

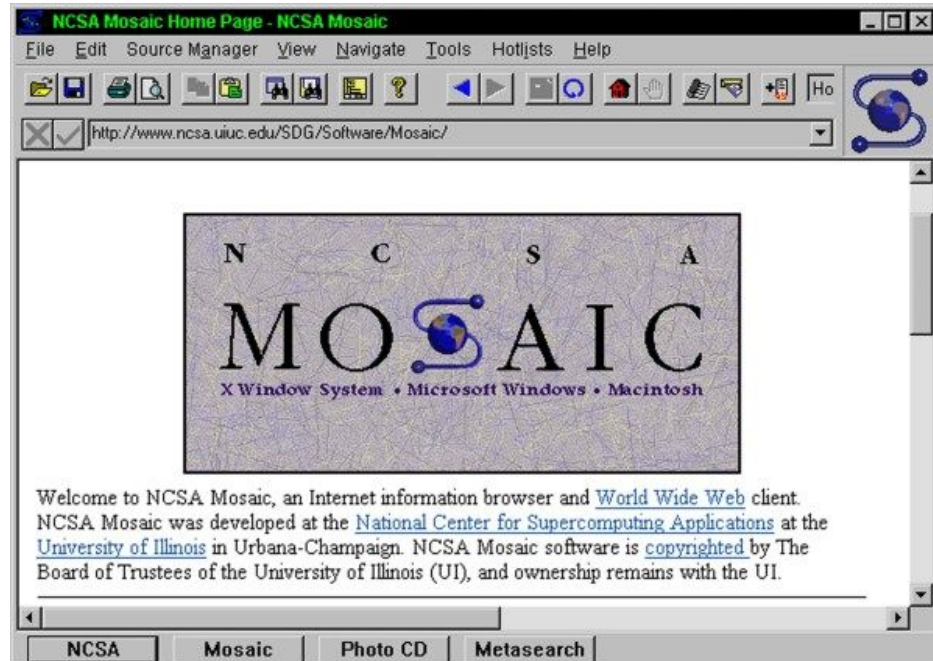
Desenvolupament web en l'entorn client

2 DAW



La web

- Un dels conjunts de protocols més exitosos d'Internet.
- Inventat per Tim Berners-Lee al CERN.
- Hipertext
- Consorci W3C
- La web = Hipertext + Protocol HTTP + Servidors i Clients + DNS



El desenvolupament web

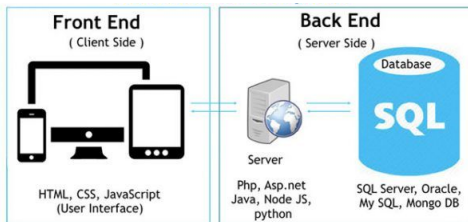
- Contingut (Organització, semàntica, presentació estructura, HTML...)
- Disseny (Aspecte visual, elements gràfics, plantilles, CSS...)
- Tecnologia Frontend (Elements interactius, Javascript...)
- Tecnologia Backend (Accés a dades, Controlador, PHP, Python, Node...)
- Distribució (Maquinari i programari servidor)
- Propòsit





Arquitectura Client - Servidor

- Server-side:
 - Maquinari: Servidors i elements de xarxa, màquines virtuals, contenidors...
 - Programari:
 - Servidors web, Apache, IIS, NGINX...
 - Llenguatges CGI: Perl, PHP, C...
 - Llenguatges i frameworks amb servidor web: Python, Java, Node.js, C++...
- Client-side:
 - Clients web: Firefox, Chrome, Vivaldi, Opera, Edge, Internet Explorer...
 - Llenguatges de marques: HTML, XHTML, HTML5, CSS...
 - Llenguatges de programació en entorn client: Javascript





Server-side vs Client-side

- Moltes coses es poden fer en els dos costats.
 - Un web **interactiva** pot ser programada totalment en el costat del servidor.
 - També pot ser programada en la major part en el costat del client.
- Cal triar la tecnologia correcta en cada cas (o la que estem més còmodes)

