

ENSE 374 – Software Engineering Management

GearShare

Linton J Dsouza (200470698)

Cirus Chakma (200495194)

Maheen Siddique (200480228)

Table of Contents

1	Introduction..	3
2	Design Problem..	6
2.1	Problem Definition/Business Case.	6
2.2	Project Charter	6
3	Solution..	7
3.1	Solution 1.	7
3.2	Solution 2.	7
3.3	Final Solution.	7
3.3.1	Components.	7
3.3.2	Features.	7
3.3.3	Environmental, Societal, Safety, and Economic Considerations.	7
3.3.4	Limitations.	7
4	Team Work.	8
4.1	Meeting 1.	8
4.2	Meeting 2.	8
4.3	Meeting 3.	8
4.4	Meeting 4.	8
4.5	Meeting 5.	8
4.6	Meeting 6.	8
4.7	Meeting 7.	8
4.8	Meeting 8.	8
5	Project Management	9
6	Conclusion and Future Work.	10
7	References.	11
8	Appendix.	12

1 Introduction

This project aims to create a web-based application that allows individuals to rent vehicles directly from other vehicle owners. Unlike traditional car rental services or rideshare platforms like Uber, users of this service would be responsible for selecting and booking a car from other users, creating a peer-to-peer marketplace. Similar to platforms like Turo, this app taps into the sharing economy, offering convenience and cost savings by enabling car owners to monetize their vehicles when not in use. The platform will include features like vehicle listing, search and booking functionality, secure payments, and user reviews.

Project Design and Background

This project aims to design a web-based application that facilitates peer-to-peer vehicle rentals. Users can list their vehicles, search for available rentals, book vehicles, and securely process payments. This platform taps into the sharing economy, where vehicle owners can monetize their unused vehicles, providing convenience and cost savings to renters by offering a decentralized alternative to traditional rental companies. Similar to platforms like Turo, this application supports direct interaction between owners and renters, offering a personalized rental experience, with the added layer of user reviews for trust and transparency.

Relevant Background Information

The rise of the sharing economy has significantly impacted various industries, including transportation. Companies like Uber and Turo have demonstrated the potential for peer-to-peer platforms to disrupt traditional services by empowering individuals to monetize their assets. This trend is driven by the increasing need for convenience, affordability, and flexibility in transportation solutions, making peer-to-peer vehicle rentals a logical extension of this model. Unlike traditional rental services, these platforms leverage user trust, reviews, and direct interactions to create a marketplace built on shared resources.

Rationale

The need for a platform like this arises from several factors:

1. **Consumer Convenience:** Renting directly from a vehicle owner may offer more flexibility, personalization, and affordability than traditional rental services.
2. **Cost Efficiency:** By cutting out the middleman, both vehicle owners and renters can benefit from lower costs.
3. **Market Demand:** As seen with Turo's success, there is clear demand for a peer-to-peer car rental service that benefits both owners and renters.

Stakeholder Requirements

Stakeholders have outlined the following requirements:

1. **Vehicle Listing and Management:** Vehicle owners should be able to list their cars, including details like availability, pricing, and photos.
2. **Search and Booking :** Renters must have a user-friendly interface to search for available vehicles by location, type, and price, with real-time availability.
3. **Secure Payment Gateway:** A trusted and secure method for handling payments between vehicle owners and renters is essential.
4. **User Reviews and Ratings:** To build trust, both renters and owners should be able to rate their experiences and leave feedback.

For further details, you can review the [Project Requirements Document].

Overview of Upcoming Sections

In the subsequent sections, the document will cover the following:

- **Detailed Design Specifications:** A breakdown of the technical architecture, user interface design, and system workflows.
- **Implementation Strategy:** A phased plan outlining the development and deployment of the platform, including testing and user feedback mechanisms.
- **Challenges and Considerations:** An analysis of potential obstacles, such as legal issues, user acquisition, and platform scalability.

