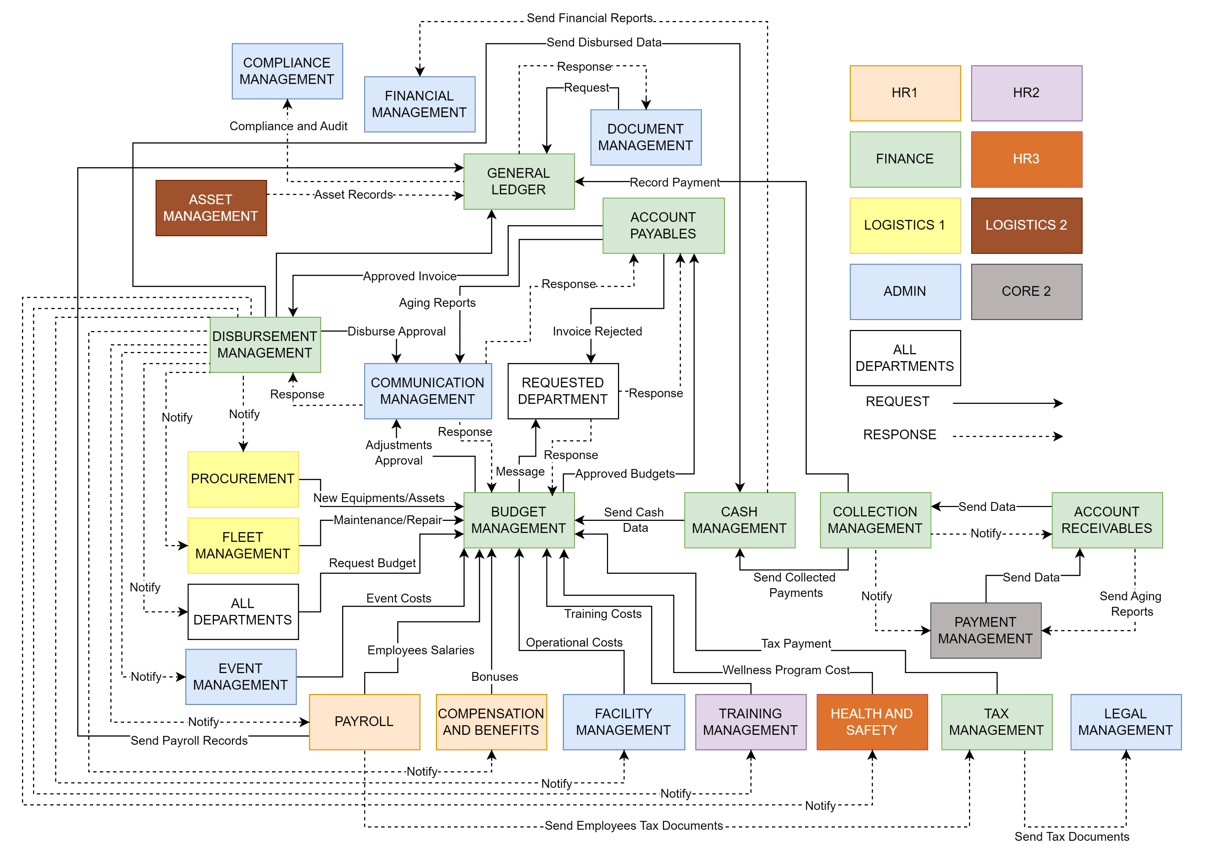
**3.8 Introduction to TOGAF and Four Architectural Domains**

The TOGAF (The Open Group Architecture Framework) structures our finance system by aligning business objectives, organizing financial data, optimizing budget calculations, and supporting the system with a full-bodied IT infrastructure for efficient financial management. It includes the Four Architectural Domains: Business Architecture, Data Architecture, Application Architecture and the Technology Architecture

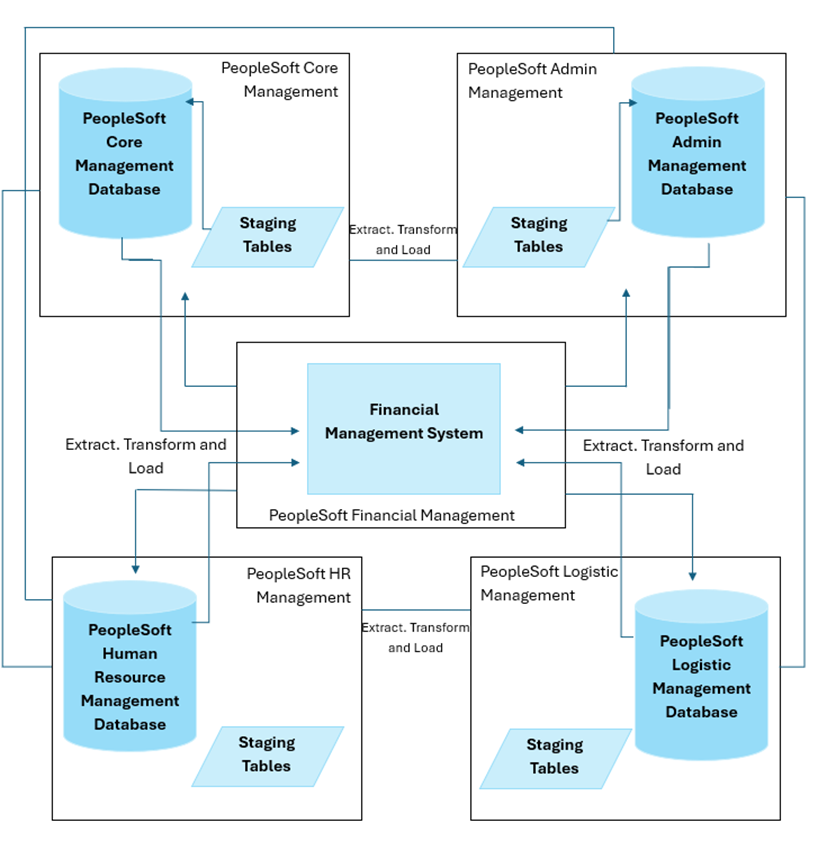
**Business Architecture**

Our Business Process Architecture organizes and connects financial processes, ensuring seamless integration of budget estimation, resource allocation, and financial planning to optimize efficiency and align with strategic goals.

****

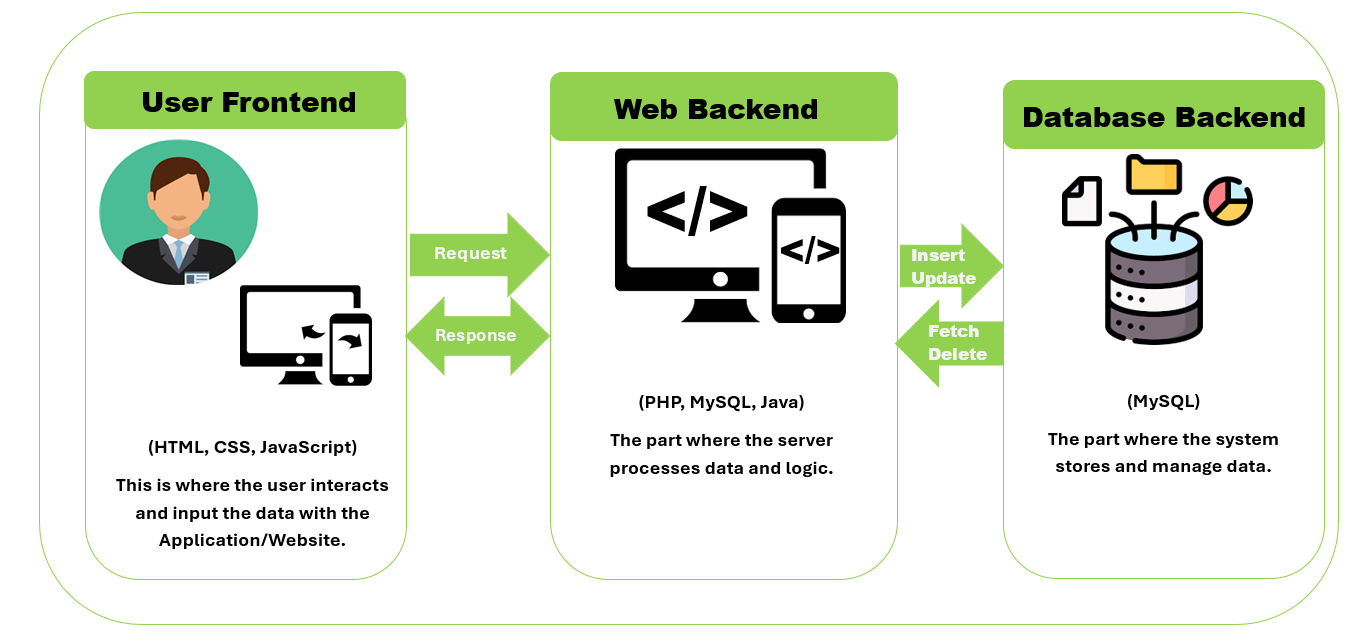
**Data Architecture**

In Data Architecture defines the structure and flow of financial data, ensuring efficient storage, processing, and accessibility to support accurate budget estimation and informed decision-making.



**Application Architecture**

It outlines how budget estimation tools, user interfaces, and data processing components interact. It ensures efficient financial calculations and reporting, providing a seamless experience for users.



**Technology Architecture**

These are the applications we used in system development to create architecture, coding, design, analysis, programming language used, icons used and others that you can also see here.

|  |  |
| --- | --- |
|  | MySQL is an open-source relational database management system (RDBMS) that uses SQL to organize and manage data. It is known for being reliable, flexible, and is often chosen for web applications due to its ability to handle large-scale databases and deliver high performance. |
|  | PHP (Hypertext Preprocessor) is a popular server-side scripting language designed primarily for web development. It’s widely used to build dynamic and interactive websites, often working in combination with databases like MySQL. PHP can efficiently handle backend logic, interact with databases, and serve data to front-end users. |
|  | CSS (Cascading Style Sheets) is the language used to define the visual presentation of a web page, determining how HTML elements are displayed in terms of layout, colors, fonts, and overall design. It separates content (HTML) from design, allowing you to create aesthetically pleasing and responsive web interfaces. |
|  | HTML (Hypertext Markup Language) is the standard language used to create and structure content on the web. It provides the foundational framework for web pages by defining elements such as text, images, links, forms, and multimedia. HTML works hand-in-hand with CSS (for styling) and JavaScript (for interactivity) to deliver fully functional and visually appealing web applications. |
|  | JavaScript is a powerful, dynamic programming language used primarily for creating interactive and responsive elements on web pages. It works alongside HTML and CSS to build dynamic web applications, enabling real-time updates, user interaction, and advanced functionality without needing to reload the page. |
|  | Tailwind CSS is a utility-first framework that offers a collection of prebuilt CSS classes, allowing developers to rapidly style web interfaces without writing custom CSS. By using utility classes for layout, typography, colors, and other design elements, Tailwind streamlines the development process, enabling consistent, responsive designs while keeping the CSS file size minimal. |