

## 1) Descrivere brevemente le principali novità introdotte da CSS3.

### Gestione Avanzata degli Sfondi:

- background-size per specificare la dimensione dell'immagine di sfondo.
- background-origin per posizionare l'immagine di sfondo in differenti box (content-box, padding-box, border-box)
- Immagini di sfondo sovrapposte.

### Gradienti Lineari e Radiali:

- Linear-gradient: gradiente lineare come sfondo, con direzione e colori personalizzabili.

### Regole Annidate e @rules:

- @font-face per definire font personalizzati.
- @support per verificare il supporto di determinate regole CSS da parte del browser.
- Transizioni: transition-property, transition-duration, transition-timing-function, e transition-delay.

### Bordi e Box:

- border-image.
- border-radius.
- box-shadow.

**Filtri per Immagini:** effetti visivi alle immagini attraverso la proprietà filter: blur, brightness, contrast, drop-shadow, grayscale, e altri, per modificare l'aspetto delle immagini in vari modi.

## 2) Descrivere in che cosa consiste l'approccio Mobile First e quali sono i vantaggi della sua adozione.

Iniziare lo sviluppo del design con un focus sui dispositivi mobili, più limitati in termini di schermo e capacità, per poi adattare e arricchire questo design per dispositivi più grandi come tablet e desktop. Questo approccio è considerato una forma di "**progressive enhancement**", dove si parte da una base funzionale per il caso d'uso più restrittivo (mobile) e si costruisce su di essa per casi d'uso meno restrittivi (desktop). Contrasta con il "**graceful degradation**", un approccio precedente che partiva da design complessi per desktop e poi li semplificava per dispositivi mobili.

### Vantaggi:

- Una maggiore attenzione all'essenziale, poiché la progettazione per schermi più piccoli costringe a focalizzarsi su ciò che è veramente importante.
- Migliore usabilità su dispositivi mobili, che sono sempre più il principale mezzo di accesso a internet.
- Un percorso di sviluppo più fluido, passando da una base semplice e funzionale a una più complessa e arricchita.

## 3) Descrivere le principali differenze tra gli approcci progressive enhancement e graceful degradation.

**Progressive Enhancement:** Base funzionale e semplice, adatta a dispositivi con limitazioni (mobile), e arricchisce il design per dispositivi più potenti (desktop). Rende il sito accessibile e utilizzabile in ogni condizione, migliorando l'esperienza utente con maggiori capacità e risorse.

**Graceful Degradation:** Si inizia con un design completo e ricco (desktop), e si semplifica per adattarlo (mobile). L'obiettivo è mantenere la funzionalità core su dispositivi meno potenti, pur avendo un'esperienza ottimale su quelli più avanzati.

4) Descrivere tutte le tecniche/modalità/crisimi, dal punto di vista della sicurezza, in modo che la trasmissione di dati sensibili dal browser al server sia "sicura".

**HTTPS (Hypertext Transfer Protocol Secure):** È la versione sicura di HTTP. Utilizza il protocollo SSL/TLS per criptare la comunicazione tra il browser e il server, proteggendo i dati sensibili durante il loro trasferimento.

**Validazione dei Dati Lato Client e Server:** È importante validare i dati sia sul lato client (JavaScript) sia lato server (PHP, Python) per prevenire attacchi come SQL injection.

**Autenticazione e Autorizzazione:** Implementare meccanismi robusti di autenticazione (es. password, autenticazione a due fattori) e autorizzazione per controllare l'accesso ai dati sensibili.

**Cookie Sicuri:** Utilizzare cookie sicuri e HTTP only, che non possono essere accessibili tramite script lato client, per minimizzare il rischio di attacchi cross-site.

**Controllo di Accesso e Gestione Sessioni:** Implementare un controllo di accesso solido e gestire le sessioni in modo sicuro per prevenire attacchi.

**Crittografia:** Criptare dati sensibili sia in transito (con HTTPS) che a riposo (sul server) per proteggerli contro accessi non autorizzati.

5) Descrivere le metodologie di codifica dei caratteri nelle pagine Web.

