Car Purchase Prediction System

Team ID : Team-592694

Team Size : 4

Team Leader : Kelvin J Anil

Team member : Amish Ranjan

Team member: Sutam Chandra

Team member : Rishi Raj Upadhyaya

Index

1. Introduction

1.1 Project Overview

The Car Affordability Prediction Web Application aims to assist users in determining their ability to afford a car based on key input parameters such as user ID, gender, annual income, and age. The application leverages a Support Vector Machine (SVM) classifier trained on historical car purchase data to make predictions. Users can input their details through a user-friendly web interface, and the system provides a prediction on whether they can afford a car or not.

1.2 Purpose

The Car Affordability Prediction Web Application serves as a user-friendly and accessible tool designed to empower individuals in making informed decisions about car purchases. Its primary purpose is to provide financial insights based on key parameters such as annual income, age, and gender, enabling users to gauge the feasibility of owning a car. The intuitive web interface simplifies the input process, ensuring accessibility for users of varying technical backgrounds. Beyond its practical utility, the application also acts as an educational resource, offering insights into the financial considerations of car affordability.

2. LITERATURE SURVEY

2.1 Existing problem

The existing problem addressed by the Car Purchase Prediction Web Application lies in the challenge individuals face when assessing their financial readiness to purchase a car. Many prospective car buyers may lack a convenient and accessible tool that considers their specific financial situation comprehensively. Traditional methods often

involve complex calculations or general financial advice that may not be tailored to individual circumstances. This can result in uncertainty and potential financial strain for users, making it crucial to bridge this gap with a solution that provides personalized and user-friendly insights. The Car Purchase Prediction Web Application aims to address this existing problem by offering a streamlined and informative platform for users to make well-informed decisions about their car purchases based on their unique financial profiles.

2.2 References

2.3 Problem Statement Definition

How might we develop an effective machine learning model for an individual that utilizes an individual's age and annual salary to accurately predict their car affordability, thereby empowering users to make informed decisions when considering a car purchase, minimize financial strain, and achieve their transportation needs and financial goals?

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

Goals

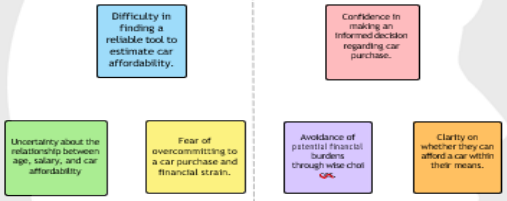
- Determine an affordable price range for a car based on my age and income.
- Make a well-informed decision to avoid financial strain.
- Find a car that aligns with my budget and lifestyle.
- Secure a car that meets my transportation needs and preferences.

Pains

- "Can I comfortably make car loan payments with my current salary?"
- "I'm not sure if my age and salary are suitable for buying a car."

Gains

- Anxious about the financial implications of buying a car.
- Eager to gain clarity on their affordability.



Thinks

- "What are the average car prices for people my age and income level?"

Feels

Says

- "I want to buy a car, but I'm not sure if I can afford it."
- "I'm concerned about making a poor financial decision."

