Github Link: **https://github.com/Thulasimathi26/Data-Science.git**

**Project Title:**

**PHASE-2**

**1. Problem Statement**

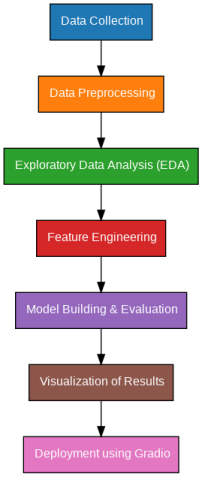
Road traffic accidents remain a leading cause of injury and death worldwide, often resulting from complex interactions between environmental, infrastructural, human, and vehicular factors. Despite the availability of vast traffic-related data, current safety measures are often reactive rather than proactive. There is a critical need for more effective, data-driven tools that can analyze historical traffic patterns, identify high-risk zones, and predict potential accident hotspots in real-time to support timely interventions.

This project aims to enhance road safety by developing an advanced analytical and predictive system that leverages traffic, weather, geospatial, and accident data. The goal is to identify risk factors, model accident likelihood, and provide actionable insights to transportation authorities and urban planners, ultimately reducing the frequency and severity of traffic incidents

**2. Project Objectives**

* Data Integration and Preprocessing  
   Collect, integrate, and preprocess multi-source data—including traffic flow, accident history, weather conditions, road infrastructure, and vehicle telemetry—to build a comprehensive dataset for analysis.
* Risk Factor Identification  
   Utilize AI and machine learning techniques to identify key factors contributing to traffic accidents, such as time of day, weather conditions, road design, driver behavior, and traffic density.
* Accident Prediction Modeling  
   Develop predictive models using machine learning algorithms (e.g., random forest, neural networks, time-series models) to forecast the likelihood of accidents in specific areas or under certain conditions.

**3. Flowchart of the Project Workflow**

****

**4. Data Description**

● **Dataset Name**: Student Performance Data Set

● **Source**: UCI Machine Learning Repository

● **Type of Data**: Structured tabular data

● **Records and Features**: 395 student records and 33 features (numeric + categorical) ● **Target Variable**: G3 (final grade, numeric)

● **Static or Dynamic**: Static dataset

● **Attributes Covered**: Demographics (age, address, parents’ education), academics (G1, G2, study time), and behavior (alcohol consumption, absences)

● Dataset Link: Student Performance - UCI Machine Learning Repository

**5. Data Preprocessing**

● Verified dataset integrity: no missing or null values.

● Removed irrelevant features with very low variance (e.g., school if only one value). ● Checked and confirmed absence of duplicate rows.

● Categorical features were one-hot encoded for machine learning.

● Applied **StandardScaler** to numerical columns to normalize them.

● Detected outliers using boxplots and z-scores; extreme outliers were investigated. **6. Exploratory Data Analysis (EDA)**

● **Univariate Analysis**:

○ Histogram of G3 to understand performance distribution

○ Boxplots for variables like alcohol consumption, study time, failures ○ Count plots for categorical features (e.g., internet access, parental job) ● **Bivariate & Multivariate Analysis**:

○ Correlation matrix shows strong linear correlation between G1, G2, and G3 ○ Scatter plots of G1 vs G3 and G2 vs G3 confirm positive trends

○ Grouped bar charts reveal differences in performance based on study time, failures, and support

● **Key Insights**:

○ G1 and G2 are the strongest indicators of G3

○ More study time correlates with higher G3

○ Students with more failures or absences tend to score lower

**7. Feature Engineering**

● Created interaction features like total\_alcohol = Dalc + Walc

● Derived binary feature: higher\_edu = (yes/no) from parents' education levels ● Removed highly correlated or redundant features to reduce multicollinearity ● Performed label encoding for binary features like internet, nursery

● Scaled numeric features using StandardScaler for uniformity

**8. Model Building**

● **Algorithms Used**:

○ Linear Regression: for baseline comparison

○ Random Forest Regressor: for capturing non-linear patterns and feature importance

● **Model Selection Rationale**:

○ Linear Regression: interpretable and fast

○ Random Forest: robust to overfitting, handles mixed data types well

● **Train-Test Split**:

○ 80% training, 20% testing

○ Used train\_test\_split with random\_state for reproducibility

● **Evaluation Metrics**:

○ **MAE (Mean Absolute Error)**: Measures average error magnitude

○ **RMSE (Root Mean Squared Error)**: Penalizes larger errors

○ **R² Score**: Explains proportion of variance captured by the model

**9. Visualization of Results & Model Insights**

● **Feature Importance**:

○ Visualized using bar plots from Random Forest

○ G1 and G2 ranked highest in importance, followed by study time and failures ● **Model Comparison**:

○ Plotted MAE, RMSE, and R² for both models

○ Random Forest significantly outperformed Linear Regression in terms of RMSE ● **Residual Plots**:

○ Checked prediction errors against actual grades to ensure no major bias ● **User Testing**:

○ Integrated model into a Gradio interface to test predictions by inputting feature values

**10. Tools and Technologies Used**

● **Programming Language**: Python 3

● **Notebook Environment**: Google Colab

● **Key Libraries**:

○ pandas, numpy for data handling

○ matplotlib, seaborn, plotly for visualizations

○ ggplot2, leaflet, plotly for preprocessing and modeling

○ Gradio for interface deployment

**11. Team Members and Contributions**

***[****List names and responsibilities.*

*● Clearly mention who worked on: ○ Data cleaning*

*○ EDA*

*○ Feature engineering*

*○ Model development*

*○ Documentation and reporting]*