



# E-commerce website for community car-rental service

# Table Of Contents

<b>1. Introduction</b>	<b>5</b>
1.1 Purpose of this document:	5
1.2 Identification:	5
1.3 Document Scope:	5
1.4 Intended Audience:	5
1.5 Stakeholder Roles/Responsibilities/Concerns:	6
1.5.1 Regional Stakeholders	6
1.5.2 Technical / Project Stakeholders	7
1.6 Languages and Tools	7
<b>2. General Overview And Design Consideration</b>	<b>8</b>
2.1 General Overview:	8
2.2 Current:	8
2.2.1 Proposed Solution - Statement of Need:	8
2.3 System Assumptions	9
2.4 System Constraints:	9
<b>3. Design Considerations</b>	<b>9</b>
3.1 Goals and Guidelines:	9
3.1.1 Architecture	9
3.1.2 Ease of Use	9
3.1.3 Development Environment	9
3.1.4 Extensibility	10
3.2 Operational Environment:	10

3.3 Development Methods & Contingencies:	10
3.1.1 Scalability	10
3.1.2 Availability	11
3.1.3 Extensibility	11
3.1.4 Security	11
3.1.5 Functionality	11
3.1.6 Separation of responsibility	11
3.4 Development Environment:	11
<b>4. System Architecture and Architecture Design</b>	<b>12</b>
4.1 Topology Diagram:	12
4.2 Hardware Architecture	12
4.2.1 System Descriptions	12
4.3 Software Architecture	14
4.3.1 Software Element	14
4.3.2 Performance Software Architecture	16
4.4 Information Architecture	17
4.4.1 Data	17
4.4.1.1 Data Inputs	17
<b>5. System Design</b>	<b>17</b>
5.1 Business Requirements:	17
5.2 Database Design:	18
5.3 User Interface Design	19
5.3.1 Renter Interface	19
5.3.2 Lender Interface	28

# 1. Introduction

## 1.1 Purpose of this document:

The purpose of this Software Design Document is to provide a descriptive enough documentation of the design of the system to allow for software development to proceed with an understanding of what is to be built and how it is expected to be built.

## 1.2 Identification:

For the database design MongoDB will be used which would also help in creating the required queries. For frontend prototyping, we've used a website creation software called Wix.

## 1.3 Document Scope:

This design document provides a description of the technical design for Renting and Lending a car. This document provides an architectural overview of the system to depict different aspects of the system. This document also functions as a foundational reference point for developers. It presents a number of different architectural views to depict different aspects of the system. It is intended to capture and convey the significant architectural decisions which have been made on the system. The primary intended audience of this document are system designers and system builders.

## 1.4 Intended Audience:

This project is intended to be used by a user who is in need of a transport facility for a few days or a user who wants to lend their car and make profit.

- The customer will use this document to verify that the developer team has created a product that is acceptable to the customer.
- The project manager of the developer team will use this document to plan milestones and a delivery date, and ensure that the developing team is on track during development of the system.
- The designer will use this document as a basis for creating the system's design. The designer will continually refer back to this document to ensure that the system they are designing will fulfill the customer's needs.
- The developer will use this document as a basis for developing the system's functionality. The

developer will link the requirements defined in this document to the software they create to ensure that they have created software that will fulfill all of the customer's documented requirements.

- The tester will use this document to derive test plans and test cases for each documented requirement. When portions of the software are complete, the tester will run his tests on that software to ensure that the software fulfills the requirements documented in this document. The tester will again run his tests on the entire system when it is complete and ensure that all requirements documented in this document have been fulfilled.

## 1.5 Stakeholder Roles/Responsibilities/Concerns:

### 1.5.1 Regional Stakeholders

- External Stakeholders
  - Car Landing Customers
  - Rentee Customers
  - Car Companies
  - Investors
  - Gas / Fuel Stations
  - Car repair and maintenance services
  - Other online car-rental services
    - AVIS India
    - Zoom Car
    - Ola Rental
    - Drivezy
    - Revv
    - Uber
- Internal Stakeholders
  - Company Managers / Board of Directors
  - Developers / Developer Team
  - Designers / Designer Team
  - Content Manager
  - Insurance Company
  - Professor Khushru Doctor
  - Anupama Nair
  - Muskan Matwani

## 1.5.2 Technical / Project Stakeholders

The following table provides the role and contact information for the key technical and project stakeholders associated with the system design

Name	Role	Email
Parth Patel	Project Manager	parth.p1@ahduni.edu.in
Jeet Shah	Database Design	jeet.s1@ahduni.edu.in
Manal Shah	System Design	manal.s@ahduni.edu.in
Yashvi Pipaliya	Documentation Lead	yashvi.p@ahduni.edu.in
Manav Vagrecha	Front End	manav.s@ahduni.edu.in
Kesha Bagadia	System Design, Documentation	kesha.b@ahduni.edu.in
Shreyansh Shah	Backend	shreyansh.s@ahduni.edu.in
Yashvi Gandhi	Backend Testing	gandhi.p@ahduni.edu.in
Mihir Chauhan	Backend Testing	mihir.c1@ahduni.edu.in

## 1.6 Languages and Tools

- Language
  - Nodejs
  - Reactjs
  - Expressjs
- Tools
  - Visual Paradigm
  - Draw.io
  - Creately
  - Gleek
  - Wix

## 2. General Overview And Design Consideration

### 2.1 General Overview:

Currently an organization carries out its daily work by providing; their service to the customers using a manual system. The organization uses a manual system for reserving, renting, register and to keep record of all the rental activities and customer information.

Our project introduces an online car renting service. So if someone needs a car for a short period of time, they can find one closest to their choice and range on here. Even if one's on a vacation in an unfamiliar place, this site will provide a trusted and convenient source of car rental. And if someone has a car lying around that they don't use, they could profit off it by lending it. So this software will be for the users who would like to book cars as well as for users who'd like to lend cars, all within a few clicks. This web application should be free and accessible from any browser on desktop and can be used by any user.

### 2.2 Current:

The Statement of Need explains why the system is being developed, what purpose it serves, and why it is necessary. The reason for this web application is to:

With the cities expanding and endless roadlines, transportation is becoming an issue for the people who do not have their own personal transport. There are people who don't use their vehicle often and which increases the maintenance and service cost rather than the fuel cost.

#### 2.1.1 Proposed Solution - Statement of Need:

E-commerce website for community car-rental service

To build an Online Car Rental Website. If someone needs a car for a short period of time, they can find one closest to their choice and range on here. Even if one's on a vacation in an unfamiliar place, this site will provide a trusted and convenient source of car rental. And if someone has a car lying around that they don't use, they could profit off it by lending it.

## 2.3 System Assumptions

- Every user who rents a car is a licensed driver.
- Every car added to the system has a valid Car registration number.
- The images of the car provided by the lender justifies the original conditions of the car.
- The payment is conducted securely offline.
- Car is successfully returned when the renter updates the booking status.

## 2.4 System Constraints:

- Verification of Car images, Incense ID and Car registration number
- Security of the car rented
- Payment security

## 3. Design Considerations

### 3.1 Goals and Guidelines:

#### 3.1.1 Architecture

The proposed solution to the problem must satisfy all the functional needs stated by the user, and is required to be developed considering all the non-functional requirements. It is supposed to be adaptive to further changes like support to additional features, functionalities and use cases.

#### 3.1.2 Ease of Use

The features of the application must be user-friendly for strong user experience. Since the application is for every type of user it is supposed to have easy and fast access.

#### 3.1.3 Development Environment

The application development is supposed to stay consistent. Adapting the technological



advances this project aims to build the application in the latest versions of technology with backward compatibility for better performances.

### 3.1.4 Extensibility

The application is aimed to be extensible to adding new features, also providing backward compatibility.

