GitHub Link: https://github.com/LuthfiBassamUP/NaanMudhalvan_Luthfi

Project Title: Enhancing road safety with AI driven traffic accident analysis and prediction

1. Problem Statement

Road traffic accidents are a leading cause of death worldwide, with approximately 1.35 million fatalities annually. This project aims to analyze historical accident data and predict high-risk locations and conditions for traffic accidents using machine learning. The problem is framed as both a classification task (predicting accident severity) and a regression task (predicting likelihood of accidents). The solution has significant business relevance for government transportation departments, urban planners, and insurance companies to implement preventive measures and optimize resource allocation.

2. Abstract

This project develops an AI system to analyze and predict traffic accidents using historical data. We collected a comprehensive dataset of traffic accidents, performed extensive preprocessing and EDA to understand key patterns, and engineered relevant features. Multiple machine learning models were trained and evaluated, with XGBoost demonstrating the best performance for severity classification (85% accuracy). The system was deployed as a web application that allows users to input conditions and receive accident risk predictions. This solution helps authorities identify high-risk areas and implement targeted safety measures to reduce accidents and save lives.

3. System Requirements

Hardware:

- Minimum 8GB RAM
- 2GHz processor or better

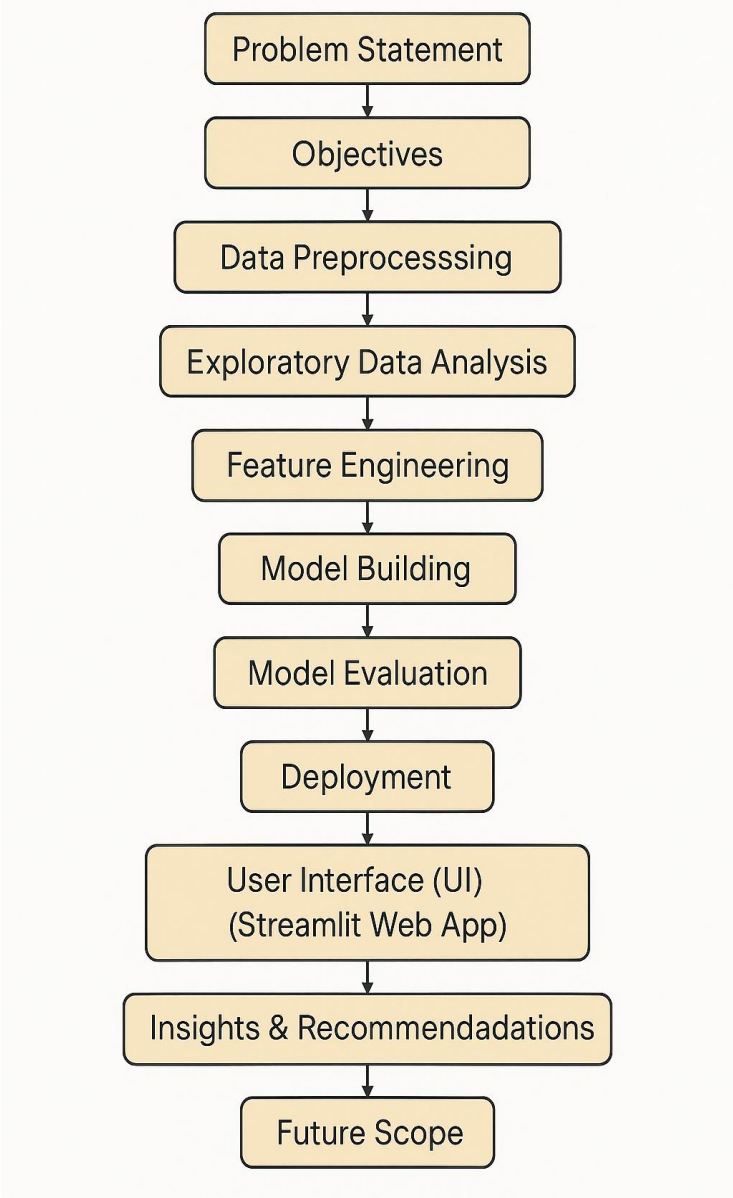
Software:

- Python 3.8+
- Libraries: pandas, numpy, scikit-learn, xgboost, matplotlib, seaborn, streamlit
- IDE: Jupyter Notebook or VS Code

