

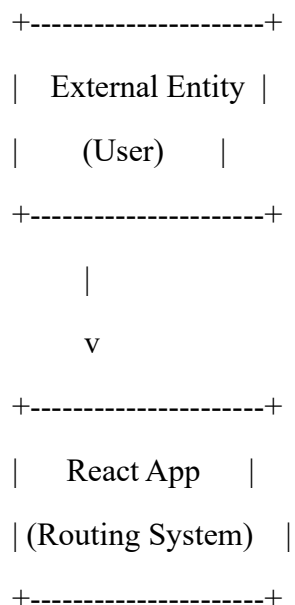
DFD for React Router-Based App

This React application uses React Router for navigation, including dynamic routing for posts and a 404 page for undefined routes. The key interactions are:

1. Users navigate between Home, About, and Contact pages.
2. Users can visit dynamic post pages using a unique id.
3. If an invalid URL is entered, the 404 page is displayed.

Level 0 (Context Diagram)

At **Level 0**, the React App is treated as a **single process** interacting with the user.

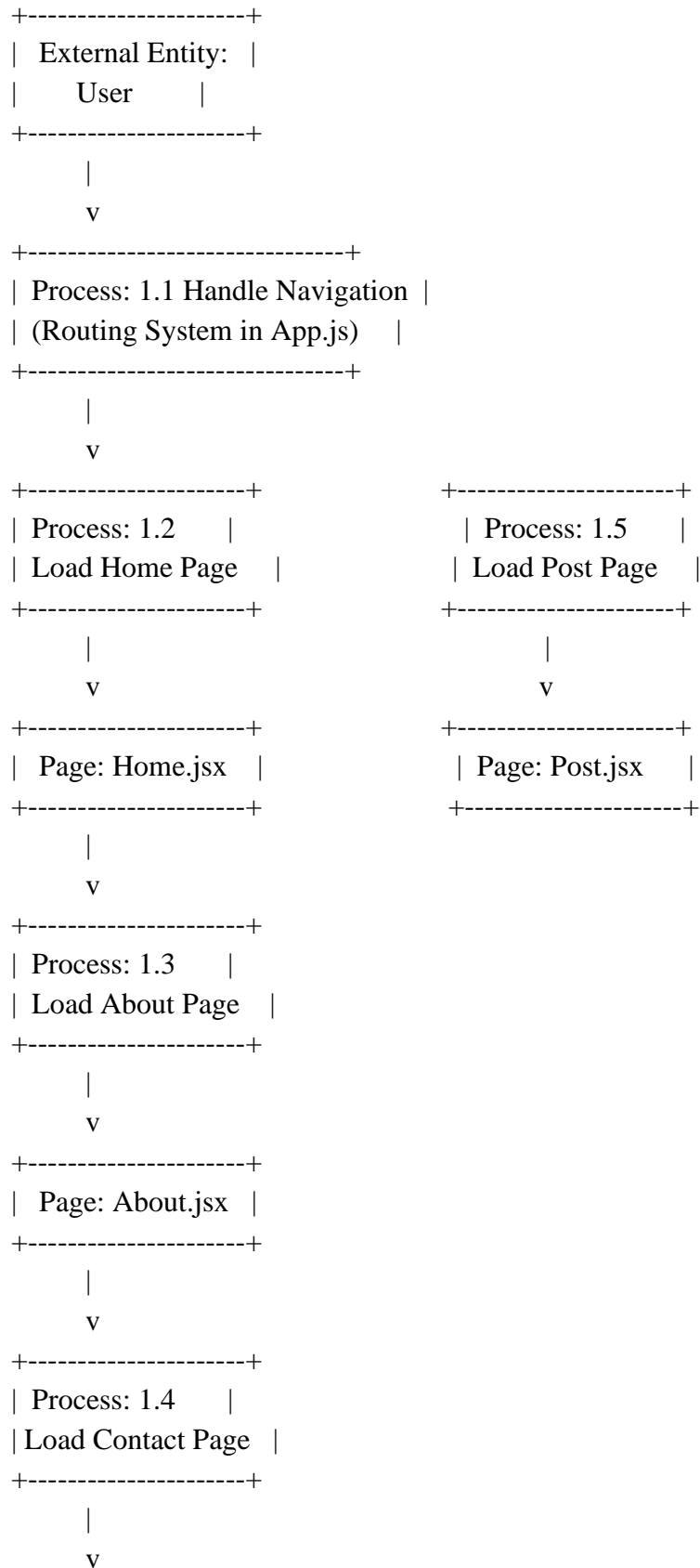


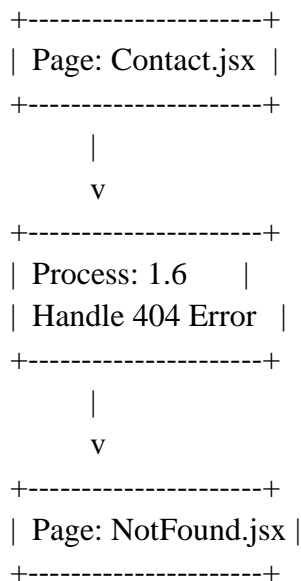
Explanation

- **User (External Entity):** The user interacts with the application by navigating pages.
- **React App (Process):** Handles user requests and dynamically loads the correct page.

Level 1 (Decomposing the Routing System)

- Breaking down the **React Router functionality** into sub-processes:





Explanation of Processes

◆ Process 1.1 - Handle Navigation (React Router in App.js)

- User clicks on a link (**Home**, **About**, **Contact**, **Post**).
 - **React Router** processes the route and determines the correct page to render.
 - The system **dynamically loads** the corresponding component.
-

◆ Process 1.2 - Load Home Page (Home.jsx)

- If the user **clicks on "Home"**, React Router navigates to the **Home page** (/).
 - **Component Rendering**: The Home.jsx component is loaded and displayed.
 - **Data Flow**: No external data is fetched.
-

◆ Process 1.3 - Load About Page (About.jsx)

- If the user **clicks on "About"**, React Router navigates to the **About page** (/about).
 - **Component Rendering**: The About.jsx component is loaded and displayed.
-

◆ Process 1.4 - Load Contact Page (Contact.jsx)

- If the user **clicks on "Contact"**, React Router navigates to the **Contact page** (/contact).
 - **Component Rendering**: The Contact.jsx component is loaded and displayed.
-

◆ Process 1.5 - Load Dynamic Post Page (Post.jsx)

- When a user visits /post/:id, React Router extracts the **post ID** from the URL.
- The **Post.jsx** component fetches and displays the specific post based on the id.
- **Data Flow**:
 - The **post ID** is extracted from the URL.
 - API call to fetch post details (fetchPostData(id)).
 - The post content is displayed.

◆ **Process 1.6 - Handle 404 Error (NotFound.jsx)**

- If the user enters an **invalid URL**, the system redirects to the **NotFound.jsx** page.
- This ensures that users don't see a blank screen for incorrect links.

Summary

Process	Action
Process 1.1 - Handle Navigation	Determines which page to render
Process 1.2 - Load Home Page	Renders <code>Home.jsx</code>
Process 1.3 - Load About Page	Renders <code>About.jsx</code>
Process 1.4 - Load Contact Page	Renders <code>Contact.jsx</code>
Process 1.5 - Load Post Page	Extracts post ID, fetches data, and renders <code>Post.jsx</code>
Process 1.6 - Handle 404 Error	Redirects to <code>NotFound.jsx</code> for invalid URLs