BRD for Data Analytics Integration in Full-Stack Organic E-Commerce Website

1. Project Overview

This project will involve integrating a data analytics solution into the existing Full-Stack Organic E-Commerce Website. The goal is to use data analytics to gain insights into user behavior, sales trends, product performance, and customer demographics. The analytics will help the business make data-driven decisions to optimize marketing strategies, improve user experience, and increase revenue. The project will leverage **Python, NumPy, Pandas, Matplotlib, Seaborn, Tableau, and Power BI** to collect, process, analyze, and visualize data from the e-commerce platform.

2. Objective

The objective is to provide a complete analytics system for tracking sales performance, customer behavior, product popularity, and other KPIs. This system will offer both real-time dashboards and historical reporting, enabling the business to identify trends and improve operational efficiency.

3. Scope of Work

The data analytics system will focus on the following:

- **Data Collection:** Extract sales, user activity, and product data from the e-commerce platform.

- **Data Processing**: Clean and prepare the data for analysis using Python libraries like Pandas and NumPy.

- **Data Visualization**: Develop interactive dashboards and reports using Tableau and Power BI to showcase key metrics.

- **Reporting:** Provide detailed reports on user behavior, product performance, and sales trends over time.

- **Predictive Analytics:** Use statistical models to forecast future sales and trends.

4. Functional Requirements

The data analytics integration will focus on the following areas:

4.1 Data Collection

- User Data: Track user registrations, demographics, session duration, and interactions with the site (product views, searches, and purchases).

- Product Data: Monitor product page views, product popularity by category, conversion rates, and stock levels.

- Sales Data: Capture real-time sales data, order history, and revenue trends.

- Transaction Data: Analyze payment gateway information and purchase success rates (failed vs. successful transactions).

4.2 Data Processing

- Data Cleaning: Ensure collected data is free from duplicates, inconsistencies, and missing values using Pandas.

- Data Aggregation: Summarize user activities and sales data for reporting, including total sales, average order value, and purchase frequency.

4.3 Data Analysis & Visualization

- KPIs:

  - Sales Performance: Total sales, average order value, product-wise sales.

  - Customer Analysis: New vs. returning users, user demographics, and purchase patterns.

  - Product Analysis: Most popular products, categories, and price points.

  - Revenue Analysis: Total revenue, monthly growth, and revenue by product category.

- Tools:

  - Matplotlib/Seaborn: For generating static visual reports (graphs, bar charts, etc.).

  - Tableau/Power BI: For creating interactive dashboards that display real-time analytics and visual reports.

4.4 Reports & Dashboards

- Real-Time Dashboards:

  - User Activity Dashboard: Track the number of users online, products viewed, and actions taken (cart additions, purchases).

  - Sales Performance Dashboard: Display current sales, most-sold products, revenue trends, and conversion rates.

- Customizable Reports:

  - Daily/Weekly Sales Reports: Summary of transactions and revenue trends.

  - Customer Segmentation Reports: Group users by behavior, demographics, and purchase history.

  - Inventory Analysis Reports: Track stock levels, product turnover rates, and demand prediction.

5. Non-Functional Requirements

5.1 Performance

- The system must handle large volumes of data efficiently and generate real-time reports with minimal delay.

- Reports and dashboards should update automatically based on new data.

5.2 Scalability

- The analytics platform should be scalable to handle growing user activity and data volume.

- Capable of adding new metrics or data points without significant changes to the architecture.

5.3 Security

- Ensure all data is anonymized and securely transferred, especially customer information.

- Apply appropriate access controls to restrict sensitive data viewing to authorized personnel.

5.4 Reliability

- The analytics system should operate without downtime and be reliable for making business-critical decisions.

- Backups and disaster recovery mechanisms should be in place for critical data.

6. Core Features

6.1 Data Pipeline

- Automated Data Extraction: Pull data from the MySQL database into the analytics environment.

- Scheduled Processing: Automate the cleaning and aggregation of data daily.

6.2 Data Visualization Dashboards

- Sales Dashboard: Visualize total revenue, product sales, and monthly trends using Tableau/Power BI.

- Customer Insights: Display user behavior data like average session duration, purchase frequency, and demographics.

6.3 Predictive Analytics

- Sales Forecasting: Use historical data and statistical methods to predict future sales and demand.

- Customer Behavior: Predict customer lifetime value and potential churn rates.

7. Technical Stack

- Python: For data extraction, processing, and analysis.

- NumPy & Pandas: For data manipulation and cleaning.

- Matplotlib & Seaborn: For static visualizations (line charts, bar graphs, etc.).

- Tableau & Power BI: For creating interactive dashboards.

- MySQL: Source of sales, product, and customer data.

- Django REST Framework: For API endpoints to access raw data and integrate with external reporting tools.

8. Deliverables

- Fully functional data analytics pipeline integrated with the e-commerce platform.

- Real-time and interactive Tableau/Power BI dashboards displaying key business metrics.

- Static reports generated with Python visualizations (Matplotlib/Seaborn).

- Documentation covering data collection methods, setup, and maintenance of the analytics system.

9. Timeline (1 Week)

- Day 1-2: Set up data pipeline, collect and clean data from the MySQL database.

- Day 3-4: Develop initial reports using Matplotlib/Seaborn and build Tableau/Power BI dashboards.

- Day 5: Implement predictive models for sales forecasting and customer behavior analysis.

- Day 6: Test dashboards for performance and scalability; refine visualizations.

- Day 7: Final review, documentation, and project handover.