Validar formularis

Guardar dades
permanents

Presentació de
diapositives

Eliminar un
paràgraf

Calcular dades

Accedir a bases
de dades



Javascript en el costat del client

- Javascript no pot directament escriure fitxers al servidor.
- No tots els navegadors suporten javascript. Una pàgina web ben feta deu funcionar sense javascript.
- Javascript pot demanar coses al servidor de forma asíncrona. La tecnologia asíncrona més famosa és AJAX (Javascript asíncron amb XML)
- Utilitzarem Javascript per a:
 - Aconseguir que la web reaccione a la interacció amb l'usuari.
 - Proporcionar una interfície agradable per a les dades.
 - Controlar finestres, marcs i applets en el document HTML.
 - Pre-processar dades abans d'enviar al servidor.
 - Modificar estil i contingut de forma dinàmica en funció de la interacció amb l'usuari.
 - Sol·licitar o enviar fitxers al servidor.

Compatibilitat

Navegadores actuales y su soporte para distintas tecnologías										
Browser	JavaScript	ECMAScript 3	DOM 1	DOM 2	DOM 3	XPath	DHTML	XMLHttpRequest	Rich editing	
Amaya	No ^[note 1]	No ^[note 1]	No ^[note 1]	No	No	No	No	No	No	No
AOL Explorer	Yes	Yes	Partial	Yes	No	No	Yes	Yes	Yes	Yes
Avant	Yes	Yes	Partial	No	No	No	Yes	Yes	Yes	Yes
Camino	Yes	Yes	Yes	Yes	No ^[note 2]	Yes	Yes	Yes	Yes	Yes
Dillo	No	No	No	No	No	No	No	No	No	No
DocZilla	Yes	No	Yes	Yes	Partial	Yes	Yes	Yes	Yes	No
ELinks	Partial	Partial	No	No	No	No	No	No	No	No
Web	Yes	Yes	Yes	Yes	No ^[note 2]	Yes	Yes	Yes	Yes	Yes
Flock	Yes	Yes	Yes	Yes	No ^[note 2]	Yes	Yes	Yes	Yes	Yes
Galeon	Yes	Yes	Yes	Yes	No ^[note 2]	Yes	Yes	Yes	Yes	Yes
Google Chrome	Yes	Yes	Yes	Yes	Partial	Yes	Yes	Yes	Yes	Yes
iCab	Yes	Yes	Partial	Partial	No	No	Yes	Yes	Yes	No
Internet Explorer	Yes	Yes	Yes	Yes ^[note 3]	Partial	Yes	Yes	Yes	Yes	Yes
Internet Explorer for Mac	Yes	Yes	Partial	No	No	No	Yes	No	No	No
K-Meleon	Yes	Yes	Yes	Yes	No ^[note 2]	Yes	Yes	Yes	Yes	Yes
Konqueror	Yes	Yes	Yes	Yes	Partial	No	Yes	Yes	Yes	No
Links	No ^[note 4]	No	No	No	No	No	No	No	No	No
Lynx	No	No	No	No	No	No	No	No	No	No
Maxthon	Yes	Yes	Partial	No	No	Yes	Yes	Yes	Yes	Yes
Midori	Yes	Yes	Yes	Yes	Partial ^[note 5]	Yes ^[note 5]	Yes	Yes	Yes	Yes
Mosaic	No	No	No	No	No	No	No	No	No	No
Mozilla	Yes	Yes	Yes	Yes	No ^[note 2]	Yes	Yes	Yes	Yes	Yes
Mozilla Firefox	Yes	Yes	Yes	Yes	Partial	Yes	Yes	Yes	Yes	Yes



Validació HTML

- Per evitar problemes de compatibilitat, és important que el codi HTML siga vàlid.

