**System Deployment & Integration**

**1. Technology Stack**

* **Frontend:** React, Redux, HTML, CSS, JavaScript
* **Backend:** Not applicable (Frontend-only project)
* **Database:** Not applicable (Using third-party APIs like TMDb for data)
* **API Integration:** The Movie Database (TMDb) API
* **Hosting & Deployment:** Netlify or Vercel

**2. Deployment Diagram**

The deployment diagram outlines how the frontend application is hosted and accessed by users. Since this is a frontend-only project, the architecture is simple:

* **Client (User’s Browser):** Accesses the application via a deployed link.
* **Hosting Service (Netlify/Vercel):** Serves the React application files.
* **Third-party API (TMDb API):** Provides movie data to the frontend via API requests.

**3. Component Diagram**

The component diagram illustrates how different parts of the application interact:

* **User Interface (UI):** Components for displaying movies, search, filtering, and recommendations.
* **State Management (Redux):** Manages global application state (movie data, filters, user preferences).
* **API Service:** Handles API requests and responses to fetch movie data from TMDb.
* **Routing (React Router):** Manages navigation between different views (home, search results, movie details).
* **Error Handling Module:** Manages errors from API requests and provides user-friendly messages.

This structure ensures a seamless, scalable, and maintainable movie recommendation application.