2 Problem Definition

2.1 Business Case

Business Case Link-

[https://github.com/Lintondsouza/ENSE-374-Project/blob/main/Business%20Case%20%20\(1\).pdf](https://github.com/Lintondsouza/ENSE-374-Project/blob/main/Business%20Case%20%20(1).pdf)

2.2 Project Charter

Project Charter Link-

<https://github.com/Lintondsouza/ENSE-374-Project/blob/81e1dd22eca7465b100a3c8bedabf4ce141f49fb/Project%20Charter.docx>

3 Solution

3.1 Solution 1

The very first iteration of our GearShare app that we brainstormed was a good base to build on. We listed the fundamentals such as:

- Users being able to select a date for their pickup
- Users being able to select a location for their pickup
- Users being able to mention a price range that they prefer to browse at
- Users being able to search for a car of their choice
- Pricing Plans page

Some of the troubles we ran into while brainstorming included:

- **The scope of the project:** Who will be the renters? Is it a peer-to-peer platform for individuals to rent to each other, or will it be a service that corporations can adopt to make their fleets available? Understanding the target audience is essential because a peer-to-peer platform would require unique considerations for trust, security, and seamless payments, whereas a business-facing app would need to cater to fleet management and bulk rental options.
- **Required features for elevated user experience:** What kind of user interface and experience do we want to offer? Should it be sleek with animations and high interactivity, or more generic and straightforward to ensure that users of all backgrounds can navigate it easily? Additionally, the first iteration may lack certain helpful features like a rating system, which are essential for improving user engagement and satisfaction.

Limitations of the First Iteration and Why It's Not Fully Viable:

While the initial version provided a solid foundation, it faced challenges that limit its potential to function effectively as a fully viable product:

1. **Basic Search and Filter Options:** While users can search for a car by name and set a price range, the search functionality is basic.
2. **No set time of pickup:** We realized that a date and location is not enough. To be more convenient, we needed to implement a way to specify the time of the meetup.

How to Improve?

We arrived at the conclusion that a standard rating system was definitely required in order to uphold integrity and standards. To widen our userbase, an easy-to-learn interface would be much better than an engaging one. We also decided to be more upfront with pricing, displaying the rates as one of the first pieces of information about a given listing.

3.2 Solution 2

1. Target Audience

- **Renters who use the service to reserve automobiles:**
The interface is specifically designed for users looking to rent vehicles, ensuring the user experience is tailored to their needs.

2. Core Features

- **Location and Date Selection:**
 - Users can select their **location**: This allows renters to search for available vehicles based on their desired location, making it easier to find cars nearby.
 - Users can choose **pick-up and drop-off dates and times**: By allowing renters to specify the time frames for renting the car, the system can provide more accurate availability and pricing options.
- **Search Function:**
 - **Initiated by the location and date selection**: Once users input their location and time preferences, the search feature is activated to start filtering available cars.
 - **Directs users to the main site content**: After initiating the search, users are guided to the primary content of the site where detailed vehicle listings and options are displayed.

3. Search Customization

- **Filter Options:**
 - Various filters are available to refine search results: Users are provided with additional filters such as vehicle type, price range, or other preferences, allowing for a more tailored and efficient search experience.

4. Additional Browsing Options

- **Alternative Exploration Page:**
 - **Provides browsing options for users without specific preferences**: For users who are uncertain about their preferences, this page offers a broader range of choices to explore.
 - **Displays popular choices made by other users**: It highlights vehicles frequently selected by other users, helping indecisive renters make informed decisions based on popularity.
 - **Organizes these options into categorized sections for easy browsing and discovery**: The content is sorted into categories, making it easier for users to navigate and find vehicles that suit their needs.

Drawbacks

1. Improvements in the Current Design

- **Progress from Initial Solution:**
The current design shows improvement over the initial version, addressing some earlier issues but still missing essential features.

2. Missing Key Functionalities

- **User Interface for Car Owners:**
The system lacks a dedicated interface for car owners, which would allow them to manage their vehicles, view bookings, and interact with the platform.
- **Car Tracking for Rented Vehicles:**
There are no capabilities to track the status or location of rented vehicles, which is critical for both security and operational oversight.
- **Payment Processing Features:**
The absence of payment processing functionalities hinders the platform's ability to handle transactions, making it incomplete as a service-oriented system.
- **Vehicle Management Tools for Owners:**
Car owners cannot manage or update vehicle details such as availability, pricing, or maintenance records, which are important for effective platform use.
- **Display of Critical Vehicle Information:**
Vital vehicle details, such as condition, mileage, and usage history, are not displayed, which could affect decision-making for both renters and owners.

