

Teorihandbok - Webbutveckling

Frågor:

1. Vad är HTML och vad används det till? Beskriv också kort historiken för HTML.

HTML står för HyperText Markup Language. Det är det språk man använder för att koda själva skelettet för hemsidor i med hjälp utav taggar i form av titlar, listor, rubriker, länkar med mera.

1991 utvecklades HTML av Tim Berners-Lee, som även skapade det första webbserverprogrammet och webbläsaren. Då var det väldigt grundläggande textformattering och länkning mellan dokument.

1995 lanserades HTML 2.0 som en ny standard av Internet Engineering Task Force (IETF). Här fanns det fler funktioner och interaktivitet.

1997 introducerades ytterligare element med HTML 3.2 som tabeller och "applet-taggen" för att inkludera javaprogram.

1999 publicerades HTML 4.01 som blev en stabil version som innehöll förbättringar för stöd av multimedia, skript och stilark (CSS, mer om det i nästa fråga).

2004 började arbetet med HTML 5 vilket används än idag. Syftet var att modernisera språket för att det skulle hänga med i den snabba utvecklingen utav webben.

2014 blev HTML5 en W3C-standard, vilket innebär att det erkändes och rekommenderades av World Wide Web Consortium (W3C). Detta markerade en viktig milstolpe, eftersom det ger en trygghet när det gäller långsiktighet och hållbarhet för webbinnehåll.

2. Vad är CSS och vad används det till? Beskriv också kort historiken för CSS.

CSS (Cascading Style Sheets) är ett designspråk som man använder för att piffa upp delarna ifrån sin HTML, här kan man ändra storlek färg och form på de elementen man skapat i sin HTML, även olika typsnitt, animationer, skuggor & effekter.

CSS introducerades på 1990-talet för att ta tag i vissa designproblem på webben. Innan CSS fanns så såg man enbart hemsidor i form av HTML vilket inte ser särskilt tilltalande ut

1996 lanserades CSS1 av W3C.

1998 lanserades CSS2 och där introducerades fler funktioner, såsom bättre medieanpassning och z-index (lager positionering).

2000 kom en stabilare version i form av 2.1 men versionen vi använder idag CSS3 släpptes 2009 och med den kom det smidiga verktyg i form av flexbox & grid.

Låter likt CSN men det är något annat.

3. Förklara vad responsiv design innebär.

Responsiv design är hur tacksam webbplatsen är att använda på olika enheter och därmed olika skärmstorlekar, man får se till att allt innehåll inte klumpar ihop sig och att det anpassar sig efter utrymmet som finns. Att allting blir dynamiskt helt enkelt, detta uppnås genom att undvika att använda en fixerad storlek på en del element och även genom "media queries". Med hjälp av dessa så kan man välja själv förändringar som ska ske på webbplatsen/appen vid olika skärmstorlekar.

4. Vad är JavaScript och vad används det till?

JavaScript är ett programmeringsspråk som används för att skapa dynamiska och interaktiva webbsidor och är känt för att vara mångsidigt och flexibelt. JavaScript gör det också möjligt att manipulera HTML & CSS innehåll i realtid, detta på grund utav att JavaScript även är ett så kallat scriptspråk vilket innebär att koden tolkas och körs direkt i webben utan att behöva kompileras i förväg.

5. Vad är ECMA-script och hur hänger det ihop med JavaScript?

ECMA-Script specificerar grundläggande regler för språkets syntax, variabler, datatyper, funktioner, operatorer mm. Det standardiserades första gången 1997 av ECMA International, en organisation som ansvarar för att utveckla tekniska standarder. Då JavaScript är ett script-språk så följer det denna standard även om det har egna funktioner som DOM-manipulation & eventListeners.

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

HTML- står för själva grunden av innehållet, skelettet som jag beskrev det innan, här kan man definiera, titlar, rubriker, "brödtext", knappar, boxar, sektioner och olika delar som hemsidan ska innehålla.

CSS- tar hand om det estetiska, här kan man sätta bakgrundsfärger, textfärger, typsnitt, ändra form & storlek på olika boxar och även om "lådan" placeras i ditt HTML dokument så kan du med hjälp av CSS flytta den dit du vill ha den.

JavaScript- kan manipulera det ovannämnda på i stort sett alla sätt och vis, vi kan dynamiskt lägga till fler boxar som innehåller element likt en box vi skapat från start, uppdatera text i realtid med hjälp av knapptryck eller mushändelser. Det har möjlighet att göra hemsidan mer levande.

7. Beskriv vad DOM är och hur vi använder det när vi skapar en hemsida.

DOM (Document Object Model) är ett programmeringsgränssnitt för att manipulera HTML & XML-dokument. Med hjälp av JavaScript kan vi manipulera DOM på olika sätt med händelser, till exempel musklick eller tangentbordshändelser. På detta sätt kan vi dynamiskt förändra innehåll, utseende & struktur på hemsidan utan att den ska laddas om. DOM är ett objektorienterat sätt att representera HTML-strukturen.

8. Vad menas med ett JavaScript-ramverk och vad tillför det till ett projekt jämfört med att bara använda ren JavaScript?

