Technical Architecture:

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

|  |  |
| --- | --- |
| Date | 17 October 2022 |
| Team ID | PNT2022TMID03254 |
| Project Name | Project - Machine Learning Based Vehicle Performance Analyzer |
| Maximum Marks | 4 Marks |

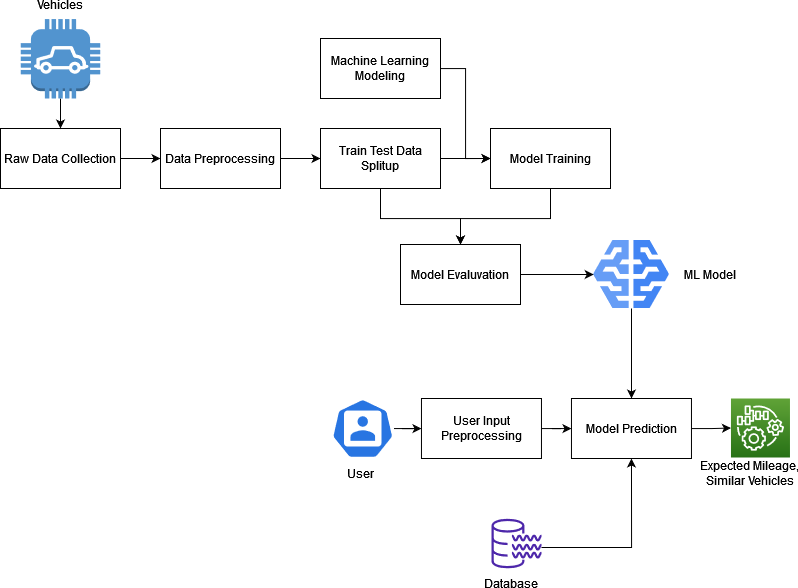


Table-1: Components & Technologies:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
| 1. | User Interface | The user interacts with the application through a Web Application that is responsive to the device that is being used. | React Js |
| 2. | Get User Data | The process collects the user input data that is  collected via a form to the server as a JSON Object | REST API |
| 3. | Model Prediction | Use the data collected from the user to make  predictions on the mileage expected. | IBM Watson ML |
| 4. | Send User Report | Send the predictions along with suggestions to the  user as JSON Object | REST API |
| 5. | Database | Database contain user information such as name, email, vehicle basic information, mileage predicted  over time. | MySQL |
| 6. | Cloud Database | Database Service on Cloud | IBM DB2 |
| 7. | External API-1 | Vehicle Details Database | <https://api.auto-data.net/> |
| 8. | Machine Learning Model | The machine learning model is used to predict mileage from the user inputs | Regression Modelling. |
| 9. | Infrastructure (Server / Cloud) | Application Deployment on Local System / Cloud Local Server Configuration: Core i5, 8GB RAM  Cloud Server Configuration: | Local, Docker |

Table-2: Application Characteristics:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | React Js, Flask, Sci-kit Learn | JavaScript, Python |
| 2. | Security Implementations | Identity and Access Management, OAUTH, WAF | IBM Cloud |
| 3. | Scalable Architecture | 3 Tier Architecture, Model-View-Controller  implementation. | Model - SQL DB, View - ReactJS,  Controller - Flask Server |
| 4. | Availability | Proxy servers, Load Balancers to help balance  traffic among servers to help improve uptime | IBM Cloud load balancers |
| 5. | Performance | The frontend is detached from the Business logic server reducing requests sent to the server. | Nginx proxy |