CSI 3150
Web and Mobile Systems

Topics of Interest

Agenda

- Topics of Interest

Type of Web Developers

Front end developer

    Client facing websites / UI apps

    HTML, CSS, JS, TypeScript, JS frameworks, DOM, HTTP,

Backend developer

    Server APIs, data, tasks

    JS, Python, C#, PHP, etc.

    Databases – relational, NoSQL, server configurations etc

Full Stack Developer

    Combine the above two

Front End

- Markup and web languages such as HTML, CSS and Javascript
- Asynchronous requests and Ajax
- Specialized web editing software
- Image editing
- Accessibility
- Cross-browser issues
- Search engine optimisation

Back End

- Programming and scripting such as Python, Ruby and/or Perl
- Server architecture
- Database administration
- Scalability
- Security
- Data transformation
- Backup

HTML AND CSS (FRONT END AND FULL STACK)

HTML 5 page structure and semantic tags

Basic CSS styling

Positioning

Alignments

Transitions and animations

Responsive design / media queries

CSS / UI FRAMEWORKS

Options available:

Tailwind CSS

BootStrap

Materialize

Bulma

Learn fundamentals of CSS first and then use these options to speed up the process of UI design

JAVASCRIPT

Basics
Variables, arrays, functions, loops

DOM and Styling
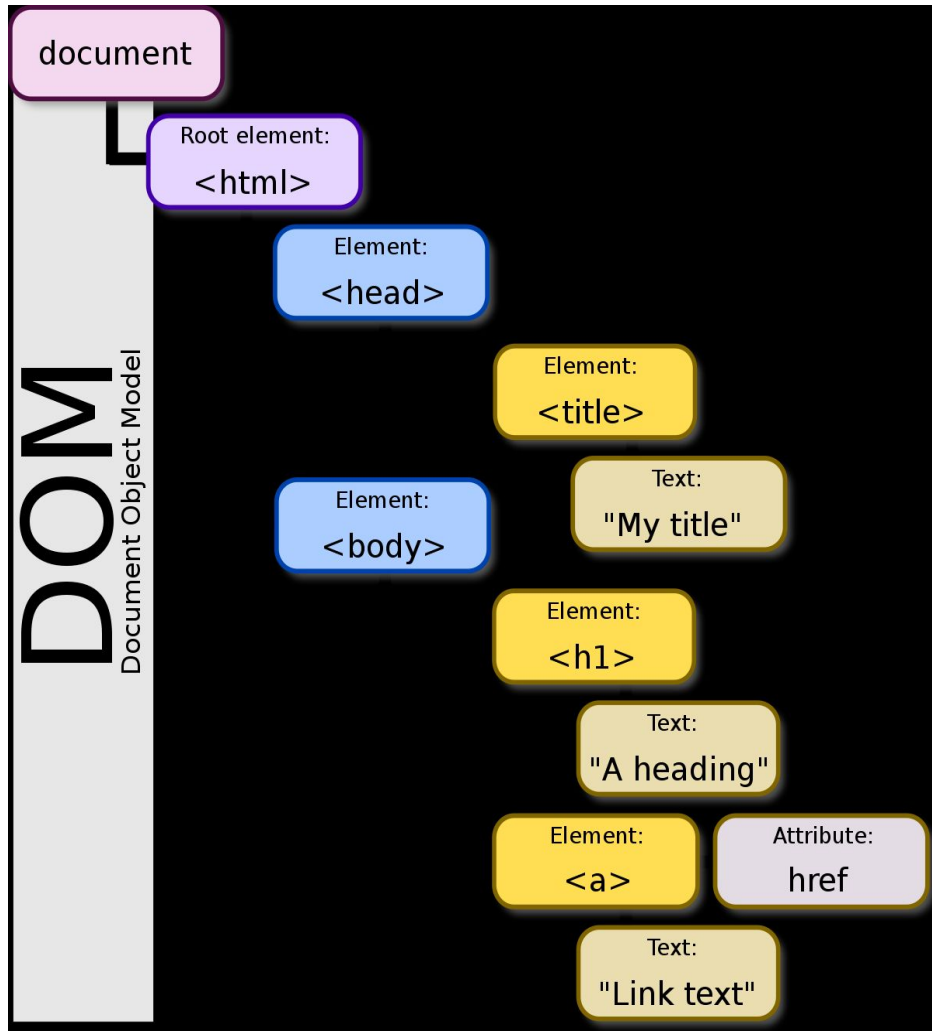Selecting and manipulating elements

Array Methods
foreach, map, filter, reduce, etc

JSON
JavaScript Object Notation

HTTP Requests
Fetch API – GET, POST, PUT, DELETE

DOM
Document Object Model

document

Root element:
<html>

Element:
<head>

Element:
<title>

Text:
"My title"

Element:
<body>

Element:
<h1>

Text:
"A heading"

Element:
<a>

Attribute:
href

Text:
"Link text"

SUPPORTING TOOLS

Version Control: Git

Repo Manager: GitHub

Package Manager

    NPM, Yarn

Module Bundler (only for Front end)

    Webpack

Editor extensions and helpers (prettier)

FRONTEND FRAMEWORKS

Become proficient in one of the following frontend frameworks

React: Currently most popular
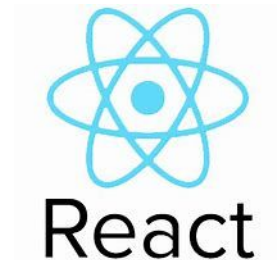
State management using: **Redux**

Vue

State management using: Vuex

Angular

State management using: Shared Service

Svelte

State management using Context API

SERVER SIDE / BACKEND

Focus on data modeling and HTTP requests/responses.