3. Additional Missing Component

- **Ranking of Automobiles by Usage Duration:**
The system does not feature the ability to rank cars based on their usage duration by renters. This component could provide valuable insights to both renters (to make informed decisions) and owners (to better manage vehicle availability).

4. Impact of Missing Functionalities

- **Consistent User Experience:**
These missing features are essential for delivering a seamless and consistent experience for both car owners and renters. Without them, the platform fails to meet user expectations.
- **Operational Efficiency:**
Critical functions, such as tracking and payment processing, are crucial for smooth operations. Their absence limits the system's ability to perform efficiently and effectively.

5. Overall System Shortcomings

- **Failing to Meet the Full Spectrum of User Demands:**
Without these fundamental components, the platform does not address the

complete range of needs for both car owners and renters, limiting its appeal and usefulness.

- **Limitations in Functionality and User Satisfaction:**

The lack of key features reduces the platform's overall functionality and negatively impacts user satisfaction, as it doesn't fully support the end-to-end process of car rental.

6. Recommendations for Future Iterations

- **Integration of Missing Features:**

Future design revisions should focus on incorporating these essential functionalities to enhance the platform's utility, user experience, and overall performance.

3.3 Final Solution

Our final solution for the GearShare car rental platform was created with both car seekers and car suppliers in mind, resulting in a smooth user experience for both parties. The platform's main feature is its capacity to establish a stable peer-to-peer connection, which was successfully implemented in the final solution.

User-Centric Design:

- The GearShare car rental platform was carefully crafted to cater to the needs of both **car seekers** and **car providers**, ensuring a user-friendly experience for both roles.
- The primary goal was to establish a seamless **peer-to-peer connection** between users, which was successfully achieved in the final implementation.

Core Features and Workflow:

- **Request Submission:**
 - Car seekers initiate the process by submitting rental requests through the website.
 - These requests are securely stored in the **database** and routed to the respective car providers or owners.
- **Request Approval:**
 - Providers receive notifications about incoming requests and can review the details.
 - Once a request is accepted by the provider, it is updated in the system as **approved**, officially confirming the booking.
- **Car Listings:**

- The home page prominently displays all available cars listed by providers, enabling car seekers to browse and select vehicles conveniently.
- **Car Listings for Providers:**
 - Providers can effortlessly list their vehicles, ensuring their offerings are visible to potential renters.

User Profiles:

- Both car seekers and providers have access to personalized **user profiles**.
- Profiles display essential details such as user information, request history, and status updates, fostering transparency and trust between users.

Scalable and Robust Solution:

- The final solution was designed to streamline the interaction between car seekers and providers while maintaining efficiency and reliability.
- The architecture ensures that the platform is capable of handling an increasing number of users and requests without compromising performance.

The initial solution was static HTML pages, with initial ideas only conceived. A second design focused on automobile providers, incorporating booking tables, dynamic listing, and overbooking check, but did not meet final design criteria. The final feature achieved peer-to-peer connectivity within the project's limits, removing some capabilities.

Comparison of GearShare Design Solutions

Aspect	First Solution	Second Solution	Final Solution
Design Approach	Static HTML pages with conceptual ideas.	Focused primarily on the car provider's perspective.	Balanced approach, addressing both car seeker and provider needs.

Key Features	None of the core components were implemented.	Introduced features like booking table, interactive listing, request tabs, and overbooking check.	Retained only essential features, emphasizing peer-to-peer connection while adhering to constraints.
Target Audience	General conceptualization, not tailored to specific user groups.	Focused solely on car providers, overlooking car seeker functionality.	Equally targeted car seekers and car providers, ensuring a seamless interaction.
Limitations	Too basic, lacked functionality to be considered a viable design.	One-sided design with limited focus on car seekers, making it incomplete.	Successfully met the project objective of enabling peer-to-peer connections.
Final Assessment	Did not qualify as a working solution.	Did not qualify as a final design due to lack of balance.	Achieved project goals and qualified as the final design.

