

Cosa vuol dire fare il front end developer

Lo sviluppatore front-end è una figura professionale che fino a qualche anno fa diciamo 7, 8 forse anche 10 anni non era così specifica perché lo sviluppo che si faceva di applicazioni web era del tipo cosiddetto a monolite.

Un tempo le applicazioni erano sviluppate e distribuite come singole unità: applicazioni monolitiche, con una sola base di codice, facili da implementare in piccole dimensioni, ma inadatte a crescere ed evolversi. I componenti essenziali di un'applicazione con architettura monolitica (interfaccia utente, sistema di autorizzazione, database, logica di business) hanno la caratteristica di essere combinati e integrati in un solo programma.

Quindi c'era chiaramente una parte di interfaccia però poi lo sviluppo delle applicazioni in sé avveniva tramite linguaggi e tecnologie server side che venivano effettivamente eseguite su server.

Confrontandomi anche con altre persone oggi questa cosa si è complicata non dico inutilmente, perché ha indubbiamente i suoi vantaggi, però questo comporta una specializzazione in un determinato ambito e che in qualche modo rende più complesso lo sviluppo di un'applicazione web: oggi parliamo quindi della parte esclusivamente di front-end.

Cosa vuol dire fare la parte front di un'applicazione web?

È quello con cui interagisce l'utente finale e che quindi comprende tutte quelle attività che vengono eseguite direttamente sul browser.

Generalmente questa è quella parte della programmazione che chiunque si approcci per la prima volta a questo ambito vede possibile immediatamente, perché è la prima cosa che si fa quando dici “mi piace la programmazione, vorrei diventare uno sviluppatore”.

La prima cosa che si inizia ad imparare, anche per via della documentazione che è presente, è l'html e il css senza sapere chiaramente tutto quello che c'è dopo che è sicuramente più complesso. Normalmente quando le persone arrivano a scontrarsi con javascript si rendono conto che non era così facile come sembrava.

Oggi la programmazione front-end è solo front-end ed è comunque qualcosa di estremamente vasto che richiede anche tanto studio, tante competenze che sembrano andare al di là di quella che è il campo di applicazione in sé perché chi fa front-end deve sviluppare interfacce e all'interno di questa interfaccia ci sono diversi elementi che rendono una persona più specializzata rispetto ad altri ambiti.

Per esempio chi lavora sul front-end e sviluppa interfacce deve essere in grado di riprodurre esattamente un layout che viene normalmente fornito da un grafico e quindi è richiesta una certa attenzione ai dettagli.

Non vuol dire che deve disegnare le interfacce (questo può essere qualcosa che avviene in determinati ambiti) ma normalmente l'attività del front-end è a stretto contatto con quella di un grafico o di un team di grafici e chiaramente deve essere in grado di riprodurre fedelmente al pixel quello che gli è stato fornito come grafica. Tutto ciò per determinate persone è una cosa che le manda fuori di testa.

L'attività del front-end developers a volte è quella di relazionarsi con utenti finali che non si preoccupano tanto di tutta la logica di programmazione che ci sta dietro ad esempio un bottone e che cosa succede quando premi quel bottone ma il fatto che quel bottone sia di un colore diverso rispetto a quello che loro vorrebbero.

Ci saranno quindi situazioni in cui il developer dopo aver lavorato per giorni su un'implementazione nel momento in cui la fate vedere al vostro utente finale questo dirà "ok bello però il bottone mi piacerebbe averlo 3 pixel più in là".

Oltre alla riproduzione fedele di interfacce potreste avere anche l'incombenza di misurarvi con le richieste, a volte anche paradossali e senza senso, di un utente finale che si concentra molto di più su quello che vede e non capisce la logica che ci sta dietro. Bisogna quindi avere anche un buon un buon livello di comunicazione per capire bene quelle che sono le necessità dell'utente e quindi riuscire a spiegare determinate cose senza "insultarli malamente" anche se a volte viene proprio voglia di fare.

Ci sono poi tutta una serie di attività che possono avvenire direttamente sul client quindi direttamente sul browser prima ancora che le informazioni dialoghino con un back-end: possono essere fatti calcoli o può essere realizzata la visualizzazione di una parte dell'applicazione o eseguiti degli algoritmi direttamente sul browser in modo tale che determinate informazioni facciano vedere un risultato in funzione di quello che ha imputato un utente senza che questo necessiti di una elaborazione back-end.

Ci sono applicazioni dove ad esempio dovrete mostrare delle chat, potreste dover implementare dei grafici, magari implementare delle timeline molto particolari o dove potreste fare un sacco di attività che non sono la classica forma con cui l'utente imputa dei dati e voi li salvate.

Cosa deve conoscere un front-end developer per fare questo mestiere oggi?

Diciamo subito che le basi di **html, css e javascript** sono assolutamente necessarie, dopodiché sulla base di queste tre nozioni principali si possono costruire questa sulle fondamenta tutte le opzioni aggiuntive che oggi vengono normalmente utilizzate nell'ambito del front-end.

Stiamo parlando degli utilizzi di framework e librerie estremamente conosciute utilizzate negli ambienti lavorativi come Angular (<https://angular.io/>), React (<https://it.reactjs.org/>), Vue (<https://vuejs.org/>) e ne potrei citare tanti, ad oggi queste sono quelle più utilizzate.

Poi tutto il mondo che riguarda i tool che un front-end ad oggi deve conoscere per poter lavorare in questo in questo ambito e parliamo sia di tool di sviluppo quindi ad esempio **Visual Studio Code** banalmente ma anche e soprattutto di **GIT** per quanto riguarda la gestione dei repository, di tool per la grafica quindi conoscere **Adobe XD** oppure **FIGMA** che sono i tool ad oggi più utilizzati per quanto riguarda la prototipazione di interfacce.

Parliamo anche di saper configurare un ambiente di sviluppo perché per poter lavorare con librerie o framework bisogna configurare un ambiente che generalmente sulle nostre macchine e che ci permette di installare dipendenze ovvero librerie stesse o librerie aggiuntive con la quale strutturiamo la nostra applicazione. Applicazione che poi non viene

rilasciata esattamente così come la scriviamo perché così come la scriviamo non può essere interpretata direttamente dal browser che è in grado di interpretare comunque sempre solo html css e javascript puro.

Quando andiamo a scrivere con altre librerie o framework noi andiamo a scrivere codice che poi deve essere in qualche modo interpretato il browser non può farlo da solo per questo motivo esistono tutta una serie di librerie che prendono quello che noi scriviamo con un determinato framework e lo convertono in javascript puro e tutto questo a questa attività che avviene dietro le quinte tipicamente è fatta attraverso delle librerie che si chiamano **bundlers** come ad esempio la più conosciuta ed utilizzata ad oggi è Webpack (<https://webpack.js.org/>).

Quanto tempo ci vuole a diventare front-end developer?

Posso dire quanto tempo non ci vuole, non bastano pochi mesi. Non bastano ad esempio tre mesi per essere pronti perché come avete visto quello che serve per essere considerati capaci di lavorare bisogna conoscere un bel po' di roba. Occorre almeno conoscere bene html css javascript, aggiungiamo anche SAS e altre tecnologie per quanto riguarda i css, ci aggiungiamo SOAP (è un protocollo per lo scambio di messaggi tra componenti software) e sicuramente almeno uno dei maggiori framework utilizzati come Angular, ecc.

Dal mio punto di vista per arrivare a conoscere destreggiarsi bene quindi arriviamo al allo step iniziale conoscere quello con cui si sta lavorando è quindi essere in grado di maneggiarlo ci vuole un annetto all'incirca ma per arrivare a proprio affacciarsi mettere la faccia dire ok ho capito dopodiché ***esperienza, esperienza, esperienza, esperienza***, esperienze sua esperienza progetti su progetti, sperimentare in continuazione perché non si è mai arrivati.