

Shokran Bahram SUT24

Frågor

1. Vad är HTML och vad används det till? Beskriv också kort historiken för HTML.
HTML som är förkortning till Hypertext markup language är det språket som används för att strukturera webbsidor, det vill säga allt text och bild innehåll genom att använda taggar. HTML kom som förslag 1986 av Tim Berners-Lee men som blev verklighet 1991 för att forskare på företaget CERN ska kunna dela dokument via internet. Språket har utvecklats över tid sedan den först kom ut och den senaste versionen som är HTML5 kom ut 2014.
2. Vad är CSS och vad används det till? Beskriv också kort historiken för CSS. **CSS som står för cascading style sheets är stilmall språk som är till för att bestämma utseendet av en hemsida samt att det används till att styra allt från design,färg,typsnitt,placering av de olika html elementen och sidans responsivitet. CSS kom som förslag 1994 av Håkan Wium Lie som också jobbade på samma företag som Berners-Lee(HTML). Förslaget blev till verklighet 1996 av företaget W3C som ville separera innehållet med designen på webbsidorna. CSS har kommit med 2 stora uppdateringar sedan den först skapades CSS2 och CSS3.**
3. Förklara vad responsiv design innebär. **Responsiv design innebär att skapa en hemsida som fungerar bra på alla skärmar oavsett om det är en mobil eller dator. Responsiv design kan uppnås främst genom media query i CSS men även genom flexibel layout. Syftet med att skapa detta är för att ge användaren möjlighet att använda hemsidan på alla plattformar.**
4. Vad är JavaScript och vad används det till? **Javascript är ett programmeringsspråk som är till för att göra webbsidor både responsiva och interaktiva, det vill säga genom att manipulera HTML och CSS genom DOM(Document Object Model). DOM hjälper oss att välja en specifik del av html filen som vi vill ändra på utan att behöva ladda om sidan varje gång. JS och DOM kan användas som exempel till att hantera knapptryckningar, uppdatera en hemsida i nutid som till exempel ett sökfält, hämta/sicka data till API sidor som till exempel väderapp samt API som kan ändra bakgrunden till en random bild.**
5. Vad är ECMA-script och hur hänger det ihop med JavaScript? **ECMAScript är en standardisering specifikation som JS bygger på, det vill säga JS följer**

ECMA-standarden som definierar funktionerna och syntaxen för programmeringsspråket samt hur JS ska fungera genom definitionerna. Varför JS hänger ihop med ECMAScript beror på att JS implementerar ECMAScript det vill säga när ECMAScript uppdateras blir JS även uppdaterad. en milstolpe för dessa var ES6 versionen som släpptes 2015 som hade många nya funktioner som till exempel const och let som nu används istället för var.

6. Förklara översiktligt vilket ansvar HTML, CSS och JavaScript har i teknikstacken inom frontend på webben.

HTML- HTML bestämmer strukturen på hemsidan, det vill säga allt vi ser på hemsidan som till exempel text och bild.

CSS- CSS bestämmer designen och layouten på hemsidan, det vill säga all färg, typsnitt och responsivitet.

JS- JS ansvarar för all innehåll som användaren kan trycka eller använda som till exempel knappar, API anrop och sökfält.

7. Beskriv vad DOM är och hur vi använder det när vi skapar en hemsida. **DOM som är förkortning till Document Object Model representerar hemsidans struktur i ett träd liknande form. Man kan säga att det fungerar som API där varje element som till exempel rubrik och knapp kan manipuleras med hjälp av events och metoder. När vi skapar en hemsida behöver vi alltid ha `(document.addEventListener("DOM Content Loaded", funktion() {}))` som låter oss att använda DOM funktionerna genom att kalla på dem inuti den efter att dokumentet är läst samt att utan den fungerar inte DOM metoderna.**

Några exempel på hur dom kan användas är `document.getElementById("test")` som hämtar ett element med id namnet "test" annat exempel är `test.addEventListener("click", () => { console.log("Knappen klickades!"); });` som lyssnar på en händelse när en knapp klickas.

