

# Internship Assignment

## Tasks:

1. UI Implementation from [Figma](#):
  - Convert the provided Figma designs into a working React application.
  - Ensure pixel-perfect accuracy and responsiveness across different screen sizes.
  - Implement all visual elements, layouts, and styles as specified in the Figma designs.
  - Pay close attention to details such as typography, spacing, colors, and component reusability.
  - In addition to the current Figma add one click to open third pane functionality.
2. Component Breakdown:
  - Break down the UI into reusable React components.
  - Organize the component structure in a logical and maintainable way.
  - Document the component hierarchy and props.
3. Interaction and Functionality:
  - Implement interactive elements as defined in the Figma designs (e.g., buttons, forms, lists).
  - Ensure that user interactions trigger appropriate state updates and API calls. The *workflow* of user interactions should be implemented in React.
4. Code Quality and Best Practices:
  - Write clean, well-documented, and maintainable code.
  - Follow React best practices and coding conventions.
  - Use meaningful variable and function names.
  - Implement proper error handling.

## Evaluation Criteria:

- Accuracy in translating Figma designs into React components
- Code quality, organization, and maintainability
- Correct implementation of functionality and user interactions, specifically the React implementation of the workflow.
- Responsiveness and cross-browser compatibility
- Adherence to best practices and coding conventions
- Proper use of Git for version control

**Submission:**

- A link to the GitHub repository containing the completed project.
- A link to the deployed worksite.
- A brief document explaining the project structure, component hierarchy, and any challenges faced.

**Deadline:** Monday April 28, 2025.

**Instructions for the Intern:**

1. Read the tasks carefully and work on each of the mentioned tasks.
2. Add interactivity: Implement the required user interactions, such as button clicks, form submissions, and data updates. This is where you implement the workflow in React.
3. Integration: Mock a couple of integration with Google or any other source, with free credits to assess the integration skills.
4. Test thoroughly: Test your application on different browsers and screen sizes to ensure it is responsive and works correctly.
5. Document your code: Write clear and concise comments to explain your code and its functionality.
6. Use Git for version control: Commit your code regularly and push it to a GitHub repository.
7. Submit your work: Once you have completed the assignment, submit the GitHub repository link and a brief document explaining your project.