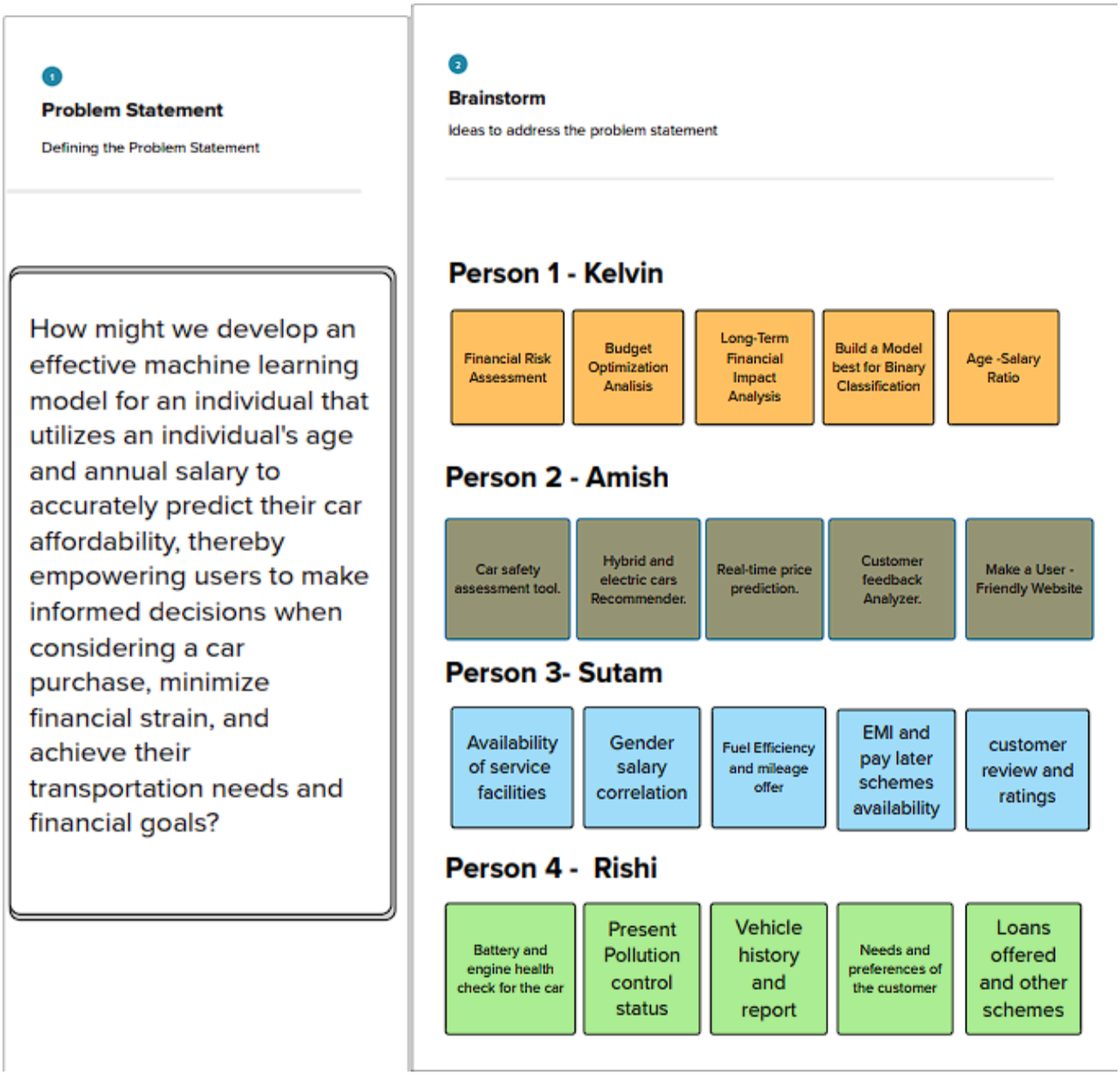
Does

- Researches car prices, loan options, and interest rates.
- Compares their age and salary to the car price ranges.
- Tries to calculate potential car loan monthly payments.

Hears

- "Financial experts recommend not spending more than 40% of your salary on a car."
- Car dealerships are offering special financing deals.
- "Recent news reports mention changes in car loan interest rates."

3.2 Ideation & Brainstorming



3

Group ideas

Organizing the essential ideas you intend to implement

Build a Model best for Binary Classification

Age -Salary Ratio

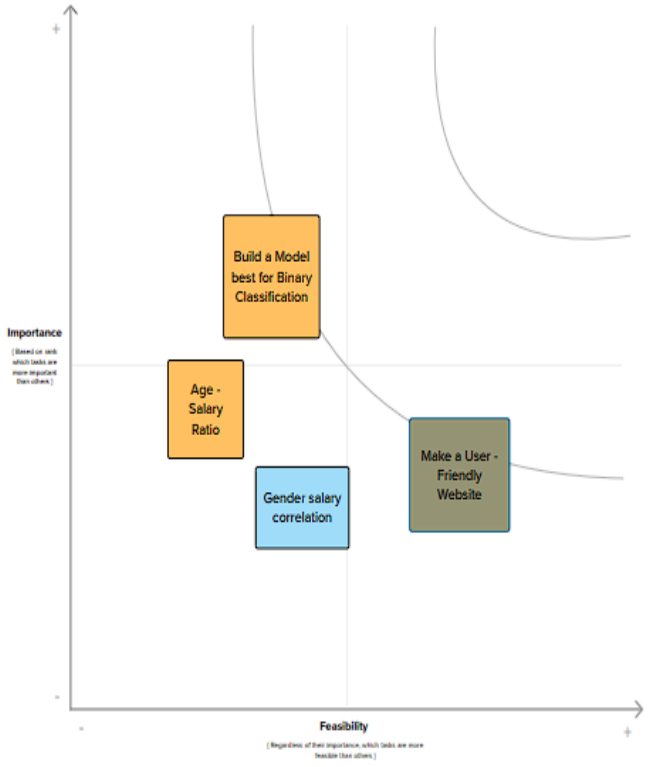
Gender salary correlation

Make a User - Friendly Website

4

Prioritize

Prioritizing our ideas based on rank



4. REQUIREMENT ANALYSIS

4.1 Functional requirement

1. Prediction Interface: The application must offer a user-friendly interface for users to input their details for car affordability prediction.

2. Machine Learning Model Integration: The system should integrate a machine learning model (Support Vector Machine, SVM) trained on historical car purchase data for accurate predictions.

3. Input Validation: Ensure that input data is validated for correctness and completeness to maintain data integrity.

4. **Prediction Output:** Provide a clear and understandable prediction outcome, indicating whether the user can afford to purchase a car or not.

5. Responsive Design: Design the web interface to be responsive, ensuring a seamless user experience across various devices.

4.2 Non-Functional requirements

1. Performance: The application should respond to user inputs within 3 seconds to provide a responsive user experience.It should handle at least 100 simultaneous users without significant degradation in performance.

2. Reliability: The system must be available 99% of the time to ensure users can access predictions reliably.

3. Usability: The user interface should be intuitive, requiring minimal guidance for users to input their details and interpret the prediction results.

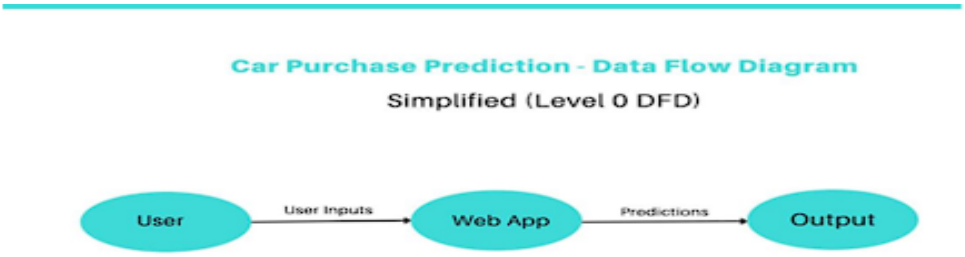
4. Compatibility: The application should be compatible with major web browsers, including Chrome, Firefox, Safari, and Edge.

