

PIP Assignment for Vishal Maheshwari

Practitioner has to go through following react assignment over the period of 30 days.

Objective:

HashedIn by Deloitte has a requirement to build a web application portal for HashedIn University to allow Linkers to add courses to their cart/bucket thereby enabling them to complete courses in a stipulated time (i.e completing their tracks in a form of courses).

Project Details:

HashedIn University is our unique, 2 month bootcamp for newly recruited engineers, conducted twice a year. It is inspired by Trilogy University and modelled on the HBR article – No Ordinary Bootcamp by Dr. Noel Tichy.

The intense, yet fun-filled training program is designed to create the next generation of cloud and software experts in the latest technologies, and transform them into industry-ready leaders.

HashedIn needs to develop the web application for HashedIn University to create an interface for a linker to become a Hasher!

User Interface & Visual Designs:

Please find the mockups for the application in the [Link](#). Mockups are for visual representation only.

Password: Hashedin@123

Quality Focus:

Even though the client doesn't mention quality, the quality focus has to be there. Whatever we deliver has to be delivered with excellent quality. This is important for getting continued business and establishing the value HashedIn can deliver to them i.e. "High Quality delivered Quickly"

General Requirements

1. Error States

Error state view refers to the view (modal in this case) displayed to the user in case any error occurs. Show appropriate modals as per design in case of errors.

2. Unit Tests

As part of best practices, we expect proper unit tests using React Testing Library for each functionality on the application with at least 80% average code coverage.

Non Functional Requirements

- At HashedIn, quality is our north star. Anyone can write code, It is the quality of code which differentiates us from others. We expect individuals to adhere to code quality and best practices. Follow the following pointers to write cleaner and better code -

Create React App

Use create react app with typescript as the template and then start the project. Refer from [this](#).

React Hooks

Use only react hooks for this assignment. Refer from [this](#).

Linting/Formatting

Use linting tools like Eslint, and ensure that there are no linting errors. Use Prettier for formatting. Refer from [this](#).

Fonts

[Montserrat](#) is the default font for this assignment. Make sure to add it in your "index.html" file.

Best React Practices:

React is pretty good at handling itself, that doesn't mean you are on bail. Take cues from the sessions and reading material provided, to ensure that you follow the best practices in React.

Helpful developer tools:

1. To check pixel perfection : [PixelPerfect plugin](#)
2. To find font size and family: [WhatFont](#)
3. To find colours from a page: [EyeDropper](#)
4. [React Dev Tools](#)

Evaluation Criteria:

Merely covering the requirements would not be enough. There would be heavy focus on code quality and robustness of code in general. The evaluation criteria as mentioned below would be adhered to at completion of given time.

Criteria	Percentage
Milestone 1	15%
Milestone 2	10%
Milestone 3	10%

Milestone 4	10%
Milestone 5	10%
Error State Handling	5%
Coding Best Practices/Code Quality 1. Component Breakdown 2. Hooks 3. Semantic HTML and CSS 4. TypeScript 5. Pixel perfection 6. Git commits and description	40%

Functional Requirements:

1. Top Navigation

- All the icons on the top bar should be clickable and should display the respective pages on the icon click.
- **HashedIn by Deloitte Icon** should be placed on the left and clicking on it should open the homepage.
 - **Courses** - It should also open the homepage on the click
 - **My Wishlist** - Should show all the courses added in the Wishlist
 - **Cart Details** - Should show all the courses added to the cart
 - **My Profile Page** - Should show the user's profile

2. Dashboard

- It should have below features:
 - **Page Banner**
 - displaying 'Discover Latest Courses on React' .
 - **List of courses**
 - Each course should show some basic details as:
 - The Complete React Developer Course
 - Build Responsive Websites with HTML5 and CSS
 - It should have two actions
 - Add To Cart -
 - Clicking on it will add the course to the cart.
 - Add to Wishlist -
 - Clicking on it will add the course to the wishlist.
 - Courses already added in the wishlist should be

marked with a yellow star.

- Clicking on the arrow should open the course details page for the selected course.
- **Show Discounted Price**, if applicable
 - Some courses should be available at discounted price. In such a scenario, strike the actual price and show the discounted price. Otherwise, just show the actual price.
- **Show Popup**
 - Popup should be shown once the course is added to the cart stating 'Course successfully added in the cart'.
 - If the user tries to add the course which is already present in the cart, an alert message should be shown to the user stating 'Already exists in the cart' with the course name.
- **Implement Pagination**
 - Course list should be paginated. It should show 4 (this number should be configurable) courses initially. Clicking on the next/previous button, should change the courses accordingly.
- **Search bar**
 - User should be able to search the course by the course title, author or tags.
- **Sorting**
 - User should be able to see the sorted data based on price i.e Sort by Lowest and Highest
- **Cart**
 - The cart widget on the right side should be updated in real time to show the list of courses added to the cart and should display the total cost. It should have an action 'GO TO CHECKOUT ', clicking on which will take them to the cart page.
 - The cart widget should not move if the page is being scrolled.

3. Shopping Cart (Checkout Screen)

- This page should list the courses added to the cart. It should have below features:
 - **Course Card**
 - Each course card should have a feature to:
 - Move to Wishlist
 - It will remove the course from the cart and will add to the wishlist.
 - Delete the course
 - User should be able to delete the course if not needed.
 - Clicking on any of the above actions should update the cart price.
- **Price Card**
 - Shown on the right should show the total cart value and savings amount.
 - It should have a Checkout button, which should show the popup on click, notifying the user about the successful order placement.
 - The cart should get empty once the user clicks on "OK" in the success modal.

4. Course Details

- This page should show the information about the selected course. It should have below features:
 - **Page Banner**
 - displaying 'Discover Latest Courses on React'
 - **Breadcrumb**
 - It should have the title as All Courses with the link to dashboard Page followed by the selected course title name.
 - **Course Content**
 - It should describe the course on the left side.
 - **Video Card**
 - This card should have any thumbnail image.
 - Clicking on which should play a video.
 - The card should have the price details (discounted price too if applicable).
 - This card should also allow the user to add the course to the cart or the wishlist and show the popup once the course is added to the cart.
 - **Implement time left feature**
 - Some courses are available on sale at a discounted price.
 - The card should show how many hours are left for the sale to get over.
 - Time should be calculated on the basis of time left for the next day to start.
 - Time should only be shown for those courses whose sale will end in less than 24 hours.

5. My Wishlist

- It should show the list of courses added to the wishlist.
 - Each course should be given an option to Add to cart.
 - Clicking on it should get an update similar to what we have on the dashboard.
 - It should have the delete option.

6. My Profile

- This page should allow the user to enter his/her basic details
 - Profile Image on the left side.
 - User details should have
 - Display name
 - First name
 - Last name
 - About yourself
 - Your area of interest (user can select multiple options)
 - Are you a student or a professional?
 - If professional, then
 - Experience
 - 0-5
 - 5-10
 - 10 & above
 - Expertise

- Backend
 - Frontend
 - Mention your role
- Add appropriate validations to each of the fields.
- Save Button should be disabled when any of the validations fail.
- On click of save, show a popup that says Your profile is updated.

Submission Checklist:

1. Make sure latest code is pushed to git in master branch
2. Deploy your app.