

Fullstack

Som fullstackutvecklare jobbar du med hela stacken inom mjukvaruutveckling på webben. Stacken som det heter är kombinationen av program, ramverk och verktyg som behövs för att bygga en webbplats eller en webbapp.

En fullstackutvecklare är en person som kan jobba med hela stacken vilket innebär backend på serversidan eller fronten på klientsidan.

HTML

Som fullstackutvecklare har du goda kunskaper i flera språk och kan anpassa ditt arbete vartefter behovet hos kunden. Dessutom hör det inte till ovanligheten att en fullstackutvecklare lär sig nya språk under sin karriär.

Ett av de språk en sådan utvecklare måste kunna tillämpa är HTML, vilket är vad jag använder för att skriva den här uppgiften. HTML är det språk som ligger längst upp på ytan under det som användaren ser.

HTML står för Hyper Text Markup Language och används för att strukturera det visuella på en hemsida, det är med html man implementerar länkar, bilder, video och textformatering.

CSS

CSS är det andra självklara språket för en fullstackutvecklare. Det används för att snygga till hemsidan. CSS står för Cascading Style Sheet och agerar som ett regelverk för HTML koden, CSS berättar hur HTML koden ska bete sig när den presenteras för en användare. Det är tack vare CSS vi kan skapa vackra men också funktionella hemsidor, då CSS gör det möjligt att separera de vackra från de funktionella och låta var och en uppträda med lika mycket uppmärksamhet på båda utan krock.

CSS är också anledningen att vi kan använda oss av internet i olika webbläsare och förvänta oss samma resultat.

JavaScript

Det tredje och mest populära av programmeringsspråken för att bygga hemsidor idag är JavaScript. JavaScript spelar en kritisk roll i skapandet av en webbsida. Det är tack vare detta programmeringsspråk vi kan förvänta oss att knappar leder där dom ska, att bilder går att klicka på och det är JavaScript som gör det möjligt att ha en trevlig, interaktiv och intuitiv upplevelse på webbsidor. Medan HTML säger vart en knapp ska finnas, CSS berättar hur den ska se ut så är det tack vare JavaScript det ger användaren en anledning att ens trycka på knappen från första början. Med hjälp av JavaScript kan en utvecklare implementera viktiga funktioner och tredjeparts ramverk för att göra upplevelsen för användaren så mycket mer lustfylld.

Backend

När en fullstackutvecklare jobbar backend pratar man om det som måste ske bortom det en användare ser. I backend jobbar man med *server side programming*. I backendutveckling används oftast andra språk än de vi har nämnt med undantag för JavaScript som numera kan köras som ett backend språk tack vare Node.JS. I backend jobbar en utvecklare med databashantering, API utveckling, servern och webbsäkerhet. I stort handlar det om att på ett säkert och effektivt sätt skicka och ta emot data.

Frontend

När vi jobbar frontend är fokus på kundupplevelse. Vi kan höra orden UI och UX design vilket är förkortningen för de som jobbar med *user experience* och *user interface*. Även om dessa två positioner kan skriva kod är deras huvudsyfte att förstå kunden och ta reda på vad som skulle göra deras upplevelse bättre. I många om inte alla fall handlar det om "hur kan vi få dem att göra ett köp" och hur en UX/UI kan manipulera användarens omgivning för att göra de mer benägna att fatta ett beslut som leder till vinst för företaget.

När en fullstackutvecklare jobbar frontend handlar det om att implementera de ändringar en UX/UI designer vill förändra, det kan därför vara fördelaktigt att vara duktig på att kommunicera.

IDE / Texteditor

En IDE, Integrated Development Environment, är en utvecklingsmiljö för programmerare där det finns en omfattande mängd nödvändiga verktyg, textredigerare, debugging verktyg och som också kompilerar din kod. Det huvudsakliga syftet med en IDE är att effektivisera arbetsprocessen genom en enda plattform med alla de essentiella verktygen en utvecklare kan tänkas använda.

IntelliJ, Eclipse, Xcode, PyCharm och Visual Studio är exempel på sådana program.

Visual Studio Code är ett textredigeringsverktyg som används för att skriva enklare kod. Det som skiljer ett textredigeringsprogram mot en integrerad utvecklingsmiljö är att textredigeraren, i det här fallet VSC, inte kommer med en omfattande mängd inbyggda verktyg. VSC fokuserar istället på enkelheten, förinstallerat är ett antal verktyg för att skriva kod på ett likvärdigt sätt som i en IDE. VCS har stöd för *syntax highlighting*, *IntelliSense*, och *debugging*.

När det kommer till mindre projekt eller prototyper passar VSC för professionella programmerare, framförallt gör VSC det möjligt för de som har mindre eller ingen erfarenhet att testa på hur det är att koda i en professionell miljö utan att överbelasta hjärna och dator.

Git och GitHub

Git är ett så kallat VCS, Version Control System eller Versionshanteringsprogram som är utformat för att utvecklare på ett enkelt sätt ska kunna samarbeta utan att förlora ändringar eller göra ändringar som skadar programmet. Genom att skapa avgreningar så kallade *branches* kan var och en jobba på en specifik del i programmet eller koden för att sedan sammanfoga den med stammen, det kallas för att *merge*.

All data som sparas med hjälp av git sparas på den lokala maskinen, vilket gör att github kan komma in i bilden. Github löser just det, genom att spara filerna i molnet hos github gör den som startar projektet en så kallad *repository* som motsvarar allt jag nämnde ovan om git. Den stora skillnaden är just att det nu blir möjligt att för vem som helst i världen att samarbeta med dig i realtid. Detta har blivit otroligt populärt i modern tid och varenda utvecklare använder sig idag av git och github för att starta nya projekt. Plattformen är open-source och har en inbyggd community plattform som gjort det möjligt för ett mer effektivt arbete hos alla.