## 3.2 Operational Environment:

- NodeJS
- ExpressJS
- ReactJS
- Robo3t
- Visual Studio Code

Functional goals of the proposed system includes:

- Extending functionality of the existing web application
- Improving application performance
- Sharing and coordinating data via a distributed model
- Completing the one-call one click deployment model

## 3.3 Development Methods & Contingencies:

The basics of a good architecture is to layer the application into multiple autocratic and autonomous applications that can be replaced individually and allow us to keep the application running while we are working on a specific layer

### 3.1.1 Scalability

Ensure that the architecture can be scaled horizontally, across multiple servers and across multiple regions. That means that once your traffic goes up, you should be able to add and remove new servers as the solution requires

### 3.1.2 Availability

The architecture should support a high availability environment. Infrastructure redundancy is required. This ensures the solution is available if multiple servers or an entire data center fail.

### 3.1.3 Extensibility

Architecture must be able to swap out modules, change layers, and add pieces to the application without having to worry about the underlying data contracts in place

### 3.1.4 Security

Solution architecture should expose only the minimal amount of code possible. Most of the back-end pieces should be hidden away. In addition to that, security of each system should be multi-layered.

### 3.1.5 Functionality

The software is capable to provide functions which meet stated and implied needs when the software is used under specified conditions.

### 3.1.6 Separation of responsibility

System should be modular enough that each piece of code has a set of responsibilities and not more. The back-end should not create front end code nor should the front-end code include business logic.

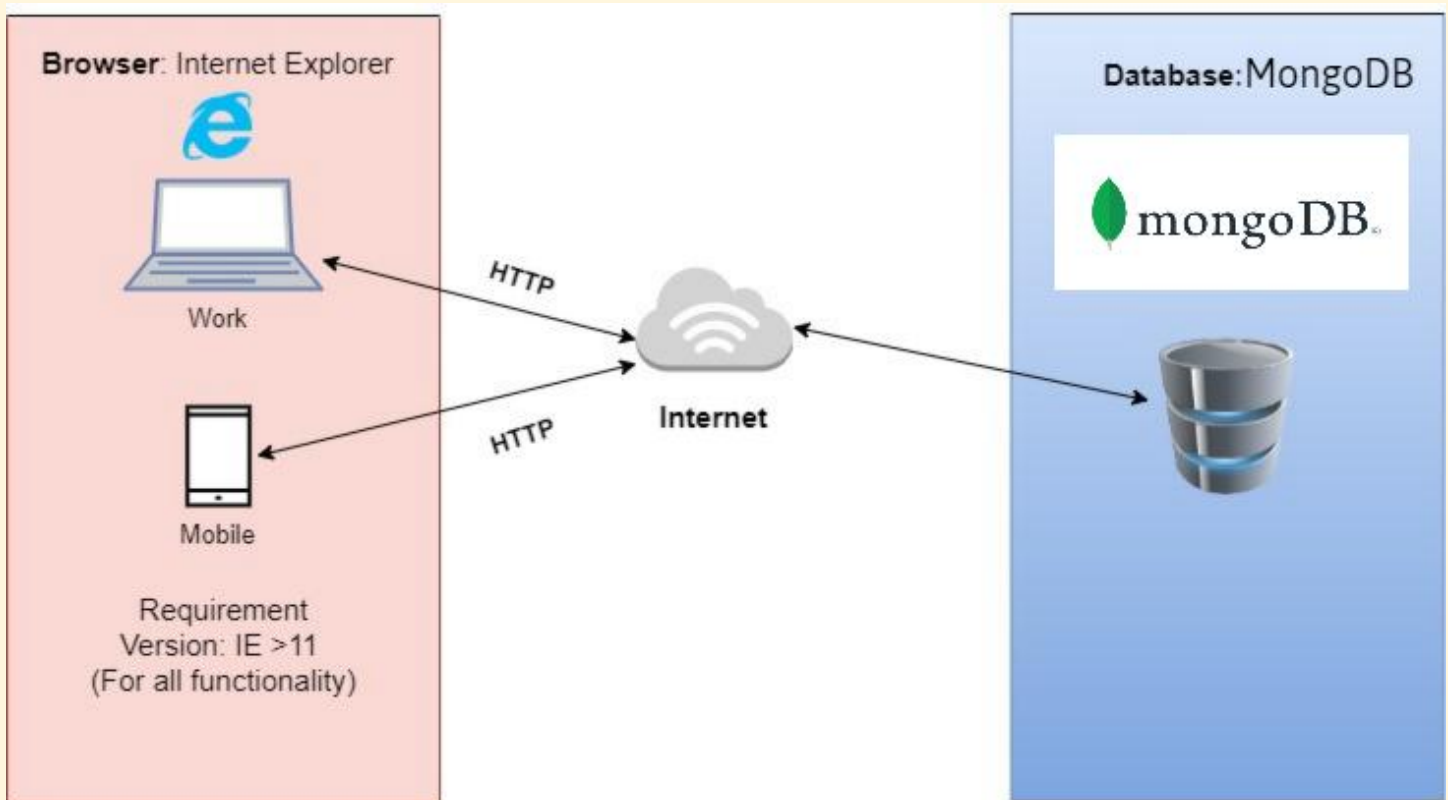
## 3.4 Development Environment:

Software	Description
NodeJS	Programming Language at server
ReactJS	Programming Framework for Web
ExpressJS	Backend Framework
MongoDB	Database

## 4. System Architecture and Architecture Design

### 4.1 Topology Diagram:

A network topology diagram shows how the elements of a computer network are arranged. It allows you to visualize how different nodes are connected and how they communicate



