

Individuell reflektion

I denna kurs, TDP003: Projekt: Egna datormiljö, har jag, Han Lin Yap, och Alexander Östman skapat en Portfolio-hemsida. Hemsidan skapades i Python med kid-mallen.

Arbetet gick väldigt bra, från start till slut. Jag och Alexander har tidigare programmerat förut. Jag har hållit på med hemsidor och Alexander med Python. Med dem vardera kunskapen så tycker jag att det var lätt att göra Portfolion. Jag är nöjd med portfolion.

Planering & skisser

Vi uppskattade hur lång tid datalagret och presentationslagret skulle ta och vi trodde dem svåraste delarna var sökning och kid-mallen. Dem uppskattningen stämde rätt så bra. Vi uppskattade att hela projektet skulle ta 46 timmar var och det tog 35 timmar för mig (sammanräknad tid från dagboken¹). Tiden för coachmöte och demo-sessionen är inte inräknad.

Våra skisser visade hur layouten på sidan skulle se ut och inte designen. Den stämde också i största del. Det var bara sökrutan som låg längre ner på hemsidan än på skissen.

Datalager

Det gick snabbt att börja bygga upp datalagret tycker jag, för att vi hade en systemspecifikation att gå efter. Vi kunde då fokusera att programmera "funktionaliteten", för att vi visste vilka funktioner, parametrar som skulle finnas och vilka returvärde det skulle vara.

Presentationslagret

Vi koda Xhtml och CSS till hemsidan först och det gick väldigt smidigt sedan gjorde vi om till kid-mall. Det var lite klurigt i början att göra om till kid men sedan när vi förstod logiken gick det bra.

Vilka problem/intressanta fenomen stötte du på?

Vi visste inte hur man skulle sortera efter en dictionary i en lista. Vi sökte på google och letade om det fanns någon inbyggd och det fanns det på sätt och vis. Det kunde lösas på en rad kod. (Se dagboken 2009-09-17 för kod exempel.)

När vi skulle testa datalagret med en test-fil så visade det sig att det var många fel. Det tog lång tid att fixa. Jag tror anledningen att vet blev fel var att det inte var tillräcklig beskrivet i systemspecifikationen.

Vi hade också strul med den avancerade sökningen när den skulle implementeras på presentationslager.

¹ Dagbok - http://google.com/notebook/public/15502429618632229265/BDWZu5goQ_M6Krbgk

Vilka insikter har du fått?

Att det blir lättare om man delar upp till två lager, datalager och presentationlager.

Jag har tidigare inte provat att använda mig av mall-system och nu förstår jag bättre hur mall-system fungerar.

Att lagra data på CSV är nytt för mig. Jag brukar annars använda mig av Mysql. På Mysql kan man ställa en fråga på vilken data man vill få tag på, men i CSV kunde man inte göra det så vi fick manuellt filtrera datan.

Vad skulle du göra annorlunda idag?

- Att ta reda på hur man låter bli att kid-mallarna blir cachade i webbservern. Kid-mallarna cachade i webbservern och det gjorde att man var tvungen att starta om webbservern varje gång man skulle ändra på en kid-fil. Det sölade ner tillverkningens processen även om det bara tar några sekunder att starta om webbservern.
- Att när man ska konfigurera Apache att fungera med mod_python modulen så skulle det vara bättre att man säger att endast py-filer är Python. Det vi hade, som stod i installationsanvisningarna, ställde in så alla filer är Python förutom css, png, gif osv. Det skulle vara bättre om vi inte exkluderade en filtyp i taget. Javascript-filer fanns inte med i exkluderingen därför funderade inte Javascripten. Apache trodde att Javascript-filen var Python. Det tog ett tag innan vi förstod att det var felet.

Jag skulle göra om datalagret. Jag skulle använda mig av högre-ordningens funktioner, som map och filter funktionerna, och list comprehension metoden mera, för att minska på så många inäslade loopar och if-sater som sökfunktionen till exempel var och eventuellt dela upp till fler små funktioner som kan återanvändas. Vi följde systemspecifikationen och trodde att man enbart skulle ha dem funktionerna som stod i systemspecifikationen. När vi programmerade datalagret kände vi inte till att funktionerna map och filter var så användbara och kraftfulla.

Om jag skulle göra en större sida så skulle jag behöva en annan mall-system än Kid, för att Kid är en liten mall-system som är väldigt begränsat. Kid har brist på dokumentation, många länkar i deras sida är trasiga och Kid utvecklas inte mera.

Källkod

Jag är nöjd med uppdelningen av koderna, men det går att förbättra. Våra namn på variabler är ibland i camelcase, ibland i med understräck och alla är inte så beskrivande. Vi hade inte bestämt oss i början om det skulle vara camelcase eller med understräck, därför blev det blandat.

I vår datalager har vi två globala variabler. Man ska undvika att ha globala variabler, men vi hade en global variabel som lagrade data från data.csv, för att i systemspecifikationen hade funktionerna ingen parameter för data, därför har vi en global variabel för det.

HTML- och CSS-koden är indenterad och i CSS har vi försökt att sortera attributen i bokstavordning så det blir lättare att hitta.