Technical Architecture

**Technologies for Front-end:**

**HTML5:**

HTML5 is the latest version of HTML (Hyper Text Markup Language) having new elements, attributes, and behaviors, and a larger set of technologies that allows the building of more diverse and powerful Web sites and applications. It is classified into several groups based on their functions like Semantics, Connectivity, Offline and storage, Multimedia, 2D/3D graphics and effects, Performance and integration, Device access, Styling.

**CSS3:**

CSS stands for Cascading Style Sheets. CSS is a standard style sheet language used for describing the layout and formatting of the web page.

We can easily apply same style rules on multiple elements. We can control the presentation of multiple pages of a website with a single style sheet. We can style dynamic states of elements such as hover, focus, etc. that isn't possible otherwise. We can change the position of an element on a web page without changing the markup.

Advantages: CSS Save Lots of Time, Easy Maintenance, Pages Load Faster, Superior Styles to HTML, Multiple Device Compatibility, etc.

**Bootstrap4:**

Bootstrap 4 is a free front-end framework for faster and easier web development. Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins Bootstrap also gives us the ability to easily create responsive designs.

We will be using Bootstrap4 CDN (Content Delivery Network).

**Data Driven Document (D3) library:**

D3 is an open-source JavaScript library developed to create custom interactive data visualizations in the web browser using SVG (Scalable Vector Graphics), HTML and CSS. Visual representations of data are the most effective means of conveying meaningful information and D3 provides a great deal of ease and flexibility to create these data visualizations. It is dynamic, intuitive and needs minimum amount of effort. D3 focuses more on interactions, transitions and transformations.

Features: Uses Web Standards, Data Driven, DOM Manipulation, Data Driven Elements, Dynamic Properties, Types of visualization, Custom Visualizations, Transitions, Interaction and animation.

Advantages: D3.js is a JavaScript library. So, it can be used with any JS framework of our choice like Angular.js, React.js or Ember.js. It works with web standards so we don't need any other technology or plugin other than a browser to make use of D3. D3 does not provide any specific feature, so it gives us complete control over our visualization to customize it the way we want. Since D3 is lightweight, and works directly with web standards, it is extremely fast and works well with large datasets.

**Back-end:**

**NodeJS:**

Node.js is an open source server environment. Node.js runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.). It uses JavaScript on the server. It uses asynchronous programming.

Advantages: Node.js can generate dynamic page content. It can create, open, read, write, delete, and close files on the server. It can collect form data and it can add, delete, modify data in our database.

Node.js files contain tasks that will be executed on certain events. A typical event is someone trying to access a port on the server. Node.js files must be initiated on the server before having any effect.

**ExpressJS:**

Express.js, or simply Express, is a back end web application framework for Node.js, released as free and open-source software. It is designed for building web applications and APIs.

**Middle-Tier (Between Back-end and database):**

**Mongoose:**

Mongoose provides a straight-forward, schema-based solution to model our application data. It includes built-in type casting, validation, query building, business logic hooks and more, out of the box.

**Data Source:**

**MongoDB:**

MongoDB is a cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas.

**Integrations:**

**ReactJS:**

React is a JavaScript library for building user interfaces. React is used to build single page applications. React allows us to create reusable UI components.

**XML Files:**

**JavaScript XML (JSX):**

JSX allows us to write HTML in React. JSX makes it easier to write and add HTML in React. JSX converts HTML tags into react elements. JSX is an extension of the JavaScript language based on ES6, and is translated into regular JavaScript at runtime.

JSX will throw an error if the HTML is not correct, or if the HTML misses a parent element. We will use it for debugging.

**Screen Designing:**

Responsive web design is about creating web pages that look good on all devices! But for our application medium to large screen devices are preferable. A responsive web design will automatically adjust for different screen sizes and viewports.

To create a responsive website, add the following <meta> tag to all your web pages:

<meta name="viewport" content="width=device-width, initial-scale=1.0">

**Testing:**

**End to End Testing:**

End-to-end testing is a technique used to test whether the flow of an application right from start to finish is behaving as expected. The purpose of performing end-to-end testing is to identify system dependencies and to ensure that the data integrity is maintained between various system components and systems.

The entire application is tested for critical functionalities such as communicating with the other systems, interfaces, database, network, and other applications.

**Jest:**

Jest JavaScript resting framework with a focus on simplicity. Jest is used for testing React projects.

Unit testing is a methodology to test software where individual units (components) of a software are tested. The purpose of unit testing is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software.

Mocking is technique where code parts are replaced by dummy implementations that emulate real code. Mocking helps achieve isolation of tests. Mocking is primarily used in unit testing.

**Servers:**

**Github server pages:**

GitHub Pages is a static site hosting service that takes HTML, CSS, and JavaScript files straight from a repository on GitHub Enterprise Server, optionally runs the files through a build process, and publishes a website. (Link:- [Visual Wizard-Magical DSA](https://priyank-mungra.github.io/Magical_DSA/Mark-1/index.html))

**Platforms:**

**Version Control System- Git:**

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is easy to learn and has a tiny footprint with lightning fast performance. It comes with features like cheap local branching, convenient staging areas, and multiple workflows.

We will use Git on Github.

**Integrated Development Environment (IDE)- Vs code:**

Visual Studio Code is a free source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

**OAuth API:**

OAuth2 is the preferred method of authenticating access to the API. OAuth2 allows authorization without the external application getting the user's email address or password. Instead, the external application gets a token that authorizes access to the user's account.

**Languages:**

**JavaScript:**

JavaScript is the world's most popular programming language. JavaScript is the programming language of the Web. JavaScript is easy to learn.