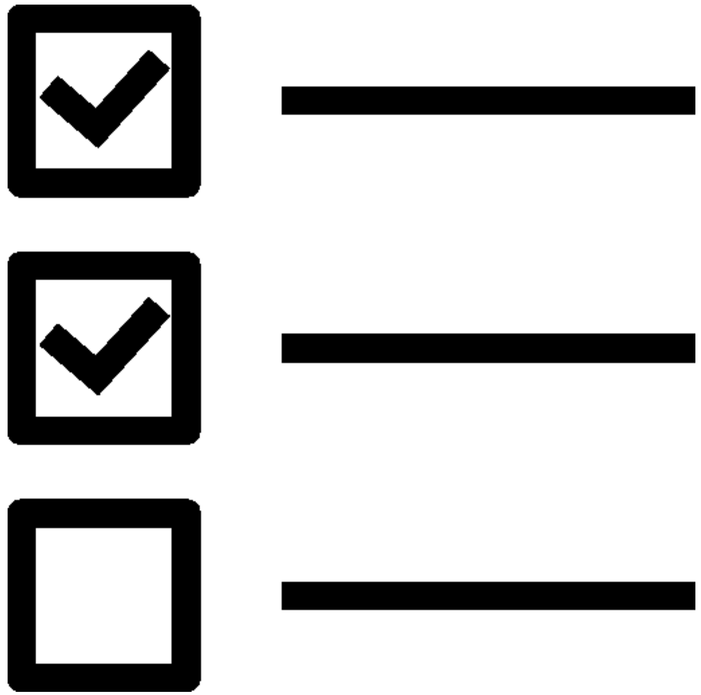
Web Application Development using Python

Web Application Development

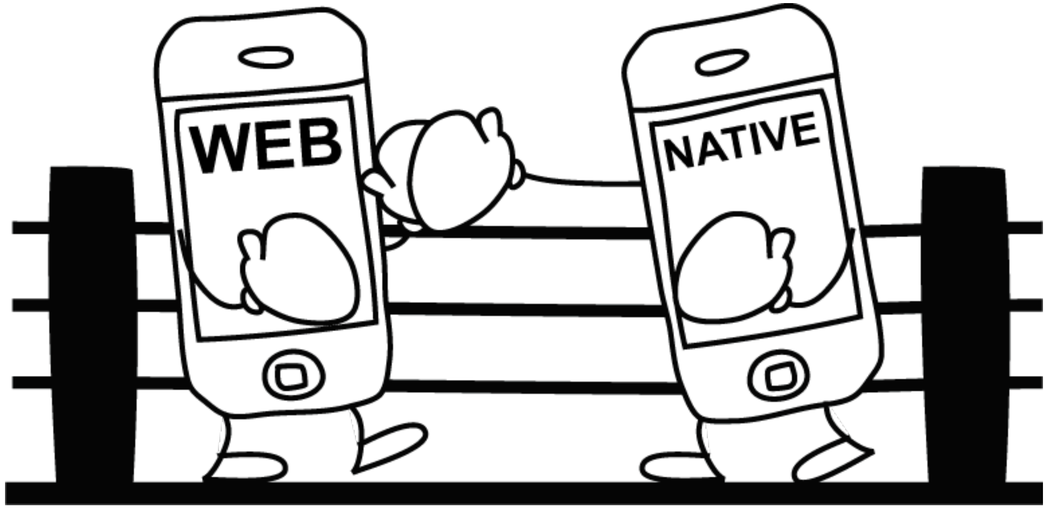
Prepared by George Khoury

Outline

- What is a Web Application?
- Web Application Development Process
- Web Application Development Frameworks
- Architectural Pattern
- HTTP Protocol



What is a Web Application?



What is a Web Application?

