Alex Williams
Fab Lab EP Week 1 Curriculum Draft
1/15/24

Full Stack Web Developer Curriculum (DRAFT)

NOTE: Omitting course description, table of contents, etc for brevity.

Curriculum Week 1

Introduction on the Internet and Web (~1 day)

The world wide web has evolved from its very first website to a myriad of feature rich experiences [1], [2], [3] accessible on computers, laptops, tablets, phones, smart TVs, and IoT devices. By leveraging Full Stack Web Development strategies and best practices, students will be able to develop such feature-rich content and serve it over the internet in a productive and efficient manner. However, before embarking on this journey it is critical for students to understand what exactly *is* "the web". In this section, we will discuss the motivation behind creating the web, how it differs from the internet, its capabilities, and where its going.

1. History
   - The Internet and the Web – Connecting Content, Connecting People (my attempt at a catchy title)
     - Internet vs. Web – are they the same?
     - Who made it?
     - What are its capabilities?
     - Who maintains it?
     - Where is it going?
   - Technology behind the web
     - HTTP
     - Content Negotiation
     - Web Servers
     - DNS & IP Addresses
     - Web Browsers
     - Web Standards
   - Web Vs Mobile Apps Vs Programs
     - A distinction between the various execution environments

Activities:

- In Class:
  - "Hello World" website first day
- Quiz on (Take Home):

- Internet vs. Web (Multiple Choice or similar)
- Web Vs Mobile Apps Vs Programs (Multiple Choice or similar)

Full Stack Web Development (~2 days)

Full Stack (Web) Development is a fulfilling career path with high earning potential and marketability. Furthermore, it gives web developers the freedom of choice over a wide array of programming languages and frameworks – think, "best tool for the job". However, one major downside to this approach in comparison to front-end focused or back-end focused web development can be in the lack of Subject Matter Expertise to any given area – although this can always be achieved with time put in! Becoming a professional full stack web developer requires knowledge on a variety of topics in regards to developing, testing, deploying, and monitoring web applications for the modern era. In this section, we will focus on the building blocks of full stack web development from an architectural and functional perspective.

1. Early days of Web Programming – Web 1.0
   - Simple, Static web pages
   - ISP-owned web servers
   - File-based content as opposed to DB
   - Loose security
2. Traditional Web Programming – Web 2.0
   - "Rich" content
   - Social Media
   - Responsive Design (Mobile and Tablet views)
   - Accessibility (WCAG)
   - Increased Security (HTTPS, TLS)
3. Current Web Programming
   1. Performance & Robustness a priority (ex. Edge Servers/CDNs, Serverless computing, Single-page apps, Adaptive Bitrate Streaming, WASM, QUIC)
   2. Micro-service based architecture
   3. Scalable Cloud Dbs & Data analytics
   4. Single-Sign On & Authentication schemes
   5. Internet of Things (IoT)
4. Full Stack Dev "Big 3" Development Building Blocks
   1. User interface (Front-end)

    2. API layer (Middleware/Backend)
    3. Database layer (Backend)
5. Full Stack Dev “Big 5” Production Building Blocks
    1. Testing
    2. Deploying
    3. Securing
    4. Monitoring
    5. Debugging

Activities:

- In Class:
    - Viewing examples of Web 1.0 sites & Web 2.0 sites
        - [Special Topic if time allows] a look at “Web 3.0”
    - A review of the web’s capabilities https://whatwebcando.today/
    - A review of the various Full Stack Development “road maps” https://roadmap.sh/full-stack that are used in the industry to guide professionals on latest trends.
- Quiz on (Take Home):
    - Web 1.0 vs Web 2.0 (Multiple Choice or similar)
    - Web capabilities (Multiple Choice or similar)
    - “Big 3” Development Building Blocks (Multiple Choice or similar)
    - “Big 5” Production Building Blocks (Multiple Choice or similar)

Software Tools (~2 days)

This section covers the review and installation of the software tools commonly used by Full Stack Web Developers. These tools will be used in future class sessions and take home assignments. All software and tools mentioned are free to download (for personal use, non commercial) unless noted otherwise. Optional tools are allowed if brought to the attention of the instructor and given approval beforehand (or buyer beware).

1. Integrated Development Environments (IDEs)
2. Command Line Interfaces (CLIs)

3. Package Managers
4. Version Control Systems
5. API Testers
6. [Special Topic – if time allows – Emerging Trends] AI Tools

Activities:

- In Class:
    - Above tool installation
- Quizzes (Take Home):
    - Tooling quiz (Multiple Choice or similar)