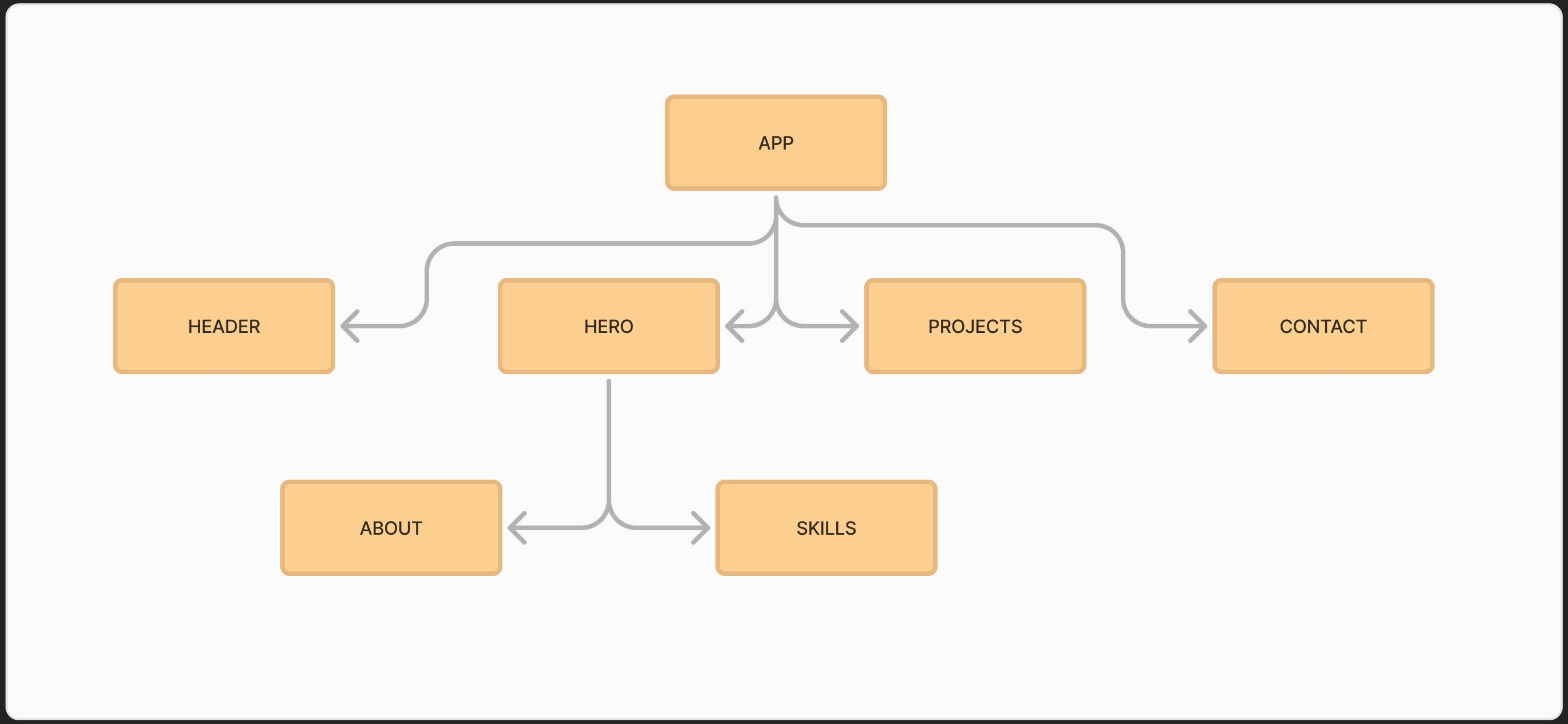
1. COMPONENTS: Parts of the user interface are defined by reusable, self-contained code segments called components. Complex interfaces can be constructed by combining them. You can make components for a portfolio's Header, About, Projects, and Contact sections—each in charge of displaying a different type of content.
2. STATE: Within a component, state is utilized to manage dynamic data. When data changes, it enables React components to update and re-render, enabling interactive features like form input management, element toggling, and project filtering in a gallery.
3. PROPS: Props are read-only data that are transferred between parent and child components to facilitate data sharing and communication. Components become more reusable and versatile as a result. To display project details dynamically, for example, a project component may receive various project details as props.

2.1.3 FLOWCHART TASK


APP
HEADER
HERO
PROJECTS
CONTACT
ABOUT
SKILLS