

What is the Web

- Constant use doesn't tell you everything
 - Lightswitches don't teach electricity
- Originally used to share and link sci papers
 - "HyperText"
- Makes use of Internet
 - Web not the same as Internet
 - One kind of traffic
 - Cars aren't Roads
 - Even if most of the traffic

Web Page

A **Web Page** is an **HTML Document** (.html)

- HTML is a way of "marking" text
- A web page may **link** to other pages
- May also refer to other **assets** (files)
 - Images, video, fonts, CSS files, JS files, etc

Browser

A **web browser** is a program

- Can make **requests** from a **web server**
 - To get web pages or assets as a **response**
- Can **render** a web page
 - Providing visual output
 - Uses CSS to decide what to show
 - NOT the raw HTML
 - Executes any **javascript** (JS) on the page
 - Or loaded by the page

Webserver

A program that **responds** to web **requests**

- Can **serve** (respond with) **static** assets (files)
- Can serve **dynamic assets**
 - responses created in the moment

HTTP (web) traffic is request-response

- Server cannot send except in response
- Later tech (Websockets) CAN violate this rule
 - But it all has to start from basic HTTP

Navigation

A browser can load a different page

- Navigation
- Triggered by user or program action
 - Ex: Click a link
 - Ex: Page loads different page after 10 seconds

Previous page is now gone

- Can navigate "back" to it
- But current page is new page

URLs

Every page and asset has a URL

- "address" of the asset
- URLs have many parts
 - protocol
 - port
 - domain
 - path
 - query string
 - hash fragment
- Some used by Internet, some by server

HTML

HTML (Hyper Text Markup Language)

- Provides **structure** to the text of the document
- Defines data about the text
- Does NOT directly define the appearance
 - Common mistake!
 - You CAN use HTML to make an appearance
 - But it ends up hard to use/change
 - Bad idea to try

Cascading Style Sheets (CSS)

CSS provides the rules for how web content LOOKS

- Appearance, visuals, **styling**
- Based on the *structure* of the document
 - Relies on **semantic HTML** for full benefit
- Completely different syntax from HTML
- Same CSS could apply to different HTML pages
- Same HTML page might have different CSS stylings
 - <https://csszengarden.com>

JavaScript (JS)

- **Javascript** is a programming language
 - No relationship to Java
 - Name was a marketing deal that fell through
 - Can be run on a computer with NodeJS
 - Usually runs **in the browser**, "on" a page
- Returned by a server, RUN by the browser
 - A different computer entirely
- Can see, read, and change the page HTML
- Can react to user or page actions
 - Click a link, a button, type a key, etc
- Provides **interactivity** beyond navigation

Accessibility (a11y)

- **a11y** because 11 letters between 'a' and 'y'
 - Programmers are lazy
- Ensuring everyone can *access* your content
- Often neglected in web efforts of the past
 - And still is neglected
- But Web has grown in importance
- New presentations also benefit
 - Watches, Augmented Reality, etc

a11y Benefits

- The benefits are wide
 - Not just blind, handicapped, etc
 - Though that should be enough
 - Also injuries, children, bouncy bus/train
 - Same solutions help new tech/programs
- Solutions not well known...yet
 - Demand is growing

a11y Technique

Much of web tech is accessible by default!

- We devs just need not to break it
- Need to know what breaks it

Servers

A server can provide **dynamic pages**

- Beyond the scope of this class (see 6250)

A server can manage dynamic data

- JS on the web page can send/get data from server
 - While staying on the same HTML page
 - Can change the HTML, a little or a lot
- Also beyond this class (still 6250)
 - But we can add UI to provided **services**

What are UI/UX?

User Interface (UI) are the visuals and controls

- Technically not just visuals
- We write UI
- Hopefully with intent

User Experience (UX) is what the user experiences

- Users have UX in reaction to UI
- May not be the experience we desire

Just kidding, the difference is actually ketchup

- Designers have a running joke about UI/UX and ketchup
 - I mention in case you run into it
 - <https://uxdesign.cc/the-infamous-ui-vs-85e3c4dab43f>
 - <https://uxplanet.org/ok-boomer-it-depends-that-ux-ketchup-meme-e89372053077>
 - <https://www.uxdesigninstitute.com/blog/user-research-tomato-ketchup/>

UI/UX

Inherent Contradiction

- Everyone has valid opinions
- Education involves surprise truths

We must restrain our ego

- We can be wrong
 - and still right!
- Goal is to empower USER
 - Not ourselves
- Users can surprise you

Audience, Context, and Goals

- UI/UX has definitely WRONG answers
- UI/UX has fewer definitely RIGHT answers
- Build with Audience, Context, and Goals in mind
 - Businesses goals often terrible for user
 - Welcome to late-stage capitalism

SO MUCH!

- So many terms
- 3 different syntaxes

Getting into modern web dev is complex!

- We will focus on UI/UX
 - Still complex
- Recognize the goal of each concept
 - HTML to provide structured content
 - CSS to give visuals based on structure
 - JS to provide interactivity