LLM Apps

Intro to Web Applications

Web Application

- What it is
- Differences with mobile application
- Types
- Development process
- A web application is like a LEGO construction
- Keys to prevent a project from getting out of control
- Main components
- Main processes
- Usual functionalities
- LLM Web Application Development: key phases

Web Application: What is it

- A web application is a software program that runs in a web browser instead of being installed locally on a device.

- It is accessible over the Internet or an internal network and runs on a web server, not on the user's device. This means that you can use it from any device with a web browser and an Internet connection.

- Web applications are popular because they do not require specific installation and are automatically updated on the server, ensuring that all users access the most recent version.

Web Application vs. Mobile Application

- Platform and Access:
  - Web Applications: Run in web browsers and are accessible from any device with an internet connection and a web browser (like Chrome, Firefox, Safari). They do not require downloading or installation.
  - Mobile Applications: Are specifically designed for mobile devices and are downloaded and installed from app stores like Google Play Store or Apple App Store.

- Development:
  - Web Applications: Are generally developed with technologies like HTML, CSS, and JavaScript. They are easier to maintain, as they are updated directly on the server and users always access the latest version.
  - Mobile Applications: Are developed using languages and frameworks specific to mobile platforms, like Swift for iOS or Kotlin for Android. They may require separate maintenance and updates for each platform.

Web Application: Types

- The uses of web applications are endless:
  - Informative/Educational.
  - E-commerce.
  - Financial Services.
  - Social Networks.
  - Entertainment.
  - Productivity and Business.
  - Communication.
  - Content Management.
  - Health and Wellness.
  - Booking and Planning.
  - Etc.

Web Application: Development Process (1)

1. Planning and Analysis: This initial phase involves defining the goals of the application, identifying the target audience, and analyzing requirements. Functionalities, application architecture, and project planning are established.

2. Design: User interface and user experience (UI/UX) designs are created. This includes developing mockups and prototypes to visualize how the application will look and function.

3. Development: During this phase, code is written to create the application. Both the front-end (the part that interacts with the user) and the back-end (server logic and database) are developed.

4. Testing: In this stage, tests are carried out to ensure that the application works correctly. This includes tests of functionality, usability, security, compatibility, and performance.

Web Application: Development Process (2)

5. Deployment: Once the application is tested and ready, it is deployed on a production server. This means that the application goes online and is available to end-users.

6. Production: The application is operational and accessible to users. During this phase, its performance is monitored to ensure optimal functioning.

7. Maintenance: This phase is a continuous process involving updating the application, fixing bugs, improving features, and ensuring it remains secure and efficient as technologies and user needs change.

8. Evolution: Based on user feedback and changes in the market or technology, the application may require major updates, new features, or even a complete restructuring.

A Web Application is like a LEGO

- Don't be overwhelmed by the number of pieces. Web development relies on many pre-designed packages. No software engineer masters them all before starting to work on a project.

- The essential thing is to understand the key concepts and the main skeleton of the application, the "empty house".

- Decorative details are secondary.

Keys to prevent a project from getting out of control

- Divide the application into small pieces.
- Use testing to ensure that subsequent changes don't break what you have already developed.

Web Application: Main Components (1)

1. Client (Front-end): The part of the application that users see and interact with. Includes everything related to the user interface, such as web pages, images, and styles. Developed using technologies like HTML, CSS, and JavaScript.

2. Server (Back-end): The part that runs on a server and manages the application logic. Processes client requests, performs database operations, and returns results to the client. Developed with programming languages like Python, Java, Ruby, or Node.js.

3. Database: Stores and manages the data needed for the web application. Can be an SQL database (like MySQL or PostgreSQL) or NoSQL (like MongoDB). The server interacts with the database to retrieve, update, and store information.

Web Application: Main Components (2)

4. Cron Job: Scheduled tasks that run automatically at specific intervals. Used for regular maintenance tasks like backups, data updates, or sending automated emails.

5. APIs (Application Programming Interfaces): Allow integration with external services or internal components. For example, a web application might use an API to get weather data or integrate a payment system.

6. File Server (Optional): Some web applications use dedicated servers to store and serve large files, like images, videos, or documents.

Web Application: Main Processes (1)

1. Page Creation: This is the process of designing and developing the individual web pages that users will see. HTML is used to structure the page, CSS for design and style, and JavaScript for interactivity.

2. Dynamic URL Creation: Modern web applications often generate URLs dynamically. This means that URLs can change based on user navigation or interactions in the application. This technique is useful for creating a personalized user experience and for improving SEO.

3. Routing: Routing is the process of defining how an application responds to a client request to a particular route or URL. In the back-end, this involves configuring the server to handle different URLs and send the correct response (such as a web page, an error, or data).

Web Application: Main Processes (2)

4. Connecting with Data Sources: Web applications often need to interact with databases to store and retrieve data. This process involves setting up and using a database, and writing server code to query, update, insert, and delete data.

5. Connecting with APIs: Web applications frequently integrate with external APIs (like Google Maps, Twitter, or payment services) to expand their functionality. This requires making requests to these APIs and processing the responses for use in the application.

6. Authentication and Security: Implementing methods for users to register, log in, and authenticate. Additionally, it is crucial to ensure that the application is secure and protected against common threats such as SQL injection attacks or cross-site scripting (XSS).

Web Application: Usual Functionalities

- **Authentication**: Logging into the application.
- **Creating a User Account**: Registering for a new account.
- **Navigation and Search**: Exploring the sections of the application and searching for information.
- **Interaction with Specific Content** or Services of the application.
- **Payment for Subscription or Services**: Entering payment details.
- **Contact with Support or Customer Service**: Sending inquiries or seeking help.
- **Setting Preferences**: Adjusting display settings and preferences.

Development of LLM Apps: Key Phases

- Phase 1: Mastering the logic of the LLM application.
  - Proof of Concept (POC).
- Phase 2: Understanding the basic environment of the web application.
  - Minimum Viable Product (MVP).
- Phase 3: Extending the basic environment of the web application.
  - Typically, with specialized roles for development and maintenance.
    - Front
    - Back
    - QA
    - Ops
    - Etc.