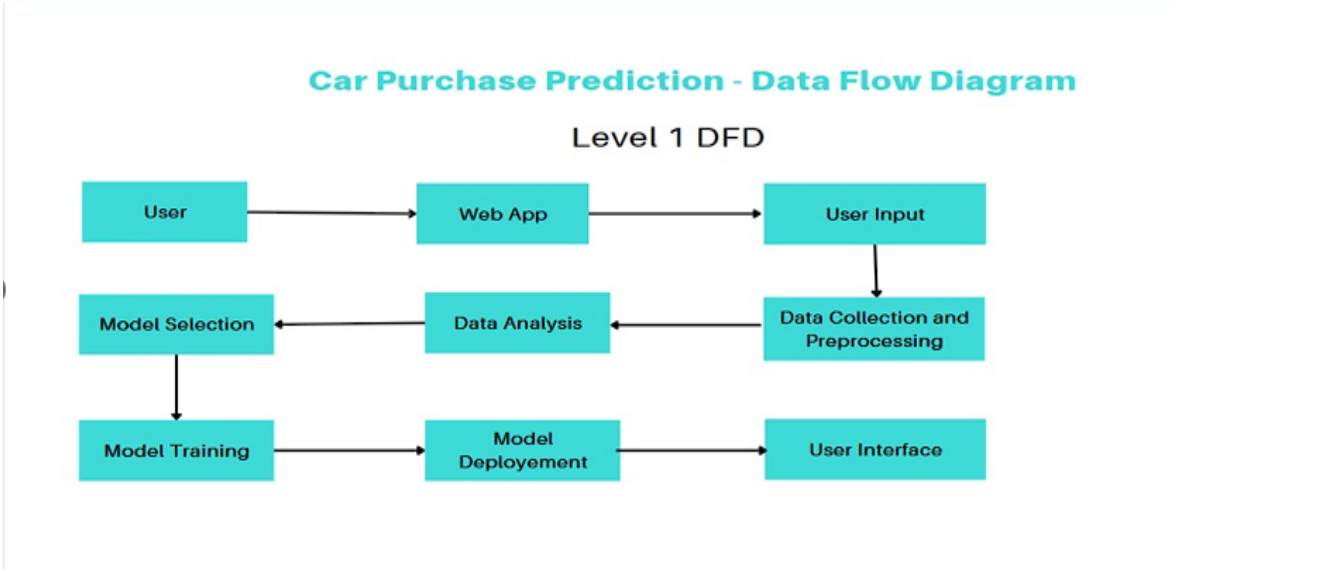
5. PROJECT DESIGN

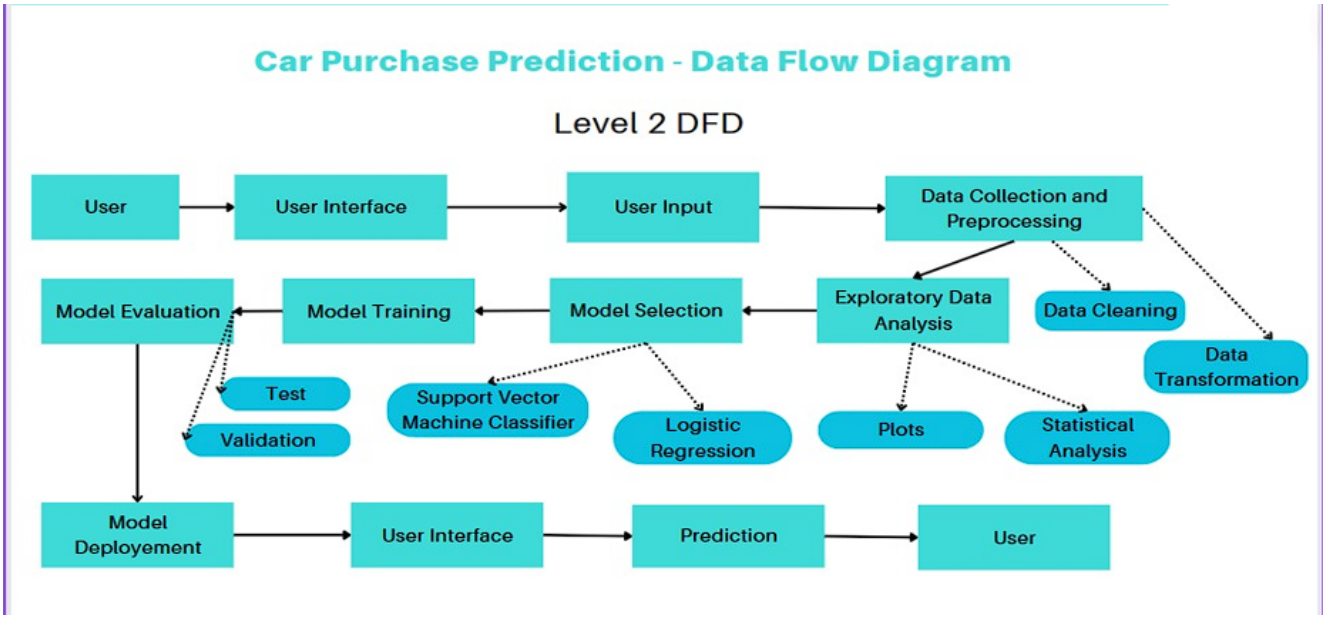
5.1 Data Flow Diagrams & User Stories

Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows data enters and leaves the system, what changes the information, and where data is stored.







