Project Idea: UK Train Rides

1.1 Project Proposal

- **Objective:**
  Build an interactive system (with dashboards and analyses) to explore UK train ride data. The project will focus on data cleaning, ridership trends, forecasting passenger volumes, and presenting insights for informed decision-making.
- **Scope:**
  - **Ridership Analysis:** Understand daily, weekly, and monthly passenger trends.
  - **Route & Station Insights:** Identify top routes/stations by passenger traffic.
  - **Dashboard Creation:** Develop interactive visualizations (in Tableau or Power BI) for stakeholders.

1.2 Project Plan

Phase	Tasks	Duration	Milestone
1 - Data Collection & Preprocessing	- Gather raw UK train ride datasets (e.g., station info, ridership logs). - Clean the data: handle missing values, correct data types, remove duplicates. - Build initial data model.	Week 1	Cleaned & modeled dataset
2 - Data Analysis & Questions	- Perform exploratory data analysis (EDA) to identify key metrics (peak hours, busiest stations). - Formulate specific questions (e.g., route popularity, seasonal usage).	Week 2	Analysis questions documented
3 - Visualization & Presentation	- Create an interactive dashboard (Tableau or Power BI). - Incorporate filters (e.g., date range, station, route). - Present final insights and recommendations.	Week 3	Final dashboard & presentation

---

1.3 Resources

- **Software & Tools:**
  - **SQL** for querying large datasets.
  - **Python** (pandas, NumPy, matplotlib, seaborn) for data processing and forecasting.
  - **Tableau** or **Power BI** for dashboard creation and data visualization.

1.4 Task Assignment & Roles

- **Data Collection:** Esraa Badwi
- **Data Cleaning & Transformation:** Basmala Ahmad
- **Forecasting & Advanced Analytics:** Yassin Walid
- **Dashboard Development:** Youssef Elbedewy
- **Review & Refinement:** Shahd Negm Eldeen
- **Presentation & Documentation:** George Emad Attia

1.5 Risk Assessment & Mitigation Plan

- **Risk:** Data inconsistency (e.g., missing station codes)
  - **Solution:** Implement thorough data validation checks.
- **Risk:** Large dataset performance in visualization tools
  - **Solution:** Pre-aggregate data or optimize queries before loading into Tableau/Power BI.

1.6 KPIs

1. **Purchase Type (Station, Online)**
2. **Journey Status (On Time, Delayed)**
3. **Top 5 Busiest Stations**
4. **Total Revenue**
5. **Total Tickets Sold**
6. **Average Ticket Price**
7. **Most Popular Route**
8. **Peak Travel Hour**
9. **Revenue per Station**
10. **Passenger Load Factor**
11. **Cancellation & Refund Rate** : Tracks ticket cancellations and refunds.

1.7 Dataset Overview

Below is a generic example of the dataset structure. Actual columns may vary based on the data source:

Field	Description
Transaction ID	Unique identifier for an individual train ticket purchase
Date of Purchase	Date the ticket was purchased
Time of Purchase	Time the ticket was purchased
Purchase Type	Whether the ticket was purchased online or directly at a train station
Payment Method	Payment method used to purchase the ticket (Contactless, Credit Card, or Debit Card)
Railcard	Whether the passenger is a National Railcard holder (Adult, Senior, or Disabled) or not (None). Railcard holders get 1/3 off their ticket purchases.
Ticket Class	Seat class for the ticket (Standard or First)
Ticket Type	When you buy or can use the ticket. Advance tickets are 1/2 off and must be purchased at least a day prior to departure. Off-Peak tickets are 1/4 off and must be used outside of peak hours (weekdays between 6-8am and 4-6pm). Anytime tickets are full price and can be bought and used at any time during the day.
Price	Final cost of the ticket
Departure Station	Station to board the train

---

2. Literature Review

2.1 Feedback & Evaluation

- **Industry Experts & Stakeholders:**
  Early feedback from railway authorities and station managers can help refine KPIs and confirm data accuracy.
- **Academic Papers & Research:**
  Studies on public transport forecasting (e.g., time series models) and passenger flow analysis.

2.2 Suggested Improvements

- **Integration with External Data:**
  Incorporate weather or special event data to refine forecasting accuracy.
- **Dashboard Enhancements:**
  Enhance dashboard functionality with interactive route maps for better visualization of station-to-station traffic.

2.3 Final Grading Criteria

- **Data Accuracy:** Proper data cleaning, validation, and cross-referencing with official sources.
- **Visualization Effectiveness:** Clarity, intuitive design, and depth of insights.
- **User Interaction & Storytelling:** Ability to drill down, filter, and extract meaningful conclusions.

---

3. Requirements Gathering

3.1 Stakeholder Analysis

- **Rail Operators/Authorities:** Need accurate ridership forecasts and route optimization.
- **Station Managers:** Want to understand daily/weekly patterns to allocate resources effectively.
- **Passengers/Commuters (Indirect Stakeholders):** Benefit from improved scheduling and reduced congestion.

3.2 User Stories

- **As a Station Manager,** I want to see daily passenger counts so that I can plan staffing needs.

3.3 Functional Requirements

- **Ridership Reporting:** Summaries of daily, weekly, and monthly passenger counts.
- **Route & Station Analysis:** Ability to filter and compare different routes or stations.
- **Dashboard Filters:** Interactive controls (date ranges, route selection, station selection).

3.4 Non-functional Requirements

- **Performance:** Dashboard should load within 5 seconds for typical queries.
- **Usability:** Intuitive layout with clear labels and tooltips.

- **Scalability:** Capable of handling data growth (e.g., multiple years of historical records).
- **Security & Data Privacy:** Adhere to data-sharing regulations if personal information is included.

---

4. System Analysis & Design

4.1 Problem Statement

Manually analyzing UK train ridership data is time-consuming and prone to error. An automated, interactive solution will streamline data processing, improve accuracy, and enable evidence-based decisions for rail operators and station managers.

4.2 Use Case Diagram

- **Admin:** Manages data sources, configures scheduled data refresh.
- **Business Users (Station Managers, Rail Operators):** Views the dashboard, applies filters, and examines KPIs.

(Diagram would show Admin and Business Users interacting with the system, which connects to the data sources and the dashboard.)

4.3 Software Architecture

1. **Data Ingestion:** Collect CSV/Excel data from UK rail authorities.
2. **Data Transformation:** Clean and model in Python (pandas).
3. **Data Storage:** Store processed data in SQL database or CSV for easy retrieval.
4. **Visualization Layer:** Build dashboards in Tableau or Power BI connected to the cleaned dataset.

4.4 Database Design & Data Modeling

- **ER Diagram:**

ER Diagram: Demonstrate relationships between key entities.
Logical Schema: Demonstrate tables and their relationships.

4.5 UI/UX Design & Prototyping

- **Wireframes:**
  - Main dashboard with key KPIs.
- **UI/UX Guidelines:**
  - Use consistent color schemes for route or station categories.

---

5. Deployment System & Integration

5.1 Technology Stack

- **Front-end Visualization:** Tableau or Power BI
- **Back-end Processing:** Python (pandas, NumPy)
- **Database/Storage:** SQL or CSV files in a secure data repository

5.2 Deployment Diagram

- **Local/Cloud Environment:**
    1. Data stored on a local server or cloud storage (GitHub.).
    2. Python scripts run locally or on a VM/container for data cleaning.
    3. Dashboard published to Tableau Server, Tableau Public, or Power BI Service.

5.3 Component Diagram

1. **Data Source Component:** CSV files, etc.
2. **Processing Component:** Python environment with relevant libraries.
3. **Visualization Component:** Tableau/Power BI.
4. **User Access Component:** Web portal or desktop application for the dashboard.

---

6. Additional Deliverables

6.1 API Documentation

- Explains how to connect data if APIs are used.

6.2 Testing & Validation

- **Data Validation**
- **Model Validation**
- **Dashboard Testing**

6.3 Deployment Strategy

How the dashboard will be published and accessed by users.