8. Vad menas med ett JavaScript-ramverk och vad tillför det till ett projekt jämfört med att bara använda ren JavaScript? **Ett JS-ramverk är som ett bibliotek med redan färdiga funktioner samt strukturer som kan göra utvecklingen av komplexa sidor lättare eftersom de har återanvändbara komponenter och lösningar på olika problem som vi kan välja att använda eller ta bort. Dessa ramverk till skillnad från ren JS gör det lite lättare och sparar oss tid eftersom vi inte behöver skriva all kod själva som vi vanligtvis gör med ren JS och kan därför återanvända en komponent på flera ställen som till exempel en knapp.**

9. Lista tre vanligt förekommande JavaScript-ramverk. För varje ramverk ska du sedan kort beskriva det. Efter din lista med de tre ramverken ska du översiktligt förklara vad som skiljer dessa åt.

SPA = Single Page applications

React: Ett ramverk med öppen källkod som är skapad av facebook som är till för att skapa användargränssnitt för speciellt SPA. React låter användarna skapa en dynamisk sida vilket gör att data kan uppdateras utan att sidan ska laddas om.

Vue: Ett ramverk som är utvecklat av Evan You som är till för att skapa webbgränssnitt och SPA. Vue funkar bra tillsammans med andra ramverk och kan därför också användas för mobila och stationära appar tack vare att JS-basen samt HTML tilläggen fungerar bra med Electron ramverket.

Angular: Ett ramverk som är skriven i TypeScript med öppen källkod som är utvecklat av Google som likt React är till för att skapa mer komplexa SPA. Angular har många inbyggda funktioner som användare kan använda utöver den vanliga strukturen, exempel på inbyggda funktioner som angular har är databindning och routing.

Skillnader: Skillnaderna mellan dessa tre ramverk är att React är den mest flexibla eftersom den är mer som ett bibliotek än ett komplett ramverk samt att den har en medelsvår inlärningskurva. Angular är ett komplett ramverk och tar därför längst tid bland dessa tre att lära sig eftersom den är skriven i TypeScript. Angular har till skillnad från de andra två ramverken en strukturerad lösning med flera inbyggda funktioner vilket gör att den har en stor kodbas till skillnad från React och Vue som använder DOM vilket gör den inte lika effektiv till små projekt. Vue är den enklaste att använda bland dessa tre och passar bra för små projekt.

10. Vad är tillgänglighetsanpassning av webbplatser och varför är det viktigt?

Tillgänglighetsanpassning innebär att utveckla hemsidor så alla kan använda det även personer med funktionsnedsättning som till exempel synnedsättning. Att tillgänglighetsanpassa handlar mer om att ha skapat en sida även anpassad för funktionsnedsatta genom att till exempel ha en tydlig navigering, stöd för skärmläsare men även alt-text för bilder m.m. Detta är viktigt främst för att alla ska kunna använda sidan oavsett om det är en person med funktionsnedsättning eller inte samt att det är krav i vissa länder om att detta ska vara tillgängligt för att kunna användas.

11. Vad är ett webb-API och vad används det till? **Man kan säga att webb-API är som ett kommunikationslänk mellan frontend och backend, det vill säga frontend är det användaren ser och upplever medan backend innehåller all data som till exempel systemlogik,beräkningar och databashantering. Backend håller i allt information som ska visas för användaren i frontend och API i det här fallet används till att ta emot förfrågningar från hemsidans användare till backend som sedan går igenom frågan och skickar sedan svar till frontend. Webb-API kan på liknande sätt användas till att hämta väderdata.**
12. Förklara REST och redogör för dess huvudprinciper. **REST som är förkortning till Representational State Transfer är en arkitekturstil till för att skapa webb-API som gör kommunikationen mellan frontend(klient) och backend(server) möjlig genom att använda HTTP-standarderna.**

Huvudprinciperna för REST är:

1. klient-server: **Innebär att frontend(klient) och backend(server) är åtskilda och kommunicerar bara via HTTP**
2. Stateless: **Innebär att varje frontend förfrågning ska innehålla allt information som behövs och att backend inte sparar tidigare förfrågningar.**
3. Cacheable: **Innebär att svar från backend(servern) ska kunna cachelagras för att få bättre prestanda.**
4. Enhetligt gränssnitt: **Innebär att använda HTTP metoder som till exempel GET som hämtar data,POST som skapar data,PATCH som uppdaterar datan och DELETE som raderar datan.**
5. Layered System: **Innebär att backend(servern) kan ha flera olika lager men frontend(klienten) vet inte vilket lager som kommer ta hand om en förfrågan.**