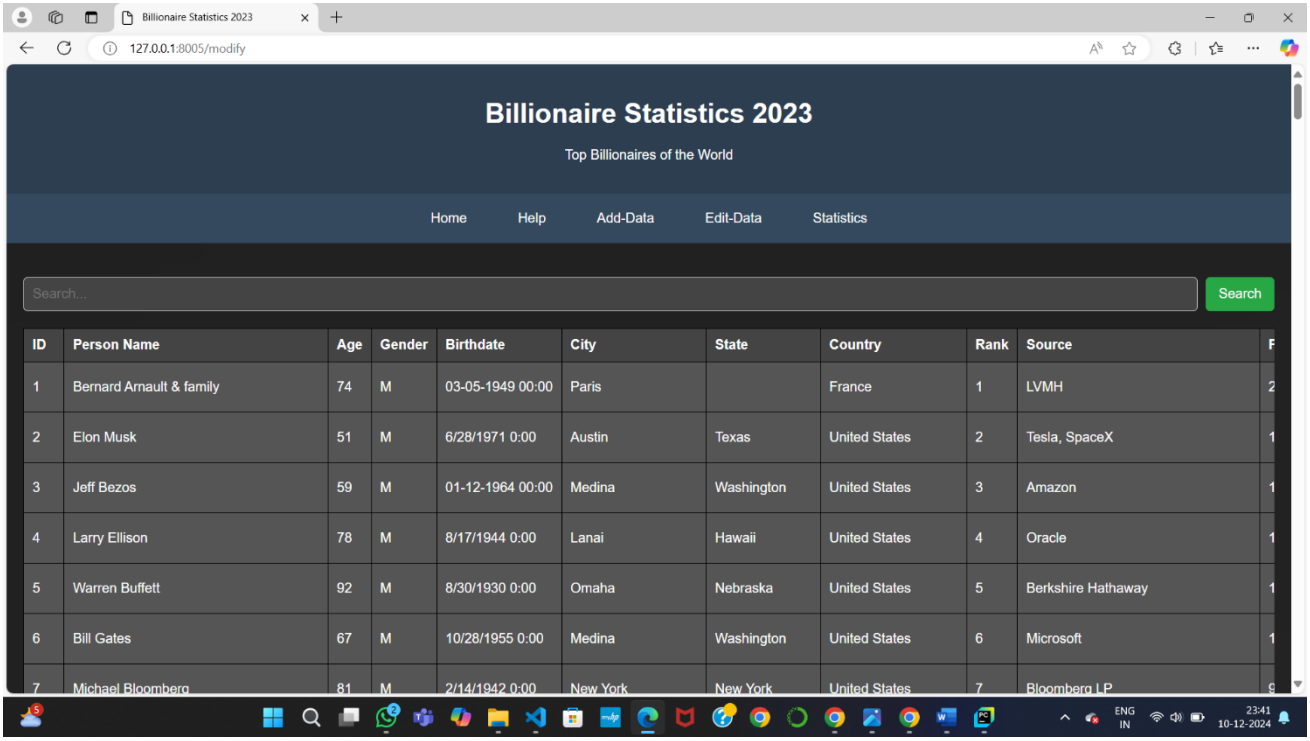


Fig 5.3 edit-data page

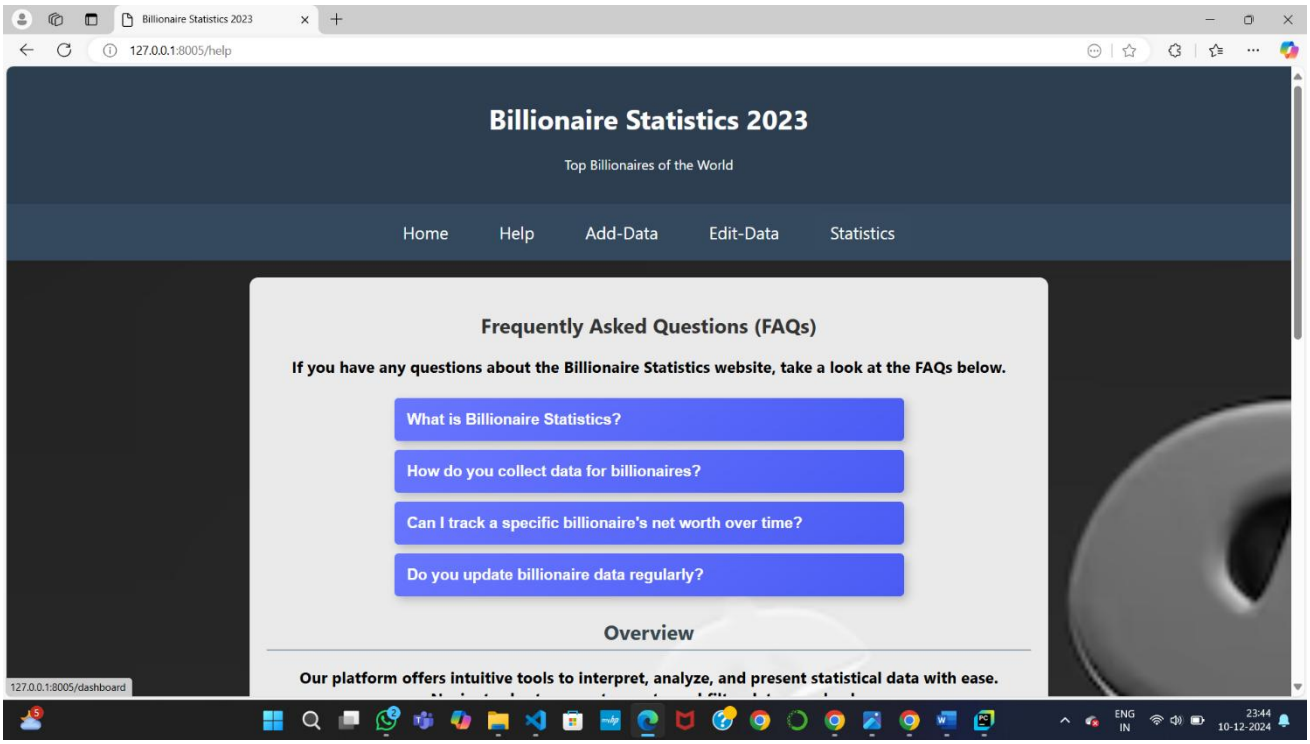


Fig 5.4 help page

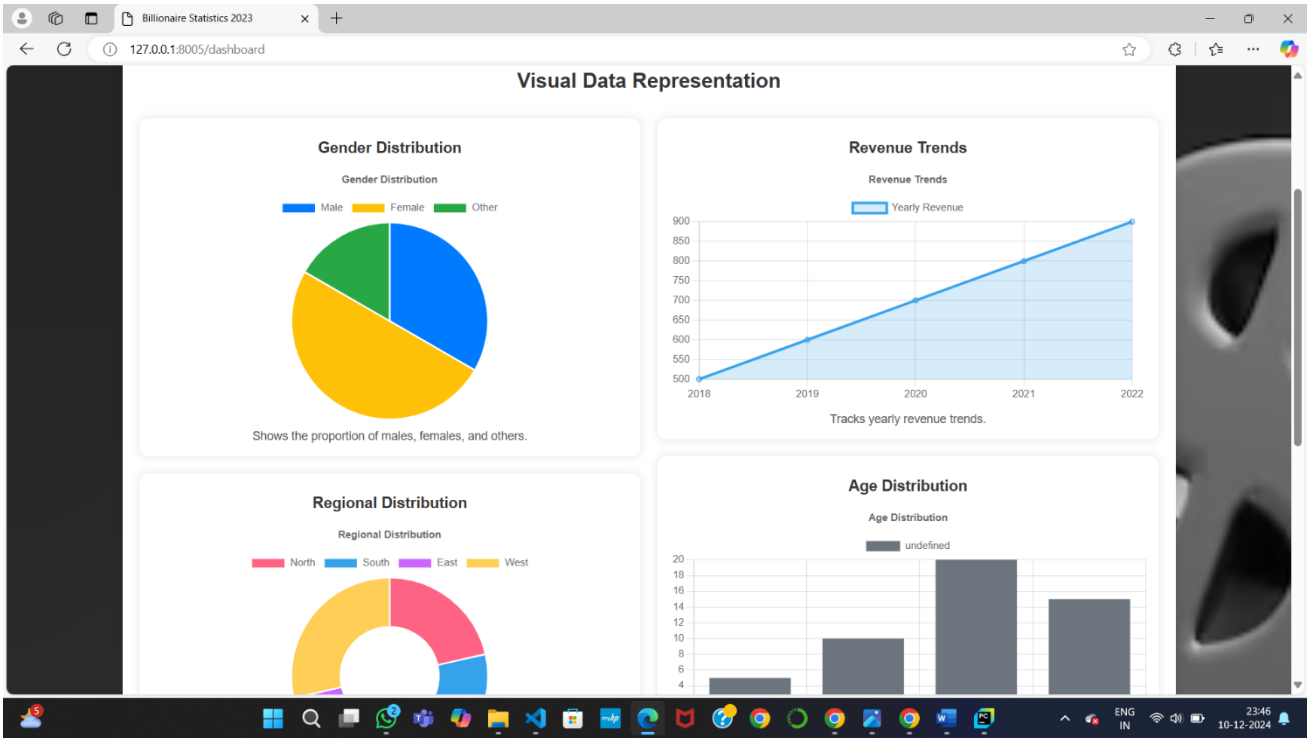


Fig 5.5 Dashboard

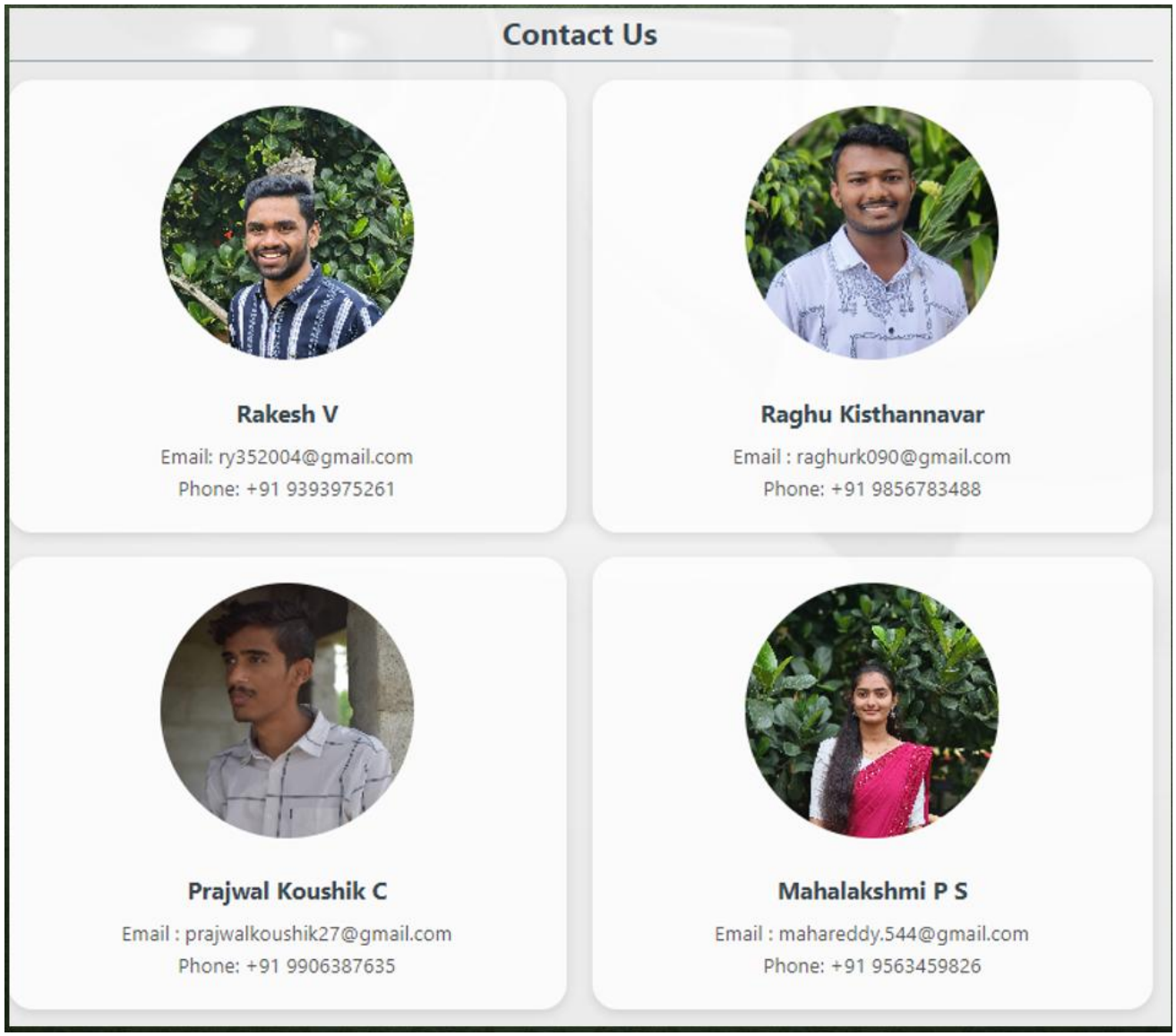


FIG 5.6 contact us

Chapter 6

CONCLUSION AND FUTURE ENHANCEMENTS

The **Billionaire Statistics 2023 Project** successfully analyzed and presented key insights into the global billionaire population using Python, SQL, and HTML technologies. The project undertook the following major steps and accomplishments:

1. **Data Collection and Cleaning**:

   Raw data about billionaires, including net worth, industries, geographic distribution, and other demographics, was collected from various reliable sources. The data was cleaned and structured using Python libraries like Pandas to ensure accuracy and consistency.

2. **Database Integration**:

   A structured SQL database was created to store the processed data efficiently. Queries were designed and executed to extract meaningful insights such as the average net worth, distribution by industries, and top-performing countries.

3. **Data Analysis**:

   Python was utilized for in-depth analysis, employing libraries such as NumPy, Pandas, and Matplotlib. Key metrics like billionaire growth rates, sector dominance, and geographic trends were calculated and visualized.

4. **Visualization**:

   Various charts, graphs, and heatmaps were generated to highlight trends in billionaire statistics, making the insights more comprehensible and visually appealing. Tools like Matplotlib and Seaborn were used extensively for this purpose.

5. **Web Interface Development**:

   An interactive and user-friendly web interface was built using HTML and CSS. This platform displayed the analyzed data and visualizations, offering a dynamic way to explore the findings.

6. **Key Findings**:

   The project identified trends such as the dominance of specific industries (like technology and

finance), regional disparities in billionaire wealth, and the impact of global economic conditions on billionaire growth.

7. **Reporting**:

A comprehensive final report was compiled in PDF and Word formats, summarizing the project's objectives, methodologies, findings, and conclusions.
This project effectively utilized modern programming and database tools to provide a detailed understanding of billionaire statistics for 2023. It offers a foundation for further exploration into wealth distribution, economic trends, and global financial disparities.

BIBLIOGRAPHY

WEBSITE LINK

1.https://github.com/Rakesh352004/Final-miniproject.git