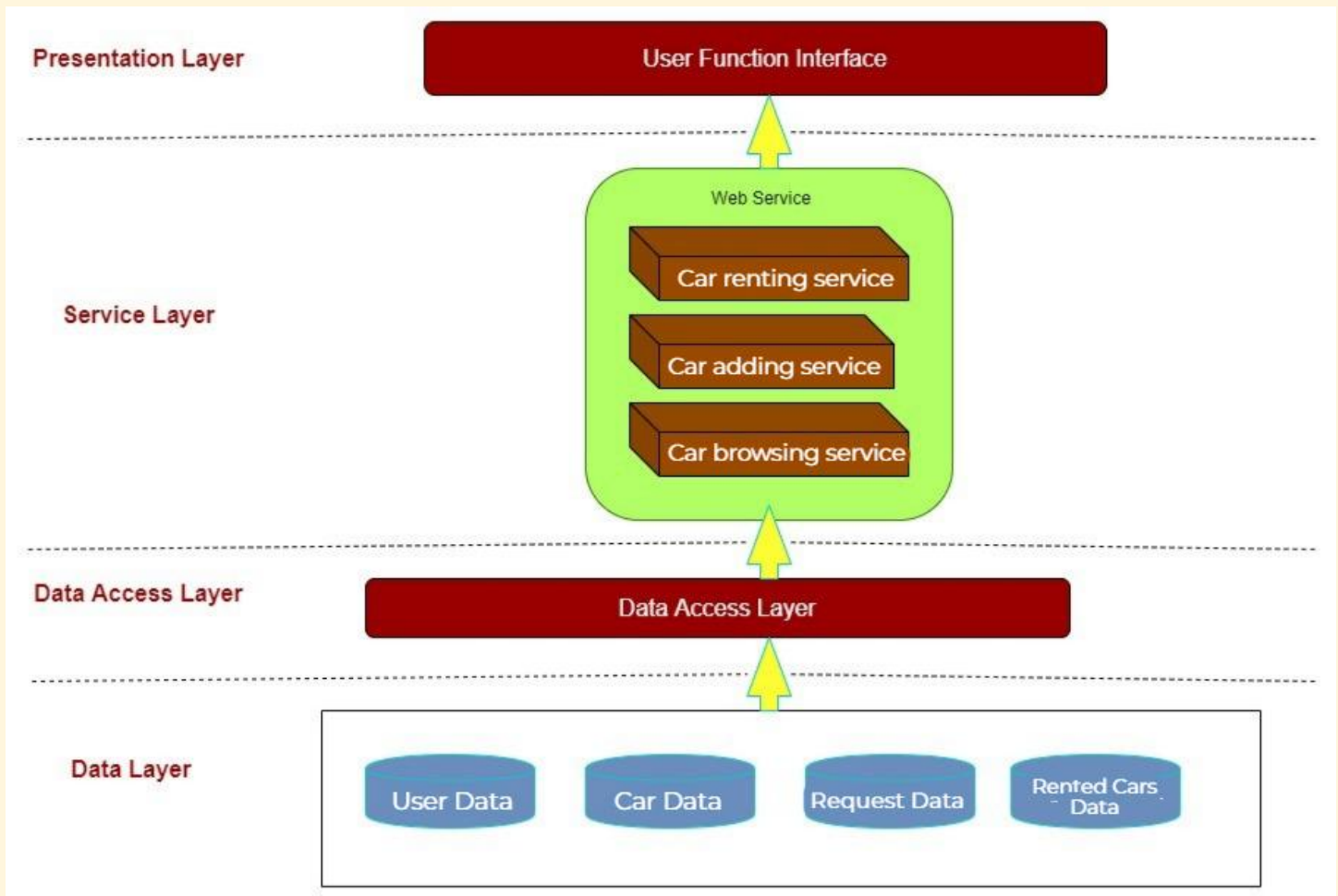
### 4.2 Hardware Architecture

#### 4.2.1 System Descriptions

	SYSTEM	NOTE
Programming Language	JavaScript	JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.
Frontend Framework	Node/React JS	ReactJS is a JavaScript library used for building reusable UI components. React can also render on the server using Node, and it can power native apps using React Native.
Backend Framework	Express JS	Express.js is a Javascript framework based on Node.js, that supports development both on the server side and the user side. Express is a very fast, essential, assertive, and moderate web framework of Node.
DATABASE	MongoDB	MongoDB is a source-available, general purpose, document-based, distributed database built for modern application developers and for the cloud era. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas.

## 4.3 Software Architecture

### 4.3.1 Software Architecture Diagram



## 4.3.2 Software Element

Function	Description
General Access	User can access generally available website using any browser
Car Renting/Discovery	User can select multiple filters to discover or rent the car they need. The filters include: <ul style="list-style-type: none"><li>• Availability date</li><li>• Location</li><li>• Fuel Type</li><li>• Number of seats</li><li>• Colour</li></ul>
Car Details Profile	User can view car details here. The details would include <ul style="list-style-type: none"><li>• Car Company</li><li>• Model</li><li>• Category Name</li><li>• Number of seats</li><li>• Car Colour</li><li>• Fuel Type</li><li>• Engine Type</li><li>• Car Images</li><li>• Additional Features</li></ul>
Car Registration/Lending	User can register their own car for lending on the website after entering the required details (mentioned above). User can edit availability dates or delete the added car.
Car Rental Details	User can view full details of a particular car rental deal. These details would include: <ul style="list-style-type: none"><li>• Corresponding user's name and contact information</li><li>• Car details and image</li><li>• Car rental period</li><li>• Car rental bill</li><li>• Car rental status</li></ul>
Car Rental Requests	User can request a car for rental. Similarly, user can accept or deny a car rental request. Users on both ends will have the provision of each other's contact information.
Car Rental History	User can view their ongoing and completed car rental deals in collation.

## 4.3.2 Performance Software Architecture

<b>Desired Quality</b>	The ability of the system to predictably execute within its mandated performance profile and to handle increased processing volumes in the future if required
<b>Applicability</b>	Any system with complex, unclear, or ambitious performance requirements; systems whose architecture includes elements whose performance is unknown; and systems where future expansion is likely to be significant
<b>Concerns</b>	<ul style="list-style-type: none"><li>▪ response time</li><li>▪ throughput</li><li>▪ scalability</li><li>▪ predictability</li><li>▪ hardware resource requirements</li><li>▪ peak load behavior</li></ul>
<b>Activities</b>	<ul style="list-style-type: none"><li>▪ capture the performance requirements</li><li>▪ create the performance models</li><li>▪ analyze the performance models</li><li>▪ conduct practical testing</li><li>▪ assess against the requirements</li><li>▪ rework the architecture</li></ul>
<b>Tactics</b>	<ul style="list-style-type: none"><li>▪ optimize repeated processing</li><li>▪ reduce contention via replication</li><li>▪ prioritize processing</li><li>▪ consolidate related workload</li><li>▪ distribute processing over time</li><li>▪ minimize the use of shared resources</li><li>▪ reuse resources and results</li><li>▪ partition and parallelize</li><li>▪ scale up or scale out</li><li>▪ degrade gracefully</li><li>▪ use asynchronous processing</li><li>▪ relax transactional consistency</li><li>▪ make design compromises</li></ul>
<b>Pitfalls</b>	<ul style="list-style-type: none"><li>▪ imprecise performance and scalability goals</li><li>▪ unrealistic models</li><li>▪ use of simple measures for complex cases</li><li>▪ inappropriate partitioning</li><li>▪ invalid environment and platform assumptions</li><li>▪ too much indirection</li><li>▪ concurrency-related contention</li><li>▪ database contention</li><li>▪ transaction overhead</li><li>▪ careless allocation of resources</li><li>▪ disregard for network and in-process invocation differences</li></ul>

## 4.4 Information Architecture

### 4.4.1 Data

Data is supplied by the end user or the consumer of the system. Data is entered into the system via the web application.

#### 4.4.1.1 Data Inputs

All inserts or upserts into the database shall be managed using industry standard data validation tools and triggers. Data validation is intended to provide certain well-defined guarantees for fitness, accuracy, and consistency for any of various kinds of user input into an application or automated system. Data validation rules can be defined and designed using any of various methodologies, and be deployed in any of various contexts.

Types:

- Data type validation
- Range and constraint validation

## 5. System Design

The proposed system tries to provide an environment for car renters and car lenders to interact with ease. Users looking to rent cars can browse and filter cars by the means of clear navigation and users looking to lend their cars can add, edit and delete their cars through simple interfaces. Proposed design may be developed to incorporate modular components that plugs into the application. Technical support and maintenance will be required for the transactional and mission critical components of the system. The system will try to provide sufficient security and credentialing methods to ensure privacy and system security.

### 5.1 Business Requirements:

For our proposed system the central requirements would be:

- Providing a list of available cars meeting user's requirements.
- Feature for users to add cars.
- Feature for users to request for a car rental.
- Feature for users to deny a request for car rental.
- Provide contact information of the respective renter or lender.
- Access to records of their previous and current car rental bookings.