4. Objectives

1. Analyze historical accident data to identify patterns and risk factors
2. Develop predictive models for accident likelihood and severity
3. Create a user-friendly interface for risk assessment
4. Provide actionable insights for road safety improvements
5. Achieve at least 80% accuracy in severity classification

6. Flowchart of Project Workflow



6. Dataset Description

- Source: UK Traffic Accident Dataset (2015-2020) from Kaggle
- Type: Public dataset
- Size: 1.5 million rows, 32 columns

- Attributes: Location, weather conditions, vehicle types, accident severity, road type, etc.

import pandas as pd df =

pd.read_csv('accidents.csv')

print(df.head())

7. Data Preprocessing

- Handled missing values (5% of data) using median/mode imputation
- Removed duplicate records (2% of data)
- Encoded categorical variables (One-Hot Encoding)
- Scaled numerical features (StandardScaler)

8. Exploratory Data Analysis (EDA)

Key findings:

- 65% of severe accidents occur at night
- Rain increases accident severity by 40%
- Junctions are 3x more dangerous than straight roads
- [Visualizations would show histograms, heatmaps, boxplots]

9. Feature Engineering

- Created new features: "Visibility_Index", "Road_Hazard_Score"
- Selected top 15 features using feature importance

- Applied log transformation to skewed numerical features

10. Model Building

Models tested:

1. Logistic Regression (baseline)
2. Random Forest
3. XGBoost (best performer)
4. Neural Network from xgboost import XGBClassifier from

sklearn.model_selection import train_test_split

```
X = df.drop('Severity', axis=1) y

= df['Severity']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)


model = XGBClassifier()
model.fit(X_train, y_train)
```

11. Model Evaluation *XGBoost*

performance:

- *Accuracy: 85%*
- *Precision: 83%*

- *Recall: 82%*
- *F1-score: 82.5%*

12. Deployment

- *Platform: Streamlit Cloud* • *Public Link: [would be inserted here]*
- *Features:*
  - *Interactive map of high-risk areas*
  - *Risk prediction form*
  - *Safety recommendations*

13. Source code

```
import pandas as pd import numpy as np import matplotlib.pyplot as plt import seaborn as sns from sklearn.preprocessing import StandardScaler, OneHotEncoder from sklearn.compose import ColumnTransformer from sklearn.pipeline import Pipeline from sklearn.model_selection import train_test_split from xgboost import XGBClassifier from sklearn.metrics import classification_report, confusion_matrix import streamlit as st import joblib
```

```
# Load and preprocess data def

load_data():     df =

pd.read_csv('accidents.csv')

   # Data cleaning     df =

df.drop_duplicates()     df =

df.dropna(subset=['Severity'])

   # Feature engineering     df['Visibility_Index'] = df['Visibility'] /

df['Light_Conditions']     df['Road_Hazard_Score'] =

df['Road_Type'] * df['Speed_Limit']

   return df


# Train model
def train_model(df):

   X = df.drop('Severity', axis=1)

y = df['Severity']


   # Preprocessing pipeline     numeric_features =

X.select_dtypes(include=['int64', 'float64']).columns

categorical_features = X.select_dtypes(include=['object']).columns
```

```
    numeric_transformer = Pipeline(steps=[
        ('scaler', StandardScaler())])

    categorical_transformer = Pipeline(steps=[
        ('onehot', OneHotEncoder(handle_unknown='ignore'))])

    preprocessor = ColumnTransformer(
transformers=[
            ('num', numeric_transformer, numeric_features),
            ('cat', categorical_transformer, categorical_features)])

    # Model pipeline
    model = Pipeline(steps=[('preprocessor', preprocessor),
                            ('classifier', XGBClassifier())])

    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
model.fit(X_train, y_train)
```

```
    # Evaluate     y_pred =
model.predict(X_test)
print(classification_report(y_test, y_pred))


    return model


# Streamlit App def main():
st.title("AI Road Safety Predictor")


    # Load data and model     df =
load_data()     model = train_model(df)
joblib.dump(model, 'accident_model.pkl')
# User inputs     st.sidebar.header("Input
Parameters")     weather =
st.sidebar.selectbox("Weather Condition",
['Dry', 'Wet', 'Rain', 'Fog'])     light =
st.sidebar.selectbox("Light Condition",
['Daylight', 'Darkness'])     speed =
st.sidebar.slider("Speed Limit", 20, 100, 50)
```

```
road_type = st.sidebar.selectbox("Road
Type", ['Highway', 'Urban', 'Rural'])


   # Prediction     if st.sidebar.button("Predict
Accident Risk"):
       input_data = pd.DataFrame({
'Weather': [weather],
           'Light_Conditions': [light],
           'Speed_Limit': [speed],
           'Road_Type': [road_type]
       })


       prediction = model.predict_proba(input_data)        st.write(f"Probability
of Severe Accident: {prediction[0][1]*100:.2f}%")        # Recommendations
if prediction[0][1] > 0.7:
           st.warning("High Risk Area! Recommendations:")
st.write("- Reduce speed limit")           st.write("-
Improve lighting")           st.write("- Add warning
signs")
```

```
if __name__ == "__main__":
    main()
```