- An interactive computer program, built with web technologies (HTML, CSS, JS), which stores (Database, Files) and manipulates data (CRUD), and is used by a team or single user to perform tasks over the internet.
- To make a data-centric web app from the bottom-up, it is advantageous to understand:
  1. Backend language (e.g. Python, Go) - control how your web app works
  2. Web front end (HTML, CSS, Javascript) - for the look and feel of your web app
  3. Deployment - Deploying / hosting your web app

What is a Web Application?

Progressive Web Applications (PWA)

- Progressive web applications are a newer type of web application which behaves like and often outperforms native applications.
- They are web applications which follow a slightly different methodology, and involve an additional set of technologies such as service workers, manifests, push notifications.
- Progressive web applications can be downloaded to your device, and stored on your homescreen which makes them accessible, and ‘native-like’, and unlike web applications they can be accessed and used offline.

What is a Web Application?

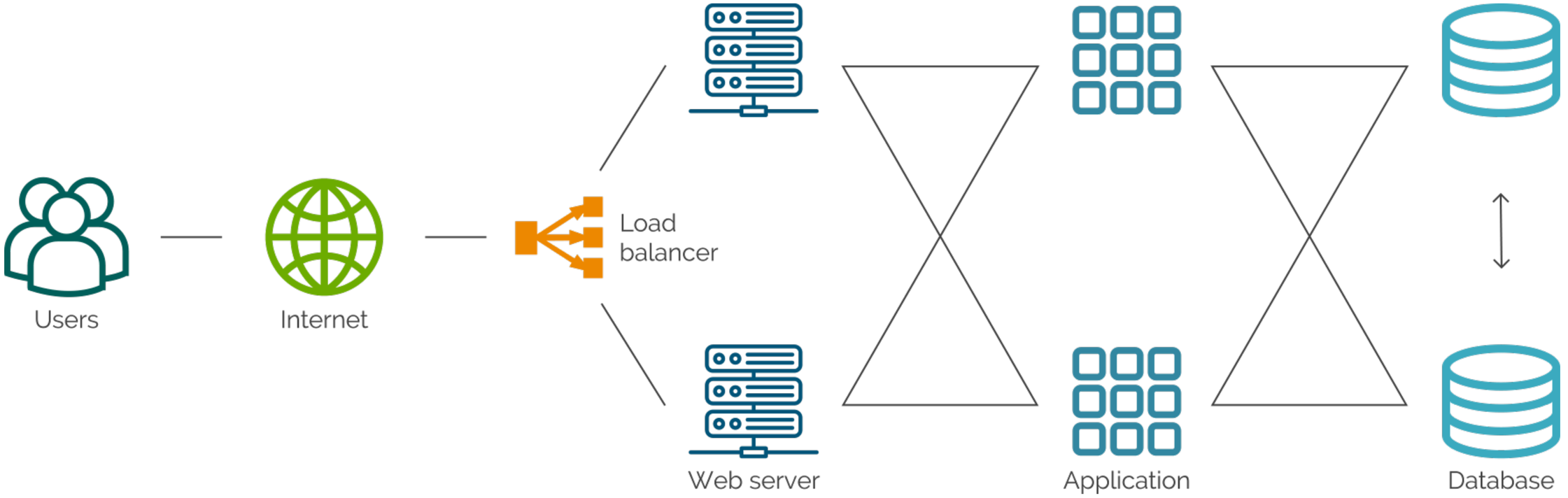
Examples

- **Google Docs** - a word processor included as part of a free, web-based Google Docs suite offered by Google within its Google Drive service.
  - Allows creating, reading, updating, and deleting documents.
- **DuckDuckGo** - is an internet search engine that emphasizes protecting searchers' privacy and avoiding the filter bubble of personalized search results.
  - Allows creating a search query for data already indexed by their engine.
- **Other examples?**

What is a Web Application?

Web applications vs. Websites

- The key difference is how we interact with each.
- **Web applications** are defined by their input - we create, read, update and delete data within a web application.
- **Websites** are defined by their output - we read the news, marketing information, FAQs on websites.


