**Project Design Phase-II**
**Technology Stack (Architecture & Stack)**

| | |
|---|---|
| Date | 15 April 2025 |
| Team ID | SWTID1743870329 |
| Project Name | Personal Expense Tracker |
| Maximum Marks | 2 Marks |

## Technology Stack :

## Frontend Technologies

ReactJS

- Core library for building dynamic and responsive user interfaces.
- Utilizes React hooks such as useState, useEffect, and useContext for efficient state management and lifecycle handling.

React Router

- Implements client-side routing to navigate between different pages and views.
- Features like useNavigate and ProtectedRoute ensure secure and seamless navigation.

React Icons

- Enhances the UI using icon libraries such as react-icons/lu, react-icons/io, react-icons/bs, and react-icons/wi.

Styling

- Tailwind CSS is likely used as a utility-first CSS framework, based on class names like bg-primary, bg-orange-500, etc.

- Google Fonts integration ensures a customized and consistent typography experience.

Toast Notifications

- react-hot-toast is used to provide user-friendly notifications and alerts across the application.

Custom Components

- Reusable and modular components like InfoCard, Modal, DeleteAlert, and layout components such as DashboardLayout and AuthLayouts help maintain clean and maintainable code architecture.

## Backend Communication

Axios

- Handles HTTP requests (GET, POST, DELETE, etc.) to communicate with the backend APIs.

- Facilitates data submission, retrieval, and file downloads such as income/expense reports.

API Paths Configuration

- A centralized API_PATHS configuration ensures efficient management and modification of backend endpoints.

## Authentication

Custom Hooks & Context

- Authentication is implemented using custom React hooks like useUserAuth and AuthProvider for managing login, logout, and route protection.

JWT Integration

- JSON Web Tokens (JWT) are used for securing API communications and user-specific route protection.

### Utilities

Helper Functions

- Includes utilities such as addThousandsSeparator for number formatting and validateEmail for input validation.

File Upload Utility

- A utility function like uploadImage is used during user signup or profile updates.

### Data Management

- React's built-in state management (useState, useContext) is extensively used throughout the app.

- While libraries like Redux or MobX are not used, the state is effectively managed via hooks and context.

### Styling Frameworks

Tailwind CSS (assumed)

- Based on utility class usage, Tailwind CSS is likely the primary styling tool for layout, spacing, and color theming.

### Development Practices

- Modular component structure and modern development practices are followed.

- Efficient folder hierarchy, centralized configs, and reusable logic contribute to clean code and scalable architecture

| **Category** | **Technology/Library** |
|---|---|
| Frontend Framework | ReactJS |
| Routing | React Router |
| Icons | React Icons(Lucide,Ionicons) |
| Backend Communication | Axios(Custom Instance) |
| Authentication | Custom Hook (useUserAuth) |
| Utilities | Helper Function (addThousandsSeparator) |
| Styling | Likely Tailwind CSS |