## 5.2 Database Design:

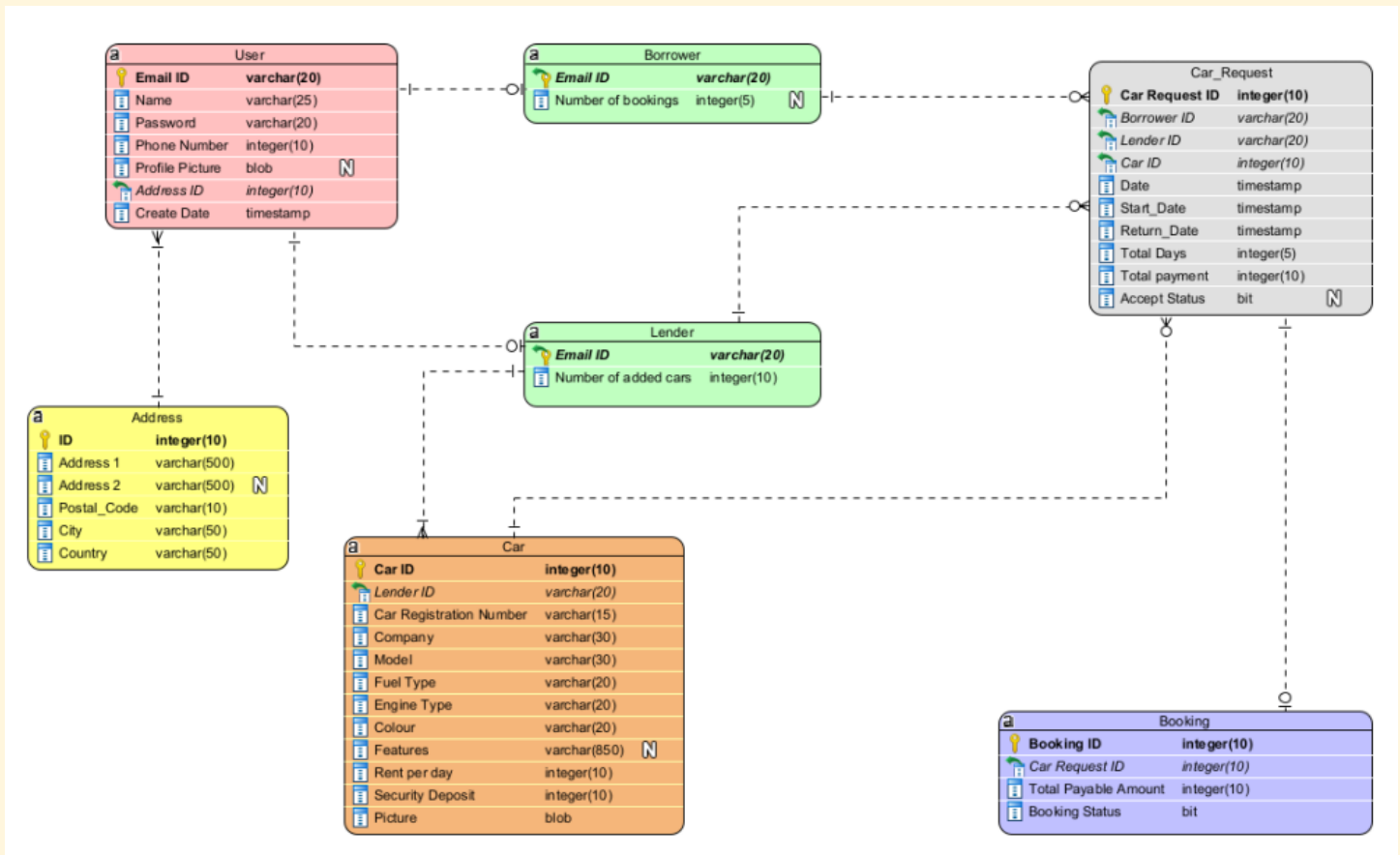
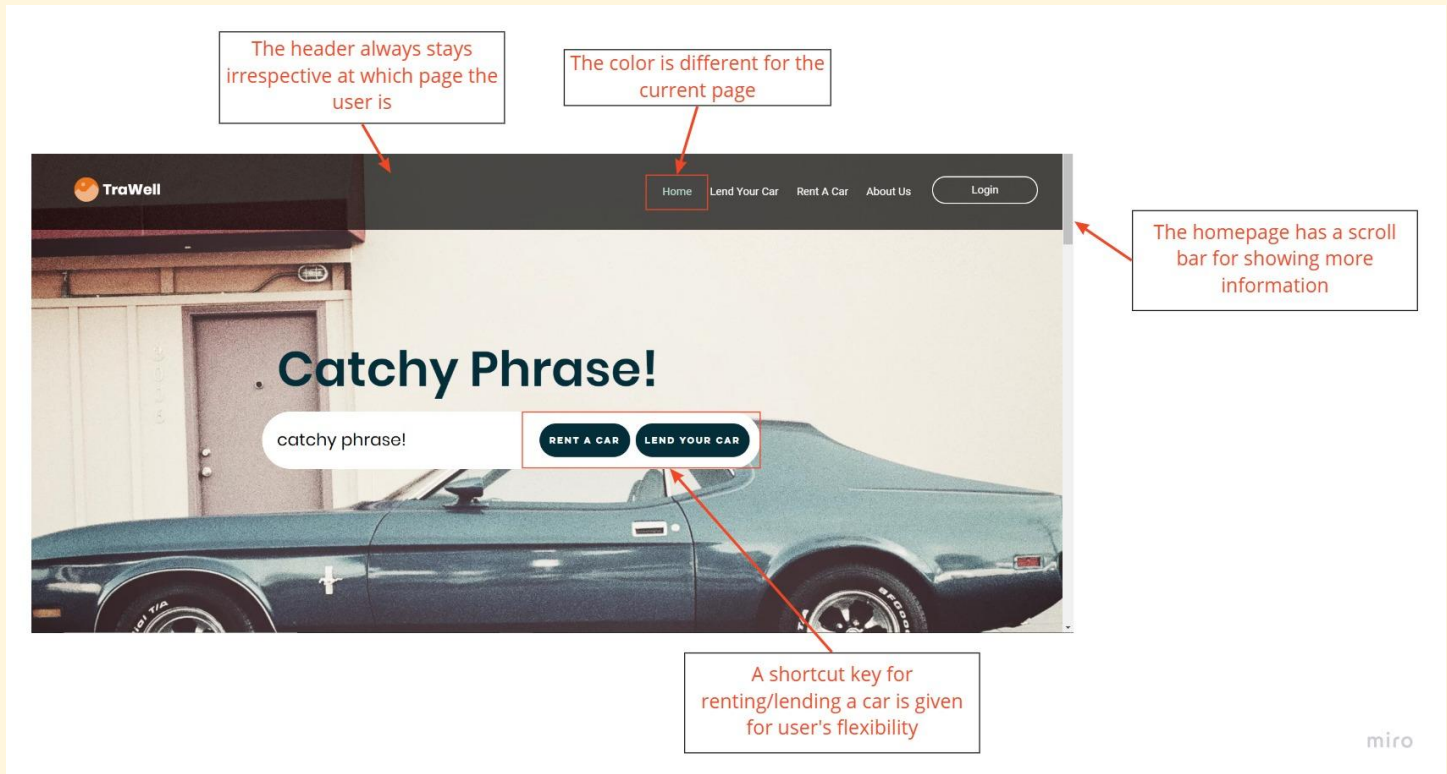


Figure: Entity Relationship Diagram (Physical)