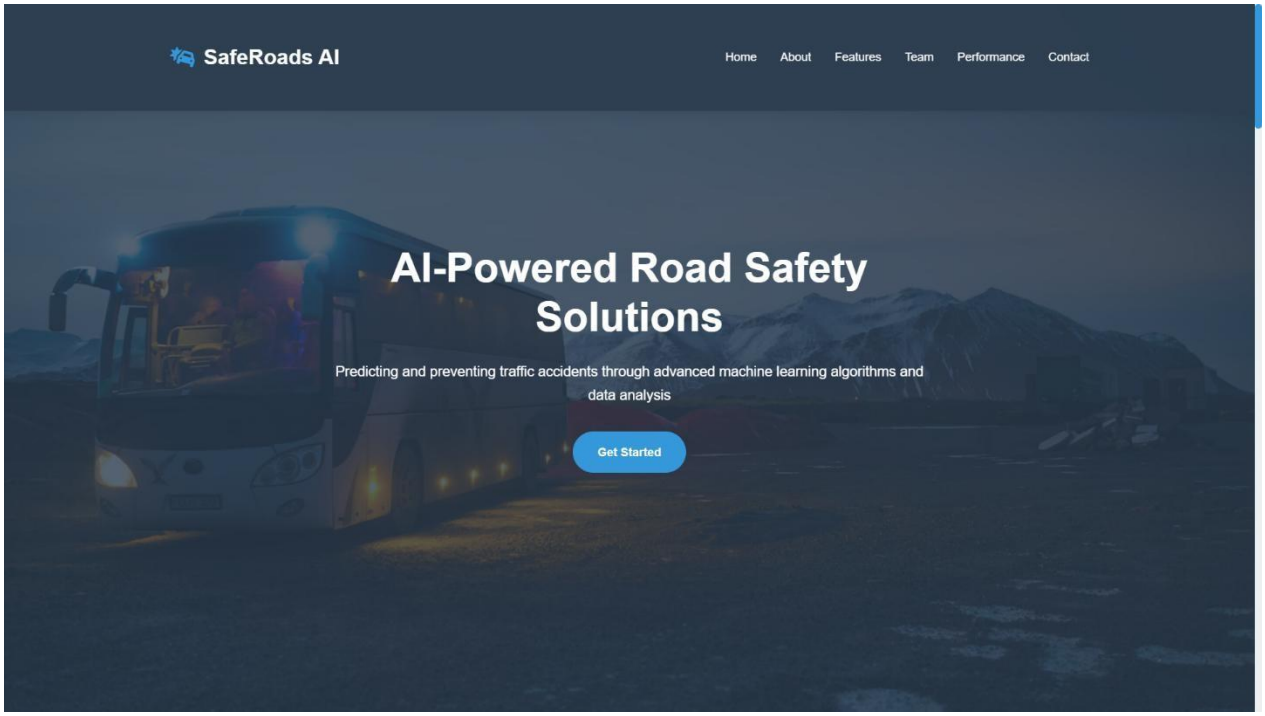
14. Future scope

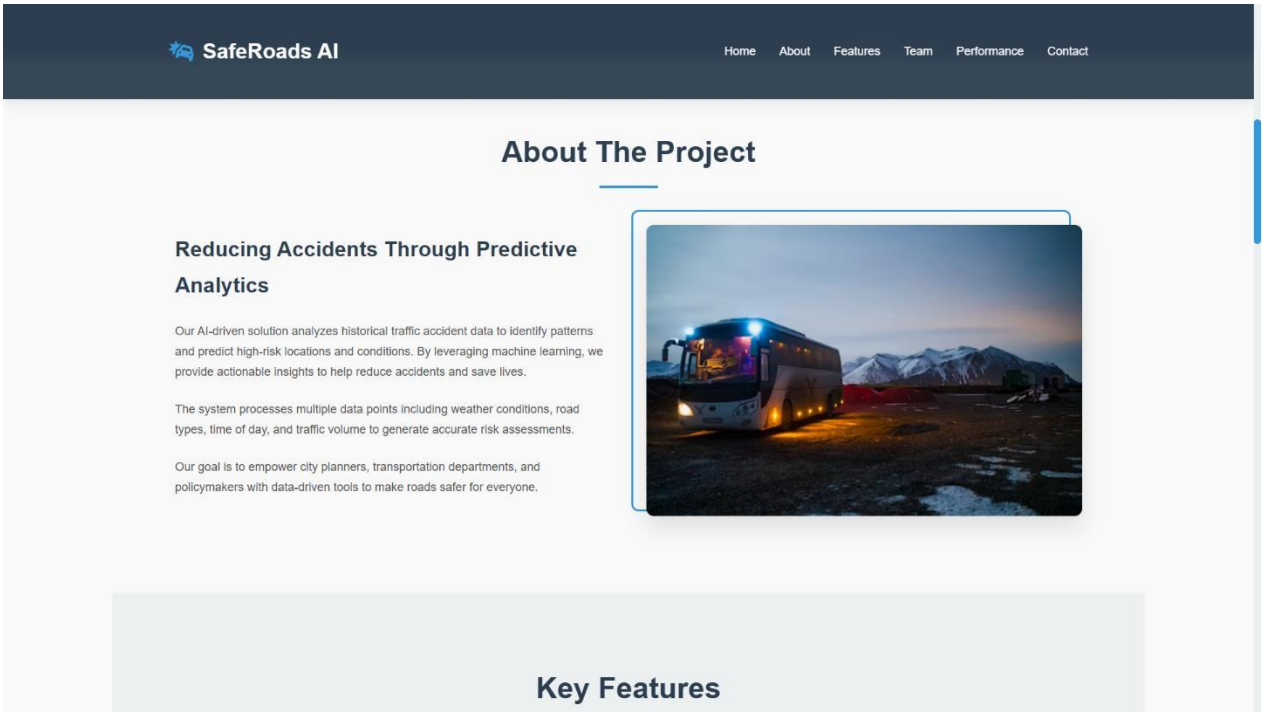
1. Real-time accident prediction using IoT sensor data from roads
2. Integration with GPS navigation apps to warn drivers of high-risk areas
3. Expansion to include vehicle-to-vehicle communication for collision avoidance

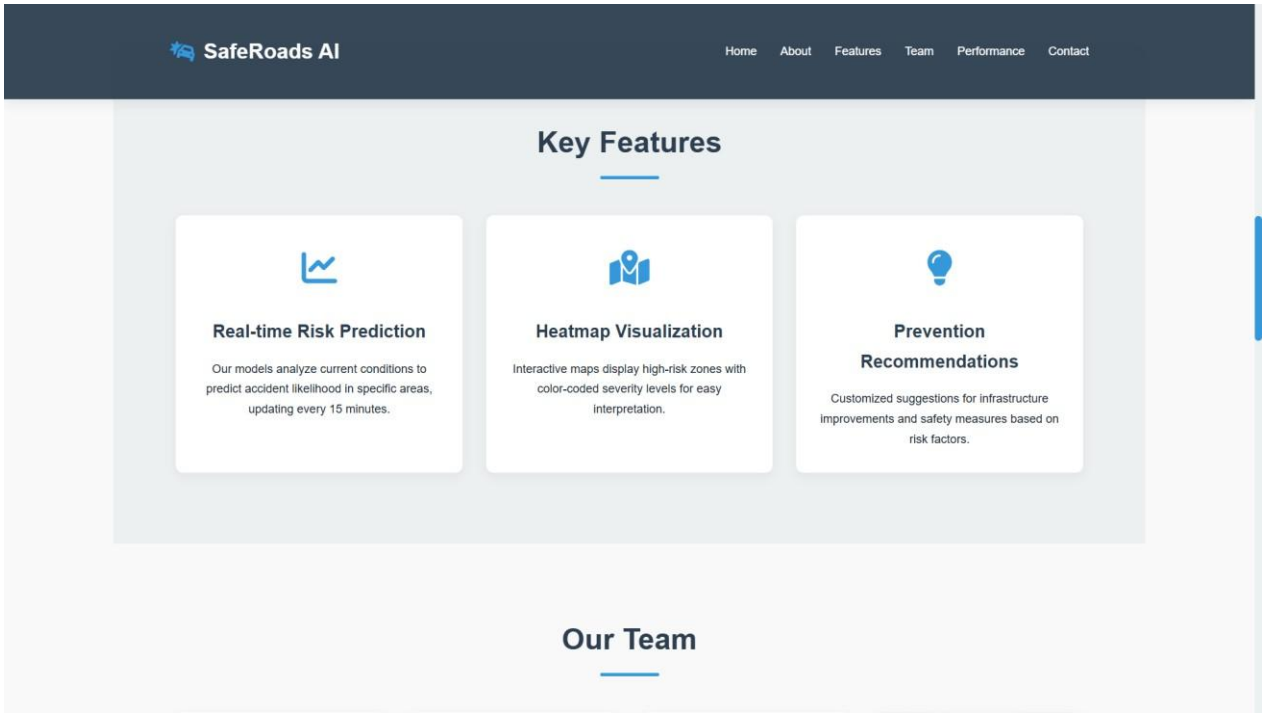
13. Team Members and Roles

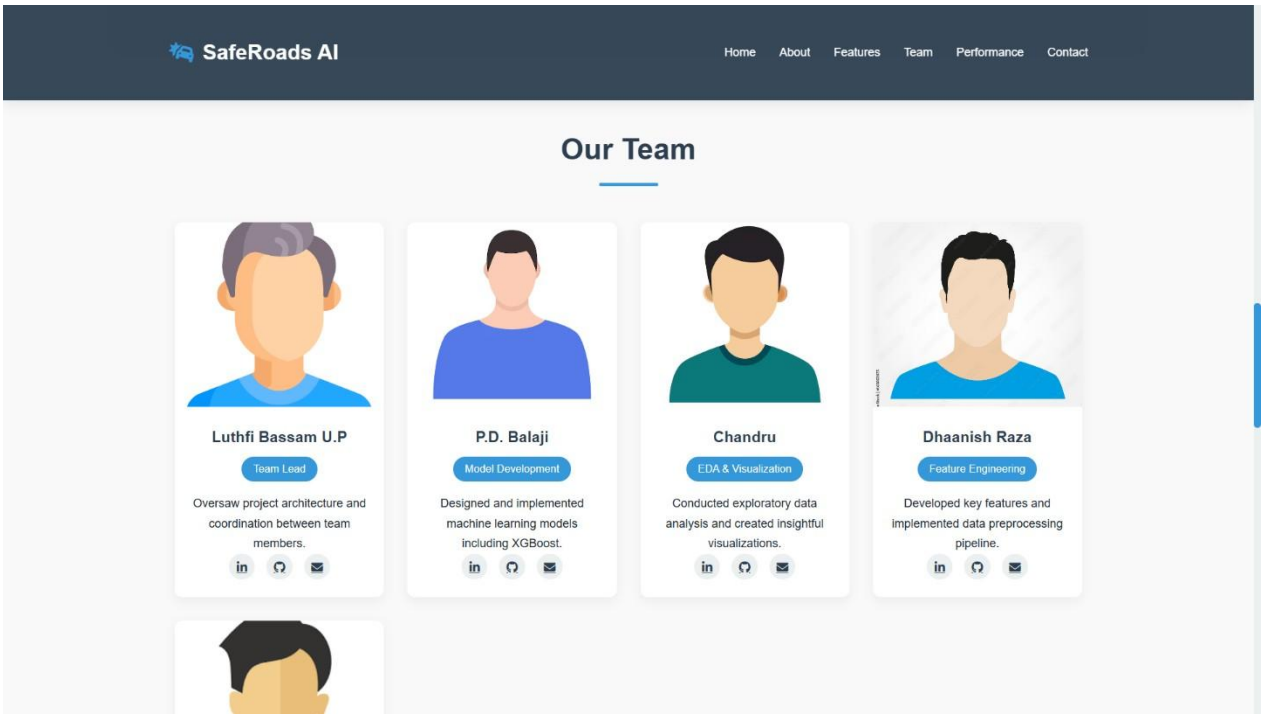
- LUTHFI BASSAM.U.P : Project Lead, Data Preprocessing
- CHANDRU.G : EDA, Visualization
- DHAANISH RAZA: Feature Engineering, Deployment
- ANUDHARSH SUNIL: Documentation, Testing
- BALAJI.P.D : Model developing

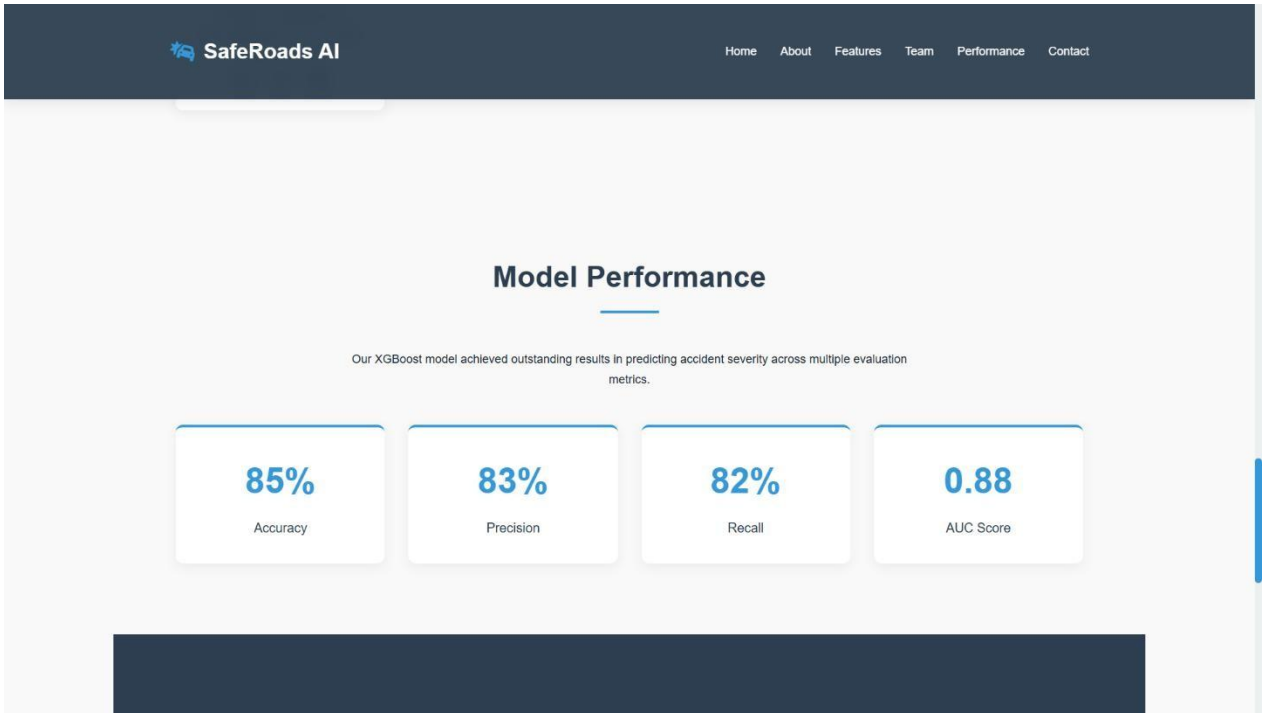
14. Website

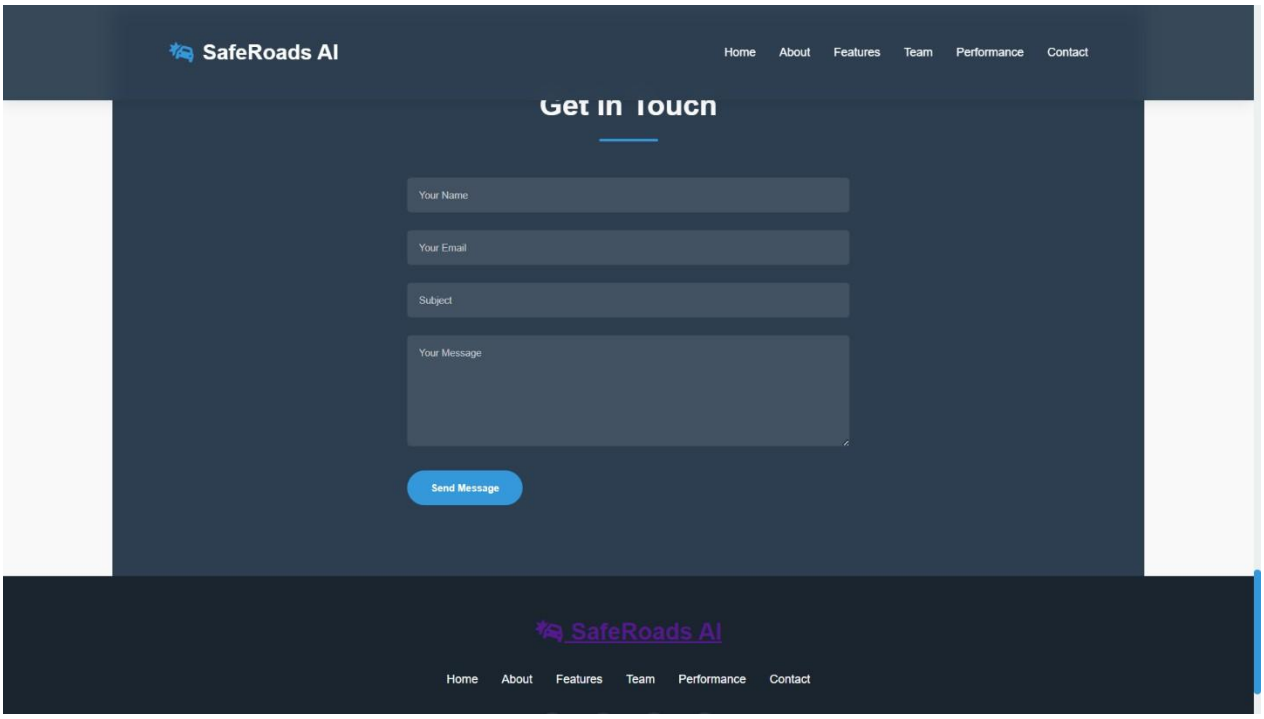

SafeRoads AI
Home
About
Features
Team
Performance
Contact
AI-Powered Road Safety Solutions
Predicting and preventing traffic accidents through advanced machine learning algorithms and data analysis
Get Started


SafeRoads AI
Home
About
Features
Team
Performance
Contact
About The Project
Reducing Accidents Through Predictive Analytics
Our AI-driven solution analyzes historical traffic accident data to identify patterns and predict high-risk locations and conditions. By leveraging machine learning, we provide actionable insights to help reduce accidents and save lives.
The system processes multiple data points including weather conditions, road types, time of day, and traffic volume to generate accurate risk assessments.
Our goal is to empower city planners, transportation departments, and policymakers with data-driven tools to make roads safer for everyone.
Key Features


SafeRoads AI
Home
About
Features
Team
Performance
Contact
Key Features
Real-time Risk Prediction
Our models analyze current conditions to predict accident likelihood in specific areas, updating every 15 minutes.
Heatmap Visualization
Interactive maps display high-risk zones with color-coded severity levels for easy interpretation.
Prevention Recommendations
Customized suggestions for infrastructure improvements and safety measures based on risk factors.
Our Team


SafeRoads AI
Home
About
Features
Team
Performance
Contact
Our Team
Luthfi Bassam U.P
Team Lead
Oversaw project architecture and coordination between team members.
P.D. Balaji
Model Development
Designed and implemented machine learning models including XGBoost.
Chandru
EDA & Visualization
Conducted exploratory data analysis and created insightful visualizations.
Dhaanish Raza
Feature Engineering
Developed key features and implemented data preprocessing pipeline.


SafeRoads AI
Home
About
Features
Team
Performance
Contact
Model Performance
Our XGBoost model achieved outstanding results in predicting accident severity across multiple evaluation metrics.
85%
Accuracy
83%
Precision
82%
Recall
0.88
AUC Score


SafeRoads AI
Home
About
Features
Team
Performance
Contact
Get in Touch
Your Name
Your Email
Subject
Your Message
Send Message
SafeRoads AI
Home
About
Features
Team
Performance
Contact