**Design process and Challenges faced**

**Frontend Design Process**

1. Framework Selection:
   * Used React.js as the frontend framework for building a responsive, component-based single-page application (SPA).
2. Component Structure:
   * Navbar: Navigation menu with links for "Home," "Cars," and "About."
   * Home Component: Displays a welcome message and a call-to-action button to explore cars.
   * CarListing Component: Displays the list of available cars (can be extended with a dynamic backend in the future).
   * About Component: Provides information about the car rental service.
3. Routing:
   * Integrated React Router for navigation between pages.
   * Routes include:
     + / → Home
     + /cars → Car Listing
     + /about → About
     + \* → Redirects to Home (fallback).
4. Styling:
   * Used CSS classes for styling with a light theme.
   * Background images, flex layouts, and hover effects for interactivity.
5. Static Assets:
   * Background and car images placed in the public/images folder.
   * Accessed in the app via relative paths.

**Link**

https://car-rental-website-kavanas-projects-b8e7adcc.vercel.app/

**Folder Tree for Accessing Files**

car-rental-website/

└── src/

├── components/ # Folder for reusable components

│ ├── Navbar.js # Navigation bar (shared across pages)

├── pages/ # Folder for main pages

│ ├── Home.js # Home page

│ ├── CarListing.js # Car listing page

│ └── About.js # About page

├── App.js # Main app component with routing

├── index.js # Entry point for React

**Challenges**

1. Routing Issues:

* Setting up proper navigation between pages (Home, Car Listing, About) and handling incorrect URLs with a fallback route.

1. Image Display Errors:

* Images may not load due to incorrect paths or deployment settings.

1. Understanding React framework and Link creation