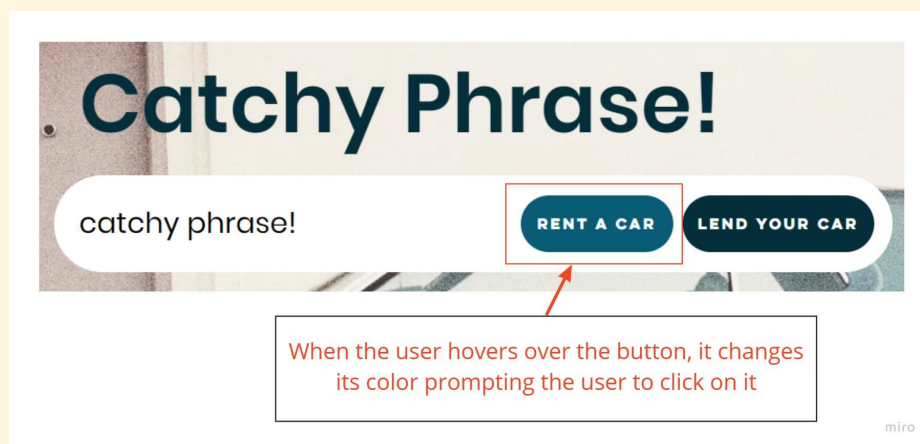
## 5.3 User Interface Design

### 5.3.1 Renter Interface

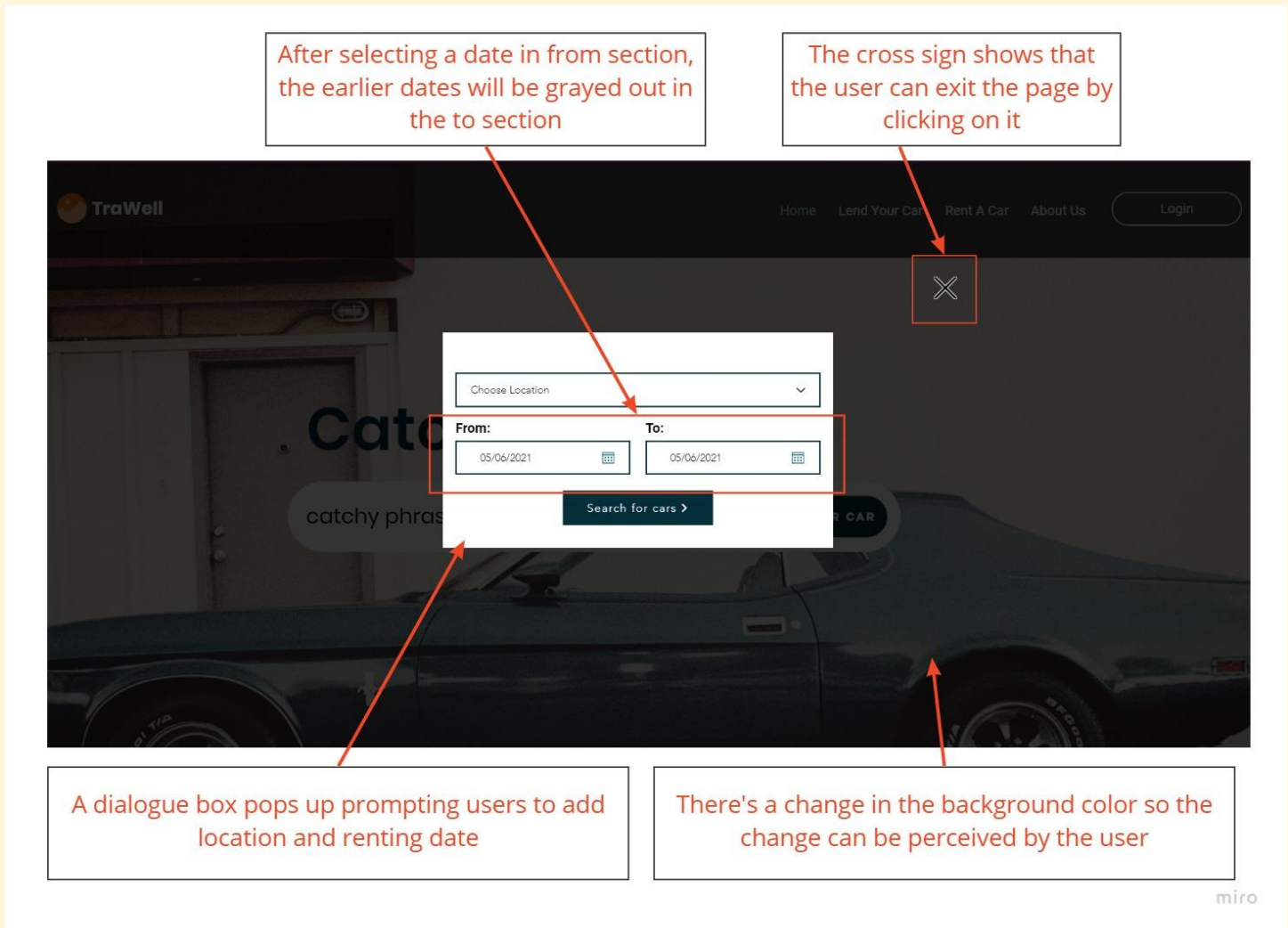
#### 5.3.1.1 The home page



#### 5.3.1.2 The user can click on the 'Rent a car' button



### 5.3.1.3 A pop-up for the user to enter details before renting a car



### 5.3.1.4 User chooses a location and date to rent a car

The user should be able to choose any one from the different locations

The chosen category should be highlighted with a darker color

Choose Location

- Ahmedabad
- Vadodara
- Rajkot
- Mumbai

Search for cars >

After filling in the details, the user can click on the search for cars button.

The button is made 3d like with a darker color to prompt users to click on it

miro

From: 05/14/2021

To: 05/06/2021

< May > < 2021 >

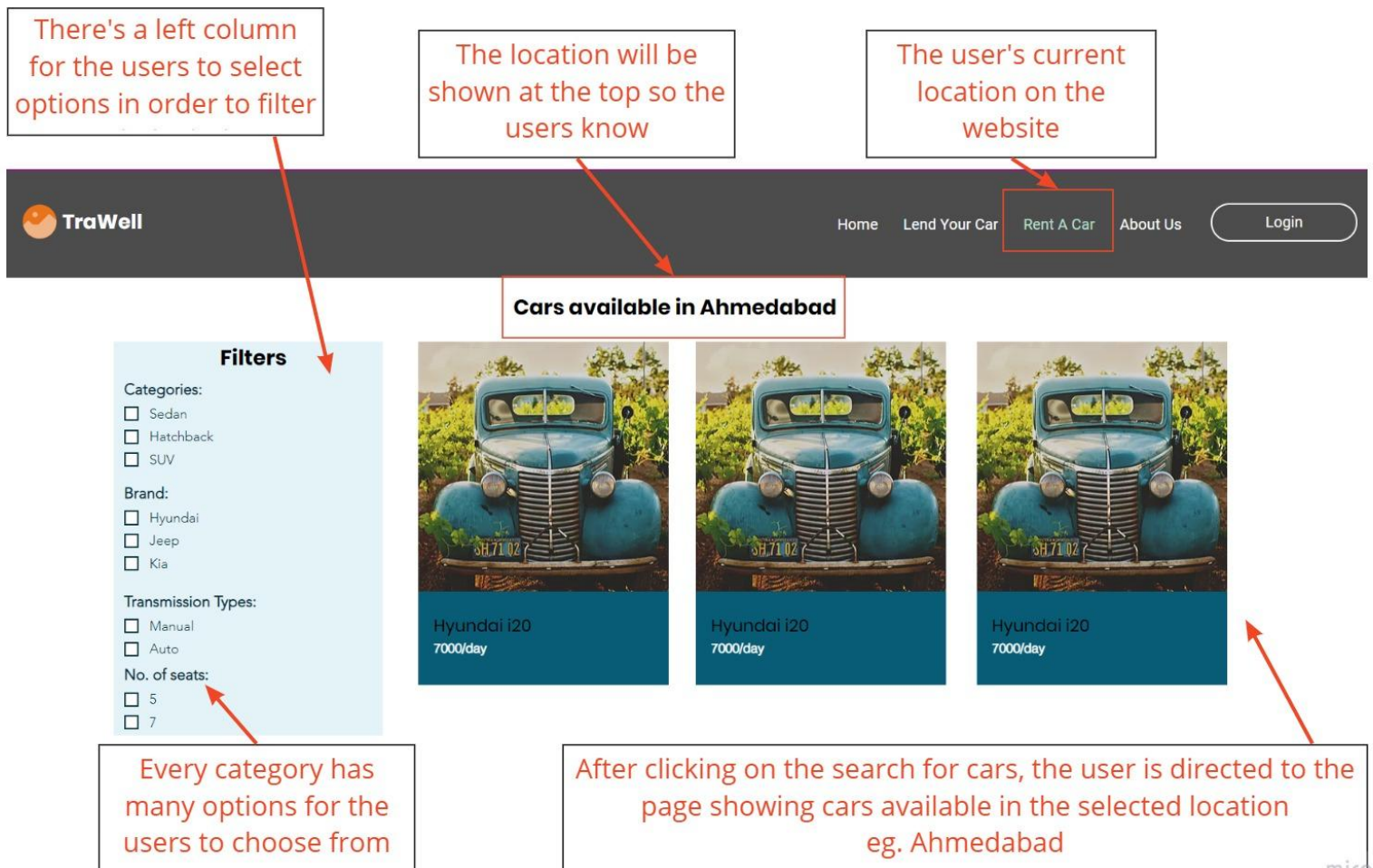
Mo	Tu	We	Th	Fr	Sa	Su
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