Users
Internet
Load
balancer
Web server
Application
Database

Web Application Development Process

Web Application Development Process

1. Ideation Phase

- Source an idea
- Market research
- Define functionality

2. Design Phase

- Sketch your web app
- Plan your workflow
- Develop wireframes
- Validate your product

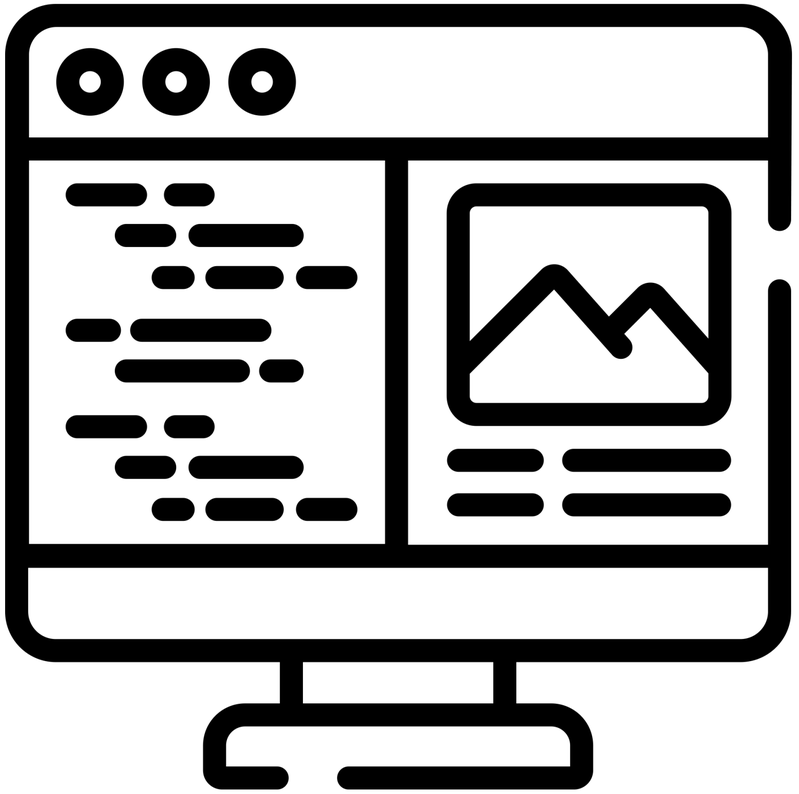
3. Development Phase

- Architect your database
- Develop your frontend
- Develop your backend
- Validate your product

