

Vid arbetet med detta program har jag stött på flera problem. Jag har under tiden fått tänka om flera gånger och adaptera min kod för att få den att fungera med varje funktion och beroende på slutfunktion. Detta har lett till att mitt program har varit under ständig förändring och även om skalet har sett densamma ut så har undersidan varit under konstant förändring. Vid början av arbetet valde jag att lägga upp en plan, där jag gick igenom de olika funktioner jag trodde jag skulle behöva. Därefter tog jag fram globala variabler för att kunna skapa ett skal till hela programmet. Det visade sig väldigt snabbt att jag hade tänkt alldeles för brett och fick tidigt börja göra om mina funktioner och skriva mindre funktioner. Vad jag menar med detta är att jag istället för få stora funktioner, gjorde om det till flera funktioner som jag delade upp. Detta gjorde jag då jag insåg att varje del av programmet behövde delas upp och bara användas vid den tänkta tidpunkt.

När jag skulle börja hantera händelsehantering och händelsehanterare så la jag grunden till dessa koder inuti min init funktion. Jag flyttade ut denna då jag inte tyckte det var värt att ha den konstant laddad och tänkte att jag skulle kunna aktivera den vid knapptryckningar. Men detta fick jag tänka om på, ännu en gång, och la tillbaka min dragstart och dragend till init, då den ska finnas laddad hela tiden. Istället valde jag att dela upp hur dragover, drop och dragleave skulle hanteras. Detta gjorde att jag ganska direkt fick en mycket bättre struktur på min kod och gjorde hela arbetet enklare att arbeta med. Dock var arbetet med mina händelsehanterare inte felfritt och jag stötte på stora problem under arbetets gång, då jag flertalet gånger fick fel slutresultat. Det resulterade i att jag fick tänka om och strukturera om min kod efter lite hjälp från Rune samt att jag själv insåg att min grundtanke inte var alldeles genomtänkt.

Något som varit genomgående med mitt arbete är att jag alltid skrivit mina funktioner på ett sätt, men avslutade med något helt annat. Detta är något som jag beskriver i början av denna rapport, men jag tycker det är viktigt att poängtera att jag har konstant behövt strukturera om min kod för att få en slutprodukt som fungerar väl ihop med varandra. Jag har hela tiden under arbetets gång insett hur viktigt det är att förstå sin kod samtidigt som man inser vikten av att kunna vara flexibel i hur programmet ska skrivas. Det har gett mig ett sorts lugn, då jag inte känt samma stress när jag arbetat med detta program i jämförelse med min slut-hemsida i Webbteknik 1.

Jag har i detta arbete arbetat med nya delar jag inte har rört tidigare. Det har gjort att min process har varit lite avig. Jag har försökt mig på att lösa ett problem, och sedan kollat på tidigare exempel från kursen och tidigare laborationer. Jag tycker personligen det har varit ett hyfsat effektivt sätt att lära mig på, men det har dock gett mig lite problem på vägen. Jag har insett hur jag måste ändra på mitt sätt att arbeta med min egna läroplan. Mina val under arbetet har med andra ord varit väldigt agila. Jag har arbetat med funktioner och variabler på ett väldigt flytande sätt, som jag därefter har fått strukturera upp.

Jag har dessutom fått testa på hur felsökning av mer programmerings-kod fungerar än att felsöka HTML och CSS, eftersom JavaScript ger helt andra typer av fel och problem än vad jag tidigare stött på. Det har medfört att jag har även här ändrat mitt sätt att arbeta. Tidigare har jag arbetat genom att skriva koder eller funktioner inom HTML och CSS som inte gett felkoder, mer än att kanske en punkt eller något är felstavat. I denna process har jag stött på väldigt många UNDEFINED eller xxx is not a function. Det har varit otroligt frustrerande då jag inte förstått vad mitt problem har varit när jag arbetat. Det har lett till väldigt mycket tid då jag stirrat in i skärmen och försökt lösa problemet genom att bara tänka om, skriva om kod och hoppas på det bästa.

