The Ride Me application is built using react js,redux,css and jsx.This is a horse ride booking application which contains some react components.

There are 4 main react components-AnimatedImageGallery- for handling horse images,BookingForm-for handling horse ride booking,Home-Landing page which uses AnimatedImageGallery component and a button 'Book Now' which navigates to Booking form ie BookingForm component.

Since React router dom is used for navigation this is a single page application which navigates to other urls smoothly using react router.

## AnimatedImageGallery

It displays a gallery of animated images of horses. The gallery uses the library 'framer-motion' for animations and consists of two main components: AnimatedImage and HorseCard.

- **AnimatedImage**: Handles the image animation using framer-motion.
- **HorseCard**: Combines an AnimatedImage with the horse's name.
- **AnimatedImageGallery**: Arranges multiple HorseCard components in a flexbox layout to create a gallery.

## Home

It is the landing page which uses AnimatedImageGallery component to display the horses's images with their names.it has a button 'Book Now' which navigates to the horse ride booking form.

## BookingForm

This component allows users to book a horse ride by filling out a form with their name, email, phone number, selecting a horse, and picking a ride date and time.for ride date and time picker it uses the component:rideDTP and uses date-fns and date-fns-tz libraries to parse and format dates and times. Implemented functions to convert and format dates into the required timezone. It validates the input, generates a calendar invite link, and provides feedback through toast notifications.

## RideDTP

It uses react-datepicker component for a user-friendly Date time Picker and it formats the time as per the requirements(week days-I hour interval times,Saturdays-3pm to 12 am) and stores the selected ride date to the redux store.

BookingForm component subscribes the store and then access ride date and generate calendar link.

## Challenges Faced during the Development

- Faced difficulty in Handling date and time inputs accurately, especially considering different time zones (specifically Asia/Kolkata timezone).

Solved it by Utilizing the date-fns and date-fns-tz libraries to parse and format dates and times. Implemented functions to convert and format dates into the required timezone, ensuring consistency and accuracy in booking times.

- It was tough to generate and send calendar invites (ICS files) for horse ride bookings.

Created a generateCalendarLink function that formats booking details into an ICS file content. Used encodeURIComponent to encode the ICS content properly for download links.

- Providing real-time feedback to users upon successful or failed Booking form submissions

Integrated the react-toastify library to display success and error popups, providing user interaction on the form submission status.