 Referat fra vejledermøde #6

**Dato:** 14-03-2016

**Tid:** 09:00 - 10:00

**Sted:** 312E

**Fremmødte:** Nicklas, Anders, Christian, Mette, Rasmus, Troels

**Udeblevet med afbud:**

**Udeblevet uden afbud:**

**Dagsorden:**

**Kommentarer fra mødet:**

**Hvad der er lavet**

Vi er midt i et sprint og er derfor kun så småt i gang med de forskellige tasks, da der på nuværende tidspunkt kun er gået tre dage siden sprintet startede.

Vi har rettet i projektformuleringen og i kravspecifikationen. Under vores user stories nævnte du at der skulle mere kød på dem, og vi er ikke helt sikre på hvordan vi skal gøre det. Så vil du kigge lidt på vores user stories, så vi kan snakke om, hvad det er, der skal på dem, til mødet. De to dokumenter ligger på Blackboard.

**Hvad der skal laves**

Vi skal have færdiggjort vores igangværende sprint og splittet vores tasks ud i mindre tasks.

**Processen**

Vi skal stadig huske at dokumentere alle de valg vi har taget. Det er den del af processen der betyder meget det her semester

**Spørgsmål**

*Dokumentering af kode - er det Doxygen?*

Kan det godt være – kan også vælge [noget som Troels ikke kan huske, men det er det der kører på C#] (Troels skriver det til os når han finder ud af hvad det hedder)

Vi skal passe på med ikke at spilde tiden på completeness – ikke smide alle mulige informationer på en meget simpel funktion

*Da du læste kravspecifikationen igennem sagde du at der skulle være en revisionshistorik - skal det være i dokumentet?*

Ja, han vil gerne have en kort tabel på forsiden af dokumentet hvor vi kan skrive når der er lavet større ændringer, hvornår de er lavet, hvem, hvad der er lavet osv.

*Har vi gjort sprintplanlægningen rigtigt nu? Vi føler at der er meget få tasks.*

Troels synes også der er meget få tasks som er meget store – specielt når vi ikke sidder og laver på det full time. Vores tasks skal have en størrelsesorden på 2-3 timer

Taskene er for store og skal brydes ned. Enten dele yderligere ud i Business Layer, Data Access Layer og Presentation Layer

Autofuldførelse – modellag, DAL, BL, få det til at virke på den anden

Finde ud af hvor varene fra indkøbslisten kan købes billigst – skal også brydes ned – evt. finde ud af om de to interfaces skal implementeres sammen

Skal sørge for at taskboardet afspejler den reelle tid der er brugt på det

*Hvad gør vi når vi finder fejl i vores produkt som er godt at få rettet hurtigt? Skal det tilføjes til taskboardet for sprintet? Vi har f.eks. lige rettet nogle ting i det vi implementerede sidste sprint, men det står ikke nogen steder.*

Ret med det samme. Fejl i det i sidste sprint har større prioritering end det her sprint

Hvis det er en lille ting (et par timer) skal det bare fixes under den task man er ved da man opdagede fejlen

Hvis det er en større ting laves der en ny task (under en story) hvor der så bliver skubbet noget andet ud af sprintet

*Er det relevant at holde møde hver uge i stedet for kun hver anden uge når vi skal afslutte sprint og starte nyt sprint?*

Det er op til os. Vi er i gang nu. Vi kan vælge at sige at vi hver fredag skriver om vi vil have møde. Eller at vi kun vil mødes hver anden uge og Troels så kan komme i løbet af ugen og kigge til os

**Generelt**

Autofuldførelse er ikke en story – det er en feature i find en vare e.l.

Vi skal være tro med modellen – stories skal give værdi for brugeren – det her er bare en funktionalitet (er en nice feature – men for sig selv giver den ikke nogen værdi, og er derfor ikke en user story)

Vores stories i taskboardet skal enten som titel eller i beskrivelsen have selve user story opsætningen – som en … vil jeg … så …

Troels er ikke umiddelbart fan af alle de infrastruktur til database – det fremgår ikke tydeligt hvad det er

Find en vare ville være smart at have så man kan trykke på varen og kan se hvor den ellers kan købes og hvad den koster der

Når sprintet er slut kan vi overveje om den måde vi har implementeret find en vare på er optimal eller om den senere skal laves om

Hvornår er noget done? Vi har sagt når der er en anden der kigger det igennem og er enig i hvad der er lavet

Troels vil gerne have at vi får vores tests på også

Automatiske tests på DAL og BL fremfor PL – det er en del nemmere end at lave automatiske tests på PL

Diagram of death – Troels vil gerne have vores packages på

Ikke tydeligt hvad unit of work er

Er det alle unit of work der skal have alle tre interfaces? Siger vi at de skal

Det minder om command pattern – skal vi lige kigge på

Autocomplete kunne i stedet hedde noget med fælles egenskaber – database funktionalitet som ikke har noget med brugerne at gøre – generisk funktionalitet

Bruger ikke interfaces IRepository nogen steder – det er implementeringsarv

Arv og repostitory på et diagram – unit of work og hvordan det anvendes på et andet diagram

Ikke arv og anvendelse på samme diagram helst – to forskellige ting man prøver på at vise

Der er en funktion der er svær at teste

Den er meget generisk og derfor svær at teste

Funktionalitet der ønskes i repositoriet er måske lagt ud i klienterne i stedet? Bør måske ligge i repositoriet for at det ikke er så generisk

Find product er defineret i BL – skal nok ligge i DAL. Forespørgslen skal være defineret i DAL

Burde gøre det nemmere at teste i BL

**Næste møde:**

*Vi mødes på fredag og holder sprint afslutning fordi vi har ferie i næste uge. Vi skal skrive til Troels hvor vi skal være og så kommer han til os.*

Med vejleder: mandag d. 18. marts 2016 klokken 12.15-13.00

**Kage:**