**UTF-8:** È una codifica a lunghezza da 1 a 4 byte per rappresentare un carattere.

I codici ASCII (0 - 127) richiedono solo un byte. Molto usato perché supporta una vasta gamma di caratteri da diversi sistemi di scrittura mantenendo la compatibilità con il set ASCII.

**UTF-16:** 2 o 4 byte per carattere, lavora in in 16 bit. È un'altra forma di Unicode e può rappresentare più caratteri rispetto all'ASCII.

**Latin 1 (ISO 8859-1):** È una codifica a singolo byte che copre la maggior parte dei caratteri delle lingue europee occidentali, inclusi alcuni caratteri speciali.

**Unicode e ISO/IEC 10646:** 4 byte per codificare singolo carattere, risolvendo i problemi di codifica delle lingue non europee, ma consumano più memoria. Non ancora "standardizzato" perché ancora alcune cose sono in ASCII.

6) Definire personas e target users e spiegarne le differenze.

**Personas:** descrizioni dettagliate di utenti rappresentativi, gruppi di utenti. Create per ogni progetto (da 3 a 7) e guidano il design in base alle esigenze e alle aspettative di questi utenti tipici. Descrivono realisticamente le azioni che compiono utilizzando un servizio, obiettivi, motivazioni e il contesto.

**Target Users:** gruppo di utenti a cui si rivolge principalmente il progetto. Riguarda il gruppo di utenti in generale. La progettazione UX user-centered si concentra su di loro, mettendo l'utente e il contesto in cui opera al centro del design e dell'applicazione.

## 7) Illustrare le diff., vantaggi e svantaggi di fogli di stile esterni, interni e inline.

### Fogli di Stile Esterni: (migliori per siti grandi, ordinati)

- Uso: Importati nell'header del documento HTML tramite il tag <link>.
- Vantaggi: Promuovono la riutilizzabilità e la coerenza tra diverse pagine. 1 file, più pagine.
- Svantaggi: Richiedono un ulteriore download, influisce sul tempo di caricamento della pagina.

### Fogli di Stile Interni:

- Uso: Posizionati all'interno del tag <style> nell'header del documento HTML.
- Vantaggi: Utile per stili specifici di una singola pagina. Non richiedono download aggiuntivi.
- Svantaggi: Non promuovono la riutilizzabilità del codice. Aumentano la dimensione del file.

### Fogli di Stile Inline:

- Uso: Direttamente nel tag di riferimento tramite l'attributo style.
- Vantaggi: Comodi per applicare stili rapidi e specifici a un singolo elemento.
- Svantaggi: Non promuovono la riutilizzabilità e possono rendere il codice HTML disordinato. Difficoltà nella manutenzione e nell'aggiornamento dello stile.

## 8) Descrivere come scambiare i valori delle variabili tra Javascript e Php.

Per scambiare i valori delle variabili tra JavaScript e PHP, si utilizza la tecnica **AJAX** (Asynchronous JavaScript and XML). Browser dialoga in modo asincrono con il server.

Definizione di un **Oggetto XMLHttpRequest** che invia richieste HTTP al server.

Si effettua una richiesta **GET o POST** al server da JS.

Vengono inviati i dati come testo semplice o JSON.

JavaScript invia dati a un file PHP sul server, e il server PHP può elaborarli e rispondere.

Funzione in JavaScript gestisce la **risposta** del server.

JavaScript aggiorna dinamicamente il contenuto della pagina senza ricaricarla.

## 9) Descrivere brevemente i passaggi e gli scambi di comunicazione che avvengono tra browser e server Web dal momento in cui viene digitato o cliccato un indirizzo Web al momento in cui la corrispondente pagina Web viene renderizzata dal browser.

Attraverso il protocollo HTTP e segue questi passaggi:

**Attivazione della Connessione:** Il client (browser) attiva la connessione al server web digitando o cliccando un indirizzo web. Il browser invia una richiesta HTTP al server per richiedere la risorsa corrispondente all'URL.

**Risposta del Server:** Il server web riceve la richiesta e la elabora. In base alla richiesta, il server identifica la risorsa richiesta. Il server invia una risposta HTTP al browser.