User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story/Task	Acceptance Criteria	Priority	Release

Customer (Mobile user)	Prediction System	USN - 1	As a customer, I can utilize a professional predictive system to assess my car affordability by inputting my key details, including my user ID, gender, annual income, and age.	I am presented with a user-friendly interface to input my details for car affordability assessment.	High	Sprint-1
Customer (Web user)	Prediction System	USN - 2	As a customer, I want to have the ability to evaluate my capacity to afford a car.	I can obtain precise predictions	High	Sprint-1
Customer (Web User)	Website	USN - 3	As a customer, I am able to enter the inputs in a very friendly website.	I can access the website	High	Sprint-1

5.2 Solution Architecture

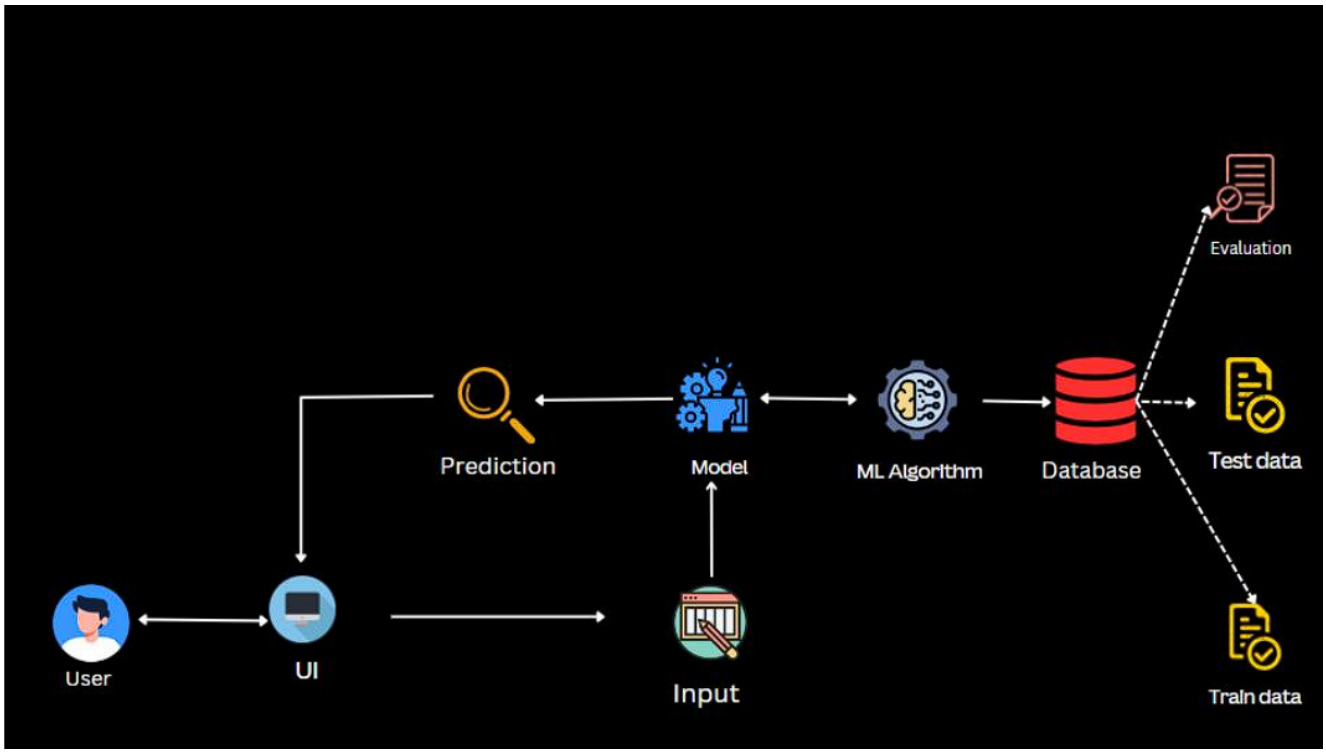
A solution architecture for car purchase prediction involves designing a system that uses data and machine learning techniques to predict whether a customer is likely to purchase a car. This system can be used by car dealerships and manufacturers to better target their marketing and sales efforts, and to improve their overall customer experience.

Our Solution leverages Machine learning algorithms and Web development technologies to address Car purchase prediction model effectively.

- Data Collection.
- Data Preprocessing.

● Model Building.
● Model Deployment.
● User-Interface.

Solution Architecture Diagram:



6. PROJECT PLANNING & SCHEDULING

6.1 Technical Architecture

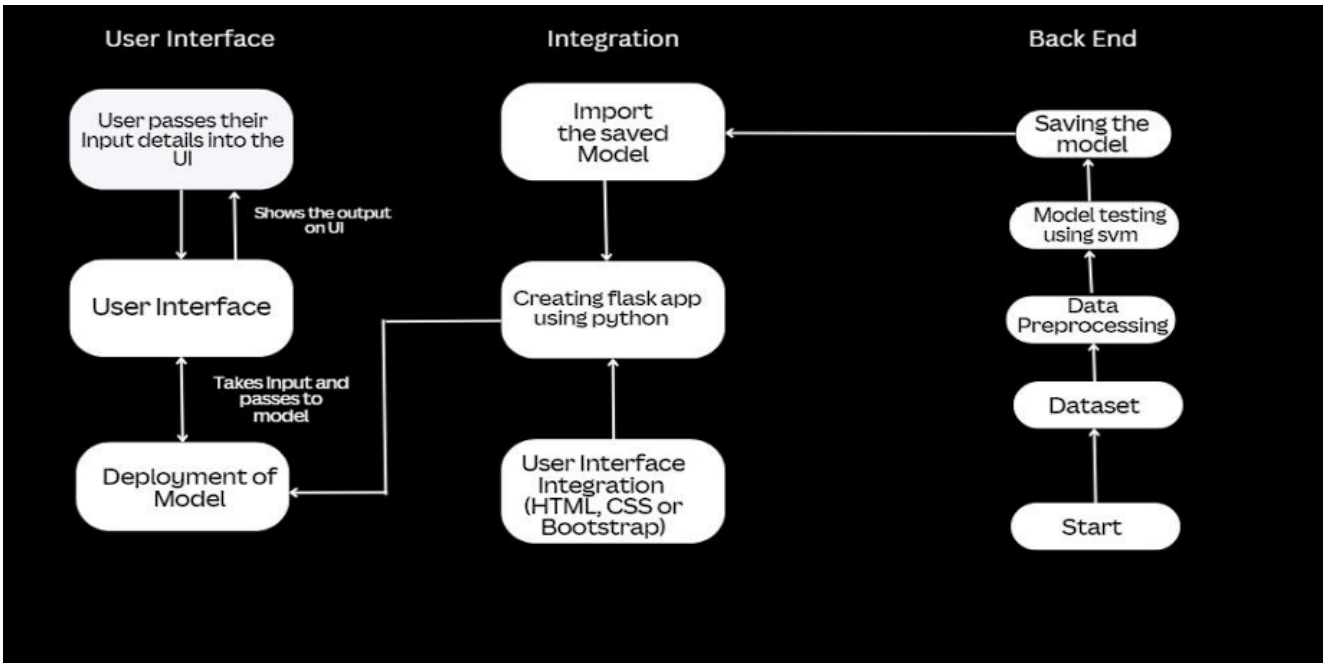


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	Python/Flask	Backend web framework for handling server requests	Python, Flask
2.	Model (SVM)	Machine learning model for car affordabilityprediction	Scikit-Learn (Python)
3.	StandardScaler	Data preprocessing for feature scaling	Scikit-Learn (Python)
4.	HTML/CSS	Frontend structure and styling	HTML, CSS
5.	Bootstrap	CSS framework for responsive web design	Bootstrap
6.	JavaScript	Scripting for smooth scrolling and GitHub links	JavaScript
7.	pickle	Serialization for saving and loading Python objects	Python (Standard Library)
8.	Pandas	Data manipulation and CSV file reading	Pandas (Python)
9.	GitHub	Hosting the project's code repository	GitHub

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Utilizes open-source frameworks for both the backend and frontend	Flask (Python), Bootstrap
2.	Security Implementations	Implements basic security measures such as forminput validation	Input validation, no detailed security implementation mentioned
3.	Scalable Architecture	The application is designed using a monolithic architecture	Monolithic architecture
4.	Availability	Availability is limited as the application runslocally	No load balancers or distributed serversare used
5.	Performance	Performance considerations includefrontend responsiveness	No specific performance optimizationtechniques mentioned

6.2 Sprint Planning & Estimation

Sprint	Functional Requirement(Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a customer, I can utilizea professional predictive system to assess my car affordability by inputting my key details, including my user ID, gender, annual income, and age.	8	High	SUTAM
Sprint-1		USN-2	As a customer, I want to have the abilityto evaluate my capacity to afford a car.	8	High	Kelvin J Anil

Sprint-2		USN-3	As a customer, I am ableto enter the inputs in a very friendly website.	4	Low	Amish Ranjan

7. CODING & SOLUTIONING

7.1 Feature 1

HTML Structure (index.html):

- Defines the layout of the web page.
- Includes navigation, user input form, about section, and footer.

Navigation Bar:

```
<nav>
    <div class="Heading">
    <div class="brand"><h1>Car Purchase Prediction </h1></div>
        <ul class="menu">
            <span id="Haboutus"><a href="#about">About Us</a></span>
        </ul>
    </div>

</nav>
```

Input Form :

```
<form action="{{url_for('predict')}}", method="post" id="calculate">
    <div class="inputBox">
        <input type ='number' required="required" name="userid">
        <span>User ID</span>
    </div>
    <div  id="Gender">
    <span>GENDER:</span>
    <span id="maleG">
    <input type="radio" name = "gender" value="Male" required="required"> Male
    </span>

    <span id="femaleG">
    <input type="radio" name = "gender" value="Female" required="required" > Female
    </span>
    </div>
```

```
    <div class="inputBox">
        <input type ='number' required="required" name="age">
        <span>Age</span>
    </div>

    <div class="inputBox">
        <input type ='number' required="required" name="salary">
        <span>Annual Salary</span>
    </div>
    <div>
        <button id="predict-button" class="predictBox">Predict</button>
    </div>
    <div class="predictionText">{{ prediction_text }}</div>

</form>
```

Footer Section:

```
<footer class="footer">
    <div class="container">
        <div class="row">
            <div class="footer-col" >
                <h4>About the Project</h4>
                <ul>
                    <li><div class="typing"><h5>Welcome to our Car Purchase Prediction project! Here,
                        we have created an intelligent system that assists users
                        in predicting their car affordability based on key details
                        like age, gender, and annual salary. Powered by machine learning,
                        our application uses a trained model to provide personalized predictions.
                        Explore the innovative work of our team members, and feel free to check
                        out the project on GitHub for more details and contributions.</h5></div></li>
                </ul>
            </div>
```

```
        <div class="footer-col" id="team">
            <h4>Team members</h4>
            <ul>
                <li><a href="#">Kelvin J Anil</a></li>
                <li><a href="#">Sutam Chandra</a></li>
                <li><a href="#">Rishi raj Upadhyaya</a></li>
                <li><a href="#">Amish Ranjan</a></li>
            </ul>
        </div>
        <div class="footer-col" id="team1">
            <h4>See our Project on GitHub</h4>
            <div class="social-links">
                <a href="https://github.com/smartinternz02/SI-GuidedProject-600946-1697539328"><i class="f
```

7.2 Feature 2

Smooth Scrolling Script (JavaScript):

- Enables smooth scrolling to the "About Us" section.

```
<script>
    // Script for smooth scrolling to the "About Us" section
    document.querySelectorAll('a[href^="#"]').forEach(anchor => {
        anchor.addEventListener('click', function (e) {
            e.preventDefault();
            document.querySelector(this.getAttribute('href')).scrollIntoView({
                behavior: 'smooth'
            });
        });
    });

    </script>
```

