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

Date	12 June 2025
Team ID	LTVIP2025TMID51124
Project Name	Cosmetic Insights: Navigating cosmetics Trends And Consumer Insights with Tableau
Maximum Marks	4 Marks

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

Example: Cosmetic Insights: Navigating cosmetics Trends And Consumer Insights with Tableau

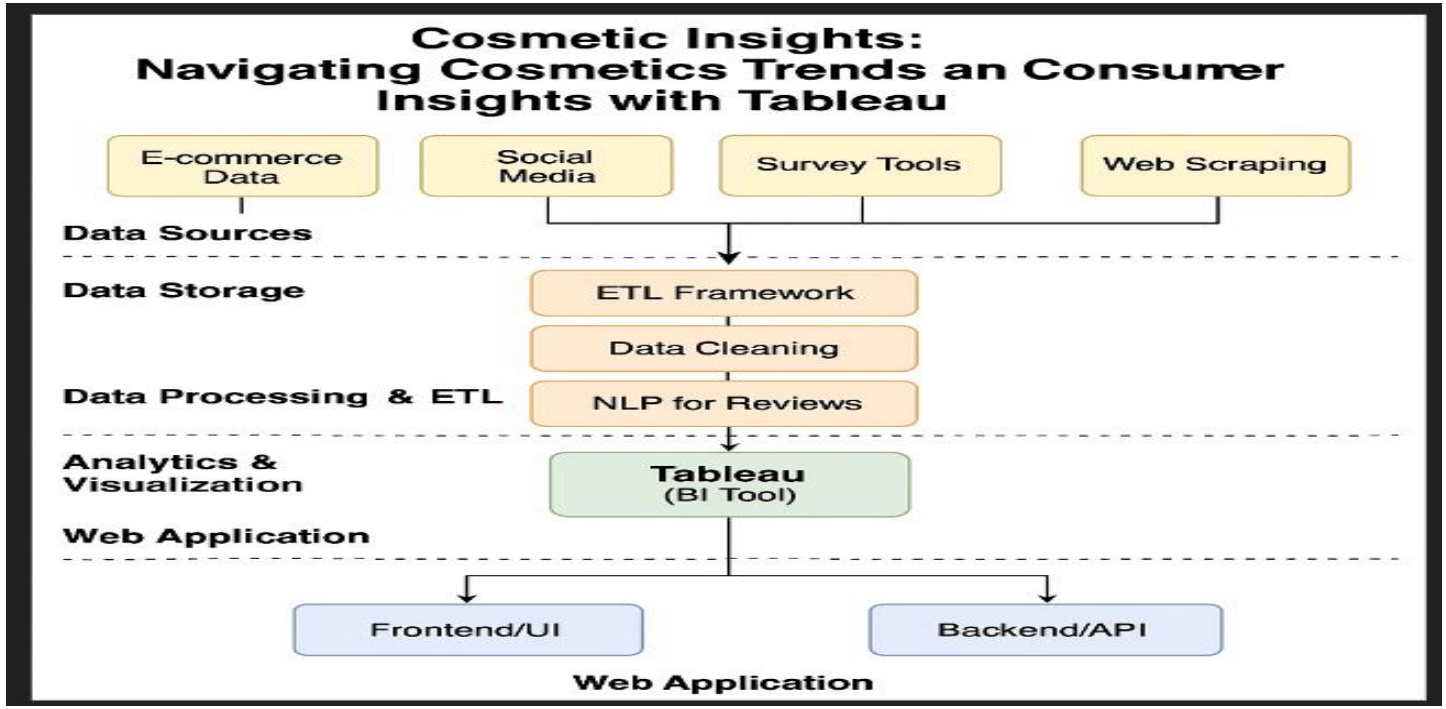

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How users interact with the application (trend viewers, feedback, insights)	HTML, CSS, JavaScript, React.js / Angular
2.	Application Logic-1	Logic for user input, product preferences, and feedback modules	Python / Node.js
3.	Application Logic-2	Data preparation and filtering for visualization	Python (Pandas, NumPy)
4.	Application Logic-3	Logic to integrate Tableau dashboards for visual analytics	Tableau JS API / Tableau Embedded Analytics
5.	Database	Stores consumer profiles, product details, reviews, and trend metrics	MySQL / MongoDB
6.	Cloud Database	Cloud-hosted version of the database for scalability and accessibility	Firebase Realtime Database / MongoDB Atlas
7.	File Storage	Stores user-uploaded files, product images, or generated reports	AWS S3 / Google Cloud Storage
8.	External API-1	Collect cosmetic product data and trends from external sources	Google Trends API / Beauty Product API (if available)
9.	External API-2	Analyze consumer sentiment from social platforms	Twitter API / Instagram Graph API
10.	Machine Learning Model	Predicts consumer trends or recommends cosmetic products based on user history	Scikit-learn / TensorFlow (Recommendation Engine or Trend Forecasting Model)
11.	Infrastructure (Server / Cloud)	Deployment of application and dashboards on cloud or local environments	AWS EC2 / Vercel / Heroku / Docker / Kubernetes

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology

1.	Open-Source Frameworks	Open-source frameworks for UI, backend, and data processing	React.js (UI), Node.js (backend), Python (Pandas/NumPy), MongoDB (NoSQL DB)
2.	Security Implementations	Secure access, authentication, and data protection	JWT Authentication, OAuth 2.0, HTTPS, Firebase Auth, OWASP Guidelines.
3.	Scalable Architecture	Application follows a modular and scalable design	3-Tier Architecture (Presentation, Logic, Data), Microservices (for APIs)
4.	Availability	Ensures the system remains available with minimal downtime	AWS Elastic Load Balancer, Multi-Zone Deployment, Auto-Scaling (EC2 / Vercel)
5.	Performance	Fast loading, dashboard responsiveness, and real-time analytics	CDN (Cloudflare), Redis Caching, Tableau Hyper Engine for Fast Rendering