3.3.1 Components

What components you used in the solution? What is the main purpose of using individual component? Provide a block diagram (with a numbered caption, such as Fig. 1) representing the connectivity and interaction between all the components.

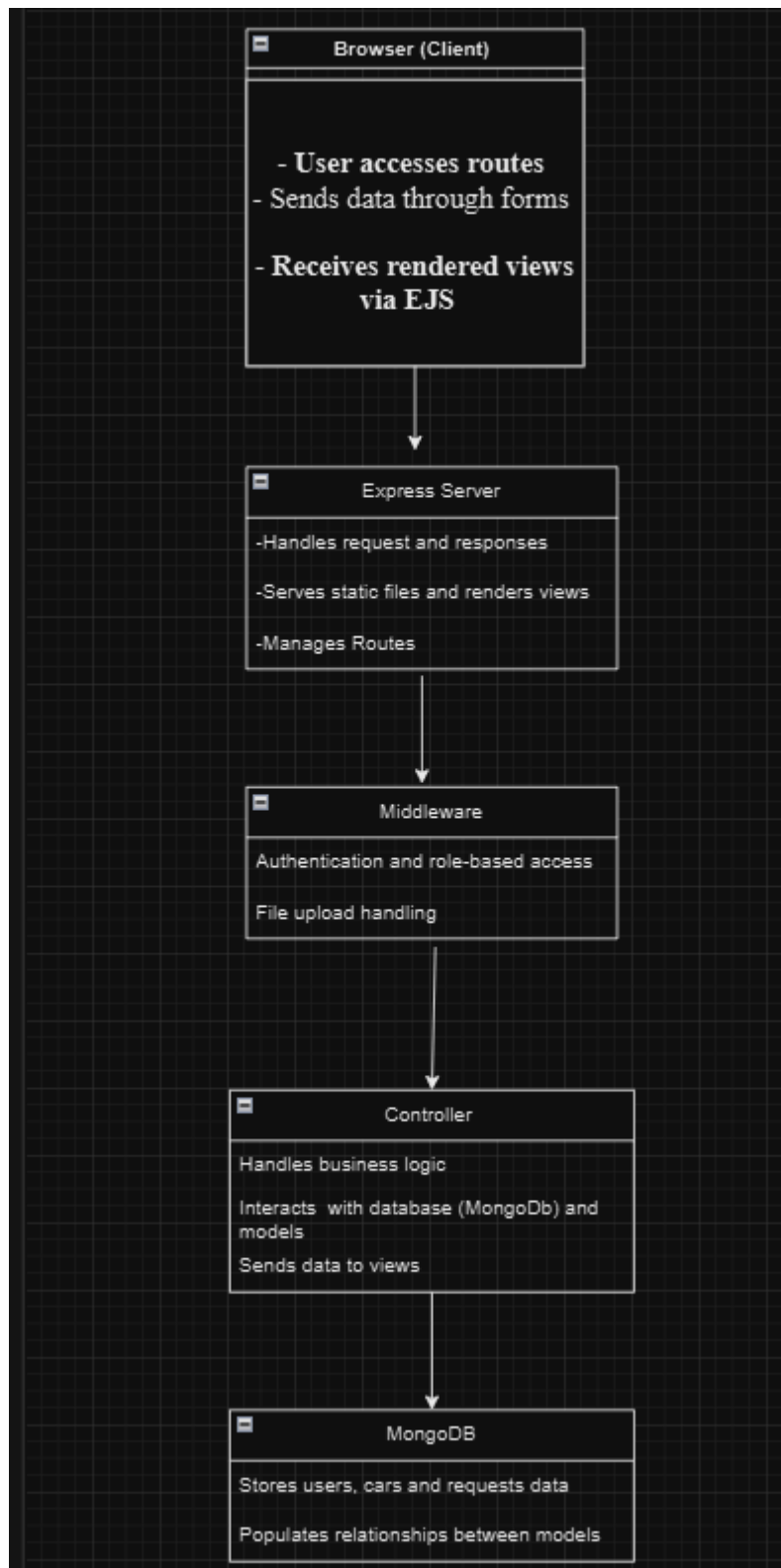
Components Used:

1. Express.js

- Purpose: To create the server and define routes for handling HTTP requests.
- Functionality: Provides a structured way to build web applications with middleware for request handling, routing, and rendering.