4. Launch Phase

- Deploy your web application

Web Application Development Frameworks



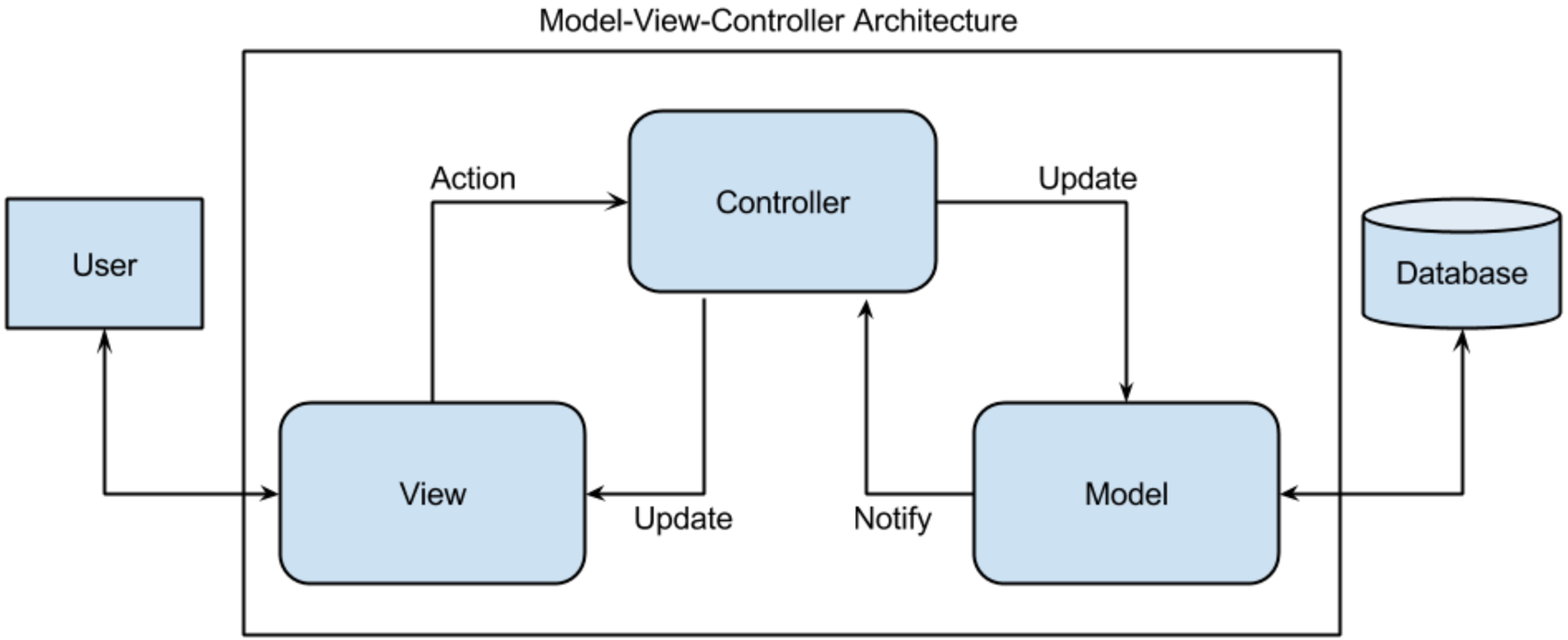
Web Application Development Frameworks

Backend frameworks
Flask with Python
Iris with Go
Rails with Ruby
Django with Python
Laravel with PHP

