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

Date	30 October 2023
Team ID	NM2023TMID06522
Project Name	Project - Competitive analysis of leading travel aggregators
Maximum Marks	4 Marks

Technical Architecture:

Example: <u>Competitive analysis of leading travel aggregators</u>

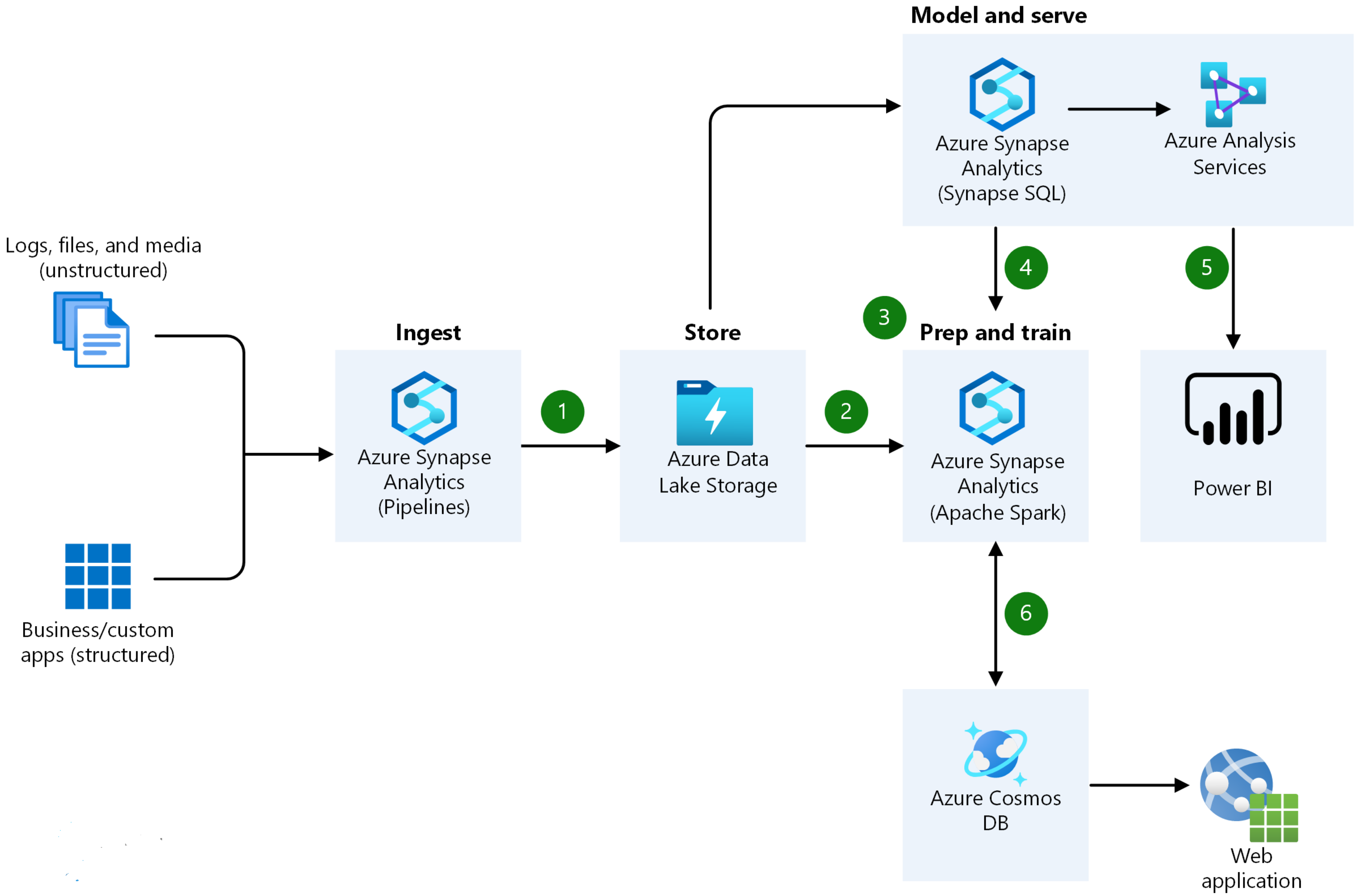
Model and serve
Azure Synapse Analytics (Synapse SQL)
Azure Analysis Services
Logs, files, and media (unstructured)
Business/custom apps (structured)
Ingest
Azure Synapse Analytics (Pipelines)
1
Store
Azure Data Lake Storage
2
3
Prep and train
Azure Synapse Analytics (Apache Spark)
4
5
Power BI
6
Azure Cosmos DB
Web application