2. EJS (Embedded JavaScript)

- Purpose: For dynamic rendering of HTML views based on server-side data.
 - Functionality: Injects server-side variables like `requests`, `user`, and `activePage` into HTML templates.
- 3. MongoDB with Mongoose**
- Purpose: To store, retrieve, and manipulate data such as user details, car listings, and requests.
 - Functionality: Mongoose simplifies the interaction with MongoDB by providing schema-based data modeling.
- 4. Session Management (Express-Session)**
- Purpose: To track user sessions (authentication and user data persistence).
 - Functionality: Stores the user's session data (e.g., `req.session.user`).
- 5. Middleware**
- AuthMiddleware:
 - Purpose: To restrict access to routes based on authentication and role.
 - Functionality: Ensures only authenticated users with the correct roles can access specific routes.
 - File Upload Middleware:
 - Purpose: To handle file uploads such as profile photos and car images.
 - Functionality: Manages the upload directory, file naming, and type restrictions.
- 6. Controllers**
- Purpose: To centralize business logic for routes like user signup, login, request handling, and car management.
 - Functionality: Queries the database and sends responses to the views or API.
- 7. Routing Layer**
- Purpose: To map URLs to specific controllers and functions.
 - Functionality: Each route (e.g., `/users/providers/signup`, `/seekers/requests`) invokes the appropriate controller.
- 8. Bootstrap CSS**
- Purpose: To provide a responsive and visually appealing frontend.
 - Functionality: Simplifies the design of components like navbar, forms, and cards.
- 9. Populated Models (via Mongoose)**
- Purpose: To fetch and link related data (e.g., car details in a request, provider information).
 - Functionality: Ensures that connected data is automatically retrieved via references.



3.3.2 Features

Feature	Functionality
Search and Filter	Simplify searching for a car that meets location and cost preferences.
Request Forms and Approve/Decline	Option for providers to approve requests more convenient for them and decline requests they can't accommodate. Sleek-looking form to fill out location and time information.
Requests Page	Bookkeeping for providers and seekers to track their approved, declined and pending requests. Contains data regarding the meeting details and duration of rental.
Provider and Seeker profile pages	Simple, yet efficient profile pages. Providers can see their listed cars along with their own information. Seekers
Create Listings Page	Page for providers to create a listing. Fields to input vehicle type, make, model and price/day.

3.3.3 Environmental, Societal, Safety, and Economic Considerations

Environmental Consideration

GearShare emphasizes the positive environmental impact of shared mobility. By enabling people to rent underutilized vehicles, the platform helps reduce the need for additional vehicle production and promotes more efficient use of existing resources. This reduces overall carbon emissions by minimizing the number of privately owned, rarely used vehicles. Furthermore, by making these vehicles accessible to a wider audience, GearShare lessens

reliance on alternative transportation methods that can strain other sectors and contribute to higher emissions.

Societal Consideration:

GearShare fosters a sense of community by encouraging resource sharing and providing a cost-effective transportation option for individuals who may not have access to a vehicle. It also supports equitable mobility by giving underserved communities access to affordable and convenient transportation solutions.

Safety Consideration:

GearShare places a strong emphasis on protecting user privacy and ensuring data security. All personal and financial information is encrypted and hashed to prevent unauthorized access. The platform implements strict data privacy measures, ensuring that user data is only used for the intended purposes of facilitating rentals. These measures ensure that both renters and vehicle owners can trust GearShare with their sensitive data.

Economic Consideration:

GearShare boosts economic opportunities for vehicle owners by turning idle vehicles into sources of income. Seekers benefit from another alternative for car rental services. The platform also supports local economies by enabling greater access to transportation, facilitating work commutes, and does its part in addressing mobility gaps in urban and suburban areas.

