Vehicle Rental Booking System

This document provides step-by-step instructions to set up and run the Vehicle Rental Booking System project.

# 1. Backend Setup

## Step 1: Install Backend Dependencies

Navigate to the backend folder and install the required dependencies using npm:  
  
```bash  
cd backend  
npm install  
```

## Step 2: Configure Database

Make sure you have PostgreSQL installed on your system. If not, please follow the official PostgreSQL documentation to install it.  
  
Create a new database for the project using the following SQL command in PostgreSQL:  
  
```sql  
CREATE DATABASE vehicle\_rental;  
```  
  
Update the `config.js` (or similar) file in the backend to reflect your PostgreSQL database credentials.  
  
Example configuration:  
  
```js  
module.exports = {  
 development: {  
 username: 'postgres',  
 password: 'your-password',  
 database: 'vehicle\_rental',  
 host: 'localhost',  
 dialect: 'postgres',  
 },  
};  
```

## Step 3: Run Migrations

Run the Sequelize migrations to create the necessary tables in the database:  
  
```bash  
npx sequelize-cli db:migrate  
```

## Step 4: Start Backend Server

Start the backend server using the following command:  
  
```bash  
node src/server.js  
```  
The backend API will run on `http://localhost:5000`.

# 2. Frontend Setup

## Step 1: Install Frontend Dependencies

Navigate to the frontend folder and install the required dependencies using npm:  
  
```bash  
cd frontend  
npm install  
```

## Step 2: Configure API Calls

Ensure that the API endpoints in the frontend (React app) are pointing to the correct backend URL.  
Example API call in `App.jsx`:  
  
```js  
const res = await axios.get('http://localhost:5000/api/vehicle-types?wheels=' + data.wheels);  
```

## Step 3: Start Frontend Server

Start the frontend development server using the following command:  
  
```bash  
npm start  
```  
The frontend will run on `http://localhost:3000`.

# 3. Test the Application

Once both the backend and frontend servers are running, navigate to `http://localhost:3000` in your browser.  
You should be able to interact with the vehicle rental booking system, fill out the form, and submit a booking.

# 4. Optional: Production Build (For Deployment)

## Step 1: Build Frontend for Production

To build the frontend for production, run the following command:  
  
```bash  
npm run build  
```

## Step 2: Serve Static Files with Backend

In your backend (Node.js), update your server to serve static files from the `frontend/build` directory:  
  
Example in `server.js`:  
  
```js  
const path = require('path');  
  
// Serve static files in production  
if (process.env.NODE\_ENV === 'production') {  
 app.use(express.static(path.join(\_\_dirname, 'frontend/build')));  
 app.get('\*', (req, res) => {  
 res.sendFile(path.join(\_\_dirname, 'frontend/build', 'index.html'));  
 });  
}  
```

# 5. Conclusion

Follow the above steps to set up and run the Vehicle Rental Booking System project.  
If you encounter any issues during setup or running the project, check the terminal for error messages.