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	Data Collection and Integration	Gathering data from various sources, including your competitors and the travel industry at large.	Web scraping tools (e.g., Python's Beautiful Soup, Scrapy) API integration for real-time data retrieval (e.g., RESTful APIs)
2.	Data Storage and Management	Storing and organizing the collected data for analysis.	Relational databases (e.g., MySQL, PostgreSQL) NoSQL databases (e.g., MongoDB, Cassandra) Data warehouses (e.g., Amazon Redshift, Google BigQuery)
3.	Data Cleaning and Preprocessing	Cleaning and preparing data for analysis, including handling missing values and data inconsistencies.	Data cleaning libraries (e.g., Python's pandas) Data transformation tools (e.g., Apache NiFi)
4.	Data Analysis and Visualization	Exploring and visualizing data to identify trends, patterns, and insights.	Data analysis tools (e.g., Python's Jupyter Notebook) Data visualization libraries (e.g., Matplotlib, Seaborn, Tableau)
5.	Competitor Profiling	Creating profiles for each competitor, including their strengths, weaknesses, and market positioning.	Text analysis tools (e.g., Natural Language Processing libraries in Python) Market research software (e.g., Statista, Nielsen)
6.	Market Segmentation	Dividing the target market into segments to identify opportunities and customer preferences.	Clustering algorithms (e.g., K-means, hierarchical clustering) Machine learning models (e.g., decision trees, random forests)
7.	Competitive Benchmarking	Measuring the performance of your company against competitors.	Key performance indicators (KPIs) tracking tools Business intelligence software (e.g., Tableau, Power BI)
8.	Predictive Analytics	Forecasting future market trends, demand, and competitor moves	Predictive modeling (e.g., regression, time series analysis) Machine learning algorithms (e.g., neural networks, XGBoost)
9.	Strategic Decision Support	Using data-driven insights to inform and support strategic decisions.	Decision support systems (DSS) Scenario analysis tools

10.	Reporting and Communication	Presenting the findings and recommendations to stakeholders.	Reporting tools (e.g., Microsoft Excel, Google Sheets) Presentation software (e.g., Microsoft PowerPoint)
11.	Data Security and Compliance	Ensuring the security and compliance of data handling and storage.	Data encryption and access control measures Compliance management tools (e.g., GDPR compliance software)
12.	Continuous Monitoring and Feedback	Continuously tracking competitors and market dynamics to adapt strategies.	Automated monitoring tools Feedback and survey platforms

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Leveraging open-source frameworks for the development of the analytics application, which can help reduce costs and foster collaboration.	Open-source programming languages (e.g., Python, R) Open-source data analytics and visualization libraries (e.g., Pandas, Matplotlib)
2.	Security Implementations	Ensuring the security of sensitive data and user information, especially in a domain where privacy is crucial.	Encryption technologies (e.g., TLS/SSL for data in transit, data-at-rest encryption) Secure authentication and authorization protocols
3.	Scalable Architecture	Designing an architecture that can accommodate growth and increased data processing needs.	Cloud-based infrastructure (e.g., AWS, Azure, Google Cloud) Microservices architecture

S.No	Characteristics	Description	Technology
4.	Availability	Ensuring that the application is available to users without significant downtime.	Load balancing solutions (e.g., HAProxy, NGINX) Redundancy and failover mechanisms
5.	Performance	Optimizing the application for high performance, especially when dealing with large datasets.	Performance monitoring tools (e.g., New Relic, Prometheus) Caching solutions (e.g., Redis, Memcached)

References:

https://c4model.com/

https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/

https://www.ibm.com/cloud/architecture

https://aws.amazon.com/architecture

https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d