<https://validator.w3.org/>



Seguretat

- Javascript s'executa en un entorn segur sense contacte amb el sistema operatiu.
- Javascript té una "política del mateix origen" on sols té accés a dades rebudes del mateix servidor.
- Javascript no pot: Modificar preferències del navegador, llançar aplicacions, llegir o escriure fitxers en el client o escriure en el servidor, retransmetre streaming, enviar emails, interactuar amb llenguatges de servidor, accedir a una web de distint domini, protegir l'origen de les imatges, implementar multiprocés,



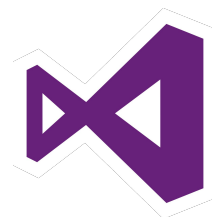
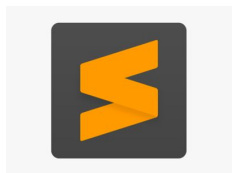
Entorns de programació web per al client





Programari

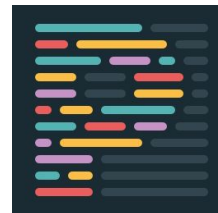
- Per a modificacions 'casuals' (carreguen molt ràpid): Gedit, Nano, Notepad...
- Editors de text avançats: Sublime, Atom, Brackets, Jedit, Visual Studio Code, Vim...
- IDEs: Netbeans, Eclipse, Webstorm...





Preparar l'entorn

- En general en classe anem a utilitzar Visual Studio amb algunes extensions:
 - Prettier: Per a formatar els fitxers. Es pot fer en el botó dret o configurar-ho per a fer-ho al guardar.
 - [ESLint](#): Ajuda a detectar errades d'estil o de programació que Javascript no pot detectar. (cal instal·lar eslint en el projecte)
 - Live Server: Per provar la web en un servidor local.





Altres ferramentes

- Tecla F12 en els navegadors
- Firefox developer edition



Firefox Browser
DEVELOPER



Tipus d'aplicacions web

- Amb Javascript 'Vanilla'
- Amb biblioteques: JQuery, Prototype, D3.js, Bootstrap, Vue.js, React
- Amb Frameworks: Angular, React, Vue.js, Phaser
- Amb Frameworks full Stack: Odoo, Blitz, Next.js, SvelteKit

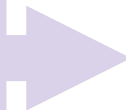


Evolució de les aplicacions web

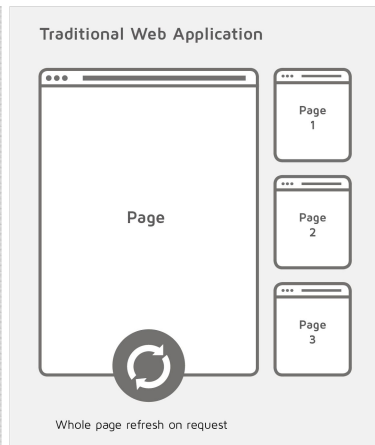
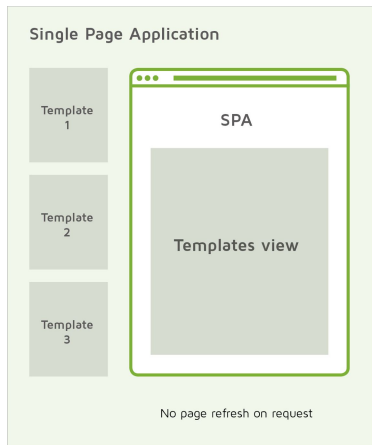
Pàgines web
estàtiques amb
hipervincles.



Pàgines web
generades pel
servidor. Un poc de
Javascript.



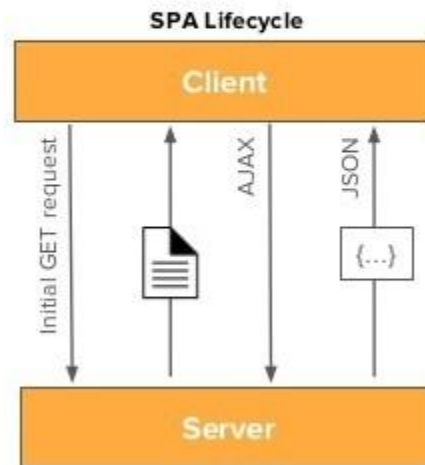
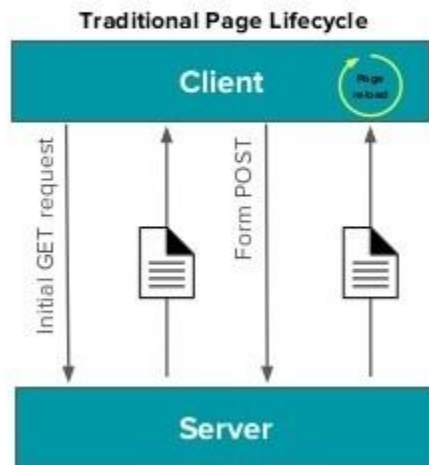
Introducció de Ajax
per carregar o pujar
coses al servidor
asíncronament.