Questa risposta può includere il contenuto richiesto (come HTML, CSS, JavaScript), un messaggio di errore, un reindirizzamento a un'altra risorsa, ecc.

**Rendering della Pagina:** Ricevuti i dati, il browser inizia a renderizzare la pagina web.

Interpretazione del codice HTML, l'applicazione stili CSS, esecuzione script JavaScript.

**Chiusura della Connessione:** Dopo che la risposta è stata inviata, il server chiude la connessione, tranne in una connessione persistente per ulteriori scambi di dati.

Il protocollo HTTP è stateless, le richieste sono indipendenti e il server non mantiene informazioni tra una connessione e la successiva sulla natura, identità, e precedenti richieste di un client. Questo processo è fondamentale per il funzionamento del web e garantisce che le risorse web possano essere richieste e fornite in modo efficiente.

10) Descrivere brevemente il concetto di “cascata” nei fogli di stile e definire come si applica e come vengono gestiti eventuali conflitti con fogli di stile inline, fogli di stile interni e fogli di stile esterni.

A "cascata" è l'ordine con cui vengono applicate le regole di stile. Quando ci sono conflitti tra regole CSS diverse, la cascata determina quale stile applicare.

**Media:** Si considera il tipo di media per cui è stato definito lo stile (ad esempio, schermi, stampe).

**Importanza di una Dichiarazione:** Il flag !important hanno priorità rispetto a quelle senza.

**Specificità del Selettore:** La specificità è un sistema di punteggio basato sulla natura del selettore utilizzato per applicare lo stile. I selettori più specifici (ad esempio, quelli con ID) hanno la precedenza su quelli meno specifici (come i selettori di tag o classe).

Stile inline ha una specificità più alta rispetto a uno stile definito in un foglio di stile esterno o interno.

11) Descrivere brevemente che cosa si intende per Personas, Focus Group e Scenarios nel contesto della User experience.

**Scenarios:** Descrivono realisticamente le azioni compiute da una persona (persona) mentre utilizza un servizio. Includono obiettivi, motivazioni e il contesto in cui le personas operano.

Aiutano a visualizzare come gli utenti interagiscono con il prodotto in situazioni reali, facilitando una progettazione più centrata sull'utente.

Questi strumenti sono centrali nel design user-centered, aiutando i progettisti a comprendere meglio gli utenti e le loro esigenze, per creare esperienze più efficaci e soddisfacenti.

12) Descrivere brevemente le principali differenze e analogie tra i solution stack LAMP e MEAN.

**Sistema Operativo:** LAMP (Linux, Apache, MySQL, PHP/Perl/Python) è specifico per un sistema operativo (Linux), mentre MEAN (MongoDB, Express.js, Angular, Node.js) è multiplatforma.

**Ambiente di Esecuzione:** MEAN usa Node.js come ambiente di esecuzione per applicazioni server-side, eseguendo codice JavaScript al di fuori del browser. LAMP, invece, utilizza PHP, Python o Perl per la programmazione lato server.

**Database:** MEAN utilizza MongoDB, un database non relazionale, al contrario di LAMP che utilizza MySQL, un database relazionale.

**Stack Full-Stack:** Entrambi sono stack full-stack per lo sviluppo web, che significa che offrono soluzioni sia per il front-end (client-side) che per il back-end (server-side).

**Popolarità e Utilizzo:** Sia LAMP che MEAN sono ampiamente utilizzati nella comunità di sviluppatori per costruire applicazioni web.

13) Descrivere le principali differenze tra le WCAG 2.0 e i requisiti definiti nell'ambito della Legge Stanca (L. 4/2004):.

**WCAG 2.0:** Linee guida internazionali sviluppate dal W3C, contenuto accessibile a disabilità.

Le WCAG 2.0 sono strutturate in principi, linee guida e criteri di successo su tre livelli di conformità (A, AA, AAA), con l'obiettivo di rendere il web percepibile, operabile, comprensibile e robusto per tutti.