7.3 Feature 3

Flask Web Application (app.py):

- Implements the server-side functionality.

```python
@app.route("/predict", methods=["POST"])
def predict():
    form_values = request.form.to_dict()
    user_id = int(form_values.get("userid",0))
    gender = 1 if form_values.get("gender", "").lower() == 'male' else 0
    age = int(form_values.get("age", 0))
    salary = int(form_values.get("salary", 0))

    # Exclude the "User ID" feature from the input data
    input = [gender, age, salary]

    # Standardize the data using the loaded scaler
    input_reshape = np.asarray(input).reshape(1, -1)
    std_input = scaler.transform(input_reshape)

    # Predicting the output
    prediction = model.predict(std_input)

    result = "purchase a car" if prediction == 1 else "not purchase a car"

    features_str = ", ".join(map(str, input))
    prediction_text = f"You should {result}"
    # prediction_text = f"You should {result}. Features: {features_str}"

    return render_template("index.html", prediction_text=prediction_text)
```

8. PERFORMANCE TESTING

8.1 Performance Metrics

Accuracy Score:

Training Accuracy Score:

```
# Evaluate the Model

# Checking the accuracy
X_train_prediction = Classifier.predict(X_train)
Training_data_accuracy = accuracy_score(X_train_prediction,Y_train)
print(Training_data_accuracy)
```

```
0.82875
```

Testing Accuracy Score:

```
# Checking the accuracy of test data
X_test_prediction = Classifier.predict(X_test)
Testing_data_accuracy = accuracy_score(X_test_prediction,Y_test)
print(Testing_data_accuracy)
```

```
0.815
```

Classification Report:

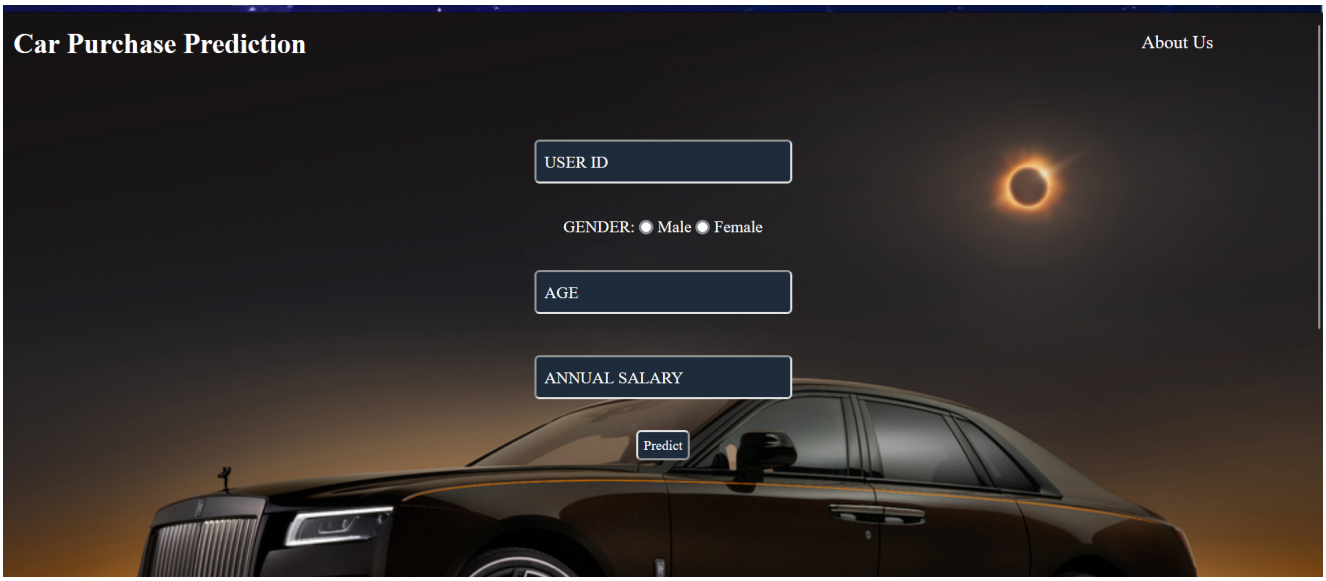
```
# classification report
# Assuming you have X_test and Y_test
predictions = Classifier.predict(X_test)

# Generate the classification report
report = classification_report(Y_test, predictions)

# Print or use the report as needed
print(report)
```

```
              precision    recall  f1-score   support

           0       0.83      0.87      0.85       120
           1       0.79      0.74      0.76        80

    accuracy                           0.81       200
   macro avg       0.81      0.80      0.81       200
weighted avg       0.81      0.81      0.81       200
```

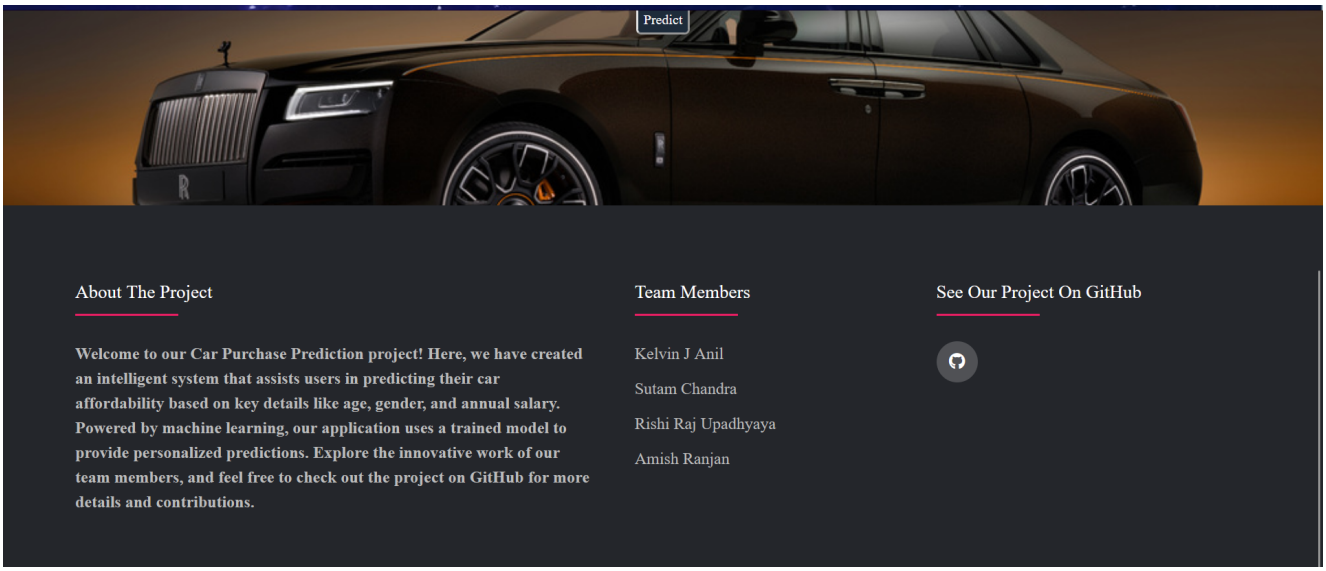
9. RESULTS

9.1 Output Screenshots

Website:

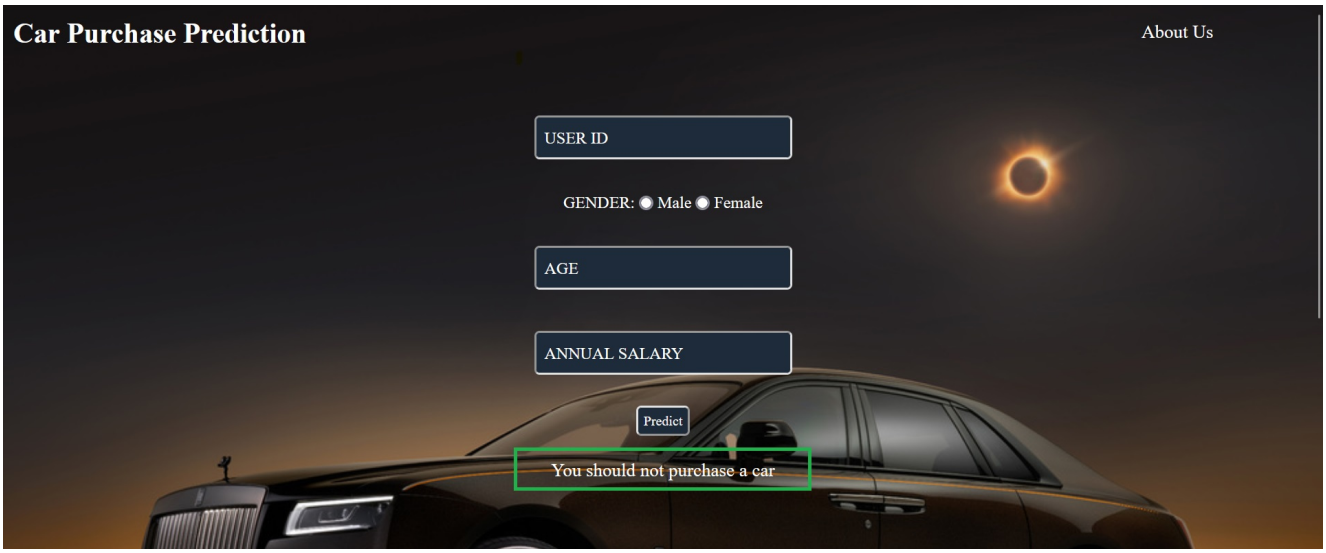


About Us:



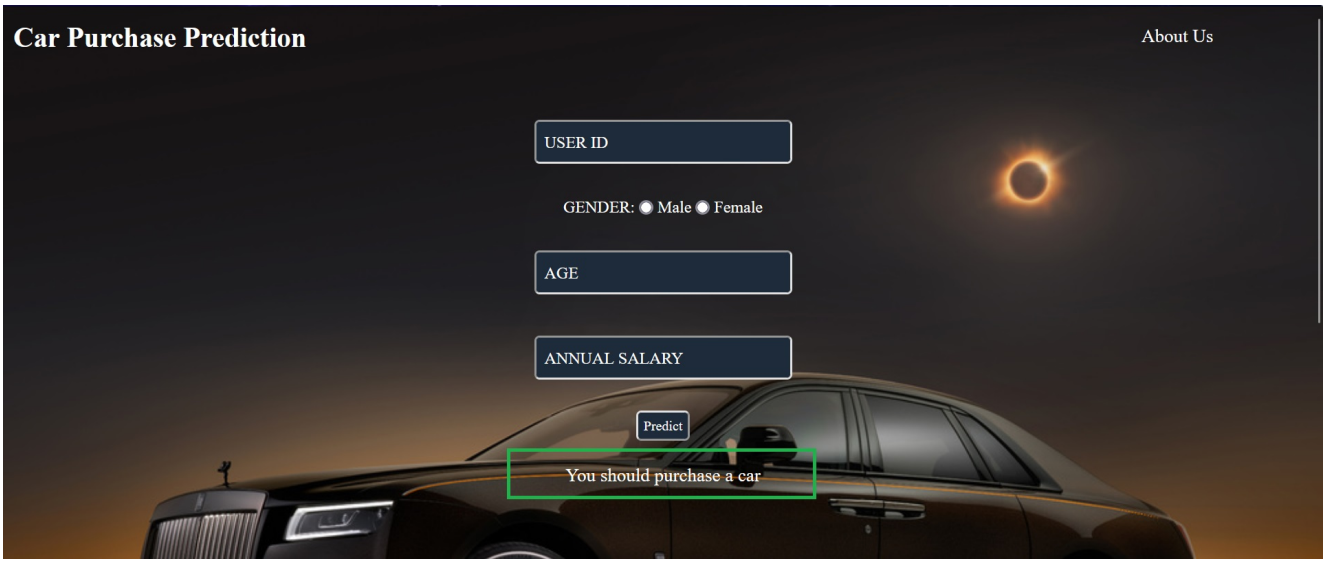
Sample Input 1:

Sample Output 1:



Sample input 2:

Sample Output 2:



10. ADVANTAGES & DISADVANTAGES

Advantages:

1. Informed Decision-Making: The project empowers users to make informed decisions about purchasing a car by providing personalized predictions based on their input details.

2. User-Friendly Interface: The web application features a user-friendly interface, making it accessible to a wide audience. The form inputs are straightforward, allowing users to easily interact with the prediction system.

3. Machine Learning Integration: The incorporation of machine learning enables accurate predictions, enhancing the reliability of the system. Users can benefit from advanced analytics without delving into complex modeling processes.

4. Open Source Collaboration: The project is hosted on GitHub, fostering collaboration and transparency. The open-source nature allows developers and contributors to enhance and optimize the application.

Disadvantages:

1. Dependency on Data Quality: The accuracy of predictions heavily relies on the quality of input data. Inaccurate or incomplete user details may lead to less reliable predictions, affecting the overall performance of the system.

2. Limited Scope: The project focuses specifically on car affordability prediction and may not cover all factors influencing a user's decision to purchase a car. Users should be aware of the project's limitations in providing a comprehensive analysis.

3. Security Concerns: While not explicitly mentioned, handling user data, even for prediction purposes, requires attention to security measures. Ensuring the confidentiality and integrity of user information is crucial to build trust among users.

4. Model Maintenance: As the project uses a machine learning model, ongoing maintenance is necessary. This includes monitoring model performance, updating the model with new data, and addressing potential biases that may emerge over time.

11. CONCLUSION

In conclusion, the Car Purchase Prediction project offers a valuable tool for users seeking clarity in their car-buying decisions. By leveraging machine learning, the application provides personalized insights based on user inputs, fostering informed choices. The user-friendly interface enhances accessibility, while the open-source collaboration on GitHub encourages community engagement and continuous improvement.

However, it's essential to acknowledge certain considerations. The project's accuracy hinges on the quality of user data, and users should be aware of the system's scope limitations. Attention to security measures is paramount to ensure the protection of user information. Ongoing model maintenance is also crucial for sustained performance and addressing potential biases.

In essence, the project presents a promising avenue for enhancing decision-making in the realm of car purchases. As technology evolves and user feedback contributes to refinement, the Car Purchase Prediction project stands as a testament to the fusion of machine learning and user-centric design in the automotive domain.

12. FUTURE SCOPE

The Car Purchase Prediction project lays a solid foundation for future enhancements and expansions. Here are some potential avenues for future development:

1. **Feature Enrichment:** Introduce additional features that may influence car affordability, such as credit score, location, or employment status. Expanding the input variables can enhance the predictive capabilities of the model.

2. **Advanced Machine Learning Models:** Explore the integration of more advanced machine learning models or ensemble methods to further improve prediction accuracy. Experimenting with different algorithms can lead to better performance and robustness.

3. **Real-time Data Integration:** Implement mechanisms for real-time data updates to keep the model informed with the latest market trends and economic changes. This ensures that predictions are based on the most current information.

4. **User Authentication and Personalization:** Introduce user authentication to create personalized profiles. This allows the system to learn from user behavior over time, refining predictions based on individual preferences and patterns.

5. **Feedback Mechanism:** Incorporate a feedback loop where users can provide feedback on the accuracy of predictions. This valuable input can be used to continuously refine and optimize the model.

6. **Mobile Application Development:** Extend the reach of the project by developing a mobile application. This provides users with the convenience of accessing the prediction tool on their smartphones, making it more accessible on the go.

7. **Comprehensive Financial Analysis:** Expand the project's scope to include a more comprehensive financial analysis, considering factors like loan interest rates, insurance costs, and potential trade-ins. This provides users with a

holistic view of the financial aspects of car ownership.

8. **Internationalization and Localization:** Adapt the application for a global audience by incorporating internationalization and localization features. This includes supporting multiple languages and accommodating regional variations in financial practices.

9. **Community Engagement:** Foster a strong community around the project by actively seeking contributions, addressing issues, and incorporating ideas from developers and users. A vibrant community can contribute to the project's sustainability and growth.

10. **Integration with External APIs:** Collaborate with external APIs (e.g., car price databases, economic indicators) to enrich the dataset and improve the accuracy of predictions with up-to-date information.

By embracing these future-oriented strategies, the Car Purchase Prediction project can evolve into a dynamic and comprehensive tool, staying relevant and beneficial to a broader user base.

13. APPENDIX

GitHub: https://github.com/smartinternz02/SI-GuidedProject-600946-1697539328/tree/main

Colab:
https://colab.research.google.com/drive/1Xr_PojaGHS-k8CxQFaw_Myvlv6MsDDEf?authuser=0#scrollTo=BVpHduuGvljk

Demo Link: https://drive.google.com/file/d/1J-7BJDVabBc9G3ngVDd70KRRrc_hIi8T/view?usp=drive_link