Single Page Application.
Comunicació amb el server
amb JSON o XML

Single page application

Single Page Applications work differently





Què és millor?

- SPA moltes vegades es sobreenginyeria i complica innecessàriament el desenvolupament web.
- Cal preguntar-se si és una aplicació web en la que el client va a estar tot el dia treballant o esporàdicament.
- Crear l'html i l'estil en el servidor sol ser menys complicat, però en aplicacions molt grans és carregar el servidor i no és fàcil de separar la vista del model i controlador.
- Si el client va a renderitzar part de l'HTML, és millor que ho faci tot.
- Nosaltres anem a fer una SPA perquè és més adequat per al mòdul.

Entre la pàgina web i l'APP

- Pàgines web.
- Aplicacions web SPA o no.
- Aplicacions web responsive.
- PWA Progressive Web App.
- App Híbrides. (Ionic)
- App Quasi Natives. (React Native o Flutter)





Les Biblioteques

- Aporten una o més funcionalitats al llenguatge de programació.
- No forcen a utilitzar-les sempre ni a canviar la manera de programar.
- Simplifiquen la programació de coses concretes.
- Poden mal acostumar al programador
- Algunes com Underscore, [Lodash](#) o inclús JQuery ja no són tan útils amb ES6



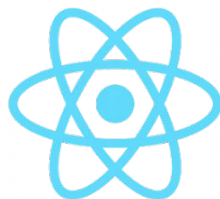
UNDERSSCORE.JS

Lo



Els frameworks

- Modifiquen totalment la manera de programar.
- S'encarreguen de tot (plantilles, comunicacions, enrutaments, components, mòduls...)
- Tenen un gran corba d'aprenentatge.
- Una vegada dominats, fan la programació molt més àgil, correcta i no redundant.
- Provoquen una gran inèrcia a l'hora de canviar de framework.
- Pot excloure altres biblioteques o tècniques.
- Si queden obsolets, la web queda obsoleta.



React



Vue.js



Utilitat dels frameworks

- Si es fan especialment per a una aplicació.
- Si són estàndard i pots exportar els components.
- Si aporten un valor afegit únic.
- Si es vol fer una aplicació no-crítica.



Obtindre biblioteques i frameworks

- Descarregar un .js o un directori comprimit de la web oficial i copiar-lo al directori de la web.
- Utilitzar un CDN dirèctament de la web oficial.
- Utilitzar un gestor de paquets tipus NPM o Yarn





Posar la web en producció

- La manera tradicional és copiar tot el contingut de les carpetes al voltant del index.html al servidor.
- Actualment es pot programar amb un 'Javascript Build Tool', açò fa una espècie de compilat del codi font i:
 - Minimitza el codi javascript i CSS resultant.
 - Unifica el Javascript i CSS en un sol arxiu.
 - Optimitza els 'assets' per millorar la velocitat de càrrega.
 - Preprocessa CSS a partir de SASS o LESS.
 - Evita duplicitats.
 - Tree Shaking: Analitzar el codi i llevar coses innecessàries.
 - Transpilar de Typescript a JS.
 - Comprimeix, de forma que el client pot descomprimir.
 - Evita incompatibilitats entre navegadors.
 - Documentació



webpack



Planificació del curs

- Javascript bàsic.
 - Llenguatge javascript.
 - Manipulació del DOM.
 - Classes i mòduls javascript.
- TDD (Tests unitaris)
- Comunicació amb el servidor
 - Càrrega asíncrona.
- npm webpack
 - Instal·lació de biblioteques
 - Preparació per a producció.
- Programació funcional, asíncrona i reactiva.
- Angular
 - Components, Pipes, Servicis, Gestió de rutes, Directives, Mòduls, Formularis reactius, Angular Material i Bootstrap, Mòduls de tercers.