**Legge Stanca (L. 4/2004):** È una legislazione italiana che specifica i requisiti di accessibilità per i siti web pubblici. Si articola in 12 requisiti principali, ciascuno dei quali verificato attraverso un certo numero di punti di controllo.

I requisiti e i punti di controllo della Legge Stanca mostrano una certa corrispondenza con le WCAG, suggerendo che la legge si basa su principi simili di accessibilità web, ma adattati al contesto normativo e tecnologico italiano.

In generale, mentre le WCAG 2.0 forniscono un quadro di riferimento globale per l'accessibilità web, la Legge Stanca si concentra su requisiti specifici e adattati per il contesto italiano, seguendo però principi simili di accessibilità e inclusione.

## 15) Descrivere brevemente le metodologie di codifica dei colori nelle pagine Web e nei fogli di stile CSS.

**RGB (Red, Green, Blue):** Sintesi additiva di luce, rappresentati da una terna di numeri tra 0 e 255. Può essere espresso in esadecimale (#RRGGBB), ogni coppia di valori è un colore primario.

**HSL (Hue, Saturation, Lightness):** Rappresenta uno spazio colorimetrico basato su tre valori: hue (tonalità), saturation (saturazione) e lightness (luminosità).

h: grado di angolazione del cerchio cromatico (valori da 0 a 360).

s: saturazione del colore (in percentuale).

l: indica la luminosità (anche in percentuale).

**HSLA:** estensione di HSL che include un canale alpha per la trasparenza.

## 16) Descrivere brevemente le principali novità introdotte da HTML5.

**Elemento <canvas>:** disegnare direttamente sulla pagina.

Fornisce API per la generazione e il rendering dinamico di grafica, diagrammi, immagini e animazioni, definendo un'area rettangolare per il disegno diretto di immagini bidimensionali.

**Elemento <video>:** prima si usavano plugin come Flash.

Stabilisce standard per includere video in una pagina web, con attributi per controlli, larghezza e altezza del video, e un meccanismo di fallback per browser che non supportano l'elemento.

**Estensioni all'Elemento <input>:** nuovi tipi di controlli per l'elemento, che includono hidden, search, tel, url, email, password, date, time, number, range, color, checkbox, radio, file, submit, image, reset. Migliorano l'esperienza utente e la validazione dei dati nelle form.

## 17) Descrivere le principali differenze tra usability e user experience.

**Usabilità:** Rendere un prodotto o un servizio funzionale e semplice da usare.

Valuta efficienza, efficacia e soddisfazione in esecuzione di compiti specifici, non deve essere noiosa.

**User Experience (UX):** Reazione complessiva dell'utente di fronte all'interazione col servizio.

Include funzionalità e usabilità del sistema, la dimensione estetica/edonistica (piacevolezza estetica, emotiva e ludica del sistema) e la dimensione simbolica (attributi sociali, forza del brand).

**In sintesi:** mentre l'usabilità si concentra sugli aspetti pragmatici dell'interazione con un prodotto (come l'efficienza e la facilità d'uso), la user experience comprende un'ampia gamma di fattori, inclusi quelli emotivi e estetici, che influenzano la percezione complessiva dell'utente.

## 18) Descrivere cosa si intende in ambito Web con usabilità e cosa con accessibilità, sottolineando le differenze.

**Usabilità:** Rendere un prodotto o un servizio funzionale e semplice da usare.

Valuta efficienza, efficacia e soddisfazione in esecuzione di compiti specifici, non deve essere noiosa.

**Accessibilità:** Conoscenze tecnologiche, di erogare servizi e fornire informazioni in modo accessibile a tutti gli utenti, indipendentemente dalle loro abilità fisiche o sensoriali.

L'accessibilità si focalizza sulla rimozione delle barriere che potrebbero impedire l'interazione o l'accesso ai contenuti web da parte di utenti con disabilità.

**In sintesi:** mentre l'usabilità riguarda l'esperienza generale di tutti gli utenti e la facilità con cui possono utilizzare un prodotto o servizio, l'accessibilità si concentra specificamente su rendere questi stessi prodotti o servizi accessibili a persone con varie forme di disabilità.