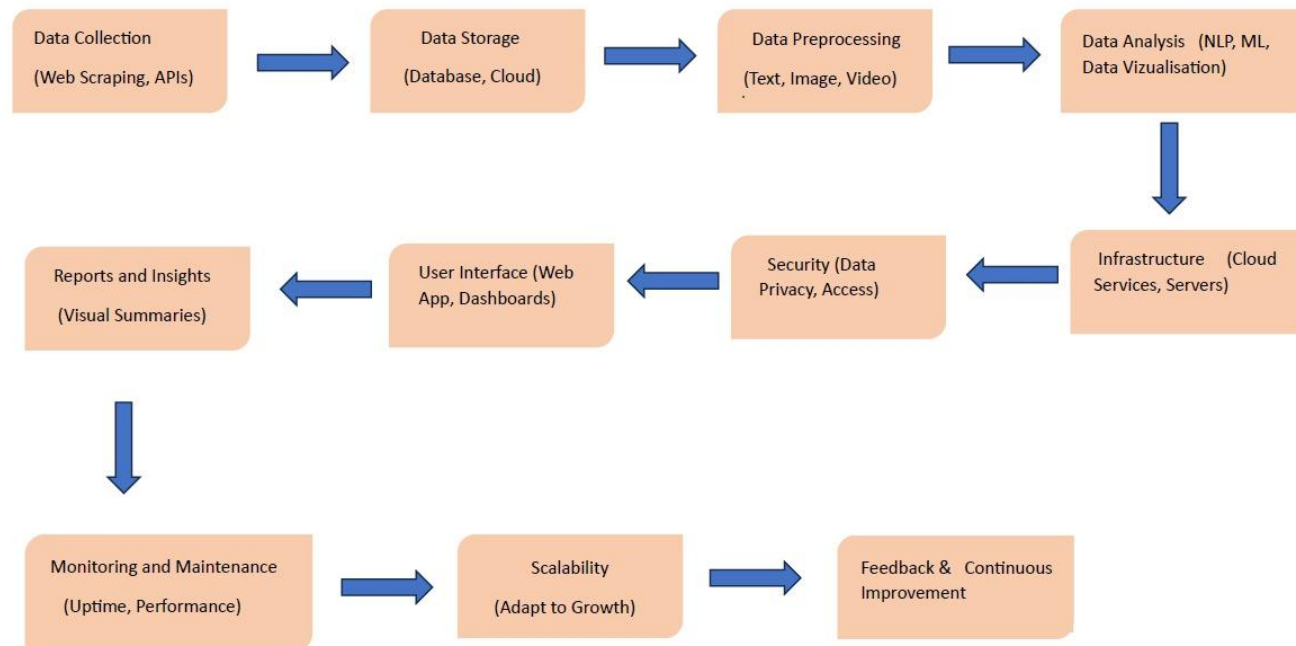


## Project Design Phase-II Technology Stack (Architecture & Stack)

Date	24 October 2023
Team ID	Team-591216
Project Name	Dissecting the Digital Landscape: A Comprehensive Analysis of Social Media
Maximum Marks	4 Marks

### Technical Architecture:



**Table-1 : Components & Technologies:**

S.No	Component	Description	Technology
1.	Data Collection	Gather data from social media platforms and other online sources.	Web scraping (Beautiful Soup, Scrapy), Social Media APIs (Twitter API, Facebook Graph API), Selenium for web automation.
2.	Data Storage	Store collected data for analysis and future reference.	Cloud-based Data Warehousing (Amazon Redshift, Google BigQuery), NoSQL Databases (MongoDB, Cassandra)
3.	Data Preprocessing and Analysis	Prepare and analyze data for insights and patterns.	Natural Language Processing (NLP): NLTK, Hugging Face Transformers.
4.	Sentiment Analysis	Analyze the emotional tone of text data	NLP libraries for sentiment analysis (VADER, TextBlob, AFINN).
5.	Machine Learning and AI	Utilize machine learning algorithms to identify trends and patterns.	Machine learning models for classification, regression, and clustering. AutoML tools (Google AutoML, H2O.ai) for automated model building.
6.	Real-time Processing	Monitor social media trends and events as they occur in real-time.	Real-time dashboards using tools like IBM
7.	Data Visualization	Create visual representations of data for exploration and presentation.	Data visualization tools (IBM)
8.	Security and Compliance	Ensure data privacy and compliance with regulations.	Encryption, access control, and compliance tools (e.g., GDPR compliance solutions).
9.	Scalability and Redundancy	Ensure the system can handle increased data loads and maintain high availability	Cloud-based infrastructure on AWS, Azure, or GCP for scalability. Redundancy through load balancers and failover mechanisms.
10.	Cloud Deployment	Host the entire system on cloud infrastructure for flexibility and scalability.	Cloud platforms like AWS, Azure, GCP. Utilize cloud services for computing, storage, and data processing.
11.	User Interface	Ensure responsible handling of social media data and user privacy	Privacy tools and policies, anonymization techniques, and ethical data practices.

**Table-2: Application Characteristics:**

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Utilizing open-source frameworks can significantly reduce development costs and time, as well as provide a community of contributors for ongoing support and improvements. These frameworks can include libraries and tools for data processing, machine learning, and data visualization.	Some popular open-source frameworks include TensorFlow and PyTorch for machine learning, Scikit-Learn for data analysis, and D3.js for data visualization.
2.	Security Implementations	Ensuring the security of user data and application resources is paramount. Robust security implementations include encryption of data at rest and in transit, access controls, authentication mechanisms, and regular security audits.	Use technologies like SSL/TLS for encryption, OAuth for user authentication, and security libraries like OWASP for vulnerability scanning.
3.	Scalable Architecture	A scalable architecture is crucial for accommodating the growing volume of social media data and users. It involves designing the application to distribute workloads, automatically adjust resources, and handle increased traffic without compromising performance.	Implement microservices architecture, containerization (e.g., Docker and Kubernetes), and use cloud-based infrastructure to ensure scalability.
4.	Availability	Maintaining high availability ensures that the application is accessible to users with minimal downtime. This involves redundancy, load balancing, and disaster recovery mechanisms.	Employ technologies like load balancers, failover systems, and content delivery networks (CDNs) to enhance availability.
5.	Performance	High performance is essential for processing and analyzing large volumes of social media data quickly. It involves optimizing code, data storage, and processing algorithms to achieve low latency and responsiveness.	Use performance monitoring tools, caching mechanisms, and code optimization techniques to improve application performance.