Pick a language:

  Node.js (JavaScript based)

  Deno (JavaScript based)

  Python

  C#

  GoLang

  Ruby

  PHP

  Java

SERVER SIDE FRAMEWORKS

Node: <u>Express</u>, Koa, Nest, loopback

Python: <u>Django</u>, Flask

PHP: <u>Laravel</u>, Symfony, Slim

C#: ASP.NET

Java: Spring MVC

Ruby: <u>Ruby on rails</u>, Sinatra

DATABASES

PostgresSQL

MongoDB

MySQL

MS SQL Server

Firebase

Elasticsearch

DEPLOYMENT

Hosting Platforms

Heroku, Digital Ocean, AWS, Azure, GCP

Web Servers

NGINX, Apache

Containers

Docker / Kubernetes, Vagrant

Image/ Video

Cloudinary, S3

Continuous Integration (CI) / Continuous Delivery/Deployment (CD)

Jenkins, Travis CI, Circle CI, AWS Automation / CI/CD

MOBILE DEVELOPMENT

You can jump into mobile app dev using web app dev related technologies

Flutter / Dart

React Native

Ionic

Xamarin

Kotlin

Swift

OTHER SUBJECTS OF IMPORTANCE

Algorithm Design and Analysis

Improve logical and critical thinking by many folds

Knowledge of data structures

DESIGN OF THIS COURSE 1/2

Frontend Development of Web and Mobile Systems

      Content and Structuring

      Presentation

      Dynamic and User Interactions

      Optimize

      Mobile UI Dev.

DESIGN OF THIS COURSE 2/2

HTML 5 -> Content and Structure

CSS 3 & TailwindCSS -> Presentation (& Optimize)

JS & Vanilla JS -> Dynamic and user interaction

React JS + React Router 6 + React Redux -> Optimize

React Native + API Calls + Google APIs -> Mobile UI Development

Demos to be covered

Web page structuring - various HTML elements (text, multimedia, lists, semantic tags, tables, embedding, buttons and input tags, etc.)

Web page presentation - Navbars, texts, fonts, layout flexbox, grid

Dynamic Interaction - user events + calculator app + quiz app (Kahoot)

React - AirBnB online experiences app, UI Cards, Hero Elements, Navigation using React Router

React Native - Uber Clone, Tasks App, Google Maps, React Redux

Connecting Frontend to Backend

RESTful API calls

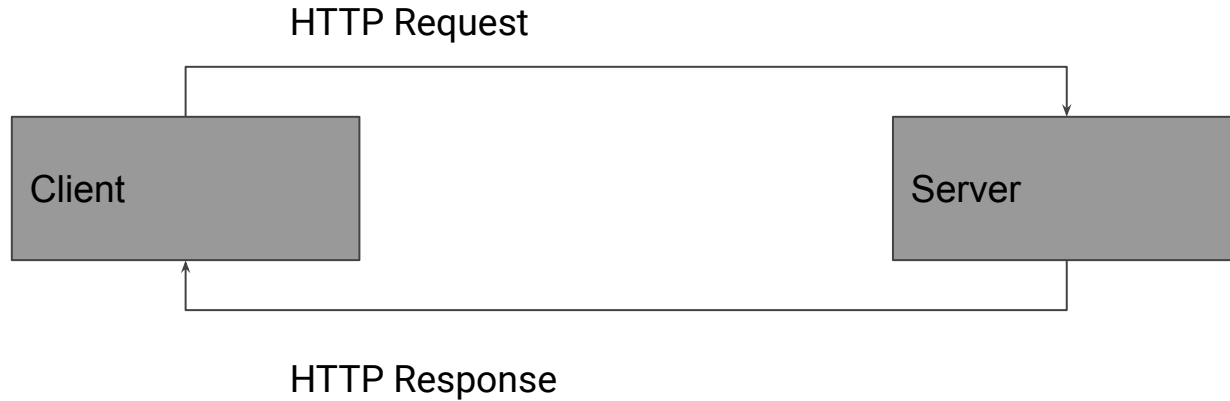A fully functional web or mobile app typically supports four basic functions:

Create == POST

Read == GET

Update == PUT or PATCH (with id)

Delete == DELETE (with id)

Connecting Frontend to Backend

HTTP Request

Client

Server

HTTP Response

Course focus on making and receiving data from API calls. API can be 3rd party like Google Maps or written by you (RESTful backend APIs)