Ett ramverk är ett hjälpmedel för att kontrollera programflödet i en applikation, om man jämför med till exempel React som är ett JavaScript bibliotek så kan det ses som en uppsättning med verktyg och funktioner som man kan använda vid behov utan att påverka flödet eller strukturen.

Ett ramverk å andra sidan ger en fördefinierad struktur och styr flödet av applikationen. Designat för att göra kodskrivningen snabbare, strukturerad och effektiv.

Både bibliotek & ramverk är smidigare att använda än ren JS då du dels skriver all funktionalitet själv och bara arbetar med grundläggande funktionalitet som erbjuds och inbyggda API:er som DOM-Manipulation, eventListeners, fetch & getElement.

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.

React- är inget fullständigt ramverk, mer som ett bibliotek med olika verktyg som hjälper oss att bygga interaktiva hemsidor med hjälp utav komponenter, dessa kan ses som legobitar som man kan stapla på varandra och återanvända för att skapa delar av en hemsida. React är byggt utav Meta.

Next.js- är ett ramverk som är byggt ovanpå React (icke självständigt), som är skapat för att göra det lättare att skapa snabba och SEO-vänliga hemsidor. SEO är kort och gott att hemsidan ska få många träffar om man "googlar" den och står för Search Engine Optimization. Next.js erbjuder även SSR (Server Side Rendering), vilket innebär att sidan kan laddas redan på servern, vilket gör att den visas och upplevs väldigt snabbt för användaren.

Vue.js- har jag inte hunnit testa men vad jag kan läsa mig till så är det ett väldigt användarvänligt ramverk med lättförståelig syntax, bra för nybörjare men har även vad som krävs för erfarna utvecklare. Med Vue kan du snabbt skapa dynamiska komponenter och hela applikationer.

Angular- är ett ramverk utvecklat av Google som är byggt på TypeScript. Det har stora komplexa företagsapplikationer med en bra struktur och många färdiga funktioner. Angular har jag heller inte testat, men det verkar vara en brantare inlärningskurva än vad Vue och React har.

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

Det är viktigt för att hemsidor ska kunna användas av så många som möjligt. Detta med hjälp utav textstorlek, kontrast, navigering via tangentbord och att allt innehåll går att förstå utan ljud eller bilder. Det finns krav på att hemsidor måste

uppfylla åtminstone AA-klassning i tillgänglighetsstandard för att få säljas och användas inom offentlig sektor.

11. Vad är ett webb-API och vad används det till?

Webb-API står för (Application Programming Interface) och är ett sätt för olika applikationer att kommunicera med varandra över webben. Det används för att skicka data eller utföra åtgärder ifrån en annan applikation eller tjänst. Detta måste utföras inom API:ets regler som man sätter när man bygger API:t

12. Förklara REST och redogör för dess huvudprinciper.

Representational State Transfer. Är en arkitekturstil som definierar en uppsättning av begränsningar och principer för att skapa webbtjänster // Malin Lindberg.

REST är en struktur för hur Backend och Frontend ska kunna kommunicera, det är baserat på http-protokollet från grunden och försöker följa dess standard.

Så när man bygger webb-API:er gör man ofta detta med REST-principerna i bakhuvudet. Då kallas de för RESTful.

1. Stateless.

Detta innebär att mellan förfrågningar så får inte servern lagra information om klientens tillstånd. Varje begäran måste vara självständig.

2. Client-Server Architecture.

Både klient och server ska vara oberoende utav varandra. Klienten skickar begäran och servern skickar svar. Detta underlättar skalbarhet och gör klientkoden mer portabel till andra plattformar. "Separation av bekymmer".

3. Cacheability.

För att förbättra prestandan kan man minska behovet av upprepade förfrågningar till servern genom att vissa svar kan markeras som cachebara eller icke-cachebara. Om svaret är cachebart, ska servern skicka rätt http-header för att markera det.

4. Uniform Interface.

REST använder ett enhetligt gränssnitt. Detta för att förenkla kommunikationen genom URIs (Uniform Resource Identifiers) som manipuleras med http-metoder som GET, POST, PUT & DELETE. (CRUD Operations).

5. Layered System.

REST-API:er kan vara uppdelat i flera lager för att hantera olika funktioner som säkerhet eller balansering utav data vid högt tryck. Detta är ingenting som klienten ser och det förbättrar skalbarhet och flexibilitet.

Ett exempel på detta skulle kunna vara en e-handel där en server hanterar användarinloggning, en annan betalningar och en tredje som hanterar produktinformationen.

6. Code on Demand.

Servern kan tillfälligt tillhandahålla exekverbar kod till klienten för att utöka funktionaliteten, tex med hjälp av JavaScript. Detta kan hjälpa till att hemsidor arbetar snabbare då mindre kod krävs för själva klientsidan.

Fördelar med REST:

- Hög skalbarhet.
- Enkelhet i design.
- Flexibilitet vid förändringar.
- Ökad prestanda genom caching.
- Modulerbarhet och oberoende mellan komponenter.