Jag tycker definitivt att mitt arbete med detta program har inte varit i närheten av så smidigt som det hade kunnat vara. Dels på grund utav att julen kommit emellan. Det hela medförde en paus från hela plugget, och gjorde att det kändes som att jag började om på noll när jag skulle börja skriva detta program. Men också för att min approach till arbetet har varit mer en "jag ändrar det på vägen"-stil, än kanske en mycket vettigare "gör det rätt från början". När jag läste igenom tipsen som fanns till programmet så läste jag "Testa vid varje steg". Jag har varit alldeles för feg och dålig på att använda Console.log() vid delar av arbetet som jag kört fast på. Vid slutet av programmet använde jag det otroligt frekvent och det gav mig ett mycket enklare sätt att hantera problem på vägen. Jag lärde mig den hårda vägen att inte använda verktygen som finns i programmering är dumt.

Mycket i arbetet har verkligen varit att testa sig fram och se om det fungerar. Har det inte fungerat har jag fått tänka om och skriva om. Och det har hänt så många gånger att jag nästan blir trött på mig själv när jag från början tänkt rätt men inte formulerat mina koder som jag egentligen vill. Mitt sätt i början av arbetet var att beskriva mina funktioner och vad jag ville de skulle göra. I slutet insåg jag att jag behövde vända på det, vad jag ville att funktionen skulle uppfylla för funktion, vilken fråga den skulle besvara typ. Det gjorde att min approach till mina funktioner blev mer systematisk och jag tog mina problem mer som en matematisk fråga.

Det hela har egentligen varit ett väldigt jobbigt arbete, inte för att programmet i sig har varit ett svårt program att skriva i slutändan, utan att jag har insett gång på gång hur min approach och mitt tillvägagångssätt har varit alldeles för dåligt.

Med det hela sagt, så tycker jag det har varit ett roligt projekt att "ta sig an". Det har under arbetet hela tiden tillkommit funktioner och plockats bort funktioner. Ett arbete som detta har jag inte tidigare stött på eftersom det har varit väldigt mycket fram och tillbaka och jag har varit ovan i mitt arbetssätt som det har medfört. Jag gick in i processen och tänkte att saker kommer bli så och så, men det har visat sig att "Det kan vara så, men det kan också vara så här". Det är något jag är väldigt ovan vid.

Jag har utvecklats som "programmerare" under hela projektet. Dels för att jag har fått prata mycket kod, både med mig själv högt och med Rune, men också för det har varit väldigt mycket trial and error i arbetet. Har det inte fungerat har jag fått börja om och se hur jag kan förbättra eller förändra mitt arbete. Genom arbetet med kommentarer som finns över hela programmet har jag kunna se hur jag förändrat min approach också till hur varje del ska hanteras.

Mina stora problem har som sagt varit händelsehanterare och hur jag ska ha räknat poängen. Det visade sig dock i slutändan att jag "övertänkte" min plan och insåg hur enkelt jag egentligen kunde komma till en slutprodukt som ger mig det resultat jag vill ha, utan att det blir massa kringelkrokar längs vägen. Det slutade med effektiva programkoder. Det är inte bara dessa funktioner som i slutändan blivit korta och effektiva, utan hela programmet har hela tiden blivit större samtidigt som koden har komprimerats och förenklats. Allt för att hålla det smidigt och enkelt, dels för mig själv men jag har även insett att det är inte bara jag som kan komma att läsa mina koder i framtiden och därför behöver jag vara tydlig i kommentarer och i strukturen av min kod. I början av denna kurs så när jag stötte på problem så kommenterade jag ut den gamla delen och skrev en ny, vilket ledde till en hel del grötighet när jag arbetade med programmet. Nu när jag stöter på problem så tittar jag istället på vad

problemet säger, och jag bemöter det med en approach som ger mig frihet att lösa problemet istället för en panik över att jag skriver fel kod från början.

Arbetet med kommentarer kring funktioner har definitivt förbättrat mitt arbete och förenklat det då jag stött på problem, med andra ord. Jag har utvecklat hur jag arbetar med programmering och hur jag hanterar problem till en ny nivå och jag tycker det är nästan roligare att felsöka kod och få den att fungera än att skriva saker från grunden.