

Assignment 008 — Hero IO

Deadline For 60 marks: 10 October , 2025 (11:59 pm)

Deadline For 50 marks : 11 October , 2025 (5:59 pm)

Deadline For 30 marks: Any time after 11 October , 2025 (5:59 pm)

Basic Requirements

- Make the Application Responsive for All the Devices
 - Do minimum 5 commits to your github Repository
 - Give a meaning full name to your application
 - Make Sure on Production Application is error Free
 - Add a Readme.md with App Name , Description & Technologies
-

Main Requirements

1. Layout & Data Design

Header

- The header must include:
 - A logo.on clicking it user will be navigated to home page.
 - A navigation bar with links [**home** , **apps** , **installation**] and active route indication.
 - A **Contribution** button as Figma linking to the your GitHub profile.

Footer

- Design a custom footer using your own creativity and style.

Data

- Create an JSON Array of minimum 12-20 objects for app data using the following structure:

```
{  
  image: string;  
  title: string;  
  companyName: string;  
  id: number;  
  description: string;  
  size: number;  
  reviews: number;  
  ratingAvg: number;  
  downloads: number;
```

```
ratings: [  
  { name: "1 star"; count: number },  
  { name: "2 star"; count: number },  
  { name: "3 star"; count: number },  
  { name: "4 star"; count: number },  
  { name: "5 star"; count: number }  
];  
}
```

2. Home Page

Banner

- Must contain a center-aligned heading, text, and two buttons.
- “App Store” button will redirect to the App Store.
- “Play Store” button will redirect to the Play Store.

States Section

- Must contain three state cards as shown in Figma.

Top Apps Section

- Display eight apps in a four-column layout.
 - Each app card should display:
 - App title
 - Image
 - Download count
 - Average rating
 - Clicking on a card should navigate the user to the App Details page.
 - Include a “Show All” button that navigates to the All Apps page.
-

3. All Apps Page

Title Section

- Include a title and subtitle following the Figma design.

Search and States

- Display the total number of apps on the left and a search bar on the right.
- Implement live search functionality
 - filters apps by title as the user types.
 - Search will be case-insensitive
 - If no app matches, display a “No App Found” message.

App Section

- Display all apps from the JSON data.
 - Each app card should include:
 - App title
 - Image
 - Download count
 - Average rating
 - Clicking on an app card should navigate to the App Details page.
-

4. App Details Page

App Information

- Show app image on the left.
- Display app details such as title, rating, downloads, reviews.
- Include an **Install** button:
 - When clicked, it becomes disabled and the text changes to **Installed**.
 - Show a Success Toast after App installed

App Review Chart

- Implement a responsive chart using the **Recharts** library.
- Visualize the app's review data as shown in the Figma design.

App Description

- Show the app details in description section as per Figma layout.
-

5. Error Page & Others

- Create a custom error page for invalid routes.
 - Show a loading animation during: **Challenge Part**
 - Page navigation.
 - Search operation.
 - Show a Relevant Not Found message app not found in app details section.
 - Ensure that reloading any route after deployment does not cause an error.
-

Challenge Requirement

LocalStorage Features

App Installation

- When the “Install” button is clicked:
 - Save the app to localStorage.
 - If the app is already installed, show a disabled button with the text **Installed**.

My Installation Page

- Create a page named “My Installation” following the Figma design.
 - Display all installed apps as cards.
 - Include an Uninstall button:
 - Clicking it
 - * removes the app from both the UI and localStorage.
 - * Show an Toast with some relevant message
-

Sort by Downloads

- Implement a dropdown for sorting apps by download count.
 - The dropdown must include:
 - **High-Low:** Sort apps in descending order by downloads.
 - **Low-High:** Sort apps in ascending order by downloads.
-

Loading Animation

- Show a loading animation during:
 - Page navigation.
 - Search operation
-

Deployment

- Deploy the project to Cloudflare / Netlify / Vercel.
 - Reloading from any route must work correctly without showing a 404 error.
-