A calendar should drop down after the user clicks on the calendar icon

The selected date will be in a black box and the date the user wishes to choose will have a lighter shade, implying it's still not selected

miro

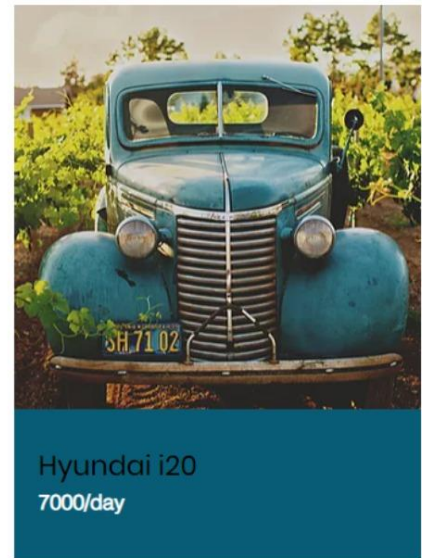
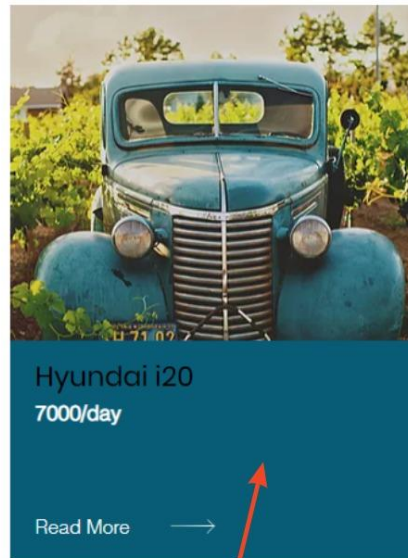
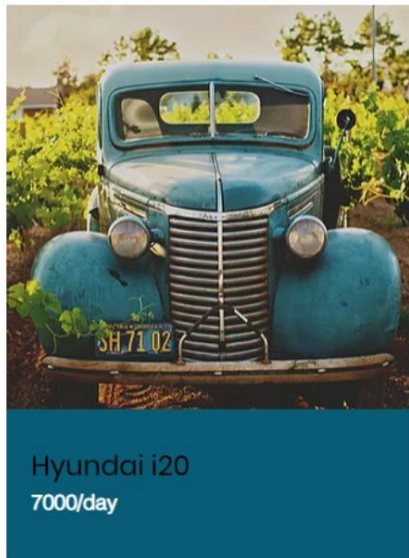
### 5.3.1.5 The user is directed to *a new page* where they can select cars and apply filters





### 5.3.1.6 User can choose a car of their choice by clicking on the raised bar

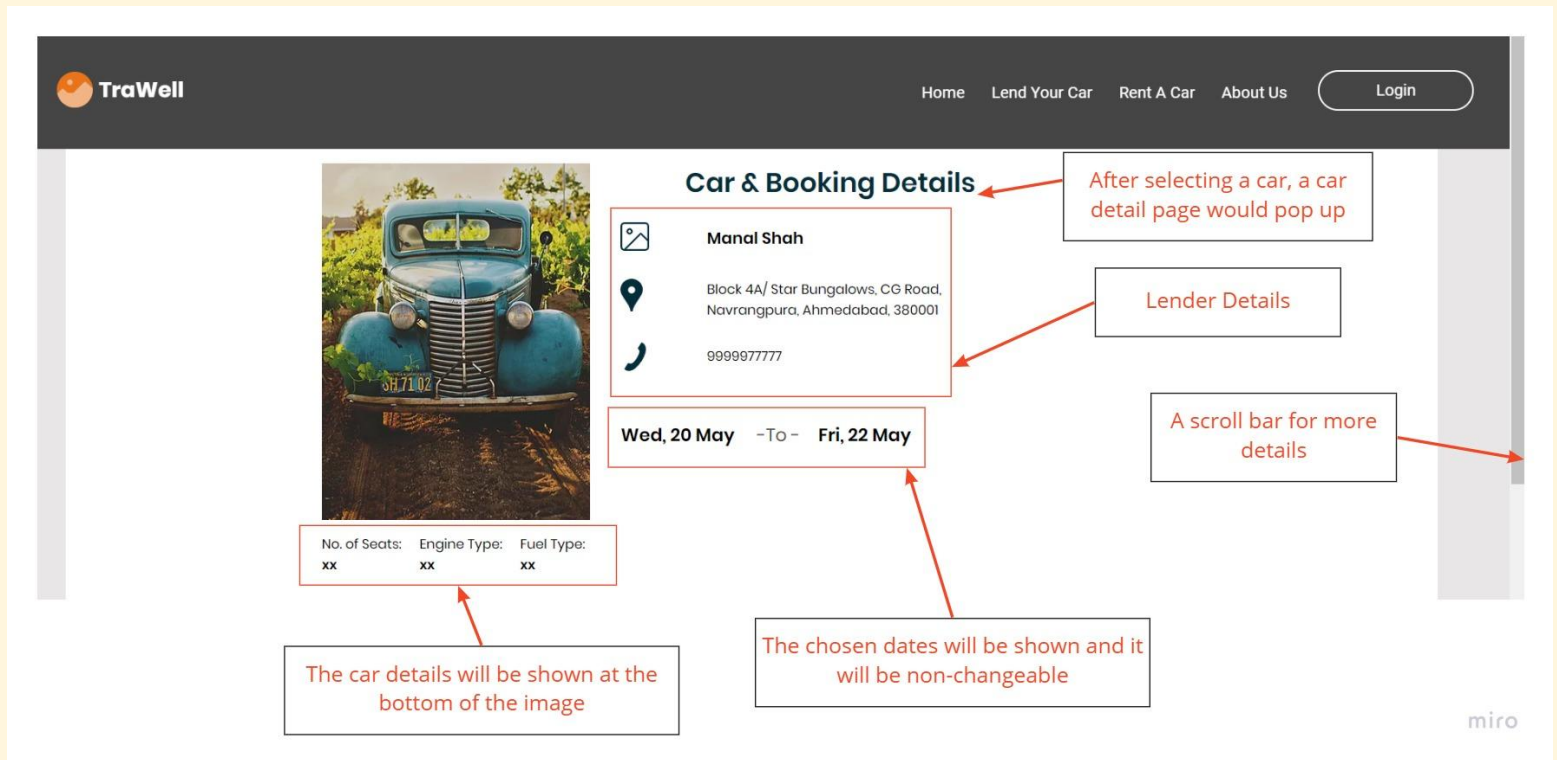
#### Cars available in Ahmedabad



When the user hovers over, the information bar is raised up prompting users to click on it

miro

### 5.3.1.7 User is directed to a car detail page where they can find all the necessary details



The user need to check on the boxes before proceeding for the request

No. of Seats: xx Engine Type: xx Fuel Type: xx

## General Instructions

### Details About Instructions

1. \_\_\_\_\_ Description: \_\_\_\_\_
2. \_\_\_\_\_ Description: \_\_\_\_\_
3. \_\_\_\_\_ Description: \_\_\_\_\_
4. \_\_\_\_\_ Description: \_\_\_\_\_
5. \_\_\_\_\_ Description: \_\_\_\_\_