3.3.4 Limitations

While GearShare offers a practical solution to underutilized vehicles, there are certain feature gaps and challenges that could affect its functionality and user experience:

- 1. Lack of Feedback System:**

Currently, GearShare does not have a feature for renters or vehicle owners to leave feedback or rate each other. This can make it difficult to build trust and ensure accountability within the community.

- 2. No Penalty for No-Shows:**

The platform does not have a mechanism to penalize users—either renters or vehicle owners—for failing to show up for agreed bookings. This could lead to frustration and wasted time for the affected parties.

- 3. Inadequate Moderation of Troll Activity:**

GearShare lacks robust moderation features to prevent or address trolling or abusive behavior. Users can potentially misuse the platform by listing fake vehicles, or making fraudulent bookings without consequences as of yet. On the plus side, these features are very much in our short term plans.

4. **No Dynamic Pricing or Incentive Structures:** There are no features to encourage vehicle availability during high-demand periods, such as dynamic pricing or incentives for owners to list their cars.

4 Team Work

Since this is a group project, you must have a fair distribution of tasks among yourselves. To this end, you must hold meetings to discuss the distribution of tasks and to keep a track of the project progress.

4.1 Meeting 1

Link: https://github.com/Lintondsouza/ENSE-374-Project/blob/81e1dd22eca7465b100a3c8bedabf4ce141f49fb/Meeting_Minutes/First%20Meeting.docx

4.2 Meeting 2

Link: https://github.com/Lintondsouza/ENSE-374-Project/blob/81e1dd22eca7465b100a3c8bedabf4ce141f49fb/Meeting_Minutes/Meeting%202.docx

4.3 Meeting 3

Link: https://github.com/Lintondsouza/ENSE-374-Project/blob/81e1dd22eca7465b100a3c8bedabf4ce141f49fb/Meeting_Minutes/Meeting%203.docx

4.4 Meeting 4

Link: https://github.com/Lintondsouza/ENSE-374-Project/blob/696be4607dc54d56f6994903ec4d9f40ef458436/Meeting_Minutes/Meeting%204.docx

4.5 Meeting 5

Link:https://github.com/Lintondsouza/ENSE-374-Project/blob/aa6a50e71d5add53d8c1c7663f899d027b2c75bf/Meeting_Minutes/Meeting%205.docx

4.6 Meeting 6

Link:https://github.com/Lintondsouza/ENSE-374-Project/blob/main/Meeting_Minutes/Meeting%206.docx

4.7 Meeting 7

Link:https://github.com/Lintondsouza/ENSE-374-Project/blob/main/Meeting_Minutes/Meeting%207.docx

4.8 Meeting 8

Link:https://github.com/Lintondsouza/ENSE-374-Project/blob/main/Meeting_Minutes/Meeting%208.docx

5 Project Management

Link:<https://github.com/Lintondsouza/ENSE-374-Project/blob/aa6a50e71d5add53d8c1c7663f899d027b2c75bf/Milestone-Based%20Schedule.pdf>

6 Conclusion and Future Work

GearShare simplifies peer-to-peer car rentals by connecting seekers with providers in a streamlined process. Through a seamless platform, seekers can explore car listings, which were listed by the providers, submit rental requests specifying duration and meeting location, and await provider approval. By offering flexibility and convenience, GearShare empowers providers to manage their listings and seekers to find suitable vehicles for their needs.

GearShare has massive potential to grow and evolve. Some ideas we believe are attainable in short term and long term include:

Short Term (5-6 months)

- An income tracker for providers
- Chat features for hashing out details to meet with ease.
- Report feature to outline suspicious individuals.
- Booking list
- Moderation features such as penalties to limit misuse of app.

Long Term (1+ year)

- Payment Portal implemented
- More seamless peer-to-peer functionality including live-location share
- Request logs for seekers.

Lesson Learned Report:

Link: <https://github.com/Lintondsouza/ENSE-374-Project/blob/main/Lessons%20Learned%20Report.docx>

7 References

- Use the IEEE reference style.
- Do not put any reference if it is not cited in the text.

8 Appendix

If you want to provide additional information, use this appendix.