

JS Skolan

Arbetsprocessen

Den fiktiva websidan: JS Skolan

Innehåll

Bakgrund	1
Processen	1
Förbättringsförslag	3

Bakgrund

Uppdraget var att ta fram en webbsida till mitt slutprojekt i kursen YHJUST16 – HTML, CSS, JavaScript och webbstandarder. Utöver tidigare nämnda teknologier så skulle webbsidan även innehålla kod från biblioteket JQuery samt ramverket Bootstrap. Baserat på detta så hämtades inspiration från tidigare skapade websidor som t.ex.

<http://www.templatemonster.com/bootstrap-website-templates/>. Detta medförde slutligen till en webbsida om hur JavaScript fungerar "JS Skolan". Här kunde jag visa mina nuvarande kunskaper vilket även underlättades då webbsidan ämne gick hand i hand.

Processen

Efter valet av ämne så etablerade jag en struktur för projektet baserat på Bootstraps "kom igång kod" <http://getbootstrap.com/getting-started/>. Då Bootstrap är beroende av JQuery så infogade jag även den min.js varianten direkt i Bootstraps egna fil (sparar en extra server request). Vidare så lades script taggarna i head med nyckelordet defer, där alternativet var att ha dem innan slutet av body taggen. En Google font kopplades även in för typografin.

Webbsidan implementerades med inspiration från tidigare websidor med flat design <http://www.creativeblog.com/graphic-design/what-flat-design-3132112>. Detta medförde således en struktur utan några större distraktioner hos logon samt innehållet. Vidare så fick "Kontakta oss" samt "Diskutera" en normal Bootstrap layout då jag ville ha en variation mot den tidigare nämnda designen. För mobiler så täckte objekten en rad vardera. Detta expanderades till två respektive tre kolumner baserat på skärmens storlek hos innehållet.

Vid ankomst på webbsidan så möts man av en stor logo. Denna implementerades med hjälp av JQuery och egen kod. Vid en scroll neråt så försvinner den om man har passerat 80 % av bilden. Detta gäller även tvärtom oavsett skärmens storlek.

Vidare så gjorde jag koden så att image links itererades ut med hjälp av JavaScript under rubriken "Innehåll". Ett klick på någon av de dynamisk skapade länkarna medförde även att de ersattes med dess barn (meny länkar). Slutligen så kunde man även klicka på ett barn och förflyttas automatiskt neråt och börja läsa om t.ex. variabler. Detta tog förmodligen längst tid i projektet då jag var osäker på hur jag skulle koppla samman relationerna pappa, barn samt dess innehåll.

Github: mow-github
Email: matwik@gmail.com

För "Kontakta oss" så ändrade jag om Bootstraps CSS för forms samt så aktiverade jag Bootstraps tooltip funktionalitet manuellt. Som ni ser i figurerna nedanför så skriver den ut ett automatiskt felmeddelande för respektive fält. Vidare så ger den även en transition effekt vid en hover på en label. Slutligen så öppnas en Bootstrap modal vid en korrekt validering som ger ett tack för din fråga.

<p>E-post</p> <div><input type="text" value="question@jsskolan.se"/></div> <p>Ange en giltig e-post</p> <p>Meddelande</p> <p>Hej, jag skulle vara tacksam ifall du kunde hjälpa mig med följande fråga... Kan jag använda <u>let</u> istället för var?</p>	<p>E-post</p> <div><input type="text" value="question@jsskolan.se"/></div> <p>Se till att e-posten har ett @, samt att den är längre än 10 tecken</p> <p>Meddelande</p> <p>Hej, jag skulle vara tacksam ifall du kunde hjälpa mig med följande fråga... Kan jag använda <u>let</u> istället för var?</p>
--	--

Det sista som implementerades var "Diskutera". Här tänkte jag att människor kunde lämna feedback på något. Här återanvände jag en bit av formuläret från "Kontakta oss". Ett submit innebar således att meddelandet lagrades i en temporär variabel. Därpå så klonades det första inlägget och ersattes med den nya texten.

JS

Tjo, kan någon förklara skillnaden på t.ex. `var x = 1;` och `let x = 1;` tack

Meddelande

`<code>let</code>` är ett nytt nyckelord i och med standarden ecmascript 6.. som i princip ersätter `<code>var</code>`
`<code>var</code>`
`<code>p>Läs gärna mera om det på : stackoverflow</code>`

Skapa inlägg

Förbättringsförslag

- Man skulle kunna ersätta den fiktiva datan i custom.js, så att den hämtas från en databas samt översätts på server sidan. I nuläget så generas en HTML som sedan itererar ut dynamisk data med JavaScript.
- Header logo skulle kunna förminkas alternativt tas bort efter ett första besök.
- Införa lite mer liv på sidan i form av bilder och övrig media som inte slår över och ger ett plottrigt uttryck.