

React Training Syllabus - VehicleExchanger

X-Force _ Version 1.0 - 2024Q2

Trainee information

Trainee	Account	Role
Tran Thanh Minh Huy	HuyTTM2	OJT Front-end developer (React)

Instructor information

Instructor	Team	Office hours
TrungNT102	X-Force	9.30AM - 4PM, Mon-Fri

Reviewers

Account	Team	Role
ToanLQ2	X-Force	Team Lead
HetTV	Shield	Team Lead

Overview

Trainees will embark on a project-based journey to create declarative user interfaces (UI components) for web applications using React, enhance state predictability in their applications with Redux in both server and client sides, and increase their apps' efficiency and robustness utilizing new React features and the Jest testing library.

Expectations and goals

- Developing interactive React components for applications by utilizing JSX for UI rendering, managing state, and handling lifecycle events.
- Creating more powerful applications through the use of Redux for global state management, handling asynchronous network requests, and efficiently passing data through the UI.
- Building more efficient and robust React applications by employing recent React features like Hooks and testing components with the Jest testing library.
- Breaking each feature into small User Stories, then estimating and breaking US into small tasks themselves.
- Having good coding conventions and Git/Github by CLI working skills.

Program information

Estimated Time

3 months at 4 days/week

Skill Level

Beginner

Prerequisites

Learners should be familiar with the following topics:

- HTML & CSS
- JavaScript
- Asynchronous JavaScript (AJAX)
- Web forms (front-end elements such as HTML, CSS, styling, and accessibility)
- React UI library with basic knowledges
- Redux state management library with core concepts

Required materials

- React 16.8 or above that includes Hooks (The most recent version - 18 - is recommended)
- React Router
- Redux core & Redux toolkit

Provided materials

- Project codebase: [Github](#)
- API: [Free Car API](#)

Training points

- React 18: [React Homepage](#)
- Next.js 14 using Pages Router: [Next.js Homepage](#)
- Redux-eggs: [Github redux-eggs](#)
- Extensible-duck: [Github extensible-duck](#)

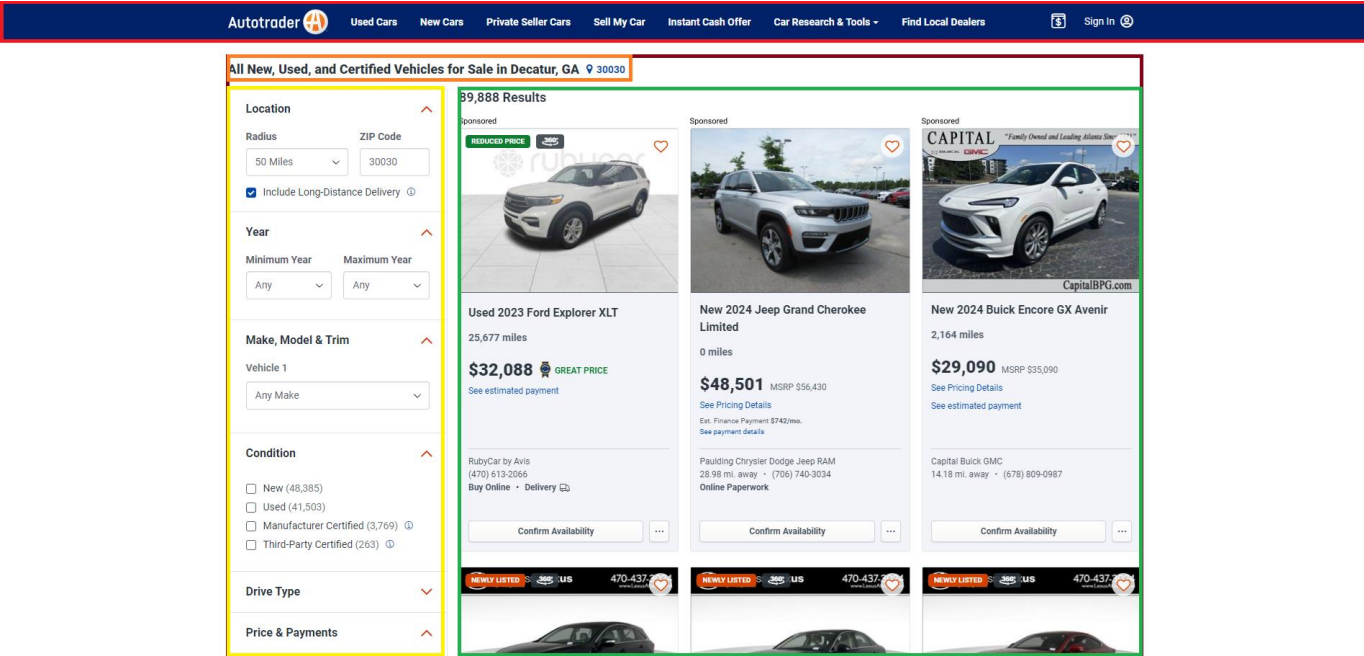
Features Overview

(Using Autotrader.com as only a reference layout)

Feature	Description	Checkpoints
F10000	Build app layout structure and enhance data from server side in redux store	Investigate project and work on defined US
F20000	Make UI/UX and filter behaviors - Search Car Page (SCP)	Will work with instructor to break down US in sprint planning
F30000	Make UI/UX and submitting email behavior - Car Detail Page (CDP)	Will work with instructor to break down US in sprint planning
F40000	Make UI/UX - Compare Car Modal	To be updated
F50000	Make UI/UX - Filter chips in SCP	To be updated
F60000	Add Spotlight car at the top of SCP	To be updated
F70000	Navigate from SCP to CDP by custom hooks	To be updated
F80000	Add filter referrer when navigate from SCP to CDP	To be updated
F90000	Previous and Next button in CDP	To be updated
F11000	Auto scrolling down to Finance section after clicking item price and navigating to SCP	To be updated

Feature details

Search Car Page (SCP)



Trainee will create a React application from codebase and utilize React components to manage the user interface. They'll create a virtual vehicle exchange site to introduce and describe everything consumers need. Using the provided Free Car API, trainee will fetch initial filters, get cars by selected filters in server side. Finally, they'll use React Hooks, Redux to build the functionality to show cars to the page.

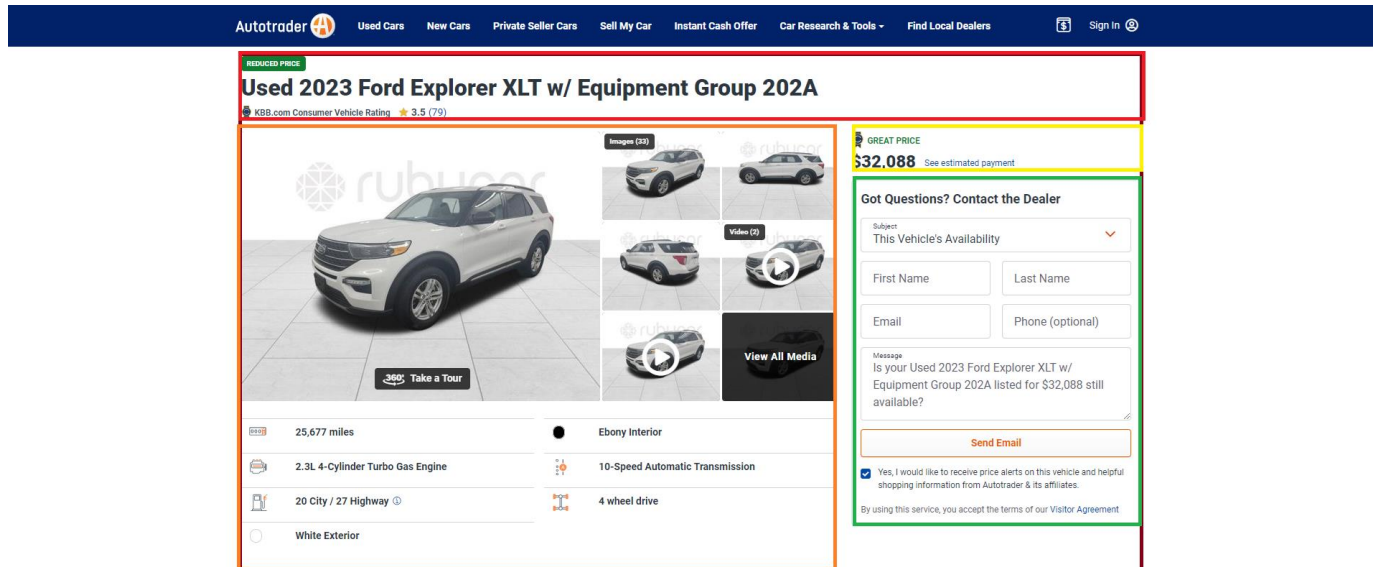
Feature F10000
Build app layout structure and enhance data from server side in redux store

- Investigate codebase and identifying how app works.
- Use borderline only to build app layout with responsive.
- Fetch and enhance vehicle filters by provided API into redux state in server side.

Feature F20000
Make UI/UX and filter behaviors - Search Car Page (SCP)

- Create reusable React components with composition.
- Use JSX to create user interface elements.
- Use PropTypes to typecheck and debug components.
- Manage state in applications

Car Detail Page (CDP)



They'll create a Car Detail Page to show up everything about selected vehicle. Using combination between the provided Free Car API and redux state, trainee will fetch a single car information, update states then displaying them to the page. Finally, they'll use free third-party to build the functionality to submit the email form, handle response and trigger notification.

Feature F30000

Make UI/UX and submitting email behavior -
Car Detail Page (CDP)

- Use React Router to add different routes to applications.
- Use state to dynamically render a different page.
- Use React Router's Route and Link components.
- Use React's useEffect hook for HTTP request and side effects for email behavior.