## Fare Details

1. \_\_\_\_\_ Rs. xx
2. \_\_\_\_\_ Rs. xx
3. \_\_\_\_\_ Rs. xx
4. \_\_\_\_\_ Rs. xx
5. \_\_\_\_\_ Rs. xx

\*These amounts are based on the per day plan

### T&C

- ☐ Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris dui ante, volutpat non interdum quis, gravida vitae metus. Praesent maximus lorem varius ipsum auctor semper.
- ☐ Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris dui ante, volutpat non interdum quis, gravida vitae metus. Praesent maximus lorem varius ipsum auctor semper.

Request Car

General details regarding fares, terms & conditions, policies are given down the scroll bar.

The button is grayed out as the user is required to first check the boxes

miro



### 5.3.1.8 The user can now click on the *request car* button

T&C

- ☒ Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris dui ante, volutpat non interdum quis, gravida vitae metus. Praesent maximus lorem varius ipsum auctor semper.
- ☒ Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris dui ante, volutpat non interdum quis, gravida vitae metus. Praesent maximus lorem varius ipsum auctor semper.

Request Car

The button's color is changed after  
ticking the checkboxes

### 5.3.1.9 The user is directed to the login page only then they can request a car.

The screenshot shows the TraWell website's login page. At the top is a dark navigation bar with the TraWell logo on the left and links for Home, Lend Your Car, Rent A Car, About Us, and a Login button on the right. The main content area has a white background. A central box contains the text "Please sign in to continue!" followed by input fields for "Email\*" and "Password\*", a dark "Login" button, a "Forgot Password?" link, and a "Don't have an account? Sign up" link. A red callout box on the right contains the text "The user is directed to the login page if the user is not logged in yet" with a red arrow pointing to the login form area. The Miro logo is in the bottom right corner.

Please sign in to continue!

Email\*

Password\*

Login

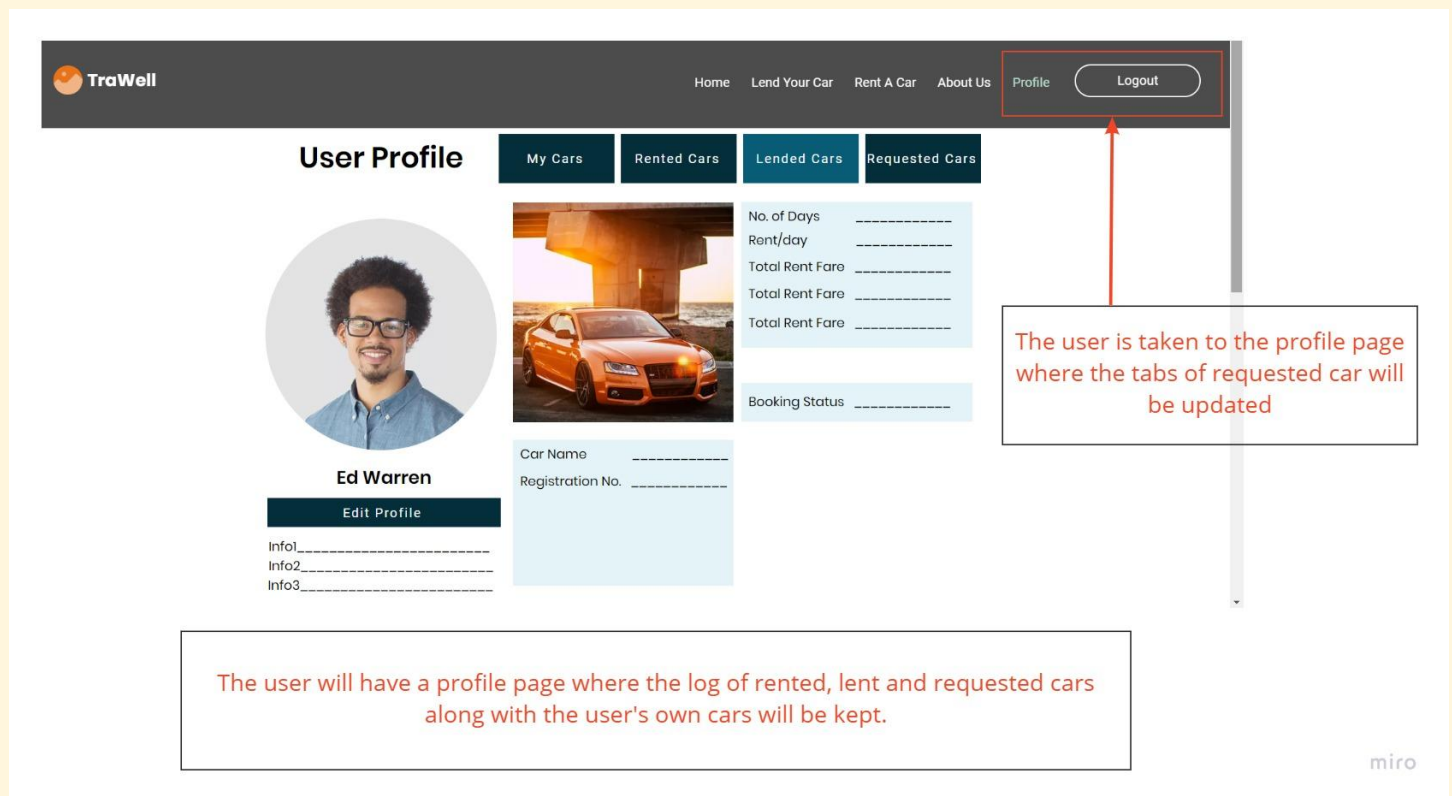
Forgot Password?

Don't have an account? [Sign up](#)