Frontend frameworks
React
Vue.js
Angular.js

Architectural Pattern

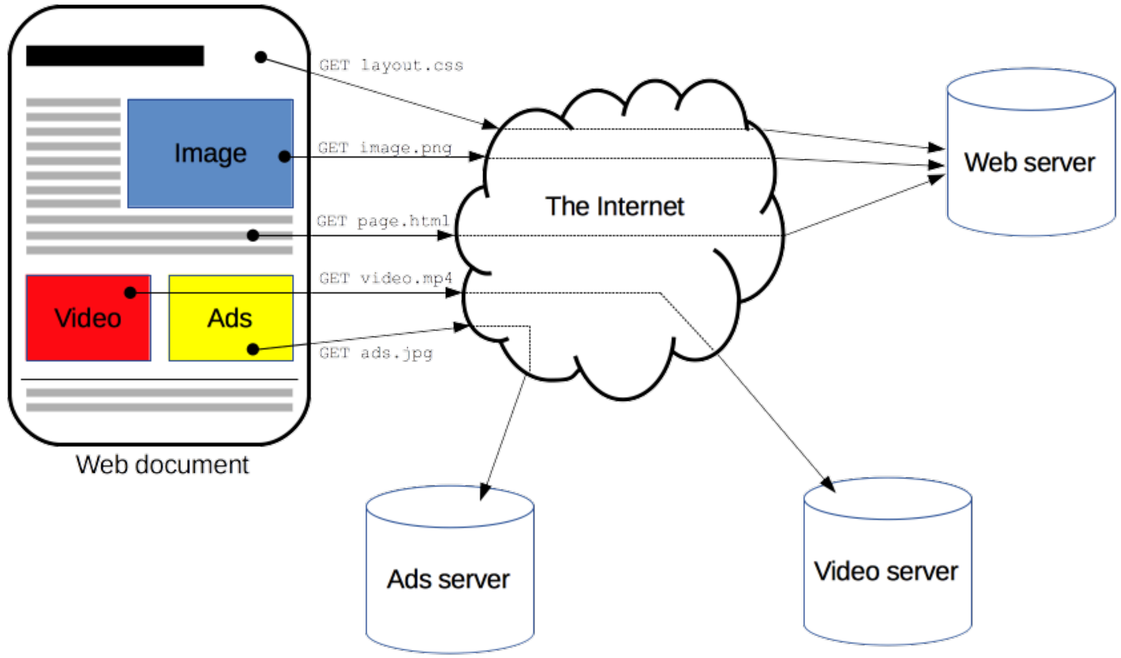
MVC Architecture



HTTP Protocol

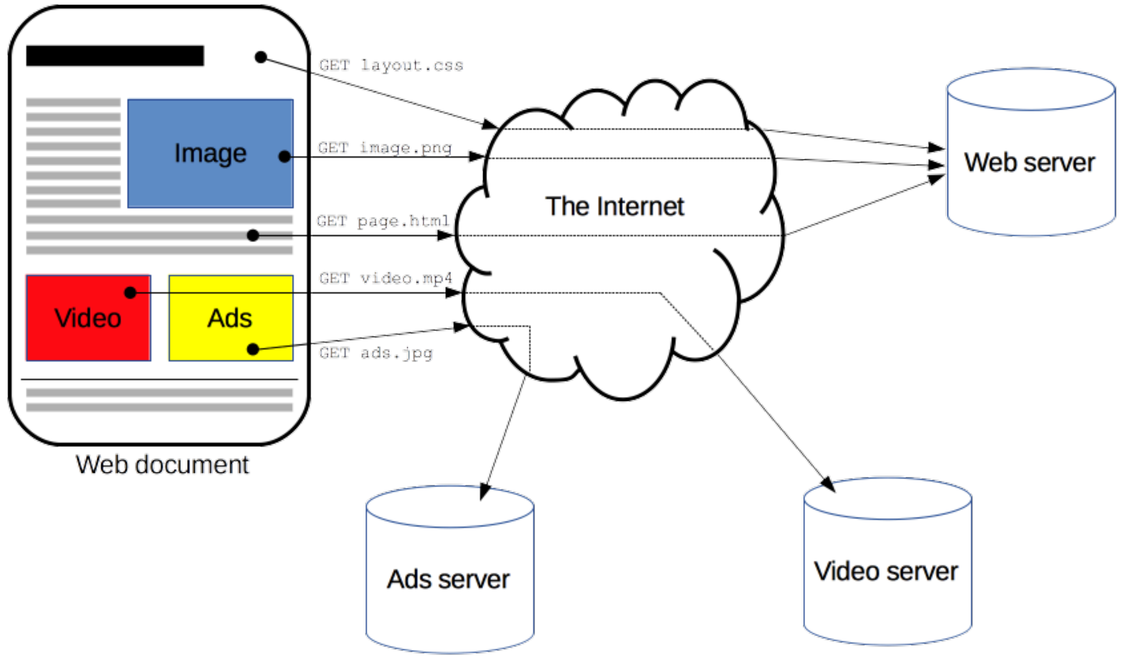
HTTP Protocol

- **HTTP** is a protocol which allows the fetching of resources, such as HTML documents.
- It is the foundation of any data exchange on the Web and it is a client-server protocol, which means requests are initiated by the recipient, usually the Web browser.
- A complete document is reconstructed from the different sub-documents fetched, for instance text, layout description, images, videos, scripts, and more.



HTTP Protocol

- **Clients** and **servers** communicate by exchanging individual messages (as opposed to a stream of data).
- The messages sent by the client, usually a Web browser, are called **requests** and the messages sent by the server as an answer are called **responses**.
- **Let us look at an example.**



Learning Resources

- https://www.lifewire.com/what-is-a-web-application-3486637
- https://www.educba.com/what-is-mvc-design-pattern/
- https://medium.com/@anshul.vyas380/mvc-pattern-3b5366e60ce4
- https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview