Mini Project Report

on

# CORPOINDIA - ASSESSING CORPORATE GROWTH IN INDIA AND ITS CONTRIBUTION TO INDIAN ECONOMY

Submitted by

Abdulla Sameer K A (20421004)
Pravaal B Nath (20921038)
Prajwal M (20220076)
Sharoon C (20221101)

In partial fulfilment of the requirements for the award of degree of Bachelor of Technology in Computer Science and Engineering.

DIVISION OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF ENGINEERING
COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY

JULY 2024

DIVISION OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF ENGINEERING
COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY

# *CERTIFICATE*

Certified that this is a Mini Project Report titled

CORPOINDIA - ASSESSING CORPORATE GROWTH IN INDIA AND ITS CONTRIBUTION TO INDIAN ECONOMY

Submitted by

Abdulla Sameer K A (20421004)
Pravaal B Nath (20921038)
Prajwal M (20220076)
Sharoon C (20221101)

of VI Semester, Computer Science and Engineering in the year 2024 in partial fulfillment requirements for the award of degree of Bachelor of Technology in Computer Science and Engineering of Cochin University of Science and Technology.

| | | |
|---|---|---|
| Dr. Sudheep Elayidom M. | Ms. Preetha S | Ms. Preetha S |
| Head of Division | Project Coordinator | Project Guide |

# Acknowledgement

We have taken a lot of effort into this project. However, it would not have been possible without the kind support and help of many. We are using this opportunity to extend our sincere thanks to all the individuals concerned with this project.

First and foremost, praises and thanks to God, the Almighty, the source of all knowledge whose blessings are our guiding light in any venture we take up. We sincerely thank Dr. Sudheep Elayidom M., Head of the Department for his invaluable assistance in selecting our project topic and for his continuous guidance throughout the process. We are also highly indebted to our Mentor Ms. Preetha S for her guidance, patience and support that helped us through our work, without which we could not have implemented this project the way it works today. We would also like to thank our project coordinators Ms. Preetha S for her guidance in helping us choose this topic. We would also like to express our special gratitude to our friends and family for their kind cooperation and encouragement which helped us develop this project.

Abdulla Sameer K A (20421004)
Pravaal B Nath (20921038)
Prajwal M (20220076)
Sharoon C (20221101)

# Declaration

We Abdulla Sameer K A, Pravaal B Nath, Prajwal M and Sharoon C hereby declare that the project "CORPOINDIA - ASSESSING CORPORATE GROWTH IN INDIA AND ITS CONTRIBUTION TO INDIAN ECONOMY" is a bonafide work done by us during the year 2024 under the guidance of Ms. Preetha S, Assistant Professor at School of Engineering, CUSAT and that no part has formed the basis for the award of any degree, diploma, associate ship, fellowship or any other similar title or recognition in any other university.

# Abstract

CorpoIndia is a cutting-edge analytical platform designed to revolutionize the way professionals and analysts assess the growth and impact of startups and corporate entities in India. This comprehensive tool delves deep into the Indian entrepreneurial ecosystem, offering unparalleled insights into key parameters such as funding dynamics, revenue trajectories, and workforce expansion.

Through sophisticated data visualizations, including interactive graphs and dashboards, CorpoIndia delivers an exhaustive overview of India's economic environment, allowing users to explore detailed state-wise funding patterns, industry-specific growth trends, and investment distributions for a granular understanding of sectoral performance and regional disparities.

It can be a resource that empowers stakeholders with the knowledge needed for strategic decision-making and market intelligence. By offering a nuanced analysis of India's startup and corporate growth.

CorpoIndia positions itself as an indispensable tool for anyone invested in the future of the Indian economy. This platform redefines economic analysis and business strategy in India, making it essential for understanding and leveraging the dynamic growth of the country's corporate sector.

# Contents

# List of Figures

# Chapter 1

# Introduction

This project aims to develop a comprehensive dashboard that delineates various economic metrics and standards defining startups and companies, optimized exclusively for desktop use. It serves as a robust platform for business analysts, potential investors, and existing stakeholders to assess the growth trajectories and prospects of Indian startups and unicorns. Addressing the gap in accessible, reliable reports, this tool mitigates the need for costly professional analysis, enabling informed decision-making.

The detailed report provides a centralized resource for individuals seeking insights into the Indian startup landscape, facilitating connections between investors and prominent startups within their preferred industries. Additionally, it offers a comprehensive evaluation framework for existing investors to assess and grade their current investments. This initiative enhances transparency and informed investment strategies within the Indian economic ecosystem.

This project analyzes the impact of startups and funding on the Indian economy, examining how increased investment fuels innovation, job creation, and economic growth. By exploring the rise of new ventures and their contributions to various sectors, the study highlights the transformative role of startups in driving technological advancements and addressing social issues. The research also delves into the evolving landscape of investor confidence and its effect on the sustainability and scalability of Indian startups.

# Chapter 2

# Literature Review

Since the advent of the 21st century India has been a hub for economic and technological growth. Many entrepreneurs with their brilliant ideas and stellar visions wish to leave a mark on this country. India has one the largest start up densities in the world and many corporate companies headquartered in India are leading stalwarts in the international business stage. In this review we look at the existing resources available to people as of now as well as research papers that have studied this aspect of the Indian economy.

1. Maradi, Mallikarjun. (2023). Growth of Indian start-up: A critical Analysis. 17. 180-186. India has become the world's second-largest startup ecosystem after the US, with Bengaluru, Mumbai, and Delhi ranking among the top 40 global startup hubs. In 2021, Indian startups raised over $23 billion across 1000+ deals, adding 33 unicorns, and 2022 has added 13 more. Investor confidence in Indian startups is growing, driven by new ventures, global interest, and advancements in infrastructure and policies. This study explores the growth trends, determinants, and challenges of Indian startups using secondary data and statistical analysis.

2. Kukreja, Megha & Makhija, Priya. (2023). Startups and their contribution towards the growth of Indian Economy, International Scientific Journal of Engineering and Management. 02. 10.55041/ISJEM00407: With an estimated 26,000 startups, India is the third-largest startup ecosystem globally, attracting over $36 billion in the last three years and producing 26 unicorns. This paper analyzes the impact of startups on India's economic growth, highlighting The Hammerlock Company, which offers innovative in-

frared heat-resistant glassware. Due to massive funding, technological advancements, and a growing domestic market, India's startup ecosystem has flourished, with startups increasing from 3,100 in 2014 to over 11,500 in 2020. This revolution is transforming Indian markets and fostering a dynamic entrepreneurial mindset amidst global economic challenges.

3. Risbud, Mrudula & Waghmare, Rahul. (2023). Sustainability Through Innovation: The Case of Indian Startup Thaely. 10.4018/978-1-6684-6123-5.ch011. Indian startups span various sectors, including traditional, technology-based, and social enterprises. Many focus on social and environmental issues through innovation. This chapter highlights Thaely, founded by Ashay Bhave, which gained international recognition for making sneakers from recycled plastic bags. Using secondary data, it explores Thaely's problem identification, innovative manufacturing process, and its social and sustainable impact.

4. Economic platforms run by News Agency: Various online economic platforms have been developed to by different news agencies to provide on demand economic news and information. For example, platforms like ETNow and CNBC TV18 groups have been widely used for broadcasting and providing economic information to people. However, these platforms tend to focus on the big players in the market as well as those companies that may be in the spotlight due to the current news at that moment. They do not provide easily visualised and personal reports to people based on their individual needs.

5. Privately run Economic Platforms: Some universities and analysis companies have developed their own economic platforms for sharing news about the corporate scene in the country. However, their functionalities are often locked behind paywalls and hefty subscription services. The data provided may not be accurate or trustworthy.

Based on the existing literature, it is evident that a comprehensive and unbiased data report detailing the economic metrics of all startups and unicorns in India is of utmost need. A common report can help establish a standard for various studies and research as well as become a benchmark for different economic development programs and platforms that may be launched by the government and other private companies. A standard, free of cost in depth report can help to bring transparency and valuable insight to different stakeholders who wish to learn more about the industrial health of our country, start their own businesses or even invest in existing, highly capable startups and LLCs.

In conclusion, the development of "CorpoIndia" will address the need for a country standard platform that provides safe, reliable information to the people of this country inviting collaboration, engagement, and resource sharing within the community.By incorporating the key features identified in existing literature as well as dealing with their existing problems, this platform has the potential to improve people's understanding of our economic scene and increase interaction between the community and the industry.

# Chapter 3

# System Analysis

## 3.1 Existing System

There are various online data dashboards that provide data about companies, their funding, revenue and employee count. They provide various services to access this data from API access to integrate into software solutions to personalised dashboards for keeping track of selected companies and industries. But there is a distinct lack of accuracy in this data. Each website has its own source which may or may not be a verified source of trustworthy and accurate information. To boot, the more reliable websites hide their data behind paywalls and expensive subscription models.
Most websites only cover the most popular and 'happening' startups like Byju's or PhysicsWallah. Other less appealing startups tend to get overshadowed. Moreover, there is no single platform where one can obtain information related to leading startups and unicorns as well as geographical data on startup funding and other metrics across Indian states and cities. Our report aims to provide a one stop solution for all startup and corporate analysis needs for the people of India and the general populace. Some of the existing economic data platforms CrunchBase, IndBiz, Economic Times Dashboard etc.

## 3.2 Proposed System

With our website, we use reliable sources which contain credible information about the funding, revenue and important metrics that help evaluate companies and startups. This data is thoroughly verified and then trans-

formed so as to be easily displayed to the user/client using visualisation and easy ranking systems. This way, the public doesn't need to have an in depth understanding of the internal, cumbersome and quite extensive company financial details to understand and know about the companies under study.
For our website, we undertake rigorous data analysis practices like web scrapping, pre-processing, data transformation and visualisation to ensure that the data presented to the user is accurate, easy to understand as well as being a light software solution that can run on household systems like PCs, laptops and even mobiles. The processes undertaken by our team to prepare this dashboard are listed below.

## WEB SCRAPPING

The first step to building a reliable dashboard of data is to find an accurate and trustworthy source. After searching across lots of databases and websites, we selected Crunchbase to scrap our data from. They provide a reliable source of data. We use the Instant Data Scraper Chrome extension to quickly scrap data from this website and save it to our local system as csv files.

## DATA PRE-PROCESSING

Web scraping tends to scrap all the information available on the website. Not all the data scrapped is necessary for the report as well as some may not be in a usable form yet. Our team undertakes three levels of data preprocessing - Bronze level, Silver level and Gold level to ensure the data used is accurate and in a usable form. We make use of the Spark engine as well as Pandas library to prepare the data.

## DATA TRANSFORMATION

Once the data is prepared for processing, we process the data, transforming the metrics into user understandable form. This may be in the form of indices or ranks, graphs and charts etc. The aim is to convert the scrapped data into usable data for our dashboard. This data is used to create insight into each company listed in our dashboard.

## CLOUD STORAGE

In order to easily access the data for our dashboard, we leverage cloud computing technology in the form of Cloud Storage to store the data. This way it is easily accessible from our dashboard no matter where the webpage may be hosted or where the python scripts are stored. Microsoft Azure, AWS Services and IBM Cloud were taken under consideration with AWS S3 bucket being chosen as the cloud storage solution due to its ease of connection with Databricks platform.

## DATA HOSTING

The data from the csv files stored in AWS S3 bucket is imported into Databricks Analytic Platform for ease of access. Databricks allows us to easily link the data stored in these csv files with our data visualisation software like PowerBI and Tableau.

## DATA VISUALIZATION

Data is best understood when depicted in a pictoral form. Using Data visualisation tools like PowerBI or Tableau, data imported from Databricks can be depicted as charts and graphs. Tableau is the our tool of choice due to its free-to-use scheme which keeps in line with our goal to provide quality data analysis with minimal cost.

## INTERACTIVE DASHBOARD

We have developed a website using React.js to provide an interactive environment where the user can freely interact with the data presented to them. This ensures that the dashboard can be accessed easily on any system without any additional specialised software.

# Chapter 4

# System Study

## 4.1 Software Requirements Specification

### Purpose

The purpose of our data analysis project, targeting the general as well as the industrial community, can be summarized as follows:

1. Analysis of Indian Economy and Startups: The project focuses on creating a comprehensive reporting system for investors, stakeholders, and business analysts. It integrates data from various sources, including financial statements, market trends, and investment portfolios, to generate detailed reports. These reports offer insights into key performance indicators, investment tracking, and other critical metrics.

2. Enhance Engagement between Startups and community: The project seeks to increase the engagement by easily providing data about up and coming startups in various industries to the people. This way, people come to know about them and can even help connect potential investors to these companies. The dashboard can also help with connecting possible collaborators for innovative projects that in the long run help improve the economic scene of our country.

3. Facilitate Information Sharing: The project aims to enable seamless sharing of information and resources within the community. By offering a news

feed, geographically linked economical data, we ensure that there is a use case for every user.

4. Support Collaboration and Networking: The project highlights the specialities and positives of companies, startups and unicorns in our country. Through this dashboard, people can come to know about companies in their industry of interest and provide ways to facilitate communication between them.

5. Empower Career Development: The project by providing valuable information about startups like funding, the industry with most funding opportunities, can help a budding entrepreneur decide which industry to base his/her startup. This also increases employment across the country which improves our GDP.

6. Analyzing the funding and growth of Indian startups: This project provides valuable insights into the dynamic landscape of the Indian economy. This study highlights the sectors experiencing rapid growth and the patterns of investment that drive innovation and job creation. By understanding these trends, we can better appreciate how startups contribute to economic expansion and technological advancement. Moreover, showcasing the potential and success stories within the Indian startup ecosystem can attract foreign investment and multinational companies, further fueling growth and integration into the global market. This analysis serves as a crucial tool for policymakers, investors, and entrepreneurs to make informed decisions that support sustainable economic development.

## Product Scope

1. Detailed economic statistics about India: - A webpage providing economic statistics about startups in India offers comprehensive data on various aspects of the startup ecosystem. It includes information on funding trends

and sector-wise distribution of startups.. The page also features analysis on the growth rates of startups and investment patterns.

2. Comprehensive analysis of economic metrics of states in India: - A website detailing economic metrics for a state provides in-depth statistics on the state's economic performance. The site often includes visualizations like charts and graphs to illustrate trends over time.

3. Company metrics in a nutshell: - Providing a brief description along with economic statistics, this site offers an overview of its background, mission, and services. It includes financial metrics such as funding amounts, revenue figures, and growth rates.

4. News Feed: - A business news feed webpage delivers up-to-the-minute updates on the latest developments in the business world. It features articles on market trends, corporate earnings, mergers and acquisitions, and economic policies.

## Project overview

The project contains following functions:-

## Functional Requirements

Home page: A brief description about the project and its functionalities.

India Statistics page: Detailed, comprehensive report on the economic factors affecting the corporate landscape of the country.

State Statistics page: Charts and graphs providing an easy to understand yet enlightening report on the health and status of startups in each state, funding ease and government support.

News Feed: Leverages NewsFeed API to deliver the latest news in the economic and industrial world of India.

Company Statistics page: Provides a comprehensive overview of a company's background, mission, services and its economic health with metrics like revenue and funding.

## Data Classes and Characteristics

1.Graphs and charts to visualise metrics like revenue, funding etc.
2. View a concise description about individual companies and startups including trivia like founder, founding year and other titbits.
3. State specific data for more in depth geographical analysis of economic growth in the country.
4. A ranking system based on revenue, funding, time taken to receive funding as well as workforce size to easily compare companies.

## Non Functional Requirements

### 4.2 Hardware and Software Requirements

## Hardware Requirements

It does not require any particular hardware. The application works on any desktop or mobile device which has an active internet connection.

## Software Requirements

Front-end development : HTML, CSS, Styled Components, React.js

Back-end development : Backend- Data visualisation using Tableau and data pre-processing using Spark. Storage of tables with AWS S3 bucket cloud storage solution and analysis of data using Databricks. Scraping of data from web using Instant Data Scraper.

### 4.3 Platforms and Tools

This a data analysis and visualisation website which is responsive to desktops. The development of this project was done using Reactjs, Databricks, AWS S3 Bucket, Apache Spark, Pandas, Beautiful Soup and Instant Data Scraper.

## Platform

System: The website runs on PC. It is responsive.

Database: Relational DBMS i.e. csv files hosted on AWS S3 bucket and imported into Tableau using Databricks.

## Tools

Azure Databricks is utilized to preprocess the dataset, ensuring data quality and consistency through various cleaning and transformation steps. This platform enables efficient handling of large datasets with its robust computational capabilities. An AWS S3 bucket is employed as the cloud storage solution for this project. This service provides scalable and secure storage for all project data, ensuring that data is easily accessible and retrievable as needed. Tableau is employed to visualize the data from Databricks, transforming it into informative graphs and charts. This integration allows for dynamic and interactive data exploration, enabling users to uncover insights and trends efficiently.

React.js : The React.js framework is an open-source JavaScript framework and library developed by Facebook. It's used for building interactive user interfaces and web applications quickly and efficiently.

Apache Spark: Apache Spark is an open-source unified analytics engine for large-scale data processing. Spark provides an interface for programming clusters with implicit data parallelism and fault tolerance.

AWS S3 Bucket: Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance. You can use Amazon S3 to store and retrieve any amount of data at any time, from anywhere.

Tableau: Tableau's powerful visualization capabilities help in presenting complex datasets in a clear and accessible manner, facilitating better decision-making and communication of findings. By leveraging Tableau, we can create a wide range of visual representations that cater to various analytical needs and audiences. We use both Tableau desktop and public.

Papa Parse: Papa Parse is a powerful, in-browser JavaScript library used to parse CSV (Comma-Separated Values) files. It simplifies the process of reading, parsing, and manipulating CSV data by providing a straightforward API. Papa Parse supports large files, multi-threading, and various configurations for handling different data formats and structures.

GPT and News APIs: The GPT API, developed by OpenAI, allows developers to integrate the powerful language model capabilities of GPT into their applications. This enables the creation of advanced natural language processing tasks, such as text generation, summarization, and translation, with just a few lines of code. News API is a service that provides access to real-time news articles from various sources across the web. It enables developers to integrate up-to-date news content into their applications, offering features like keyword searches, category filtering, and source selection.

# Chapter 5

# System Design

## 5.1 Introduction

Designing requires a careful planning and thinking on the part of the system designer. Designing a system means to plan how the various parts of it are going to achieve the desired goal. After the software requirements have been analysed and specified, design is the first of the three technical activities. Designing, coding and testing are required to build and verify the application.

## 5.2 Data Flow Diagrams

Data Flow Diagram is a pictorial way of showing the flow of data in to/within the system, around the system and out of the system. It is a graphical representation of flow of data within a system. Unlike flowcharts, DFDs do not give detailed descriptions of modules but graphically describe data and how the data interacts with the system. The DFD enables us to visualize how the system operates, its final output and the implementation of the system as a whole including modification if any. The purpose of data flow diagram is to provide a semantic bridge between users and system developers.

## 5.3 Main DFD

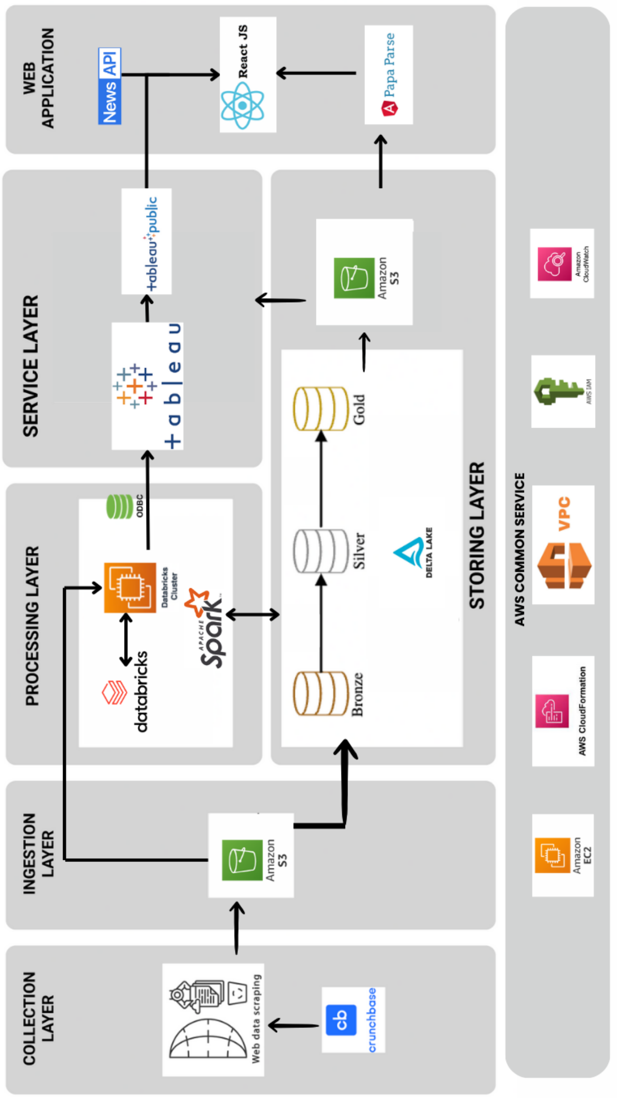


Figure 5.1: Project Architecture DFD

## 5.4 Use case Diagrams

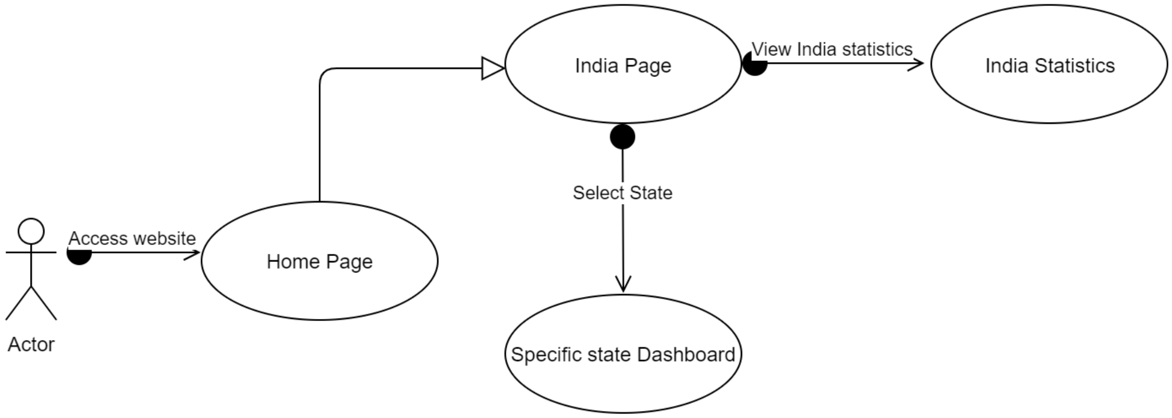


Figure 5.2: Use case diagram for accessing India and State statistics

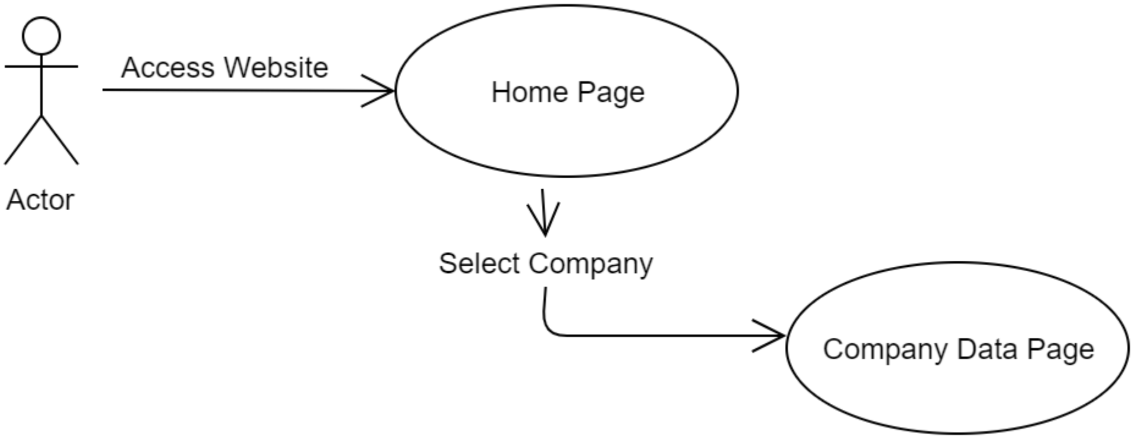


Figure 5.3: Use case diagram for accessing Company Statistics

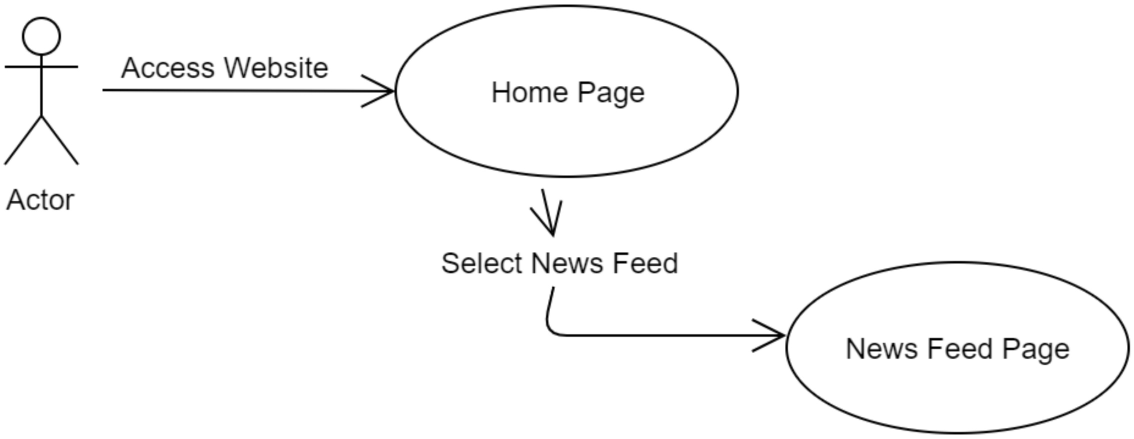


Figure 5.4: Use case diagram for accessing News Feed

## 5.5 Database Design

The data to be displayed on the dashboard is stored in csv files. These csv files are stored in AWS S3 Bucket cloud storage. They are imported into Azure Databricks from where they are accessed by Tableau for visualisation.

## 5.6 Modular Design

User Side:

The Company view module displays statistics, data and graphs about individual companies and startups. News Feed module displays personalized news feed about startups. India and state statistics module enables users to view individual state statistics via the India page as well as general stats about the economic scene in the country.

## 5.7 Input Output Design

- India Page
  Input : None
  Output : The user can view general economic statistics and metrics about the country
- State button
  Input : Select the state.
  Output : State specific statistics and graph are displayed.
- News
  Input : User selects the news button.
  Output : News feed with popular economic news is displayed.

- Company Page
  Input : User selects company name.
  Output : The user can see details and statistics pertaining to the specific company selected.
- About Us
  Input : User clicks About us link.
  Output : Details about the project developing team is displayed.
- Ranking Companies
  Input : None
  Output : The companies are ranked according to our in house formula.

# Chapter 6

# System Implementation

This website provides an easy to understand and use dashboard containing popular economic metrics and statistics about different startups in our country as well as geographically linked economics data about different states in the country. The different layers used to prepare our data for visualisation and publishing in the dashboard are presented below.

## 6.1 Collection Layer

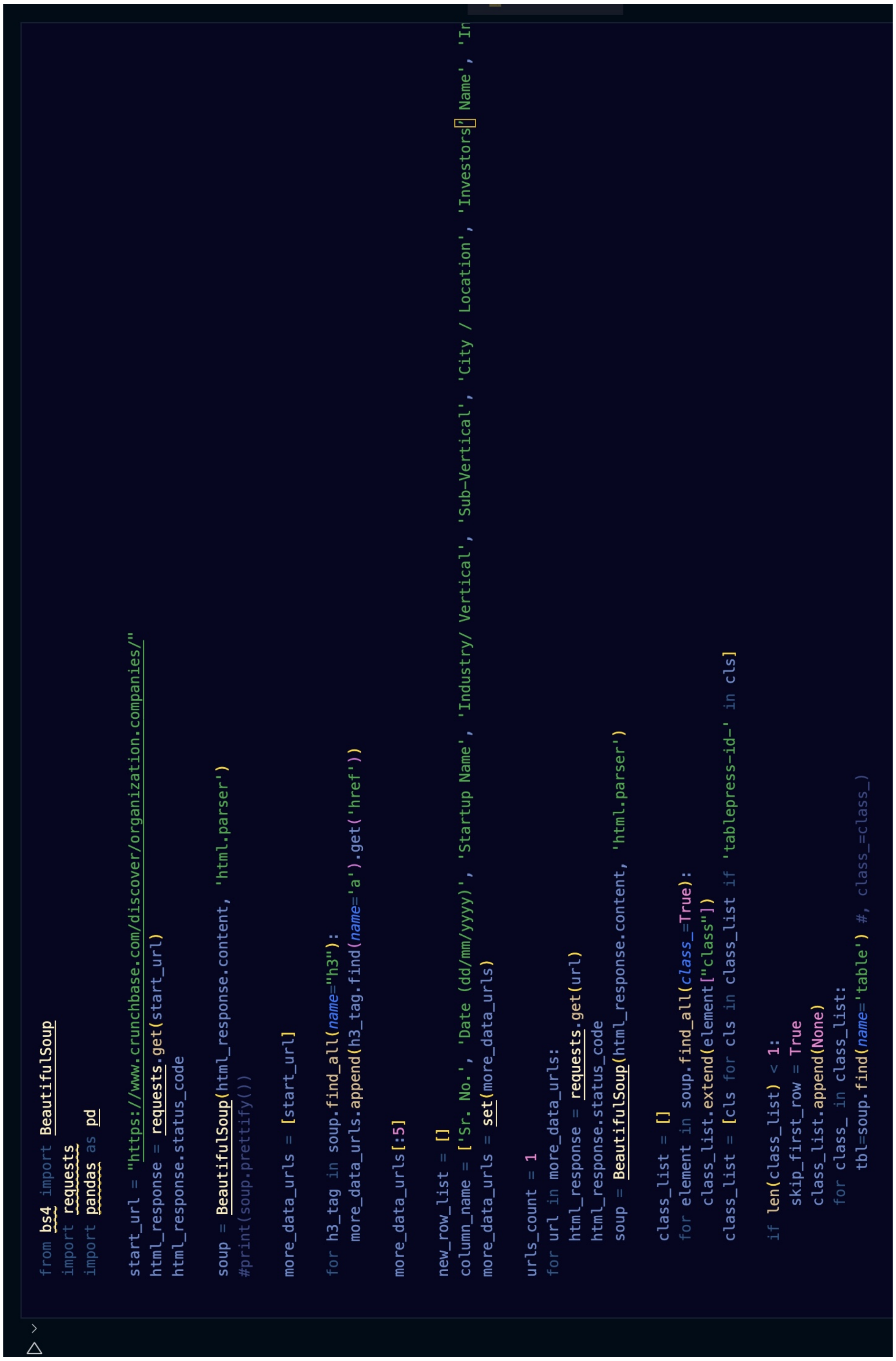
The first step for any data analysis project is to obtain the data that is to be analysed. The quality and accuracy of the data is vital to the validity of this project. We have used the data from trusted data publishing websites like crunchbase. There are multiple tools available on the market to scrap data from a website. Some tools are Beautiful Soup and Instant Data Scraper. We have used both these tools to scrap data to ensure that we get accurate and reliable data for our analysis.

We have scrapped tables containing company details like name, description, revenue, total funding amount and other metrics from crunchbase and other websites. We obtain a total of 7 csv files containing around 1000 companies each.

Similarly we scrap the funding data of companies which include data like funding type, amount etc. into separate csv files. With this, we obtain 10 csv files of 1000 entries each detailing the funding amount, date of funding and round of funding for over 6000 companies. These files are ready for preprocessing since scraping tends to collect unwanted data in our tables too. These fields must be eliminated to ensure the right data is used in our report.

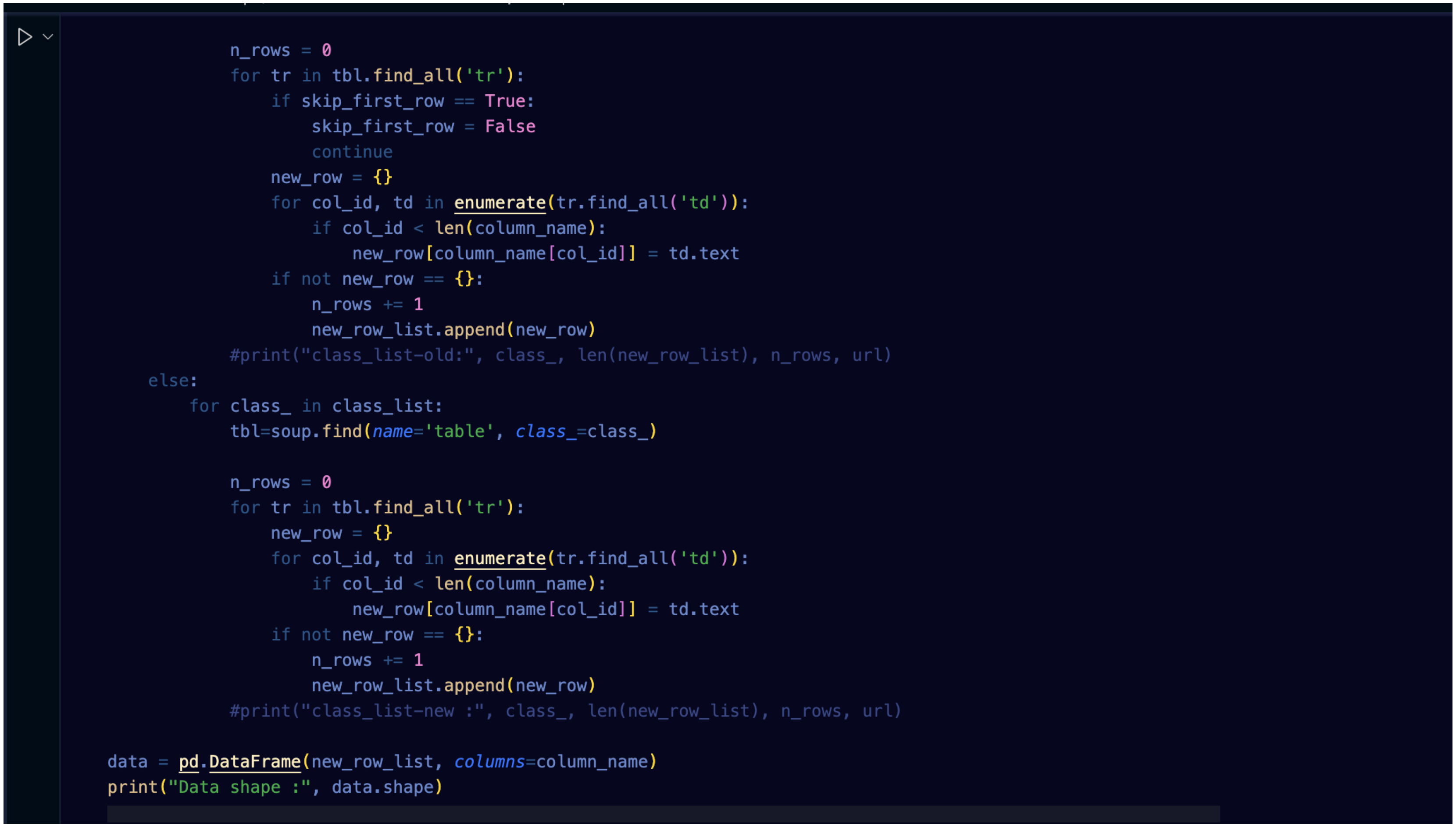
## Beautiful Soup

Data scraping using Beautiful Soup involves extracting information from web pages. Beautiful Soup is a Python library designed to parse HTML and XML documents easily, allowing for the navigation and searching of the parse tree. Beautiful Soup is favored for its ease of use, flexibility in handling different encodings, and powerful navigation methods, making it a popular choice for web scraping tasks.



```
from bs4 import BeautifulSoup
import requests
import pandas as pd

start_url = "https://www.crunchbase.com/discover/organization.companies/"
html_response = requests.get(start_url)
html_response.status_code

soup = BeautifulSoup(html_response.content, 'html.parser')
#print(soup.prettify())

more_data_urls = [start_url]

for h3_tag in soup.find_all(name="h3"):
    more_data_urls.append(h3_tag.find(name='a').get('href'))

more_data_urls[:5]

new_row_list = []
column_name = ['Sr. No.', 'Date (dd/mm/yyyy)', 'Startup Name', 'Industry/ Vertical', 'Sub-Vertical', 'City / Location', 'Investors’ Name', 'In
more_data_urls = set(more_data_urls)

urls_count = 1
for url in more_data_urls:
    html_response = requests.get(url)
    html_response.status_code
    soup = BeautifulSoup(html_response.content, 'html.parser')

    class_list = []
    for element in soup.find_all(class_=True):
        class_list.extend(element["class"])
    class_list = [cls for cls in class_list if 'tablepress-id-' in cls]

    if len(class_list) < 1:
        skip_first_row = True
        class_list.append(None)
        for class_ in class_list:
            tbl=soup.find(name='table') #, class_=class_)
```

Figure 6.1: Web Scraping with Beautiful Soup
Beautiful soup is used to scrap data from the internet and save it in easily interpretable filetypes like csv.

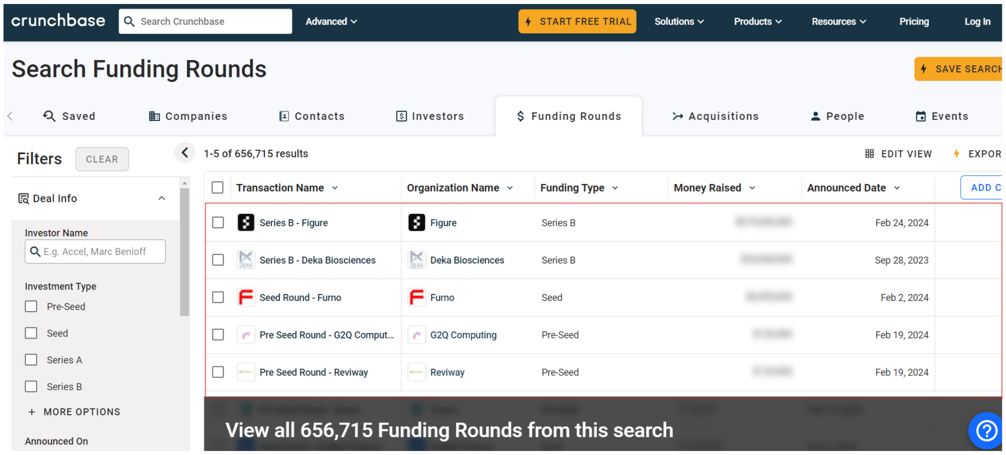


```
            n_rows = 0
            for tr in tbl.find_all('tr'):
                if skip_first_row == True:
                    skip_first_row = False
                    continue
                new_row = {}
                for col_id, td in enumerate(tr.find_all('td')):
                    if col_id < len(column_name):
                        new_row[column_name[col_id]] = td.text
                if not new_row == {}:
                    n_rows += 1
                    new_row_list.append(new_row)
            #print("class_list-old:", class_, len(new_row_list), n_rows, url)
    else:
        for class_ in class_list:
            tbl=soup.find(name='table', class_=class_)

            n_rows = 0
            for tr in tbl.find_all('tr'):
                new_row = {}
                for col_id, td in enumerate(tr.find_all('td')):
                    if col_id < len(column_name):
                        new_row[column_name[col_id]] = td.text
                if not new_row == {}:
                    n_rows += 1
                    new_row_list.append(new_row)
            #print("class_list-new :", class_, len(new_row_list), n_rows, url)

data = pd.DataFrame(new_row_list, columns=column_name)
print("Data shape :", data.shape)
```

Figure 6.2: Web Scraping with Beautiful Soup contd.
Beautiful soup is used to scrap data from the internet and save it in easily interpretable filetypes like csv.

# Instant Data Scraper

The Instant Data Scraper Chrome extension is a tool designed for quickly extracting data from web pages without the need for coding. It automatically detects patterns in the data presented on a webpage, such as tables and lists, and allows users to export this data into formats like CSV or Excel.
The Instant Data Scraper Chrome extension is ideal for users who need to quickly gather data from websites without the complexity of writing code or using more advanced scraping tools.



1-5 of 656,715 results

| Transaction Name | Organization Name | Funding Type | Money Raised | Announced Date |
|---|---|---|---|---|
| Series B - Figure | Figure | Series B | [illegible] | Feb 24, 2024 |
| Series B - Deka Biosciences | Deka Biosciences | Series B | [illegible] | Sep 28, 2023 |
| Seed Round - Furno | Furno | Seed | [illegible] | Feb 2, 2024 |
| Pre Seed Round - G2Q Comput... | G2Q Computing | Pre-Seed | [illegible] | Feb 19, 2024 |
| Pre Seed Round - Reviway | Reviway | Pre-Seed | [illegible] | Feb 19, 2024 |

Figure 6.3: Web Scraping with Instant Data Scraper
The data is scraped from crunchbase using Instant Data Scraper chrome extension. The data is stored as individual csv files. The data is locked behind a paywall. We used the free trial of one week to scrap the required tables.

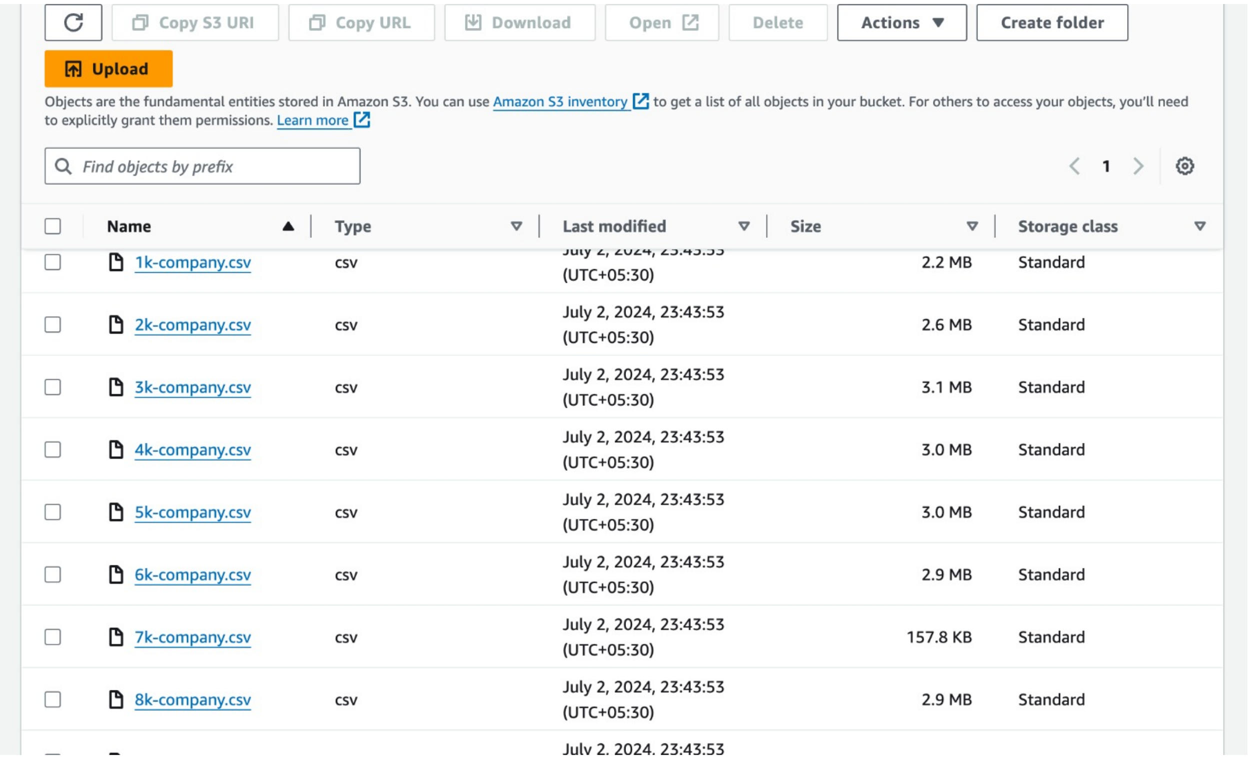
## 6.2 Ingestion Layer or Cloud Storage

Cloud storage is a service that allows users to save data on remote servers accessed via the internet. It offers scalable storage solutions, enabling individuals and businesses to store large amounts of data without investing in physical hardware. Cloud storage enhances data accessibility, as users can retrieve and manage their files from any device with an internet connection. It also provides data protection through encryption and redundancy, ensuring that data is secure and backed up. Additionally, cloud storage supports collaboration by allowing multiple users to share and work on files simultaneously.
Our collected data is stored in an AWS S3 bucket, facilitating seamless inte-

gration with Azure Databricks for efficient data processing and analytics.

## Amazon Web Services: AWS

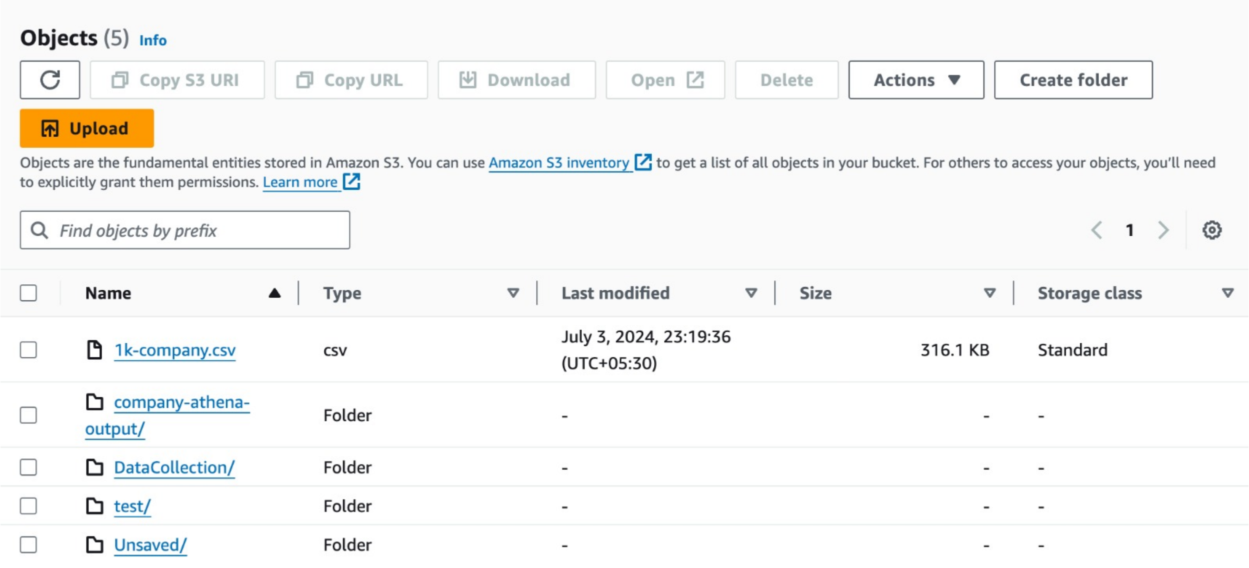
Amazon Web Services (AWS) is a leading cloud platform that offers a vast array of services, including computing power, storage options, and databases, all delivered over the internet with pay-as-you-go pricing. We have used AWS due to its ease of access and the variety of tools available for use in our project.



Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

| □ | Name | Type | Last modified | Size | Storage class |
|---|---|---|---|---|---|
| □ | 1k-company.csv | csv | July 2, 2024, 23:43:53 (UTC+05:30) | 2.2 MB | Standard |
| □ | 2k-company.csv | csv | July 2, 2024, 23:43:53 (UTC+05:30) | 2.6 MB | Standard |
| □ | 3k-company.csv | csv | July 2, 2024, 23:43:53 (UTC+05:30) | 3.1 MB | Standard |
| □ | 4k-company.csv | csv | July 2, 2024, 23:43:53 (UTC+05:30) | 3.0 MB | Standard |
| □ | 5k-company.csv | csv | July 2, 2024, 23:43:53 (UTC+05:30) | 3.0 MB | Standard |
| □ | 6k-company.csv | csv | July 2, 2024, 23:43:53 (UTC+05:30) | 2.9 MB | Standard |
| □ | 7k-company.csv | csv | July 2, 2024, 23:43:53 (UTC+05:30) | 157.8 KB | Standard |
| □ | 8k-company.csv | csv | July 2, 2024, 23:43:53 (UTC+05:30) | 2.9 MB | Standard |
|  |  |  | July 2, 2024, 23:43:53 |  |  |

Figure 6.4: Storing the csv files in cloud for easy access and deployment
We store the files in AWS S3 bucket cloud to facilitate easy integration with Databricks.



Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

| □ | Name ▲ | Type | Last modified | Size | Storage class |
|---|---|---|---|---|---|
| □ | 1k-company.csv | csv | July 3, 2024, 23:19:36 (UTC+05:30) | 316.1 KB | Standard |
| □ | company-athena-output/ | Folder | - | - | - |
| □ | DataCollection/ | Folder | - | - | - |
| □ | test/ | Folder | - | - | - |
| □ | Unsaved/ | Folder | - | - | - |

Figure 6.5: Storing the csv files in cloud for easy access and deployment
We store the files in AWS S3 bucket cloud to facilitate easy integration with Databricks.

## 6.3 Processing Layer

Data must be cleaned before processing to ensure accuracy, reliability, and efficiency in analysis. Cleaning data involves correcting errors, filling in missing values, and removing inconsistencies, which helps prevent skewed results and misleading conclusions. It enhances the quality of insights derived from the data, supports better decision-making, and ensures the smooth operation of data processing workflows. Clean data also improves the performance of algorithms and models, making the overall analytical process more effective and trustworthy.

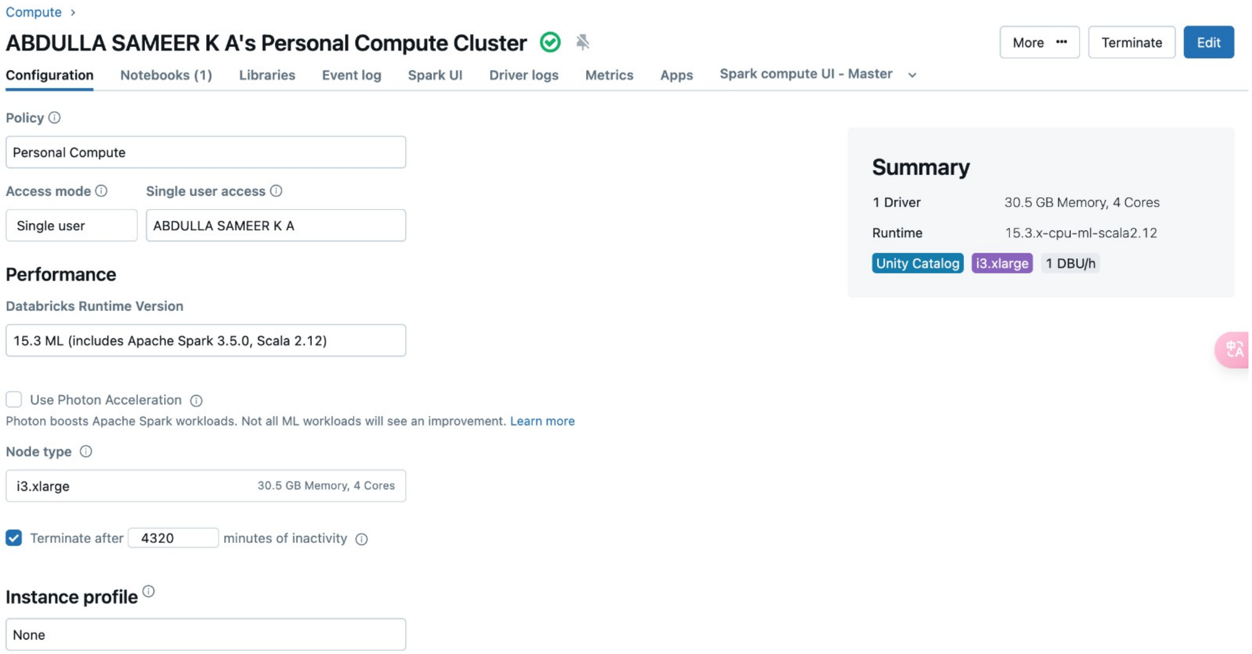
In our model we undertake three levels of data pre-processing or data cleaning.

1. Bronze Layer
2. Silver Layer
3. Gold Layer

## Azure Databricks

Azure Databricks is a unified analytics platform designed for large-scale data processing and machine learning tasks. Built on top of Apache Spark, it

provides an interactive workspace for data engineers, data scientists, and analysts to collaborate seamlessly. Databricks simplifies the management and scaling of Spark clusters, offering optimized performance for processing massive datasets in real-time. Its integrated notebooks support Python, R, SQL, and Scala, facilitating exploratory data analysis, model development, and deployment. Moreover, Databricks supports advanced analytics and machine learning workflows with built-in libraries and integration with popular tools like TensorFlow and PyTorch. Overall, Azure Databricks empowers organizations to accelerate innovation and derive actionable insights from their data efficiently.
We have made use of Databricks to easily import data from csv files stored in AWS S3 bucket and easy export into Tableau for visualisation. We use this platform to run python and pyspark scripts to preprocess the dataset. All scripts used for processing data while written on Visual Studio Code, are designed to be executed on Databricks.



ABDULLA SAMEER K A's Personal Compute Cluster

Policy: Personal Compute

Access mode: Single user

Single user access: ABDULLA SAMEER K A

Performance

Databricks Runtime Version: 15.3 ML (includes Apache Spark 3.5.0, Scala 2.12)

□ Use Photon Acceleration

Photon boosts Apache Spark workloads. Not all ML workloads will see an improvement. Learn more

Node type: i3.xlarge 30.5 GB Memory, 4 Cores

☑ Terminate after 4320 minutes of inactivity

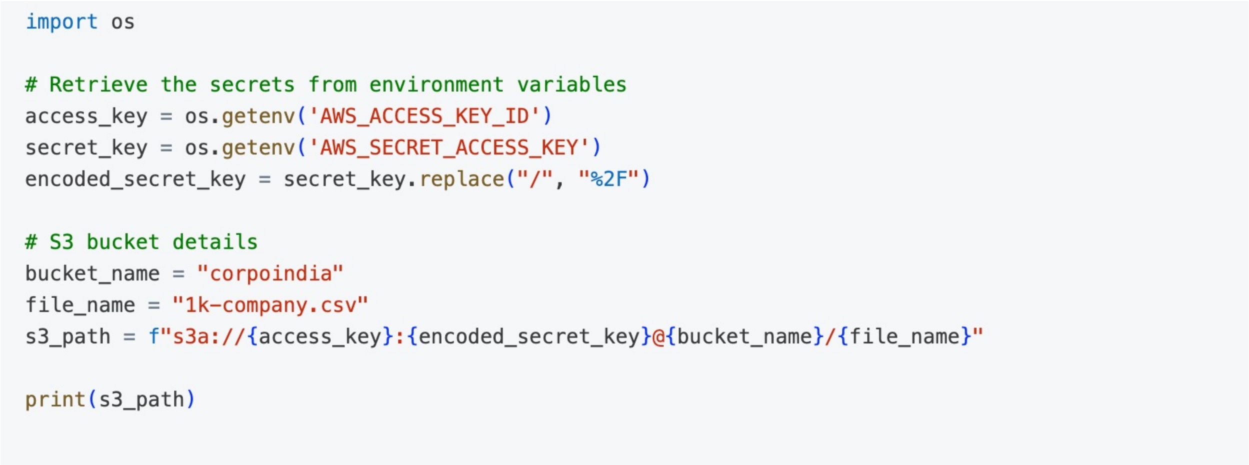
Instance profile: None

Summary

| | |
|---|---|
| 1 Driver | 30.5 GB Memory, 4 Cores |
| Runtime | 15.3.x-cpu-ml-scala2.12 |

Unity Catalog i3.xlarge 1 DBU/h

Figure 6.6: Databricks Cluster formation
Cluster Formation in Azure Databricks



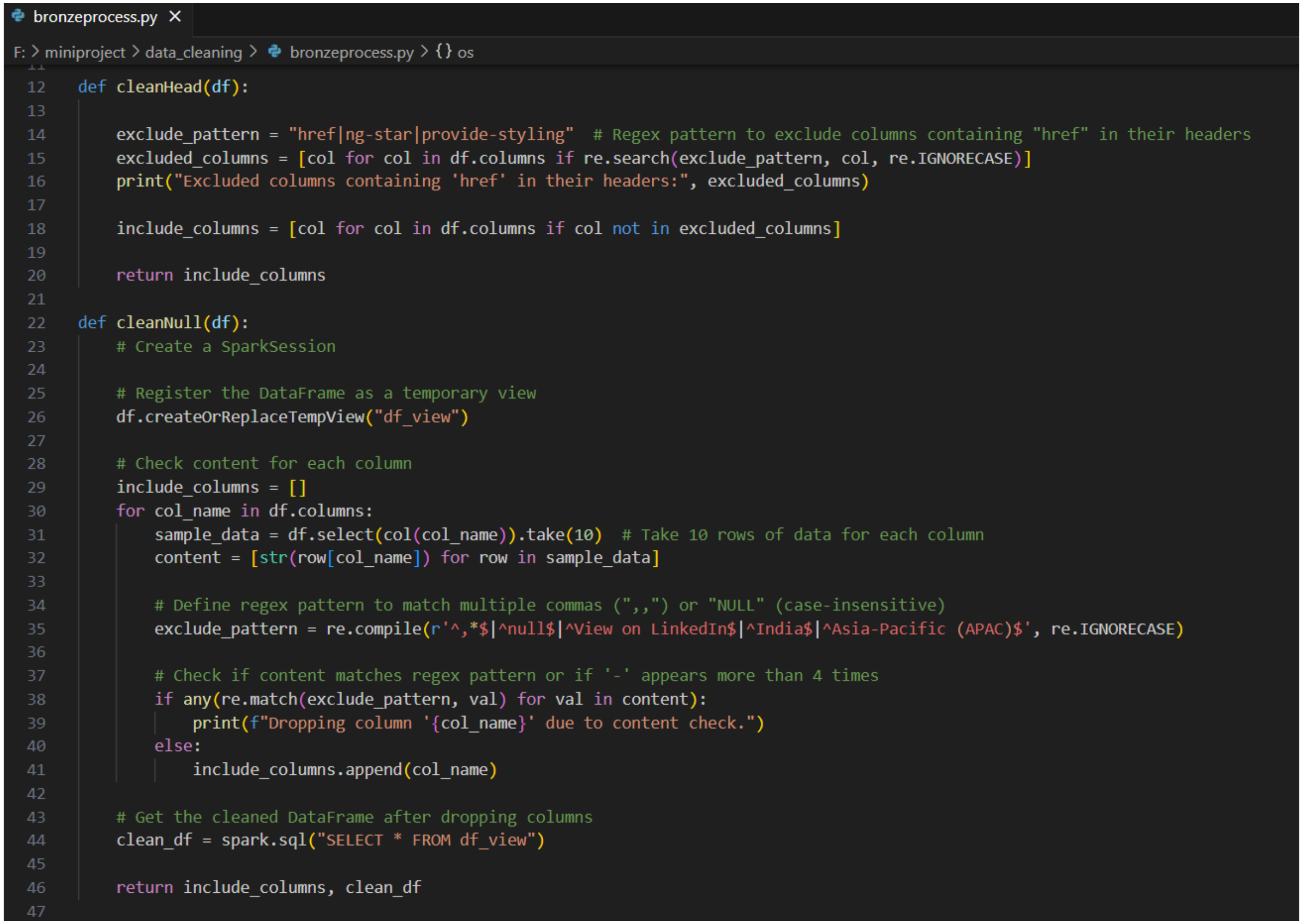
```
import os

# Retrieve the secrets from environment variables
access_key = os.getenv('AWS_ACCESS_KEY_ID')
secret_key = os.getenv('AWS_SECRET_ACCESS_KEY')
encoded_secret_key = secret_key.replace("/", "%2F")

# S3 bucket details
bucket_name = "corpoindia"
file_name = "1k-company.csv"
s3_path = f"s3a://{access_key}:{encoded_secret_key}@{bucket_name}/{file_name}"

print(s3_path)
```

Figure 6.7: Databricks AWS connection
Connecting AWS S3 bucket and Databricks

## Bronze Level layer

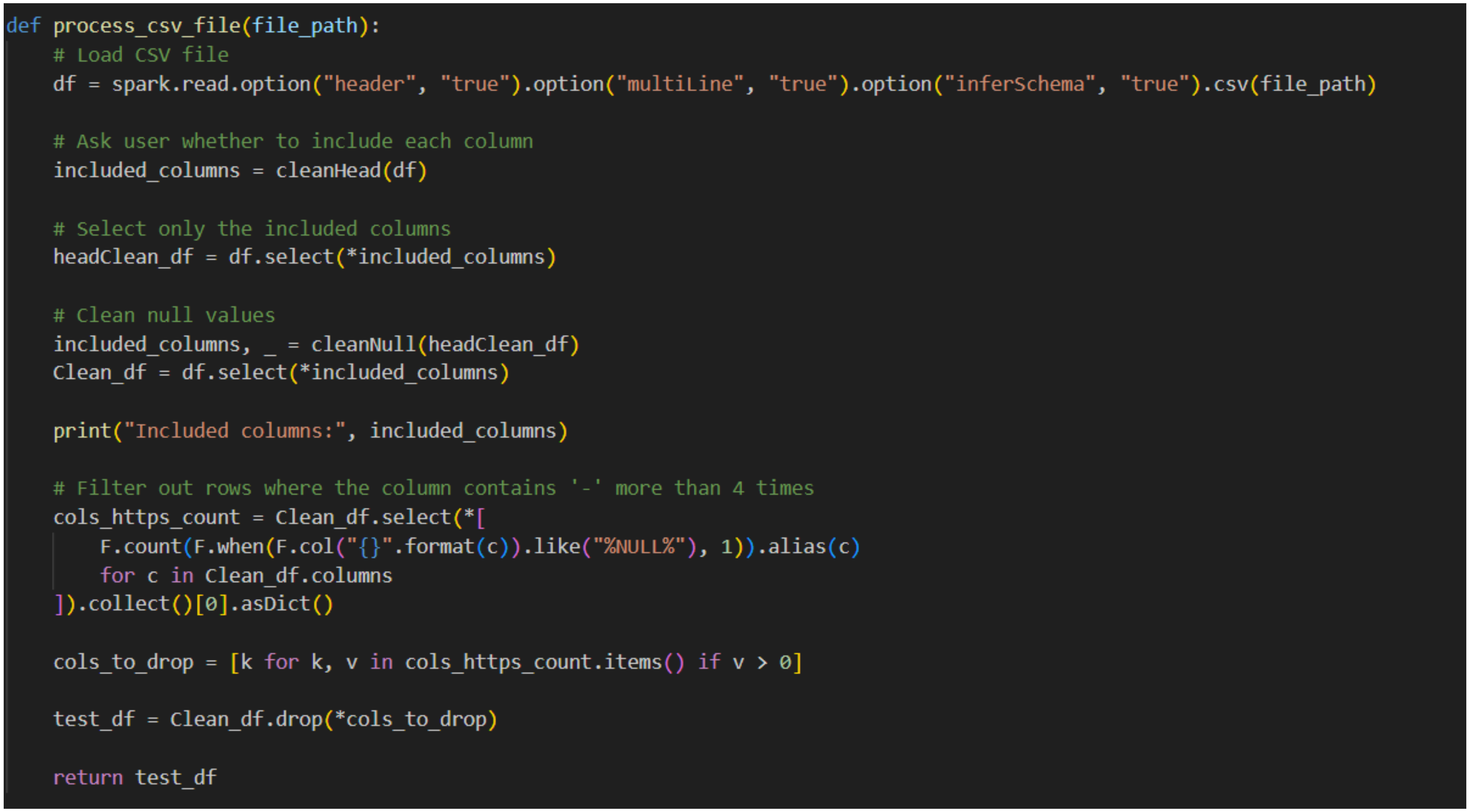
In data mining, bronze cleaning refers to the initial stage of data processing where raw, unrefined data is collected and stored in its most basic form, often termed as the ”bronze” layer. This stage involves minimal processing, primarily focusing on capturing and storing data as-is from various sources. The goal is to create a comprehensive and accurate repository of raw data that can later be refined and processed into more useful forms. By maintaining this unaltered data, organizations ensure they have a reliable foundation for further data cleaning, transformation, and analysis in subsequent stages, leading to more accurate and insightful results.

For our project, we started by eliminating unwanted links and invalid values from the data so as to reduce the size of our database. This increases performance of our cleaning scripts since it has to work on a smaller table. We also rename the columns to logical values to ensure easy access by future scrips.



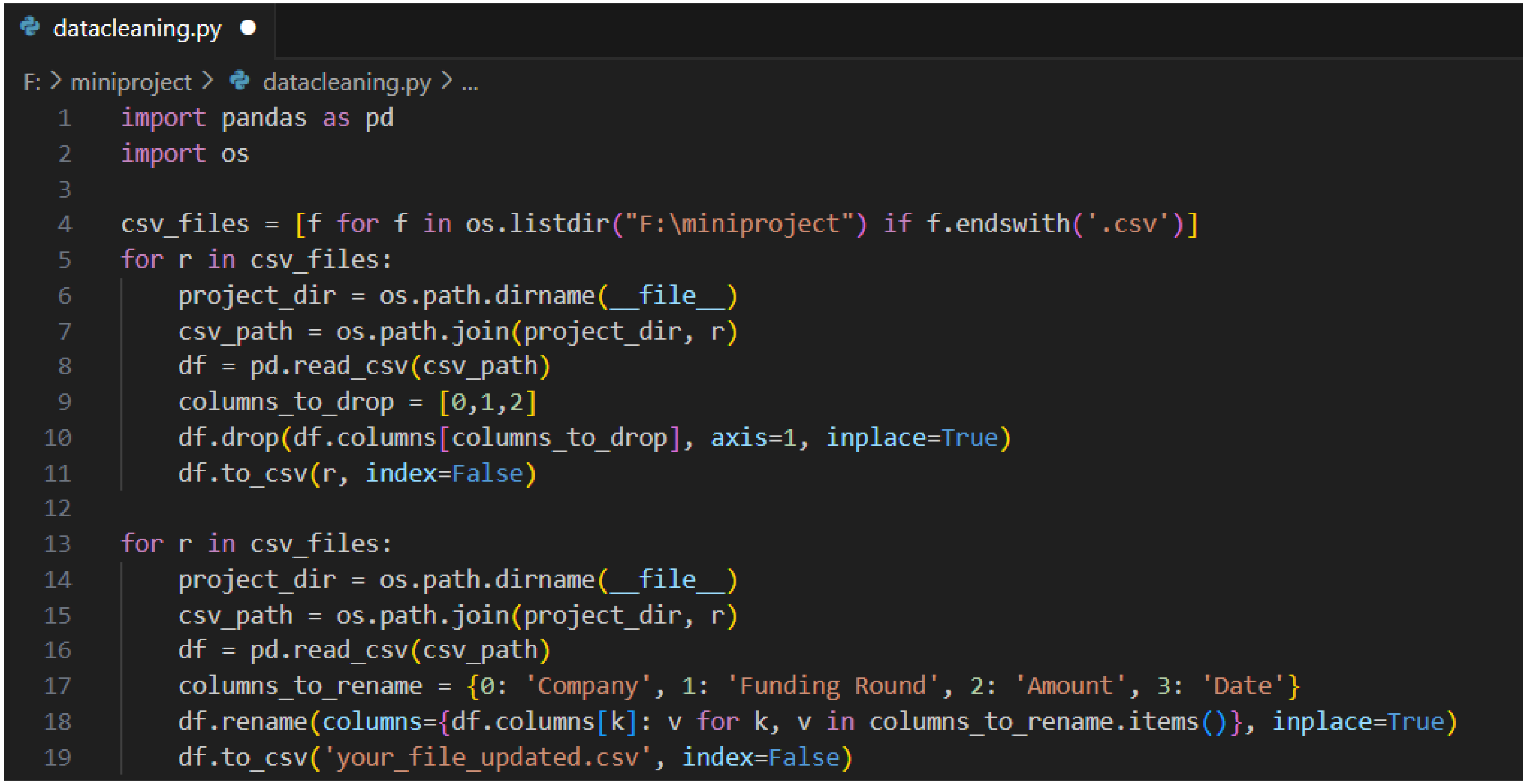
```python
def cleanHead(df):

    exclude_pattern = "href|ng-star|provide-styling"  # Regex pattern to exclude columns containing "href" in their headers
    excluded_columns = [col for col in df.columns if re.search(exclude_pattern, col, re.IGNORECASE)]
    print("Excluded columns containing 'href' in their headers:", excluded_columns)

    include_columns = [col for col in df.columns if col not in excluded_columns]

    return include_columns

def cleanNull(df):
    # Create a SparkSession

    # Register the DataFrame as a temporary view
    df.createOrReplaceTempView("df_view")

    # Check content for each column
    include_columns = []
    for col_name in df.columns:
        sample_data = df.select(col(col_name)).take(10)  # Take 10 rows of data for each column
        content = [str(row[col_name]) for row in sample_data]

        # Define regex pattern to match multiple commas (",,") or "NULL" (case-insensitive)
        exclude_pattern = re.compile(r'^,*$|^null$|^View on LinkedIn$|^India$|^Asia-Pacific (APAC)$', re.IGNORECASE)

        # Check if content matches regex pattern or if '-' appears more than 4 times
        if any(re.match(exclude_pattern, val) for val in content):
            print(f"Dropping column '{col_name}' due to content check.")
        else:
            include_columns.append(col_name)

    # Get the cleaned DataFrame after dropping columns
    clean_df = spark.sql("SELECT * FROM df_view")

    return include_columns, clean_df
```

Figure 6.8: Bronze Level cleaning
This script uses Spark to eliminate unwanted columns and data values from the csv files. Garbage values like unwanted weblinks and other unicode characters are eliminated here.



```python
def process_csv_file(file_path):
    # Load CSV file
    df = spark.read.option("header", "true").option("multiLine", "true").option("inferSchema", "true").csv(file_path)

    # Ask user whether to include each column
    included_columns = cleanHead(df)

    # Select only the included columns
    headClean_df = df.select(*included_columns)

    # Clean null values
    included_columns, _ = cleanNull(headClean_df)
    Clean_df = df.select(*included_columns)

    print("Included columns:", included_columns)

    # Filter out rows where the column contains '-' more than 4 times
    cols_https_count = Clean_df.select(*[
        F.count(F.when(F.col("{}".format(c)).like("%NULL%"), 1)).alias(c)
        for c in Clean_df.columns
    ]).collect()[0].asDict()

    cols_to_drop = [k for k, v in cols_https_count.items() if v > 0]

    test_df = Clean_df.drop(*cols_to_drop)

    return test_df
```

Figure 6.9: Bronze Level data cleaning
This script uses Spark to eliminate unwanted columns and data values from the csv files. Garbage values like unwanted weblinks and other unicode characters are eliminated here.

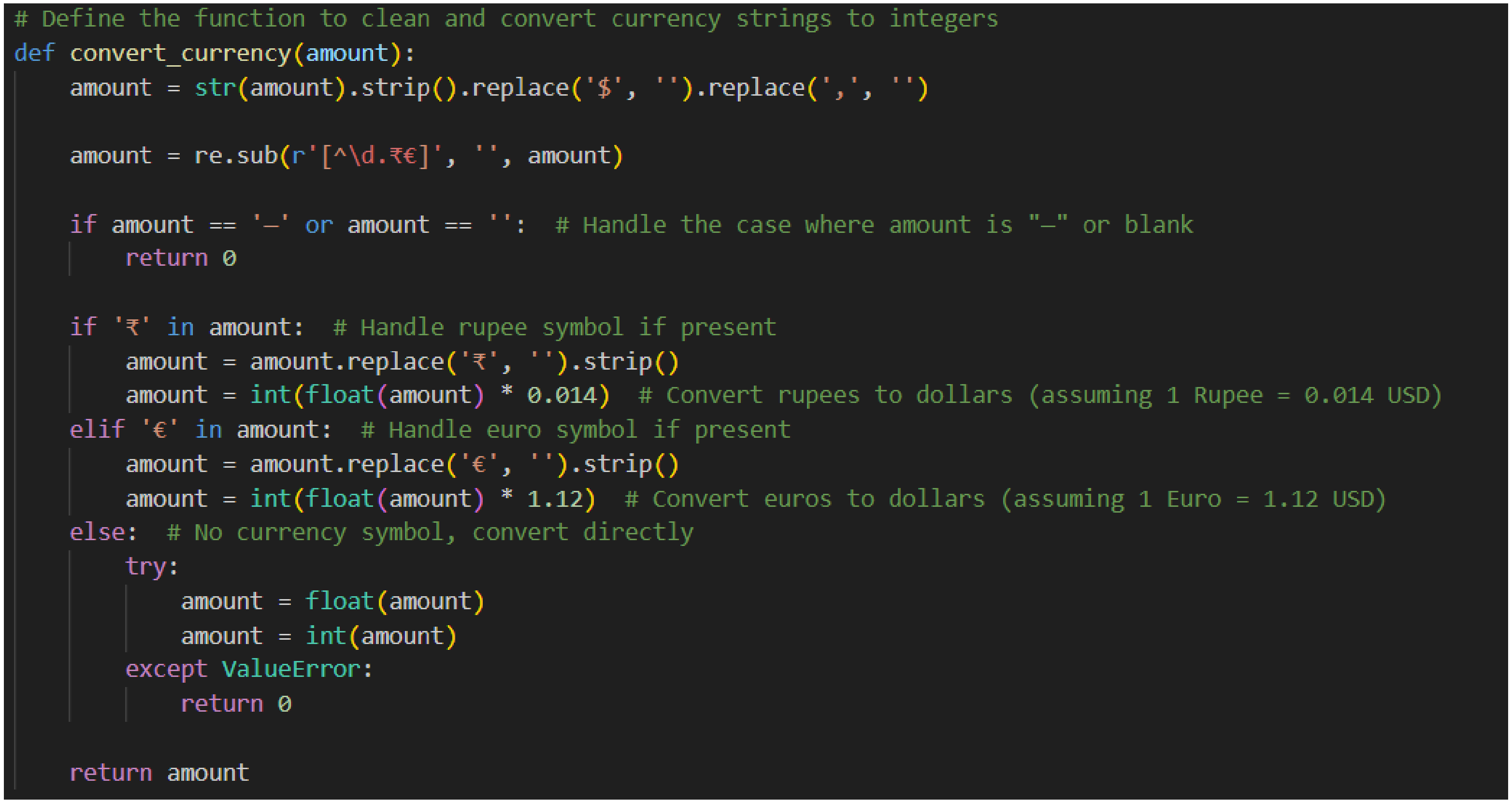


```python
import pandas as pd
import os

csv_files = [f for f in os.listdir("F:\miniproject") if f.endswith('.csv')]
for r in csv_files:
    project_dir = os.path.dirname(__file__)
    csv_path = os.path.join(project_dir, r)
    df = pd.read_csv(csv_path)
    columns_to_drop = [0,1,2]
    df.drop(df.columns[columns_to_drop], axis=1, inplace=True)
    df.to_csv(r, index=False)

for r in csv_files:
    project_dir = os.path.dirname(__file__)
    csv_path = os.path.join(project_dir, r)
    df = pd.read_csv(csv_path)
    columns_to_rename = {0: 'Company', 1: 'Funding Round', 2: 'Amount', 3: 'Date'}
    df.rename(columns={df.columns[k]: v for k, v in columns_to_rename.items()}, inplace=True)
    df.to_csv('your_file_updated.csv', index=False)
```

Figure 6.10: Bronze Level cleaning link deletion
Similarly we clean the funding csv files by deleting the unwanted columns containing links and we rename the columns to valid names for easy access in the coming scripts.

## Silver Level Layer

Silver level cleaning in data mining involves transforming and refining the raw data collected during the bronze cleaning stage. This intermediate step focuses on enhancing data quality by addressing issues such as missing values, duplicate records, and inconsistencies. The data is standardized and structured to make it more suitable for analysis, ensuring that it is accurate, consistent, and complete. Silver level cleaning prepares the data for more advanced processing and analysis, creating a more reliable and insightful dataset that can be used for decision-making and deeper analytics in the subsequent ”gold” stage.
In our csv files, there are multiple columns which represent money or currency. These include statistics like Last funding amount, Total funding amount etc. The raw data contains many inconsistencies like different currencies like dollars, euros, rupees etc. They also are in the form of a string with commas separating digits. These values cannot be used to visualise and analyse the data. So we clean them by converting them into dollars and then converting the string values into integer values, removing the commas and the currency symbol at the beginning. Any missing value is replaced by 0 since we cannot take the average of other values to replace it (decreases the accuracy of the data). We utilise the pandas library and selective formatting of strings in python to achieve our objectives.

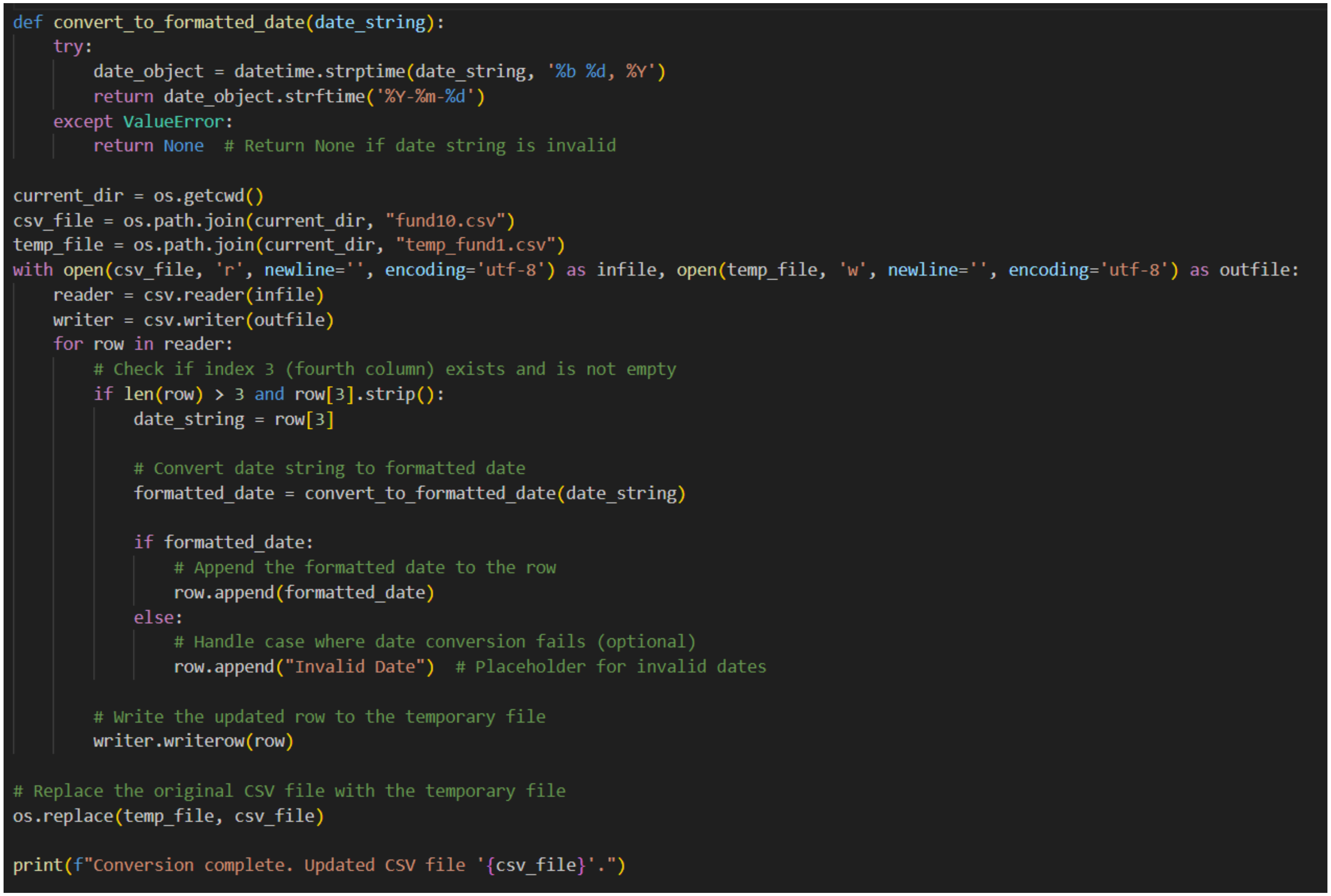


```python
# Define the function to clean and convert currency strings to integers
def convert_currency(amount):
    amount = str(amount).strip().replace('$', '').replace(',', '')

    amount = re.sub(r'[^\d.₹€]', '', amount)

    if amount == '–' or amount == '':  # Handle the case where amount is "–" or blank
        return 0

    if '₹' in amount:  # Handle rupee symbol if present
        amount = amount.replace('₹', '').strip()
        amount = int(float(amount) * 0.014)  # Convert rupees to dollars (assuming 1 Rupee = 0.014 USD)
    elif '€' in amount:  # Handle euro symbol if present
        amount = amount.replace('€', '').strip()
        amount = int(float(amount) * 1.12)  # Convert euros to dollars (assuming 1 Euro = 1.12 USD)
    else:  # No currency symbol, convert directly
        try:
            amount = float(amount)
            amount = int(amount)
        except ValueError:
            return 0

    return amount
```

Figure 6.11: Currency cleaning

Once bronze level cleaning is complete, we start the silver cleaning level. We transform the data into usable datatype. In both the funding and company detail csv files we convert all columns representing money into dollar denomination and then transform them into integer for visualisation.

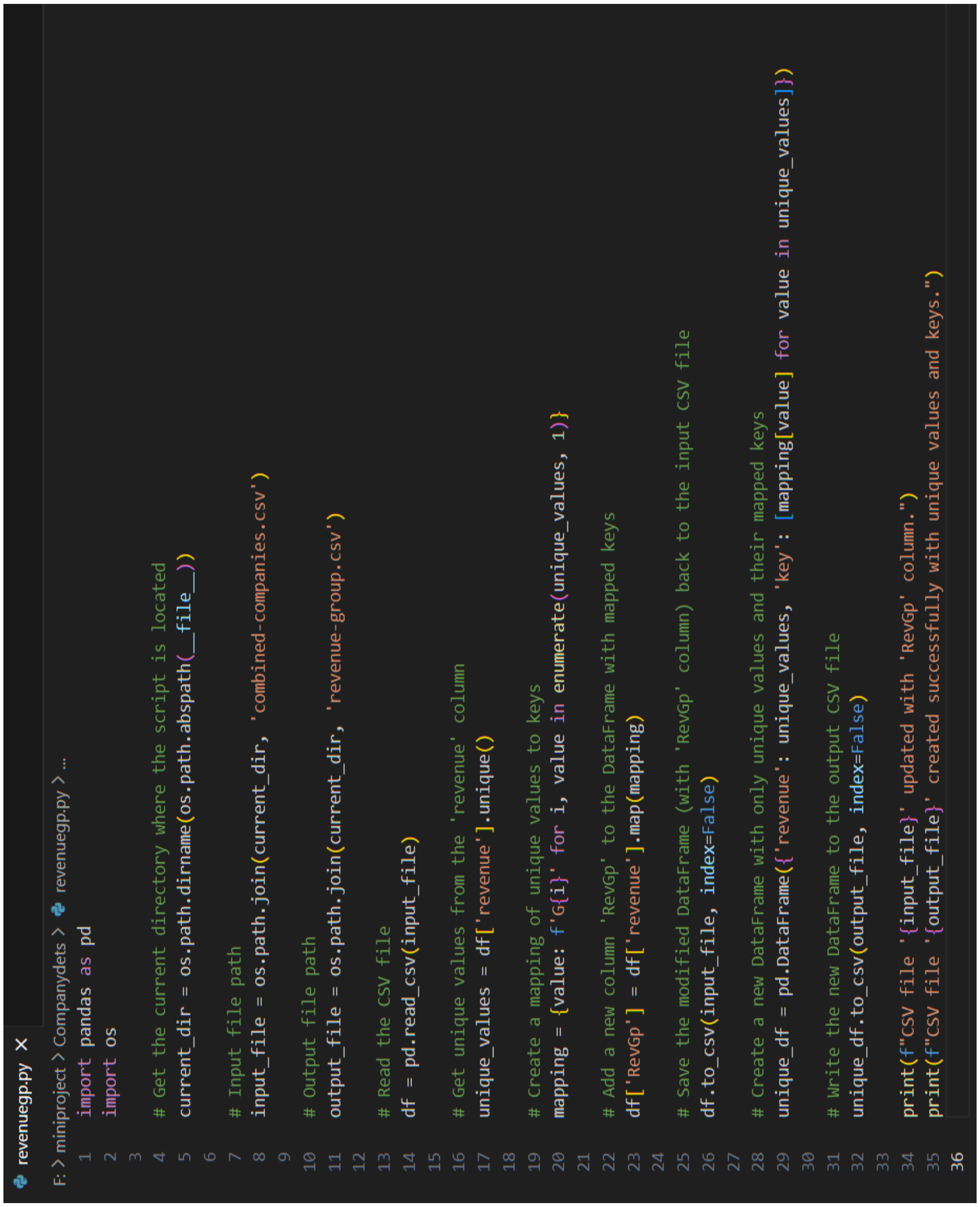
Another major cleaning operation to be undertaken here is to extract the year values from any date fields in the csv files. The year value can be used for visualisation and for our self defined ranking metric. We extract the year from fields like founding date and last funding date. These years can be useful to depict the funding age of a company i.e. how many years was the company active to validate funding.



```python
def convert_to_formatted_date(date_string):
    try:
        date_object = datetime.strptime(date_string, '%b %d, %Y')
        return date_object.strftime('%Y-%m-%d')
    except ValueError:
        return None  # Return None if date string is invalid

current_dir = os.getcwd()
csv_file = os.path.join(current_dir, "fund10.csv")
temp_file = os.path.join(current_dir, "temp_fund1.csv")
with open(csv_file, 'r', newline='', encoding='utf-8') as infile, open(temp_file, 'w', newline='', encoding='utf-8') as outfile:
    reader = csv.reader(infile)
    writer = csv.writer(outfile)
    for row in reader:
        # Check if index 3 (fourth column) exists and is not empty
        if len(row) > 3 and row[3].strip():
            date_string = row[3]

            # Convert date string to formatted date
            formatted_date = convert_to_formatted_date(date_string)

            if formatted_date:
                # Append the formatted date to the row
                row.append(formatted_date)
            else:
                # Handle case where date conversion fails (optional)
                row.append("Invalid Date")  # Placeholder for invalid dates

        # Write the updated row to the temporary file
        writer.writerow(row)

# Replace the original CSV file with the temporary file
os.replace(temp_file, csv_file)

print(f"Conversion complete. Updated CSV file '{csv_file}'.")
```

Figure 6.12: Date data cleaning
Continuing the silver level cleaning, we extract the year from all given date columns for usage in our metrics for use in our ranking formula as well as for use in data visualisation.

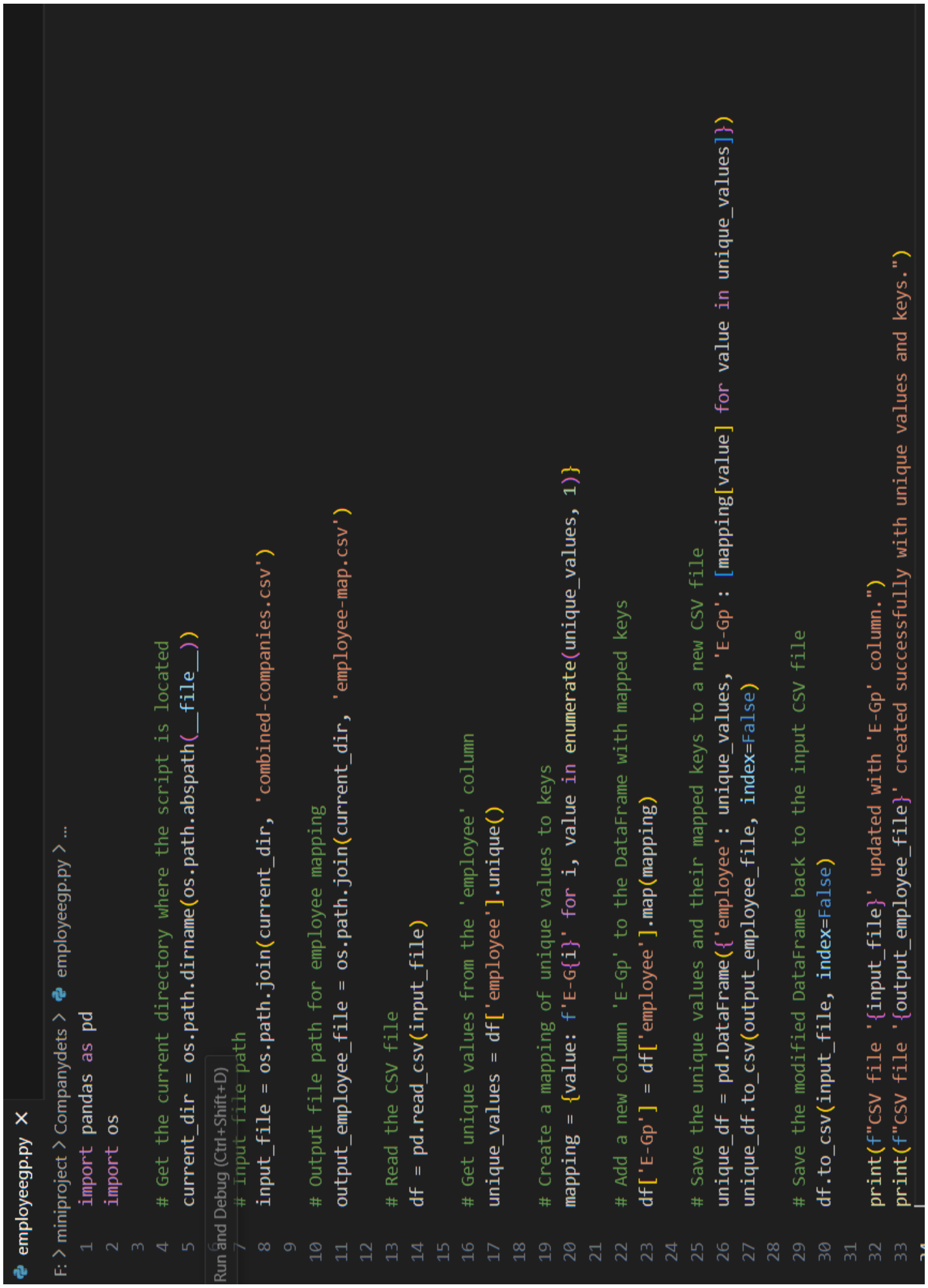
Now our next aim is to make the revenue field useful for our model. The revenue is represented as a range figure since companies rarely disclose these details in public databases. But we can still use them by assigning each unique value a key. This key is used to segregate the companies into revenue groups i.e. each company in the same revenue group has a revenue in that range. For example a company ABC in group R1 (10M to 50M) makes money in that range. So we assign these unique keys to each row and create a key to revenue range mapping csv file as well.



```python
import pandas as pd
import os

# Get the current directory where the script is located
current_dir = os.path.dirname(os.path.abspath(__file__))

# Input file path
input_file = os.path.join(current_dir, 'combined-companies.csv')

# Output file path
output_file = os.path.join(current_dir, 'revenue-group.csv')

# Read the CSV file
df = pd.read_csv(input_file)

# Get unique values from the 'revenue' column
unique_values = df['revenue'].unique()

# Create a mapping of unique values to keys
mapping = {value: f'G{i}' for i, value in enumerate(unique_values, 1)}

# Add a new column 'RevGp' to the DataFrame with mapped keys
df['RevGp'] = df['revenue'].map(mapping)

# Save the modified DataFrame (with 'RevGp' column) back to the input CSV file
df.to_csv(input_file, index=False)

# Create a new DataFrame with only unique values and their mapped keys
unique_df = pd.DataFrame({'revenue': unique_values, 'key': [mapping[value] for value in unique_values]})

# Write the new DataFrame to the output CSV file
unique_df.to_csv(output_file, index=False)

print(f"CSV file '{input_file}' updated with 'RevGp' column.")
print(f"CSV file '{output_file}' created successfully with unique values and keys.")

```

Figure 6.13: Revenue grouping
The code maps unique keys to each revenue group and then assigns the key to each row based on their revenue group.

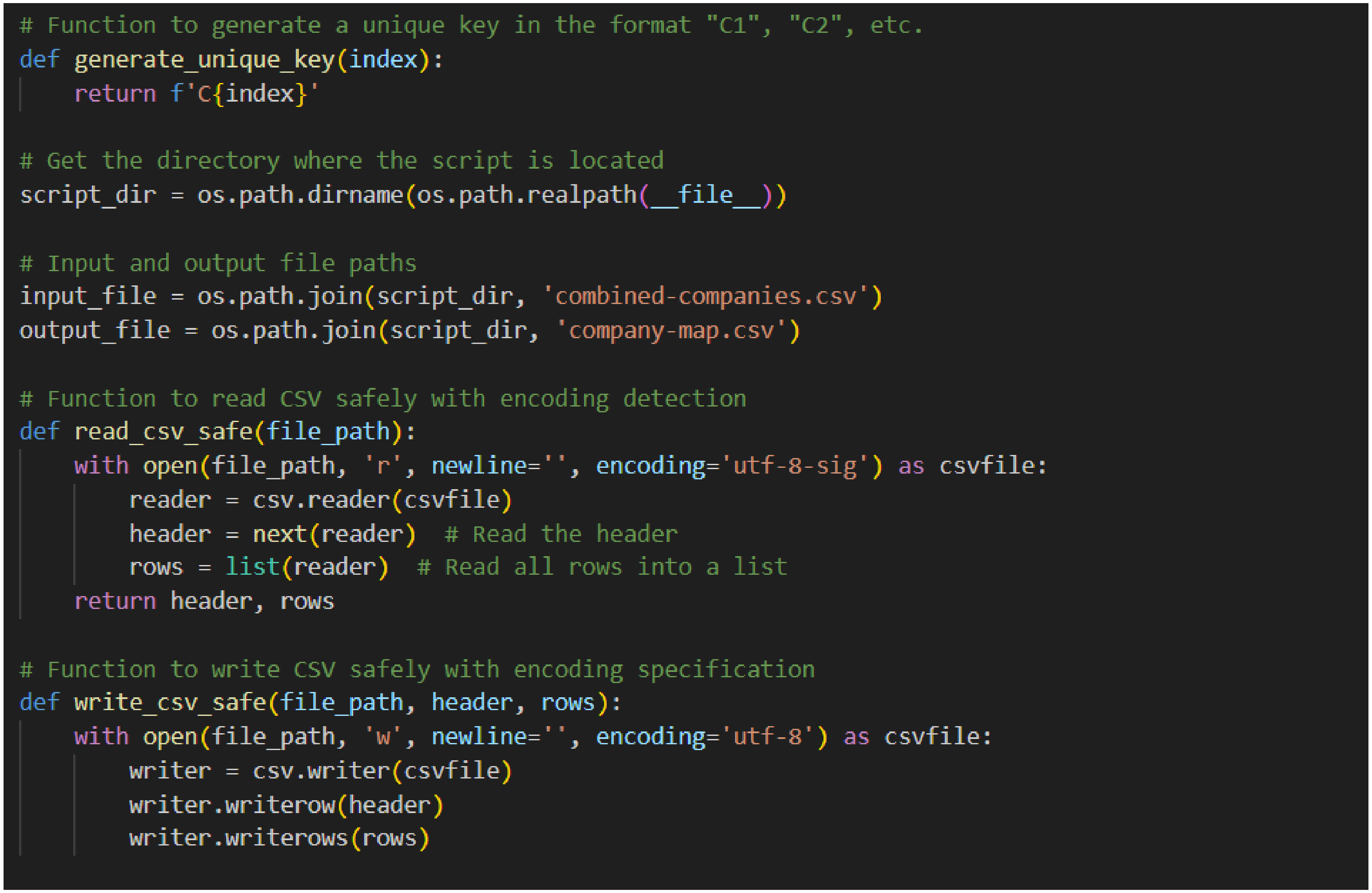
Similarly we create a key-range map for our employee groups. This way we can easily represent them and call them for our formula. Similar to the revenue group, a separate csv file with the key-range mapping is created.


employeegp.py
F: > miniproject > Companydets > employeegp.py > ...
Run and Debug (Ctrl+Shift+D)

```python
import pandas as pd
import os

# Get the current directory where the script is located
current_dir = os.path.dirname(os.path.abspath(__file__))

# input file path
input_file = os.path.join(current_dir, 'combined-companies.csv')

# Output file path for employee mapping
output_employee_file = os.path.join(current_dir, 'employee-map.csv')

# Read the CSV file
df = pd.read_csv(input_file)

# Get unique values from the 'employee' column
unique_values = df['employee'].unique()

# Create a mapping of unique values to keys
mapping = {value: f'E-G{i}' for i, value in enumerate(unique_values, 1)}

# Add a new column 'E-Gp' to the DataFrame with mapped keys
df['E-Gp'] = df['employee'].map(mapping)

# Save the unique values and their mapped keys to a new CSV file
unique_df = pd.DataFrame({'employee': unique_values, 'E-Gp': [mapping[value] for value in unique_values]})
unique_df.to_csv(output_employee_file, index=False)

# Save the modified DataFrame back to the input CSV file
df.to_csv(input_file, index=False)

print(f"CSV file '{input_file}' updated with 'E-Gp' column.")
print(f"CSV file '{output_employee_file}' created successfully with unique values and keys.")
```

Figure 6.14: Employee range grouping
Employee group is mapped to key value pairs.

The final processing left in our silver level cleaning operation is to give a unique company ID to all companies in the csv file. This way we have a primary key which can be used to easily call company data. This is also done using pandas library. A primary key is vital in database management as it uniquely identifies each record, ensuring data accuracy and preventing duplicates. It facilitates efficient data retrieval and updates while maintaining entity integrity. Additionally, primary keys help establish relationships between tables, enhancing database organization and performance.



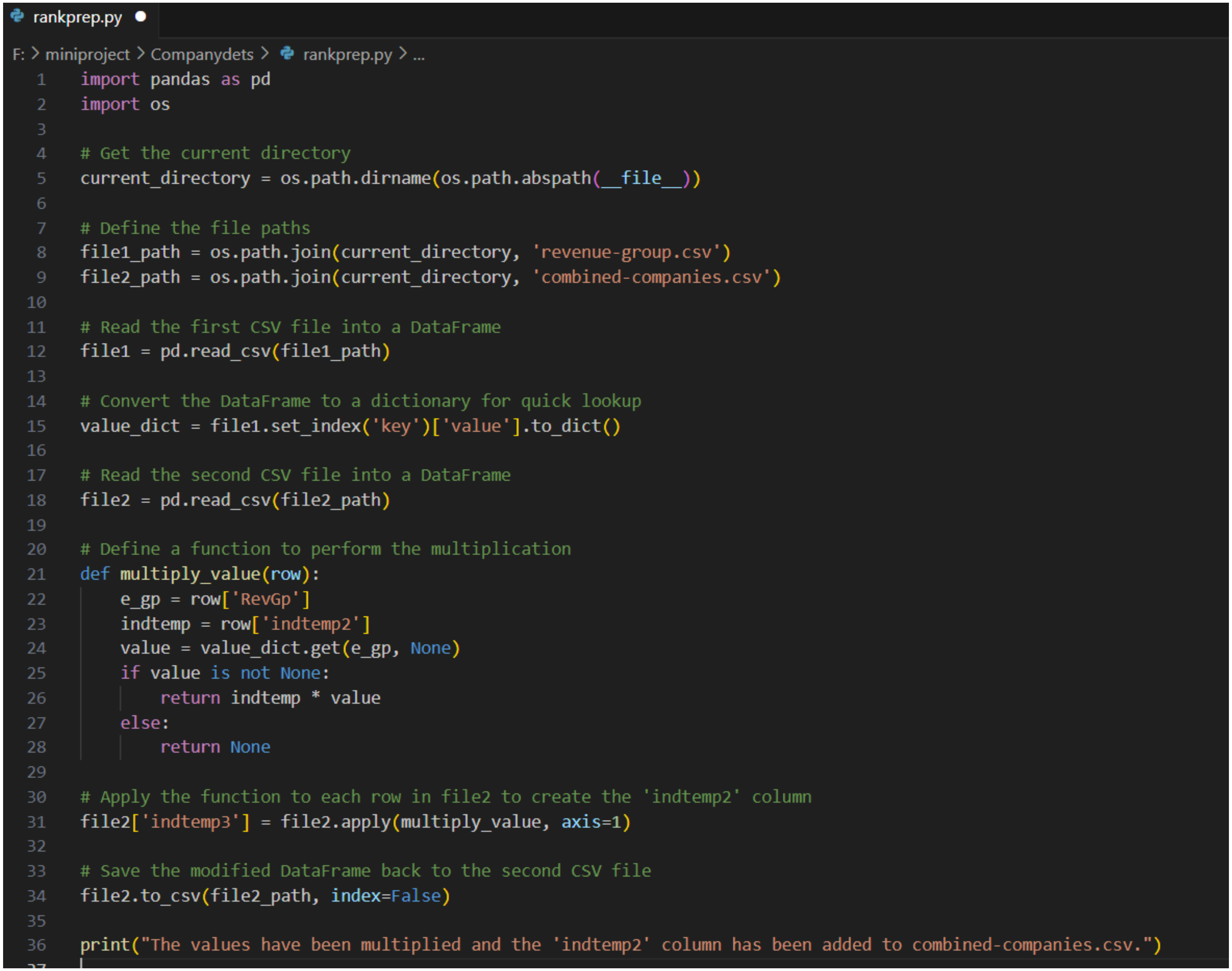
```python
# Function to generate a unique key in the format "C1", "C2", etc.
def generate_unique_key(index):
    return f'C{index}'

# Get the directory where the script is located
script_dir = os.path.dirname(os.path.realpath(__file__))

# Input and output file paths
input_file = os.path.join(script_dir, 'combined-companies.csv')
output_file = os.path.join(script_dir, 'company-map.csv')

# Function to read CSV safely with encoding detection
def read_csv_safe(file_path):
    with open(file_path, 'r', newline='', encoding='utf-8-sig') as csvfile:
        reader = csv.reader(csvfile)
        header = next(reader)  # Read the header
        rows = list(reader)  # Read all rows into a list
    return header, rows

# Function to write CSV safely with encoding specification
def write_csv_safe(file_path, header, rows):
    with open(file_path, 'w', newline='', encoding='utf-8') as csvfile:
        writer = csv.writer(csvfile)
        writer.writerow(header)
        writer.writerows(rows)
```

Figure 6.15: Primary key assignment
A unique Company ID C¡ID¿ is assigned to each company in our csv file to ensure that they can be identified uniquely by an attribute.

## Gold Level Layer

The gold cleaning layer in data mining represents the final stage of data refinement. At this level, the data undergoes advanced processing, including thorough validation, enrichment, and aggregation, to ensure it is of the highest quality. This stage involves deriving new metrics, enhancing data with additional context, and ensuring consistency across datasets. The result is a highly reliable, clean, and enriched dataset ready for in-depth analysis and accurate insights, supporting robust decision-making and strategic planning. As mentioned many times throughout our project, we must provide users with an easy to understand ranking method by which they can compare companies. Such a metric is not easy to create since several years of research would be necessary before one can claim any such metric worthy enough to rank companies by. For the scope of this project, we have researched various articles and consulted chatGPT to decide some valuable statistics to measure the worth of a company for our project. We use the total funded amount divided by the funding age of that company. The resultant value is multiplied with factors assigned to that company's respective revenue and employee groups. The factor representing revenue is greater than that of employee since revenue plays a greater role in general comparison.
We use the formula,

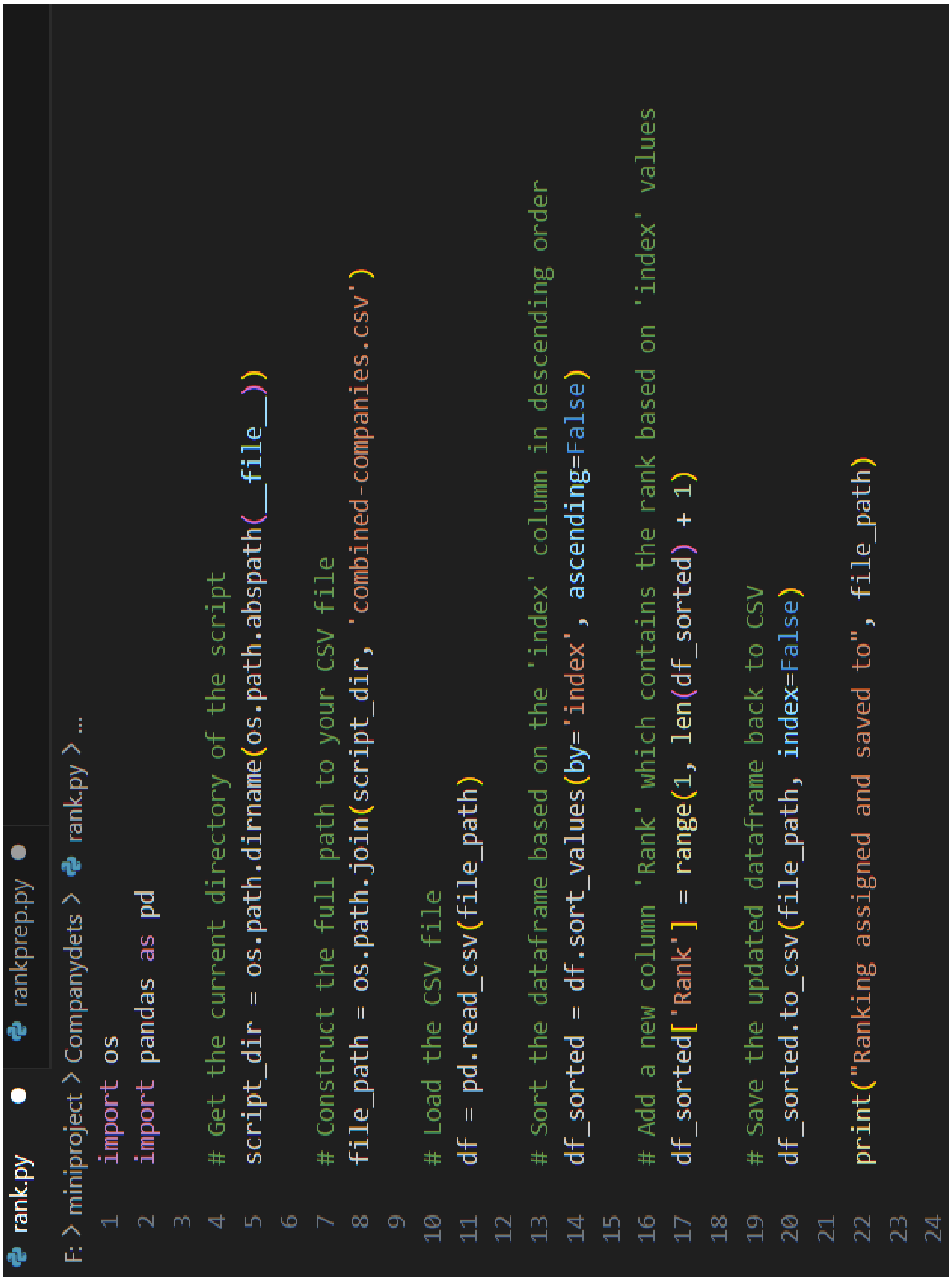
Index = Funding amount x (1/funding age) x revenue factor x employee factor
After gold layer processing, two tables are formed, funding gold and company details gold table. These are pushed to AWS S3 bucket where there are stored.



```python
import pandas as pd
import os

# Get the current directory
current_directory = os.path.dirname(os.path.abspath(__file__))

# Define the file paths
file1_path = os.path.join(current_directory, 'revenue-group.csv')
file2_path = os.path.join(current_directory, 'combined-companies.csv')

# Read the first CSV file into a DataFrame
file1 = pd.read_csv(file1_path)

# Convert the DataFrame to a dictionary for quick lookup
value_dict = file1.set_index('key')['value'].to_dict()

# Read the second CSV file into a DataFrame
file2 = pd.read_csv(file2_path)

# Define a function to perform the multiplication
def multiply_value(row):
    e_gp = row['RevGp']
    indtemp = row['indtemp2']
    value = value_dict.get(e_gp, None)
    if value is not None:
        return indtemp * value
    else:
        return None

# Apply the function to each row in file2 to create the 'indtemp2' column
file2['indtemp3'] = file2.apply(multiply_value, axis=1)

# Save the modified DataFrame back to the second CSV file
file2.to_csv(file2_path, index=False)

print("The values have been multiplied and the 'indtemp2' column has been added to combined-companies.csv.")
```

Figure 6.16: Index calculation
We prepare the index for ranking the company.

Once the index is ready, we call a script to rank the companies based on their index value.



```
import os
import pandas as pd

# Get the current directory of the script
script_dir = os.path.dirname(os.path.abspath(__file__))

# Construct the full path to your CSV file
file_path = os.path.join(script_dir, 'combined-companies.csv')

# Load the CSV file
df = pd.read_csv(file_path)

# Sort the dataframe based on the 'index' column in descending order
df_sorted = df.sort_values(by='index', ascending=False)

# Add a new column 'Rank' which contains the rank based on 'index' values
df_sorted['Rank'] = range(1, len(df_sorted) + 1)

# Save the updated dataframe back to CSV
df_sorted.to_csv(file_path, index=False)

print("Ranking assigned and saved to", file_path)
```

Figure 6.17: Ranking companies
Ranking each company

## 6.4 Service Layer

In data analysis, the service layer serves as a crucial intermediary between raw data and meaningful insights. It functions as a structured framework where data processing, transformation, and analysis occur, ensuring data integrity and efficiency. This layer typically involves data cleaning, aggregation, and statistical operations, preparing data for visualization or further modeling. Moreover, it facilitates seamless integration of data sources and enhances the overall reliability and scalability of analytical processes. Ultimately, the service layer bridges the gap between raw data and actionable intelligence, empowering organizations to make informed decisions based on robust data-driven insights.

For our project, data must be represented in such a way that it is easy to understand, useful and meaningful. The best way to represent data such that even the common man can understand it without a great degree of technical knowledge is through charts and graphs.

We use Tableau to visualise data in such a way that people can easily understand and draw insights/inferences from it.

Using an ODBC driver to connect Azure Databricks and Tableau enables seamless integration and real-time data visualization. This connection allows users to leverage the powerful data processing capabilities of Databricks alongside Tableau's advanced analytical and visualization tools. By configuring the ODBC driver, data can be securely and efficiently transferred from Azure Databricks to Tableau, facilitating dynamic and interactive dashboards.

## Tableau

Tableau is a powerful data visualization tool that enables users to transform complex datasets into easily understandable visual representations. It supports a wide range of visualizations, from basic charts to interactive dashboards, allowing analysts and decision-makers to explore trends, patterns, and relationships within data intuitively. Tableau's drag-and-drop interface simplifies the creation of dynamic visualizations without needing extensive programming knowledge. It integrates with various data sources, including

databases, spreadsheets, and cloud services, facilitating real-time data analysis and decision-making.
There are two versions of Tableau that we have made use of to display dynamic and interactive dashboards in our web application.

### Tableau Desktop

For our project we have used Tableau Desktop to connect to Azure Databricks using ODBC drivers. We import the two gold tables from Databricks. This data is then used to visualise and analyse the data into interactive charts and graphs.

### Tableau Public

Once we have prepared our charts in Tableau Desktop, we publish the workbook to Tableau Public to easily embed the charts in our front end.

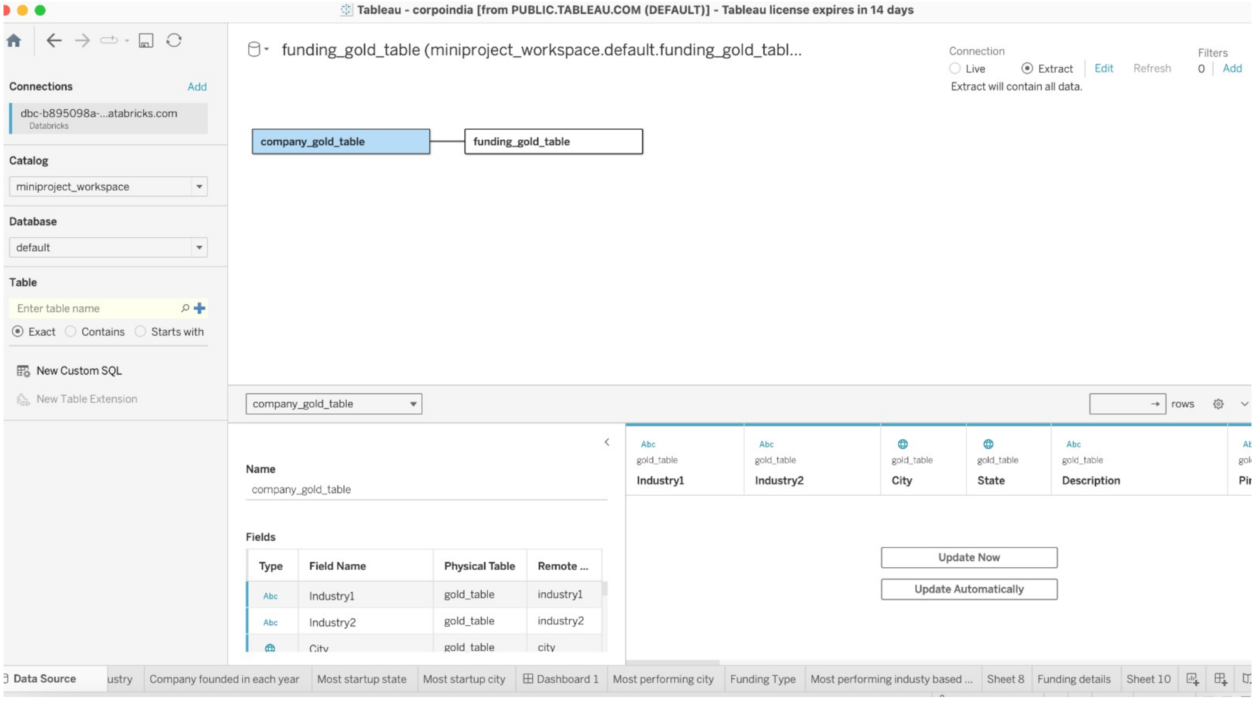


Tableau - corpoindia [from PUBLIC.TABLEAU.COM (DEFAULT)] - Tableau license expires in 14 days

Connections: dbc-b895098a-...atabricks.com (Databricks)

Catalog: miniproject_workspace

Database: default

Table: Enter table name

● Exact ○ Contains ○ Starts with

funding_gold_table (miniproject_workspace.default.funding_gold_tabl...

Connection: ○ Live ● Extract

Extract will contain all data.

Filters: 0

Name: company_gold_table

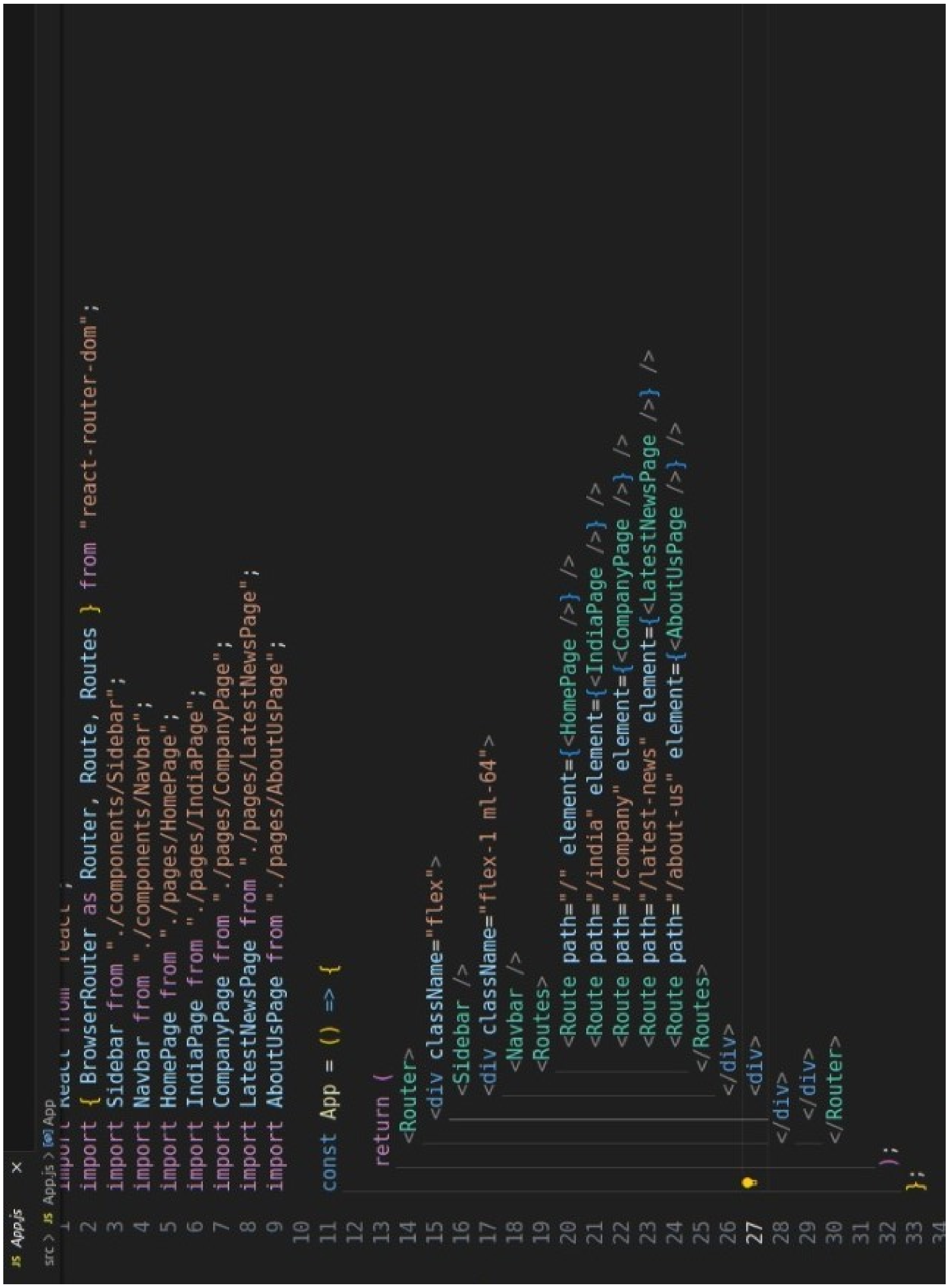
Fields

| Type | Field Name | Physical Table | Remote ... |
|---|---|---|---|
| Abc | Industry1 | gold_table | industry1 |
| Abc | Industry2 | gold_table | industry2 |
| 🌐 | City | gold_table | city |

Figure 6.18: Tableau data visualisation
Representing data in the form of charts and graphs using Tableau.

## 6.5 Web Application

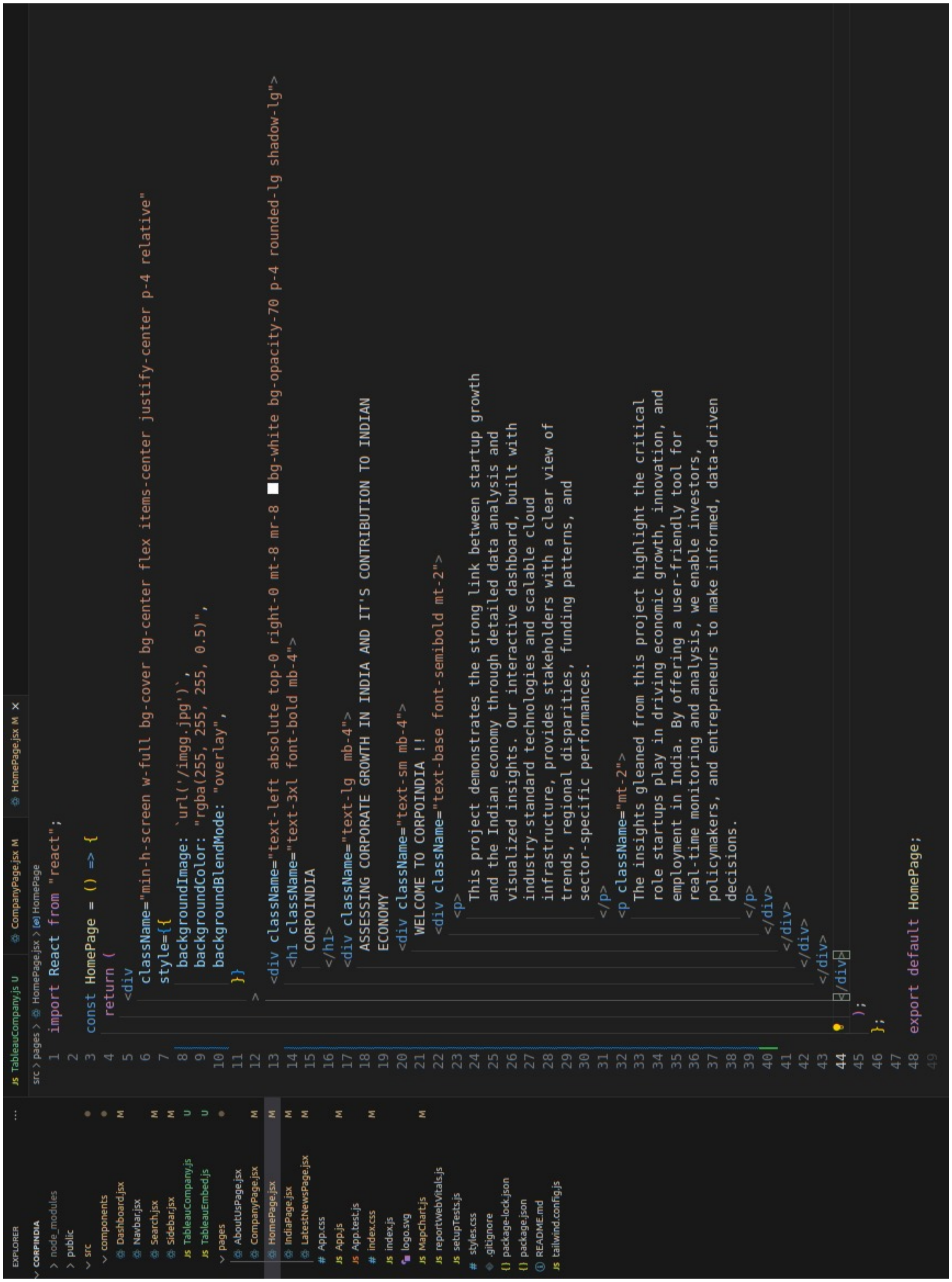
Presenting a data analysis dashboard on a web application enables users to interactively explore and visualize complex datasets. By integrating intuitive charts, graphs, and tables, a well-designed dashboard allows stakeholders to quickly grasp insights and trends, facilitating data-driven decision-making. Leveraging web technologies ensures accessibility from various devices and locations, enhancing collaboration and real-time data updates. A dynamic dashboard not only simplifies data interpretation but also empowers users to customize their view, filter information, and drill down into specifics, making data analysis more engaging and effective.



```
import React from "react";
import { BrowserRouter as Router, Route, Routes } from "react-router-dom";
import Sidebar from "./components/Sidebar";
import Navbar from "./components/Navbar";
import HomePage from "./pages/HomePage";
import IndiaPage from "./pages/IndiaPage";
import CompanyPage from "./pages/CompanyPage";
import LatestNewsPage from "./pages/LatestNewsPage";
import AboutUsPage from "./pages/AboutUsPage";

const App = () => {

  return (
    <Router>
      <div className="flex">
        <Sidebar />
        <div className="flex-1 ml-64">
          <Navbar />
          <Routes>
            <Route path="/" element={<HomePage />} />
            <Route path="/india" element={<IndiaPage />} />
            <Route path="/company" element={<CompanyPage />} />
            <Route path="/latest-news" element={<LatestNewsPage />} />
            <Route path="/about-us" element={<AboutUsPage />} />
          </Routes>
        </div>
        <div>
    </div>
      </div>
    </Router>

  );
};
```

Figure 6.19: Code Snippet of App
Code Snippet depicting the structure of the web application.

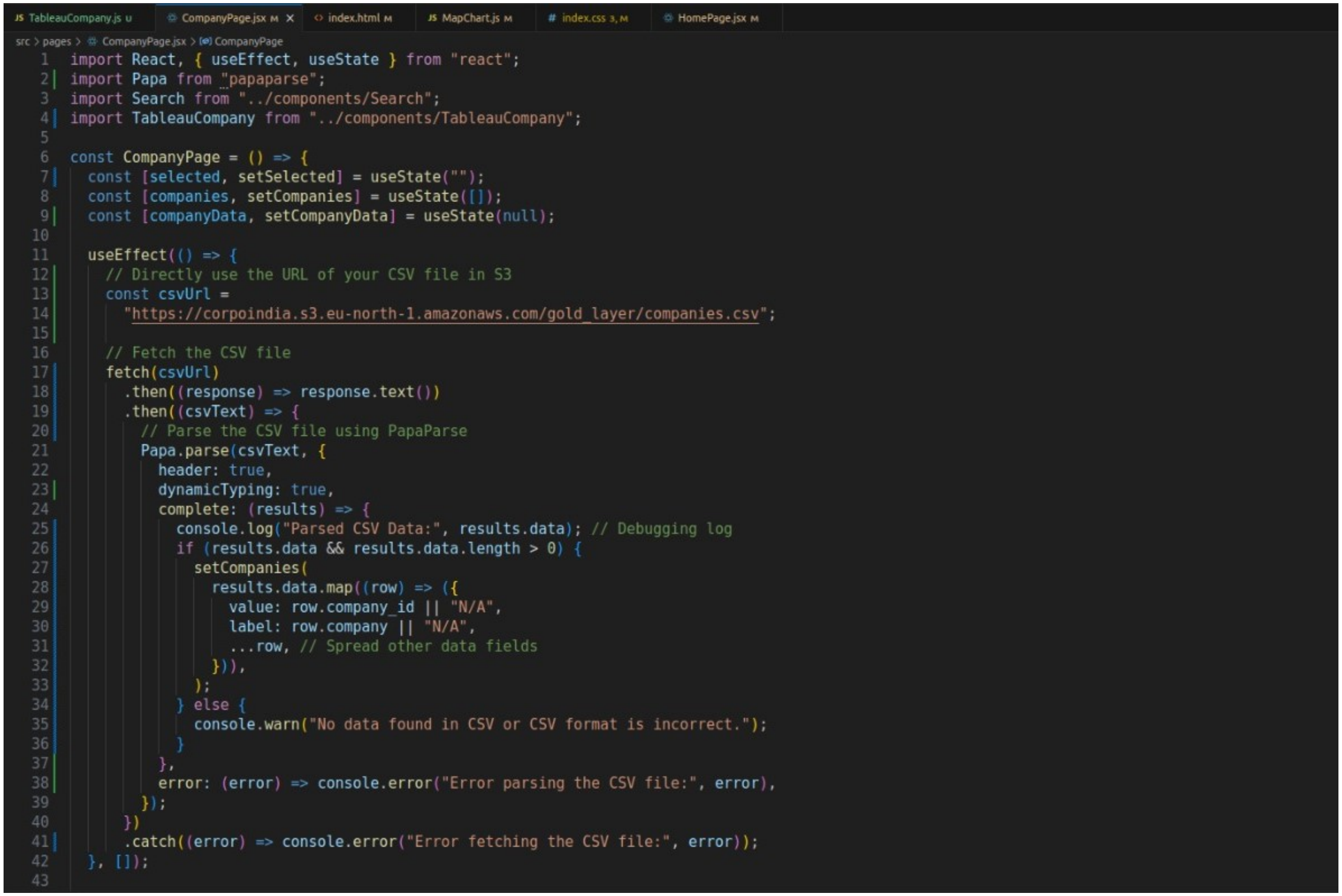
The homepage of our web application provides a comprehensive overview of the project and its objectives. This introductory section outlines the purpose and scope of the data analysis dashboard, highlighting its significance in facilitating informed decision-making.



```
import React from "react";

const HomePage = () => {
  return (
    <div
      className="min-h-screen w-full bg-cover bg-center flex items-center justify-center p-4 relative"
      style={{
        backgroundImage: `url('/imgg.jpg')`,
        backgroundColor: "rgba(255, 255, 255, 0.5)",
        backgroundBlendMode: "overlay",
      }}
    >
      <div className="text-left absolute top-0 right-0 mt-8 mr-8 □bg-white bg-opacity-70 p-4 rounded-lg shadow-lg">
        <h1 className="text-3xl font-bold mb-4">
          CORPOINDIA
        </h1>
        <div className="text-lg  mb-4">
          ASSESSING CORPORATE GROWTH IN INDIA AND IT'S CONTRIBUTION TO INDIAN
          ECONOMY
          <div className="text-sm mb-4">
            WELCOME TO CORPOINDIA !!
            <div className="text-base font-semibold mt-2">
              <p>
                This project demonstrates the strong link between startup growth
                and the Indian economy through detailed data analysis and
                visualized insights. Our interactive dashboard, built with
                industry-standard technologies and scalable cloud
                infrastructure, provides stakeholders with a clear view of
                trends, regional disparities, funding patterns, and
                sector-specific performances.
              </p>
              <p className="mt-2">
                The insights gleaned from this project highlight the critical
                role startups play in driving economic growth, innovation, and
                employment in India. By offering a user-friendly tool for
                real-time monitoring and analysis, we enable investors,
                policymakers, and entrepreneurs to make informed, data-driven
                decisions.
              </p>
            </div>
          </div>
        </div>
      </div>
    </div>
  );
};

export default HomePage;
```

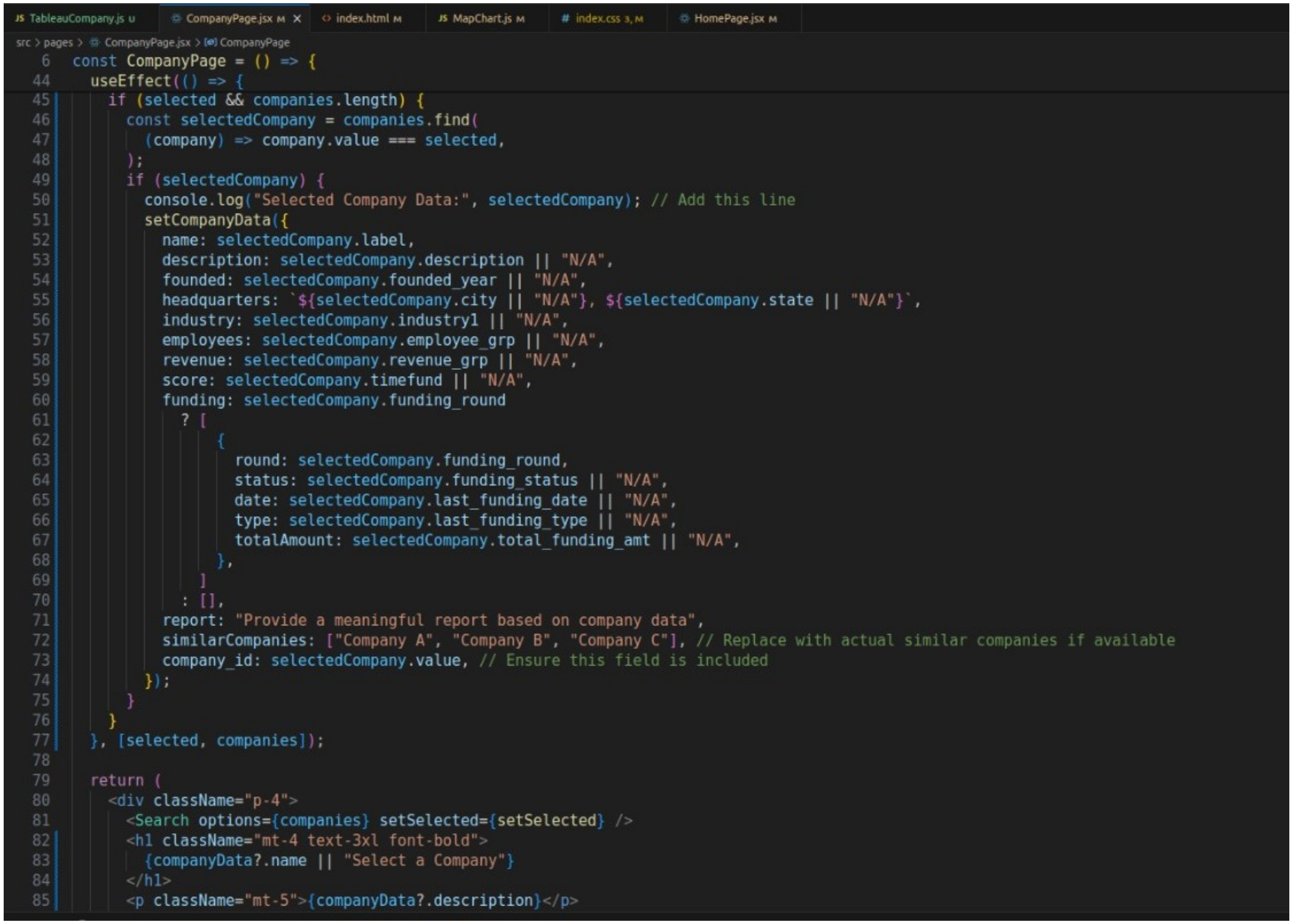
Figure 6.20: Code Snippet of Home Page
Code Snippet depicting the Home page of the web application.

The Company Statistics page offers an in-depth look at the company's key metrics and performance indicators. This page features a detailed profile of the company, including its mission, vision, and core values. Accompanying this information are dynamic graphs and charts that visually represent vital statistics like funding amounts and rounds.



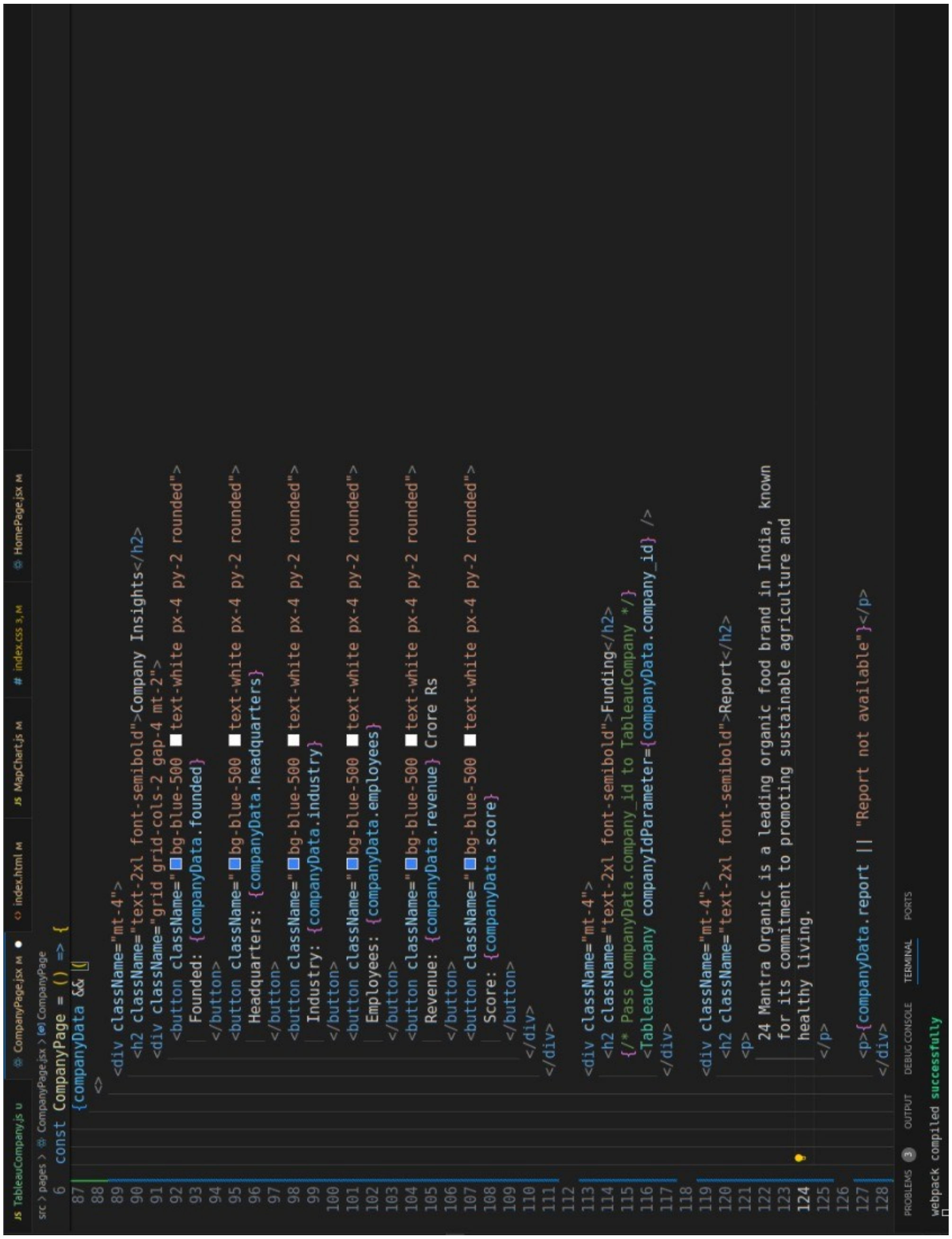
```
import React, { useEffect, useState } from "react";
import Papa from "papaparse";
import Search from "../components/Search";
import TableauCompany from "../components/TableauCompany";

const CompanyPage = () => {
  const [selected, setSelected] = useState("");
  const [companies, setCompanies] = useState([]);
  const [companyData, setCompanyData] = useState(null);

  useEffect(() => {
    // Directly use the URL of your CSV file in S3
    const csvUrl =
      "https://corpoindia.s3.eu-north-1.amazonaws.com/gold_layer/companies.csv";

    // Fetch the CSV file
    fetch(csvUrl)
      .then((response) => response.text())
      .then((csvText) => {
        // Parse the CSV file using PapaParse
        Papa.parse(csvText, {
          header: true,
          dynamicTyping: true,
          complete: (results) => {
            console.log("Parsed CSV Data:", results.data); // Debugging log
            if (results.data && results.data.length > 0) {
              setCompanies(
                results.data.map((row) => ({
                  value: row.company_id || "N/A",
                  label: row.company || "N/A",
                  ...row, // Spread other data fields
                })),
              );
            } else {
              console.warn("No data found in CSV or CSV format is incorrect.");
            }
          },
          error: (error) => console.error("Error parsing the CSV file:", error),
        });
      })
      .catch((error) => console.error("Error fetching the CSV file:", error));
  }, []);

```

Figure 6.21: Code Snippet of Company Statistics Page
Code Snippet depicting the Company Statistics page of the web application.



```
const CompanyPage = () => {
  useEffect(() => {
    if (selected && companies.length) {
      const selectedCompany = companies.find(
        (company) => company.value === selected,
      );
      if (selectedCompany) {
        console.log("Selected Company Data:", selectedCompany); // Add this line
        setCompanyData({
          name: selectedCompany.label,
          description: selectedCompany.description || "N/A",
          founded: selectedCompany.founded_year || "N/A",
          headquarters: `${selectedCompany.city || "N/A"}, ${selectedCompany.state || "N/A"}`,
          industry: selectedCompany.industry1 || "N/A",
          employees: selectedCompany.employee_grp || "N/A",
          revenue: selectedCompany.revenue_grp || "N/A",
          score: selectedCompany.timefund || "N/A",
          funding: selectedCompany.funding_round
            ? [
                {
                  round: selectedCompany.funding_round,
                  status: selectedCompany.funding_status || "N/A",
                  date: selectedCompany.last_funding_date || "N/A",
                  type: selectedCompany.last_funding_type || "N/A",
                  totalAmount: selectedCompany.total_funding_amt || "N/A",
                },
              ]
            : [],
          report: "Provide a meaningful report based on company data",
          similarCompanies: ["Company A", "Company B", "Company C"], // Replace with actual similar companies if available
          company_id: selectedCompany.value, // Ensure this field is included
        });
      }
    }
  }, [selected, companies]);

  return (
    <div className="p-4">
      <Search options={companies} setSelected={setSelected} />
      <h1 className="mt-4 text-3xl font-bold">
        {companyData?.name || "Select a Company"}
      </h1>
      <p className="mt-5">{companyData?.description}</p>
```

Figure 6.22: Code Snippet of Company Statistics Page contd.
Code Snippet depicting the Company Statistics page of the web application.

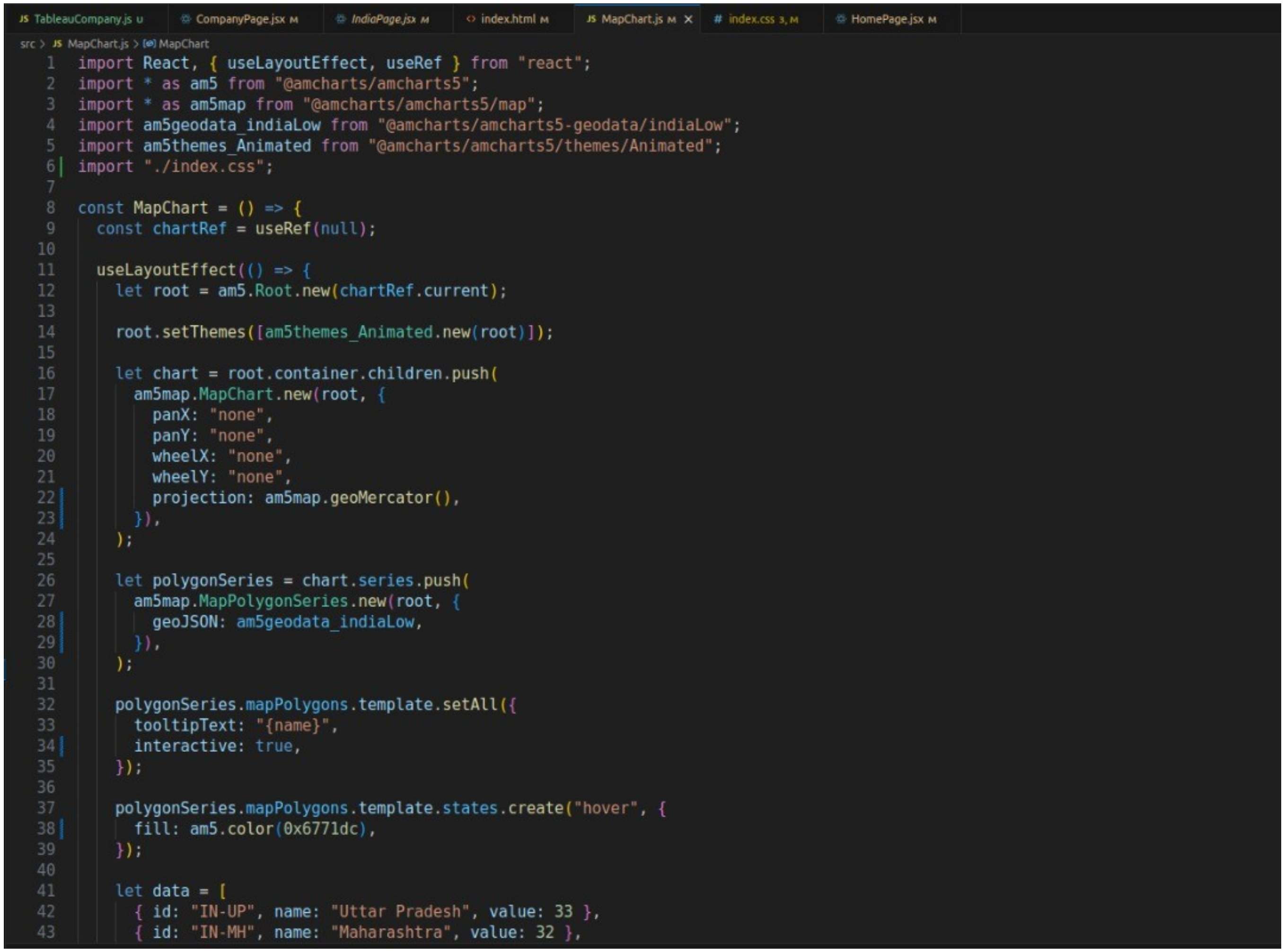


```
const CompanyPage = () => {
      {companyData && (
        <>
          <div className="mt-4">
            <h2 className="text-2xl font-semibold">Company Insights</h2>
            <div className="grid grid-cols-2 gap-4 mt-2">
              <button className="🟦bg-blue-500 ⬜text-white px-4 py-2 rounded">
                Founded: {companyData.founded}
              </button>
              <button className="🟦bg-blue-500 ⬜text-white px-4 py-2 rounded">
                Headquarters: {companyData.headquarters}
              </button>
              <button className="🟦bg-blue-500 ⬜text-white px-4 py-2 rounded">
                Industry: {companyData.industry}
              </button>
              <button className="🟦bg-blue-500 ⬜text-white px-4 py-2 rounded">
                Employees: {companyData.employees}
              </button>
              <button className="🟦bg-blue-500 ⬜text-white px-4 py-2 rounded">
                Revenue: {companyData.revenue} Crore Rs
              </button>
              <button className="🟦bg-blue-500 ⬜text-white px-4 py-2 rounded">
                Score: {companyData.score}
              </button>
            </div>
          </div>

          <div className="mt-4">
            <h2 className="text-2xl font-semibold">Funding</h2>
            {/* Pass companyData.company_id to TableauCompany */}
            <TableauCompany companyIdParameter={companyData.company_id} />
          </div>

          <div className="mt-4">
            <h2 className="text-2xl font-semibold">Report</h2>
            <p>
              24 Mantra Organic is a leading organic food brand in India, known
              for its commitment to promoting sustainable agriculture and
              healthy living.
            </p>

            <p>{companyData.report || "Report not available"}</p>
          </div>
```

```
webpack compiled successfully
```

Figure 6.23: Code Snippet of Company Statistics Page contd.
Code Snippet depicting the Company Statistics page of the web application.

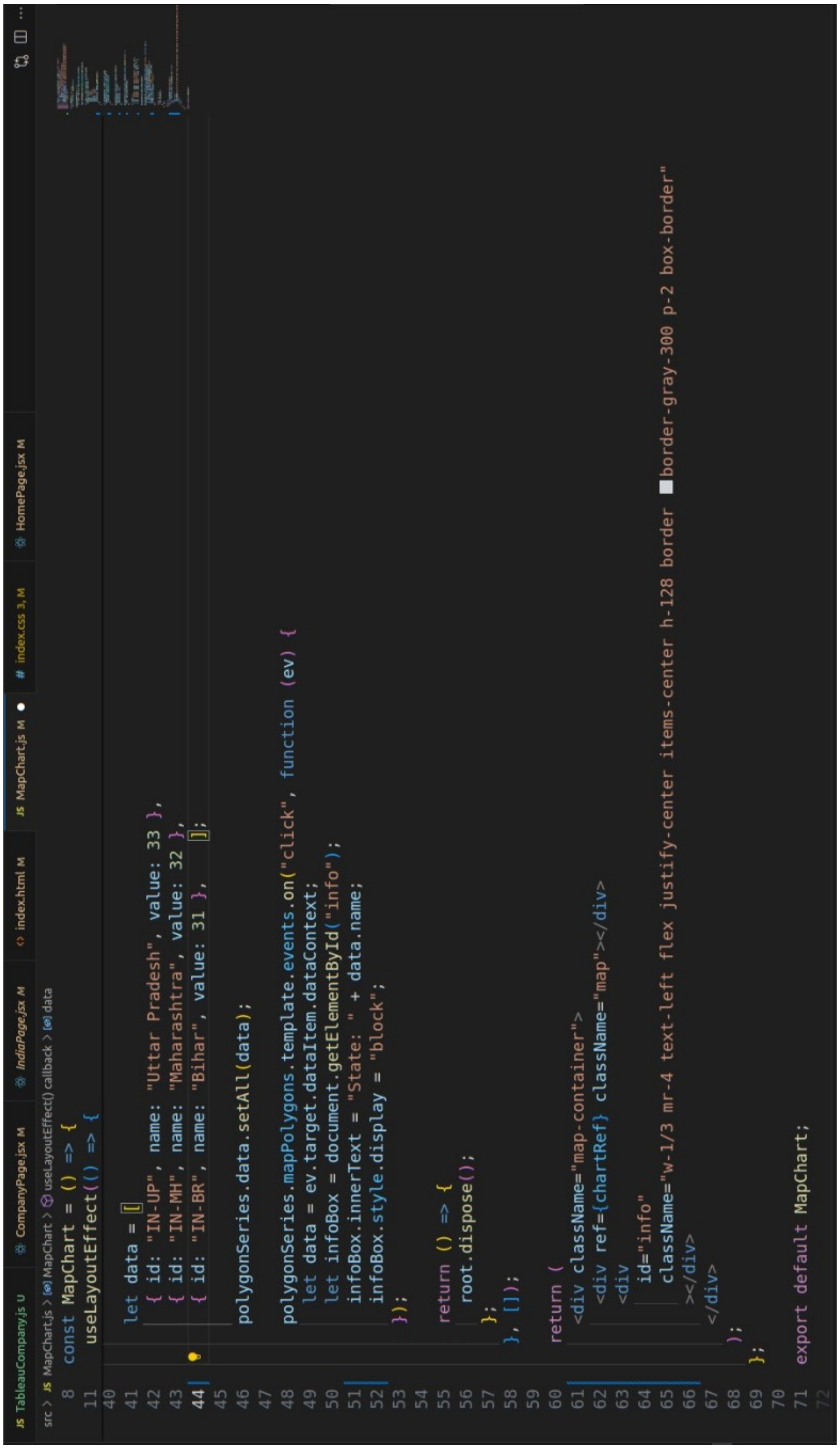
The Economic Overview or India page presents a dynamic map of India, accompanied by comprehensive graphs detailing key economic statistics. This page showcases essential data on funding, revenues, and other economic indicators, providing a clear visualization of the country's financial landscape. The interactive map allows users to explore regional economic activities, while the graphs offer insights into national trends and sectoral performance.



```javascript
import React, { useLayoutEffect, useRef } from "react";
import * as am5 from "@amcharts/amcharts5";
import * as am5map from "@amcharts/amcharts5/map";
import am5geodata_indiaLow from "@amcharts/amcharts5-geodata/indiaLow";
import am5themes_Animated from "@amcharts/amcharts5/themes/Animated";
import "./index.css";

const MapChart = () => {
  const chartRef = useRef(null);

  useLayoutEffect(() => {
    let root = am5.Root.new(chartRef.current);

    root.setThemes([am5themes_Animated.new(root)]);

    let chart = root.container.children.push(
      am5map.MapChart.new(root, {
        panX: "none",
        panY: "none",
        wheelX: "none",
        wheelY: "none",
        projection: am5map.geoMercator(),
      }),
    );

    let polygonSeries = chart.series.push(
      am5map.MapPolygonSeries.new(root, {
        geoJSON: am5geodata_indiaLow,
      }),
    );

    polygonSeries.mapPolygons.template.setAll({
      tooltipText: "{name}",
      interactive: true,
    });

    polygonSeries.mapPolygons.template.states.create("hover", {
      fill: am5.color(0x6771dc),
    });

    let data = [
      { id: "IN-UP", name: "Uttar Pradesh", value: 33 },
      { id: "IN-MH", name: "Maharashtra", value: 32 },
```

Figure 6.24: Code Snippet of India Statistics Page
Code Snippet depicting the Economic Overview of India page of the web application.

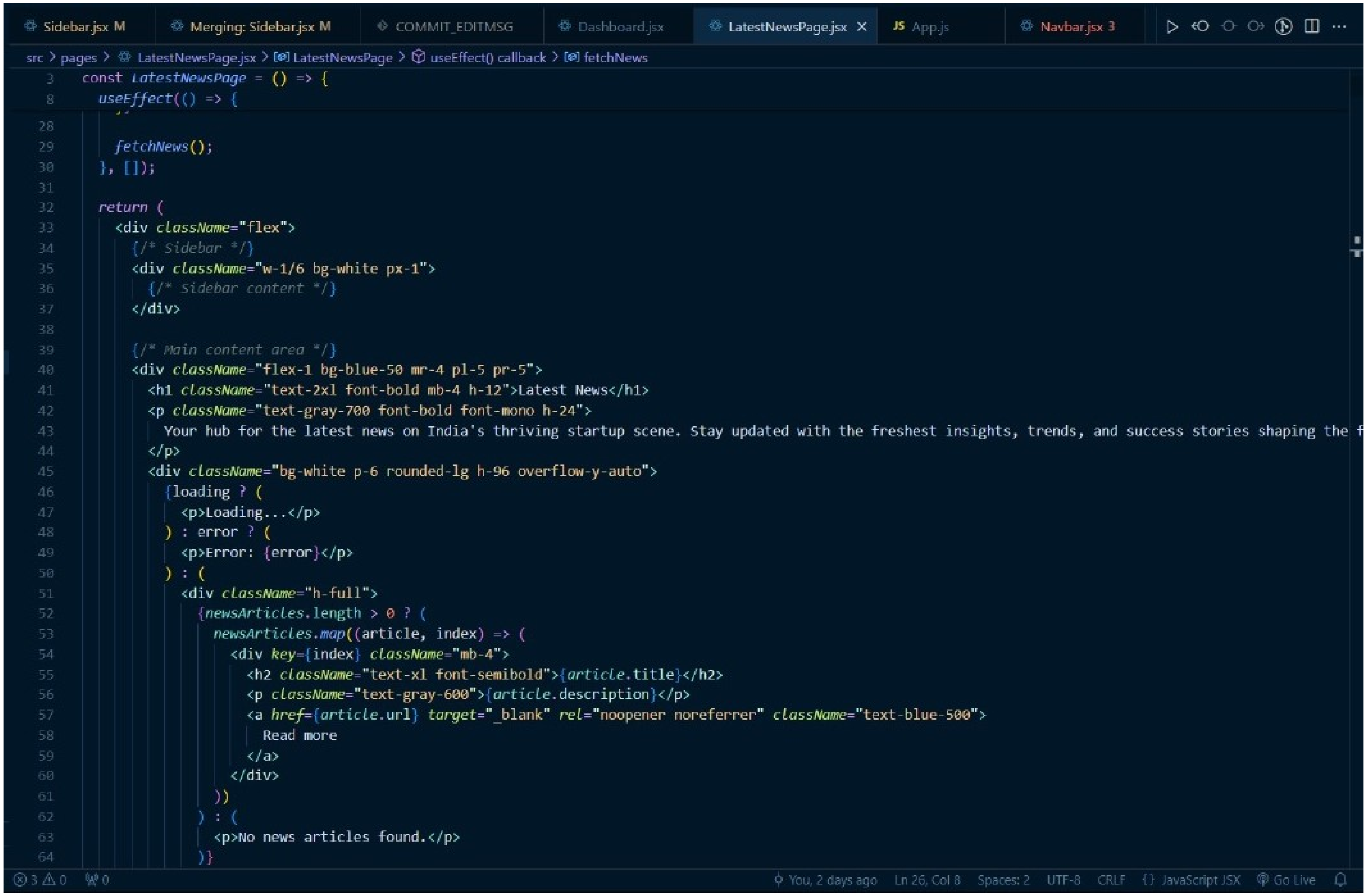
The Interactive Economic Map page features a dynamic map of India, allowing users to click on individual states to reveal detailed economic data and charts specific to that region. This interactive functionality enables a granular view of key metrics such as funding, revenue, and industry performance for each state.



```
const MapChart = () => {
  useLayoutEffect(() => {
    let data = [
      { id: "IN-UP", name: "Uttar Pradesh", value: 33 },
      { id: "IN-MH", name: "Maharashtra", value: 32 },
      { id: "IN-BR", name: "Bihar", value: 31 },
    ];

    polygonSeries.data.setAll(data);

    polygonSeries.mapPolygons.template.events.on("click", function (ev) {
      let data = ev.target.dataItem.dataContext;
      let infoBox = document.getElementById("info");
      infoBox.innerText = "State: " + data.name;
      infoBox.style.display = "block";
    });

    return () => {
      root.dispose();
    };
  }, []);

  return (
    <div className="map-container">
      <div ref={chartRef} className="map"></div>
      <div
        id="info"
        className="w-1/3 mr-4 text-left flex justify-center items-center h-128 border border-gray-300 p-2 box-border"
      ></div>
    </div>
  );
};

export default MapChart;
```

Figure 6.25: Code Snippet of State Statistics Page
Code Snippet depicting the dynamic Economic Overview of states in India page of the web application.

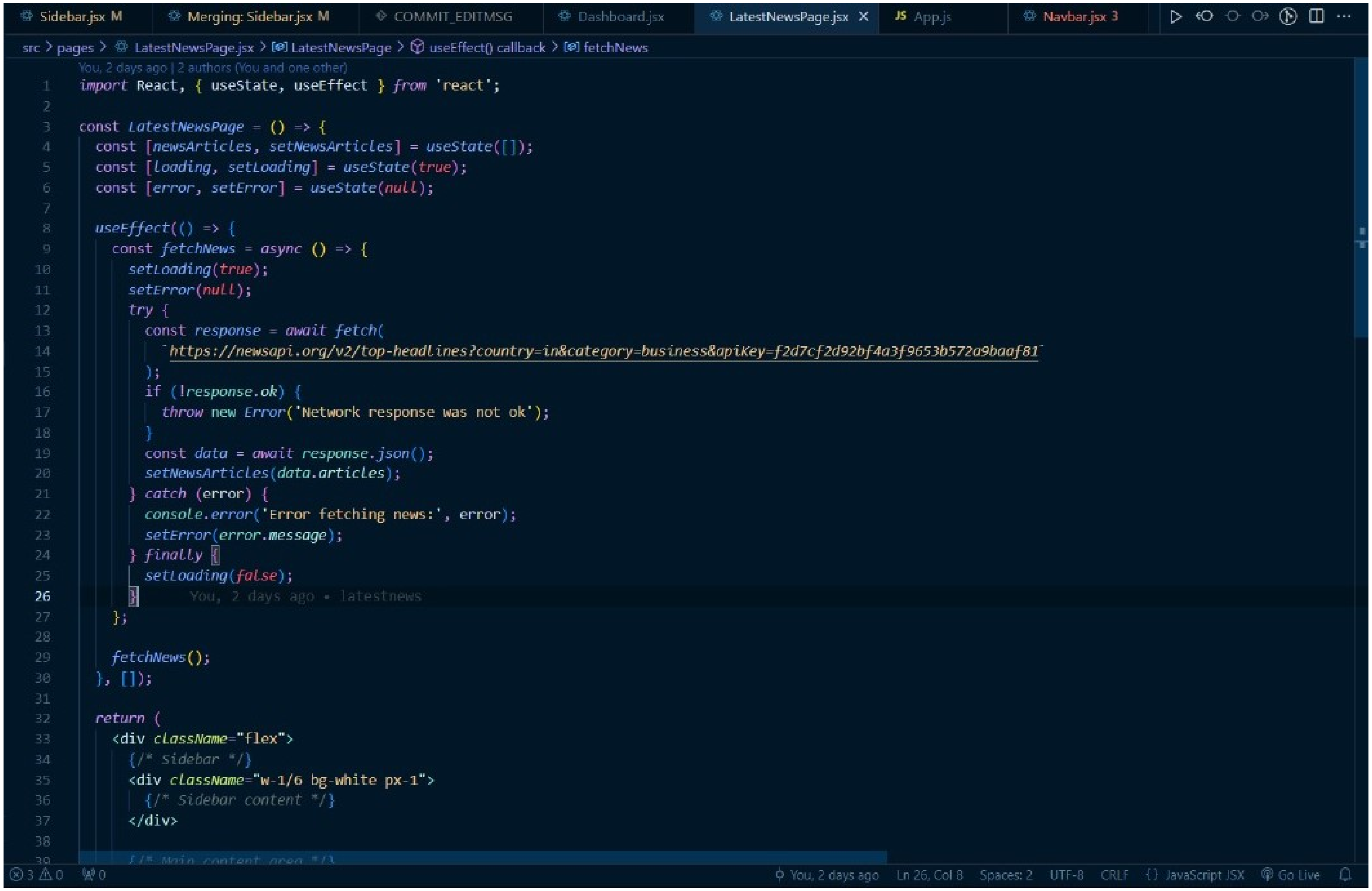
The Latest Business News page offers real-time updates on the most significant business developments in India. This page features curated articles, breaking news, and in-depth analyses covering a wide range of topics, including market trends, corporate earnings, policy changes, and industry advancements.



Sidebar.jsx M
Merging: Sidebar.jsx M
COMMIT_EDITMSG
Dashboard.jsx
LatestNewsPage.jsx
JS App.js
Navbar.jsx 3
src > pages > LatestNewsPage.jsx > LatestNewsPage > useEffect() callback > fetchNews
⊗ 3 ⚠ 0 ⚡0
You, 2 days ago Ln 26, Col 8 Spaces: 2 UTF-8 CRLF {} JavaScript JSX Go Live

```
const LatestNewsPage = () => {
  useEffect(() => {

    fetchNews();
  }, []);

  return (
    <div className="flex">
      {/* Sidebar */}
      <div className="w-1/6 bg-white px-1">
        {/* Sidebar content */}
      </div>

      {/* Main content area */}
      <div className="flex-1 bg-blue-50 mr-4 pl-5 pr-5">
        <h1 className="text-2xl font-bold mb-4 h-12">Latest News</h1>
        <p className="text-gray-700 font-bold font-mono h-24">
          Your hub for the latest news on India's thriving startup scene. Stay updated with the freshest insights, trends, and success stories shaping the f
        </p>
        <div className="bg-white p-6 rounded-lg h-96 overflow-y-auto">
          {loading ? (
            <p>Loading...</p>
          ) : error ? (
            <p>Error: {error}</p>
          ) : (
            <div className="h-full">
              {newsArticles.length > 0 ? (
                newsArticles.map((article, index) => (
                  <div key={index} className="mb-4">
                    <h2 className="text-xl font-semibold">{article.title}</h2>
                    <p className="text-gray-600">{article.description}</p>
                    <a href={article.url} target="_blank" rel="noopener noreferrer" className="text-blue-500">
                      Read more
                    </a>
                  </div>
                ))
              ) : (
                <p>No news articles found.</p>
              )}
```

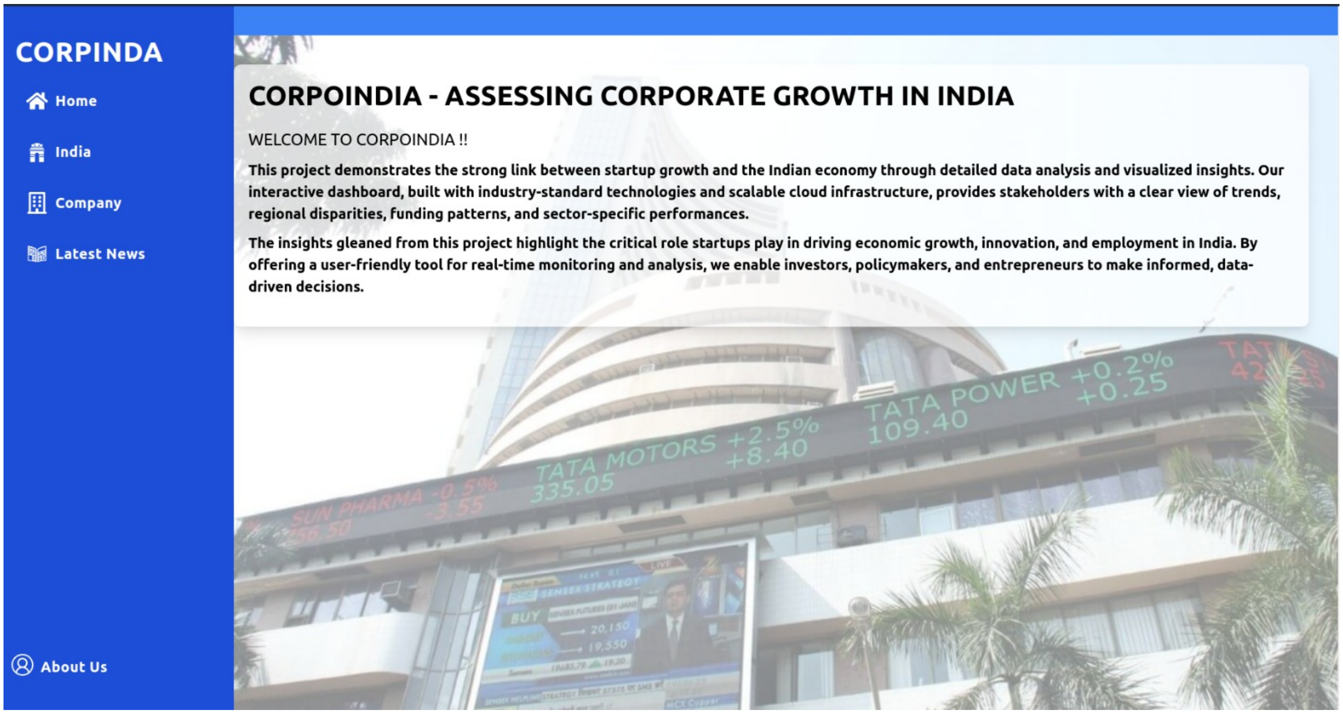
Figure 6.26: Code Snippet of News Feed Page
Code Snippet depicting the News Feed page of the web application.



```jsx
import React, { useState, useEffect } from 'react';

const LatestNewsPage = () => {
  const [newsArticles, setNewsArticles] = useState([]);
  const [loading, setLoading] = useState(true);
  const [error, setError] = useState(null);

  useEffect(() => {
    const fetchNews = async () => {
      setLoading(true);
      setError(null);
      try {
        const response = await fetch(
          `https://newsapi.org/v2/top-headlines?country=in&category=business&apiKey=f2d7cf2d92bf4a3f9653b572a9baaf81`
        );
        if (!response.ok) {
          throw new Error('Network response was not ok');
        }
        const data = await response.json();
        setNewsArticles(data.articles);
      } catch (error) {
        console.error('Error fetching news:', error);
        setError(error.message);
      } finally {
        setLoading(false);
      }      You, 2 days ago • latestnews
    };

    fetchNews();
  }, []);

  return (
    <div className="flex">
      {/* Sidebar */}
      <div className="w-1/6 bg-white px-1">
        {/* Sidebar content */}
      </div>

      {/* Main content area */}
```

Figure 6.27: Code Snippet of News Feed Page contd.
Code Snippet depicting the News Feed page of the web application.

## 6.6 Screenshots



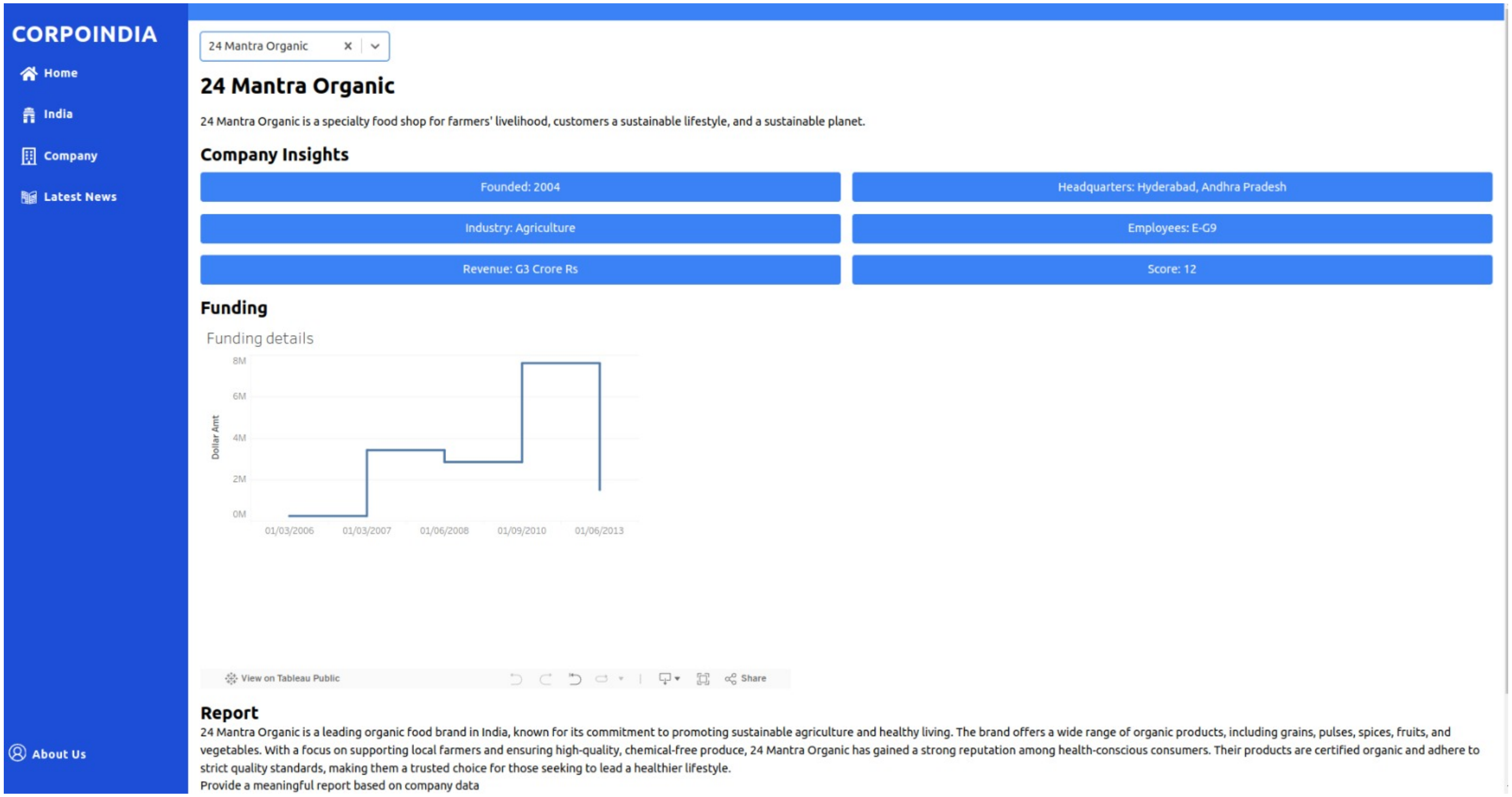
CORPOINDIA - ASSESSING CORPORATE GROWTH IN INDIA

WELCOME TO CORPOINDIA !!

This project demonstrates the strong link between startup growth and the Indian economy through detailed data analysis and visualized insights. Our interactive dashboard, built with industry-standard technologies and scalable cloud infrastructure, provides stakeholders with a clear view of trends, regional disparities, funding patterns, and sector-specific performances.

The insights gleaned from this project highlight the critical role startups play in driving economic growth, innovation, and employment in India. By offering a user-friendly tool for real-time monitoring and analysis, we enable investors, policymakers, and entrepreneurs to make informed, data-driven decisions.

Figure 6.28: Home page
Home page detailing a comprehensive yet concise summary of the project's scope and objectives.



## 24 Mantra Organic

24 Mantra Organic is a specialty food shop for farmers' livelihood, customers a sustainable lifestyle, and a sustainable planet.

### Company Insights

Founded: 2004

Headquarters: Hyderabad, Andhra Pradesh

Industry: Agriculture

Employees: E-G9

Revenue: G3 Crore Rs

Score: 12

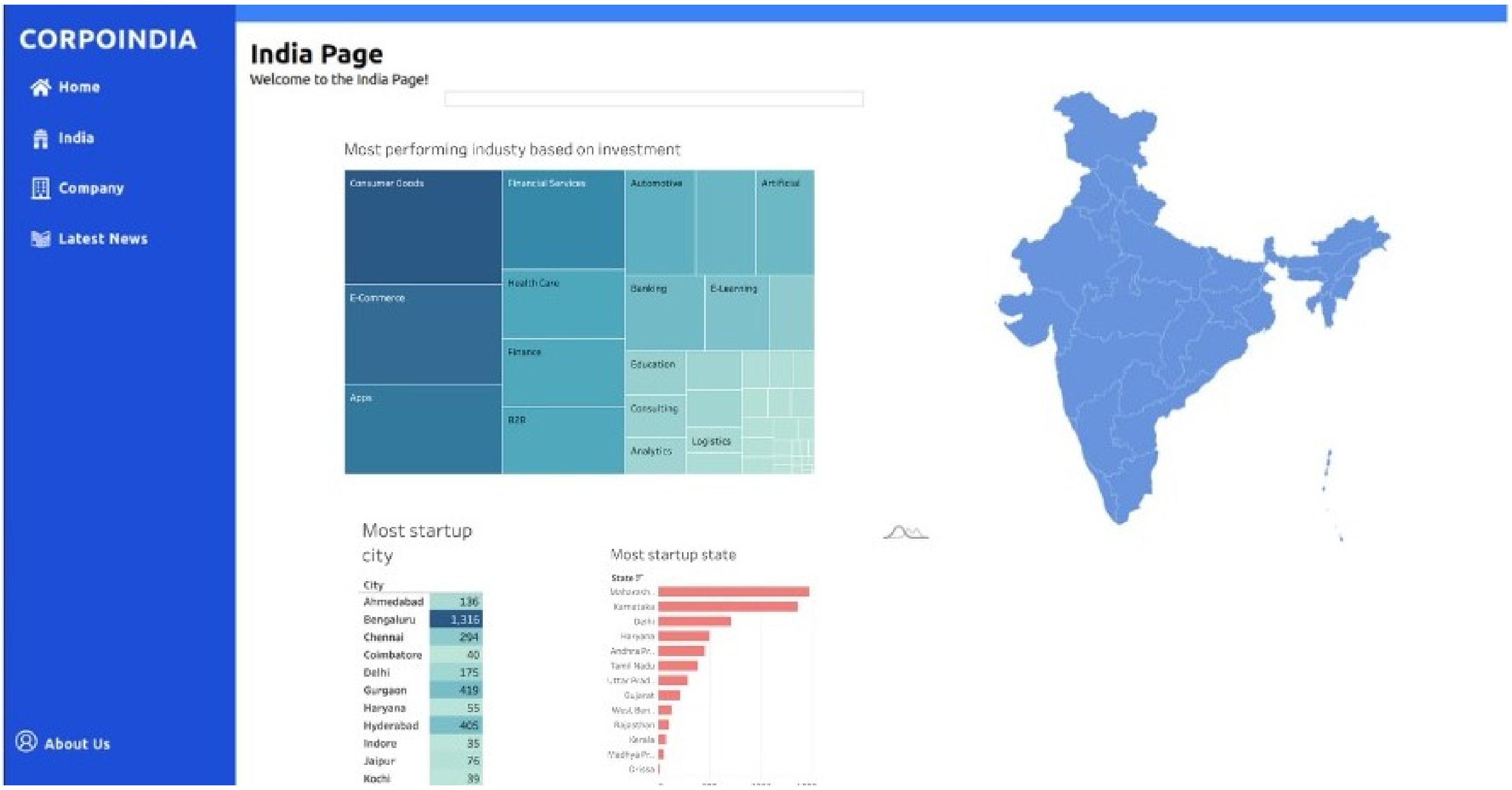
### Funding

### Report

24 Mantra Organic is a leading organic food brand in India, known for its commitment to promoting sustainable agriculture and healthy living. The brand offers a wide range of organic products, including grains, pulses, spices, fruits, and vegetables. With a focus on supporting local farmers and ensuring high-quality, chemical-free produce, 24 Mantra Organic has gained a strong reputation among health-conscious consumers. Their products are certified organic and adhere to strict quality standards, making them a trusted choice for those seeking to lead a healthier lifestyle.

Provide a meaningful report based on company data

Figure 6.29: Company Statistics Page



India Page

Welcome to the India Page!

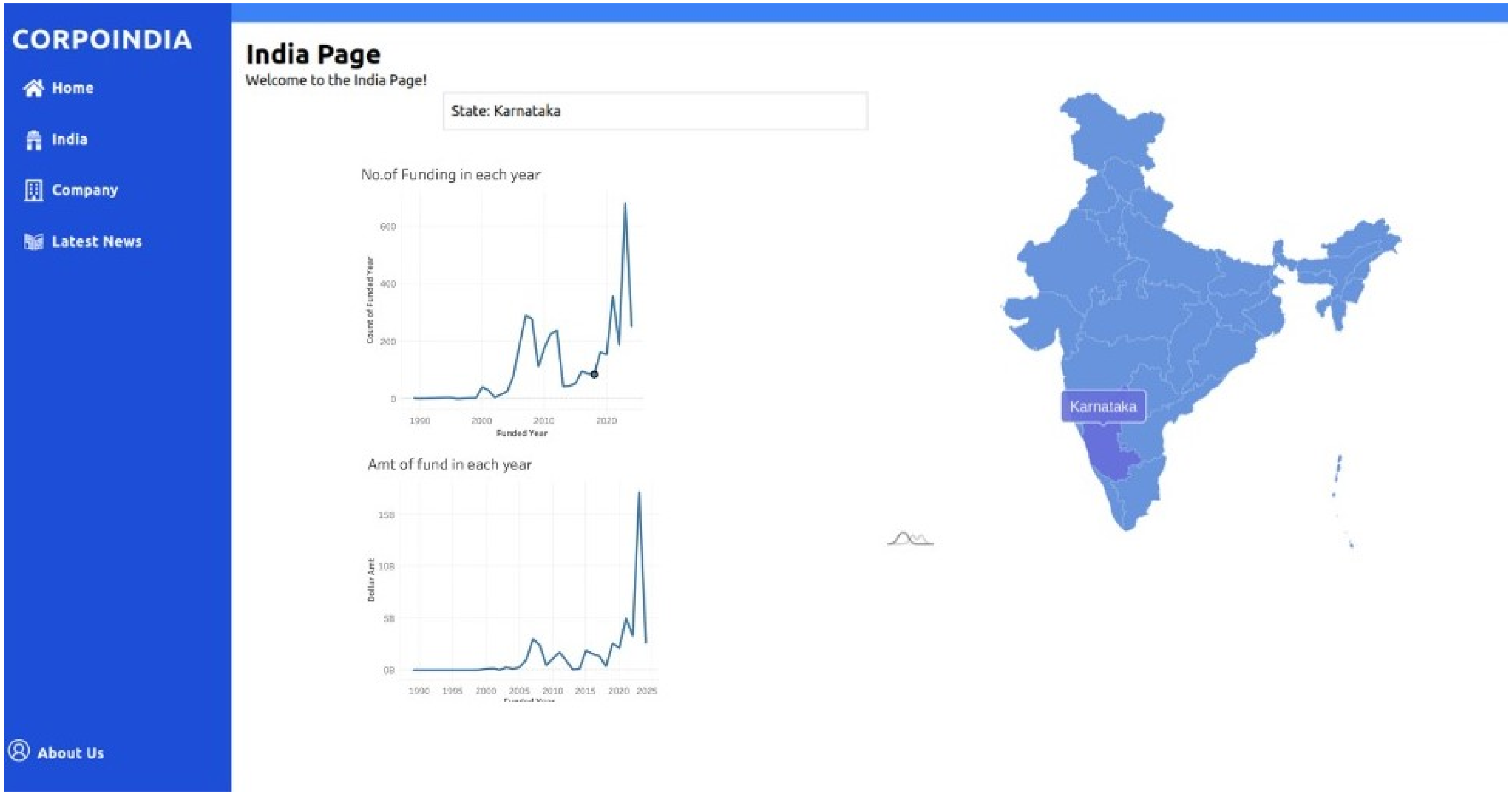
Most performing industy based on investment

Most startup city

| City | |
|---|---|
| Ahmedabad | 136 |
| Bengaluru | 1,316 |
| Chennai | 294 |
| Coimbatore | 40 |
| Delhi | 175 |
| Gurgaon | 419 |
| Haryana | 55 |
| Hyderabad | 405 |
| Indore | 35 |
| Jaipur | 76 |
| Kochi | 89 |

Most startup state

Figure 6.30: India Statistics Page

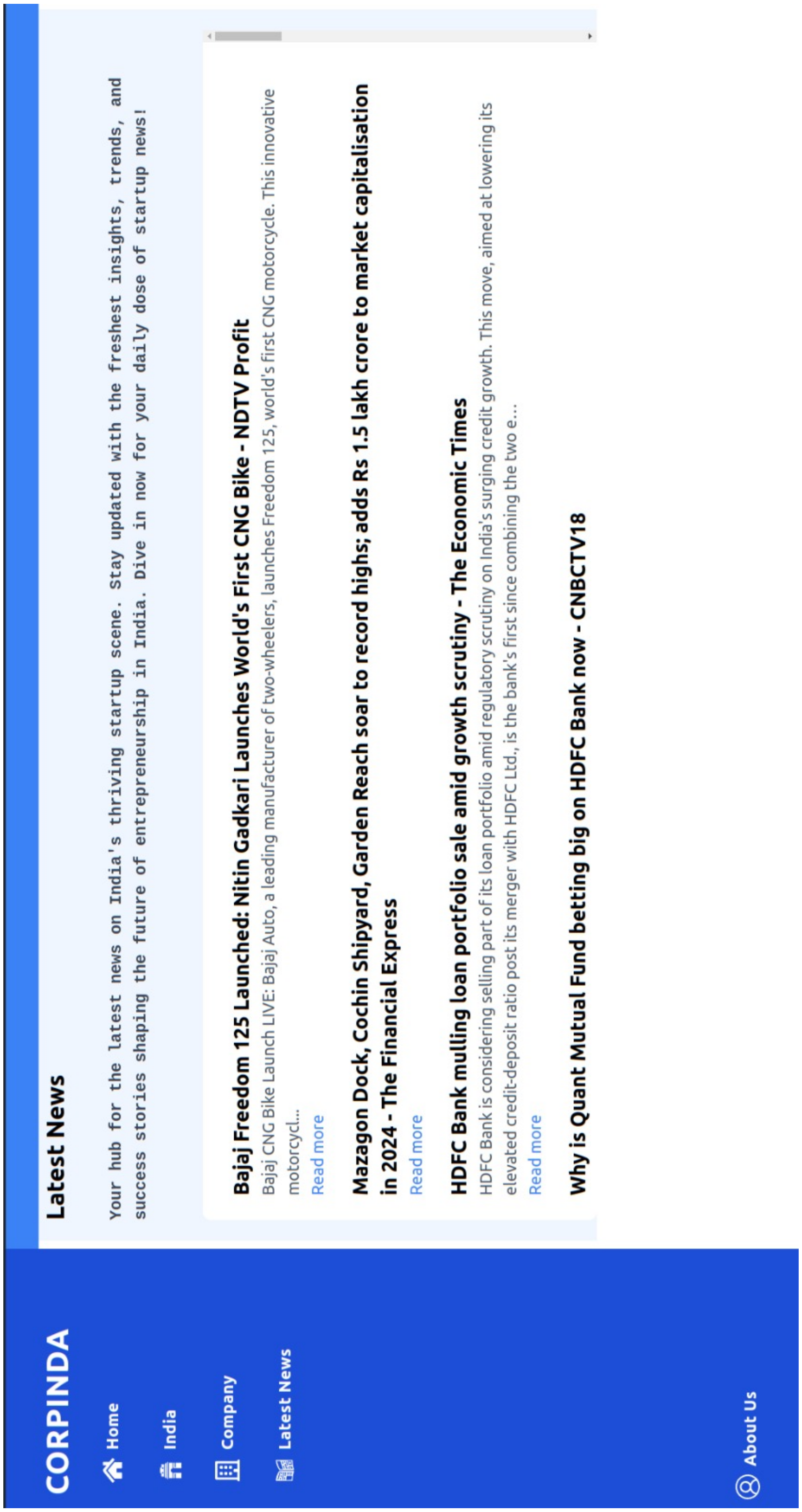


India Page

Welcome to the India Page!

State: Karnataka

Figure 6.31: State Statistics Page



## Latest News

Your hub for the latest news on India's thriving startup scene. Stay updated with the freshest insights, trends, and success stories shaping the future of entrepreneurship in India. Dive in now for your daily dose of startup news!

**Bajaj Freedom 125 Launched: Nitin Gadkari Launches World's First CNG Bike - NDTV Profit**

Bajaj CNG Bike Launch LIVE: Bajaj Auto, a leading manufacturer of two-wheelers, launches Freedom 125, world's first CNG motorcycle. This innovative motorcycl...

Read more

**Mazagon Dock, Cochin Shipyard, Garden Reach soar to record highs; adds Rs 1.5 lakh crore to market capitalisation in 2024 - The Financial Express**

Read more

**HDFC Bank mulling loan portfolio sale amid growth scrutiny - The Economic Times**

HDFC Bank is considering selling part of its loan portfolio amid regulatory scrutiny on India's surging credit growth. This move, aimed at lowering its elevated credit-deposit ratio post its merger with HDFC Ltd., is the bank's first since combining the two e...

Read more

**Why is Quant Mutual Fund betting big on HDFC Bank now - CNBCTV18**

Figure 6.32: News feed page
Utilizing the capabilities of the NewsFeed API, we deliver the latest news and updates on business and economic developments in India.

# Chapter 7

# Result

The project succeeded in bringing a comprehensive website for the general populace to provide quality data analysis report on the economic landscape of India. Interactive graphs, easy to understand metrics and overall simplicity of the dashboard makes it a must have for analysts and investors hoping to learn more about the status of funding in companies and start ups across India.

User can successfully view the metrics of different states as well as the general statistics of India about different industries. The client can also view an indepth visualised report on each company under study.

# Chapter 8

# Future Scope

The future scope of this project for the general community can encompass several potential enhancements and expansions.

1. Personalised Dashboard: By introducing personalised dashboard with a user login and signup system, we can curate the report for each registered user. This includes features like a watchlist with the user's choice of companies

2. More accurate Ranking system: The scope of our project does not allow us to conduct extensive research to validate our index formula. Proper analysis and deduction of factors that directly and indirectly effect the performance of a company and their inclusion in the formula can increase the significance and value of the ranking system.

3. Expanding companies in each city: With more time and integration of more tools, we can implement features like listing the companies registered in a state when selected by the user

4. Pay-as-you-use: As mentioned before, the lack of a proper database managed by a central governing body that readily provides information on companies means our report must refer to data hidden behind pay to use software. By working with the government to establish such a central repository, we can reduce the operational costs of running this dashboard further.

5. Mobile App Enhancements: A mobile version of this dashboard can be

developed so that the user can avail our services even on the move.

7. Accessibility and Inclusivity: Focus on improving accessibility features to ensure that users with disabilities can access and navigate the platform effectively. Consider features such as screen reader compatibility, color contrast options, and keyboard navigation.

# Chapter 9

# Conclusion

This project demonstrates the strong link between startup growth and the Indian economy through detailed data analysis and visualized insights. Our interactive dashboard, built with industry-standard technologies and scalable cloud infrastructure, provides stakeholders with a clear view of trends, regional disparities, funding patterns, and sector-specific performances.

The insights gleaned from this project highlight the critical role startups play in driving economic growth, innovation, and employment in India. By offering a user-friendly tool for real-time monitoring and analysis, we enable investors, policymakers, and entrepreneurs to make informed, data-driven decisions.

In summary, this project equips stakeholders with essential intelligence to support the sustainable growth of India's startup ecosystem, contributing to the nation's economic development. The use of cloud technology ensures that the solution is scalable to meet increasing demand, ensuring robust performance and flexibility.

# Chapter 10

# References

Books & Papers:

1. Big Data: Concepts, Technology and Architecture, Balamarugan Balusamy, Nandhini Abirami R, Seifedine Kadry and Amir Gandomi.
2. Data Analytics Made Accessible: 2020 Edition, Anil K. Maheshwari, Ph.D.
3. Data Lakehouse in Action: Architecting a Modern and Scalable Data Analytics Platform, Pradeep Menon
4. Fundamentals of software engineering, Rajib Mall, Pearson Education, 2011.
5. Kukreja, Megha & Makhija, Priya. (2023). Startups and their contribution towards the growth of Indian Economy. International Scientific Journal of Engineering and Management. 02. 10.55041/ISJEM00407.
6. Maradi, Mallikarjun. (2023). Growth of Indian start-up: A critical Analysis. 17. 180-186.
7. Risbud, Mrudula & Waghmare, Rahul. (2023). Sustainability Through Innovation: The Case of Indian Startup Thaely. 10.4018/978-1-6684-6123-5.ch011.

Web Links :

1. Understanding Startup Metrics - https://foundersnetwork.com/blog/startup-metrics/
2. Essential financial KPIS for startup success - https://www.linkedin.com/pulse/11-essential-financial-kpis-startup-success-marshall-hargrave/
3. IEEE Xplore - https://ieeexplore.ieee.org/Xplore/home.jsp