The user is directed to the login page if the user is not logged in yet

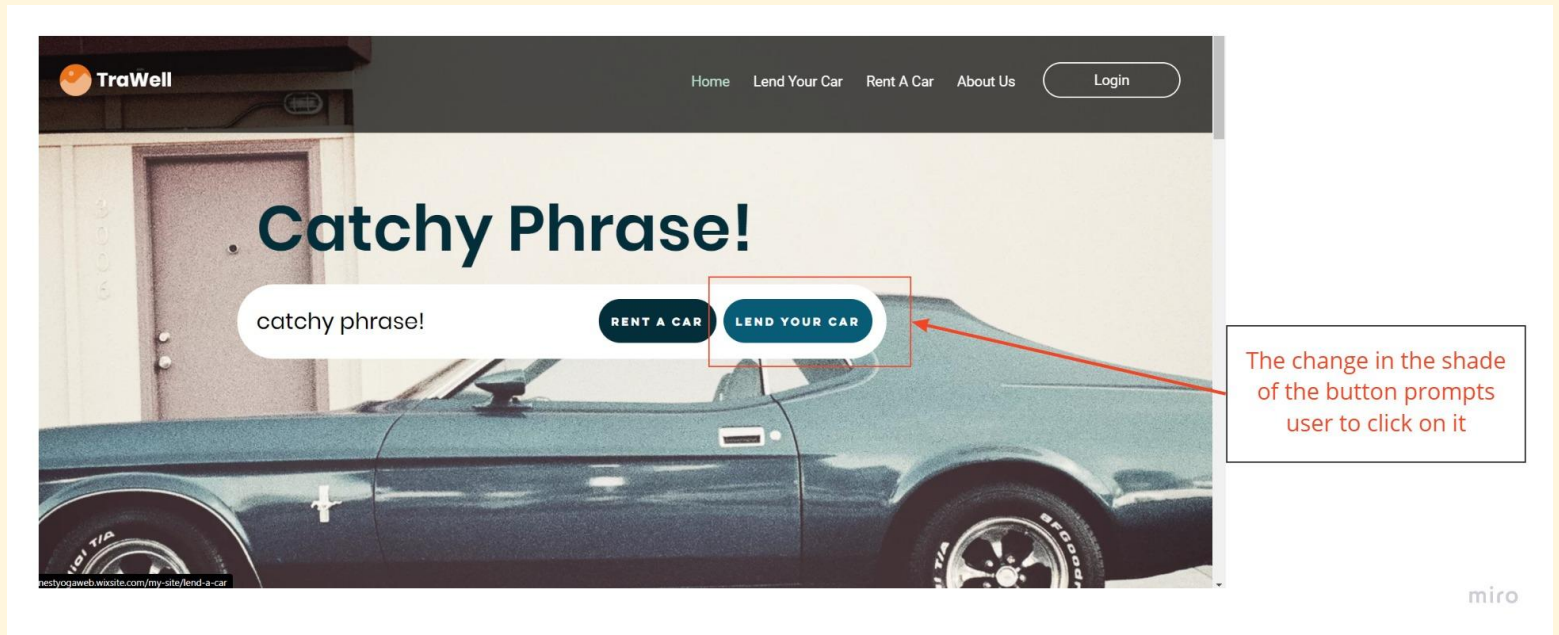
miro

### 5.3.1.10 The user is directed to the profile page where they can check the log of their activity

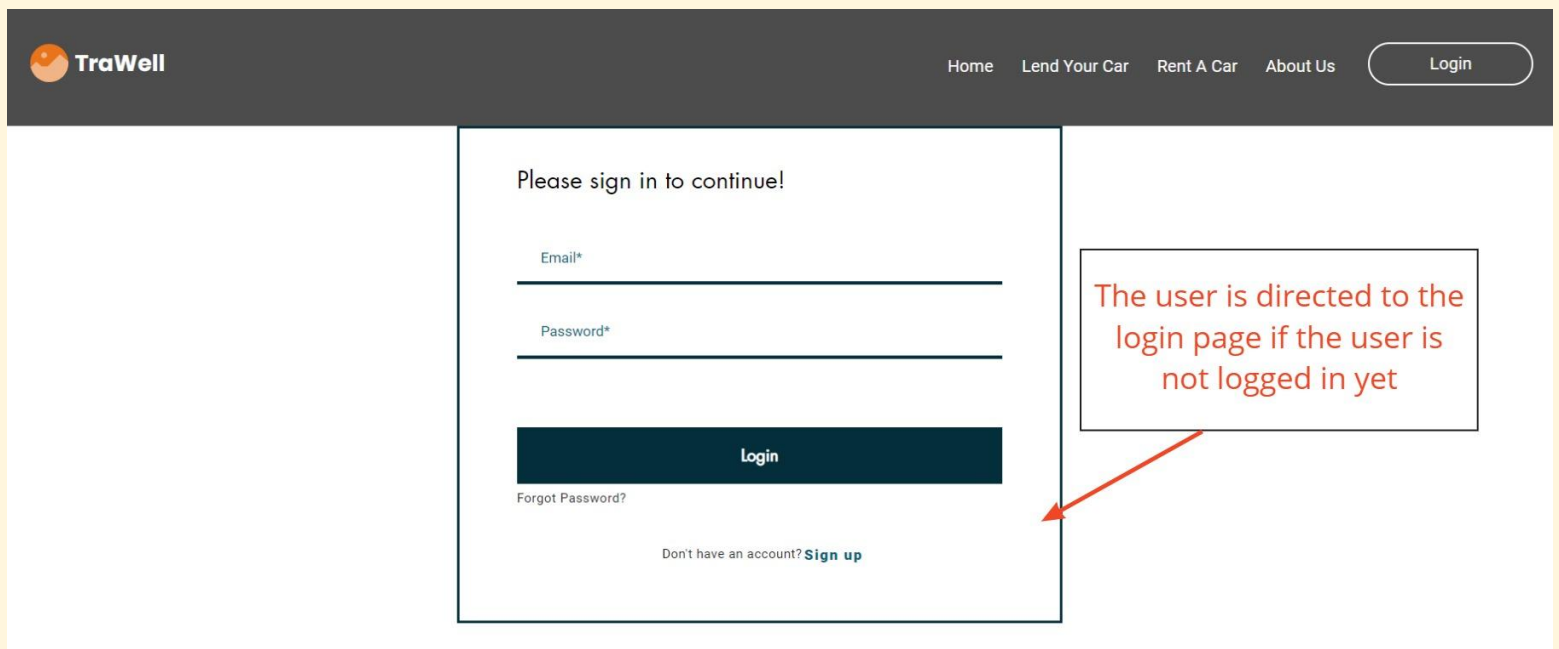


## 5.3.2 Lender Interface

### 5.3.2.1 When the user clicks on the lend your car button



### 5.3.2.2 The user is required to log in before lending a car



### 5.3.2.3 After the user is logged in, the lender form is opened for the users to fill in the details

There's a drop down menu. The first menu which is 'company name' should be filled in first. Therefore, other fields are grayed out

The user is in the lend your car page

Please fill in the details to lend your car

Company name

Modal name

Category name

No. of seats

Registration No.

Select Color

Fuel Type

The category name and no. of seats will be auto-filled after the user enter modal name

TraWell

Home Lend Your Car Rent A Car About Us Login

miro

### 5.3.2.4 The user can choose options from the drop down list and can only fill the details in accordance with the constraints given.

The screenshot shows the 'Lend Your Car' form on the TraWell website. The form is titled 'Please fill in the details to lend your car' and includes a navigation bar with links to Home, Lend Your Car, Rent A Car, About Us, and a Login button. The form fields are as follows:

- Brand: A dropdown menu with 'Hyundai' selected.
- Model: A dropdown menu with 'skoda' selected.
- Category: A dropdown menu with 'Sedan' selected.
- Seats: A text input field with '5' entered.
- Registration No.: A text input field with 'fdsa321' entered.
- Color: A dropdown menu with 'Silver Daze' selected.
- Engine Type: A dropdown menu.

Annotations on the form include:

- A box on the left states: 'A drop down menu to select from the given options' with an arrow pointing to the Color dropdown.
- A box on the right states: 'After selecting the modal name, the category name and no. seats is shown' with an arrow pointing to the Category and Seats fields.
- A box on the right states: 'Registration no. can only accept alpha-numeric data' with an arrow pointing to the Registration No. field.

The TraWell logo is in the top left corner, and the Miro logo is in the bottom right corner.

### 5.3.2.5 The user is required to fill the remaining details keeping the constraints in mind.

The user is required to enter the amount which should fulfill the minimum amount criteria

Silver Daze

Petrol

Engine Type

Add Features

Upload Images +

Max File Size 15MB

Rent (>500)

Deposit (>5000)

Add your car

The user is required to upload the images of the car

As the form is vertical and quite big, a scroll bar is required

The button is grayed out as all the fields are not filled in

miro

### 5.3.2.5 The user directed to the user profile where they can check the log of their activities.

