Fog Carporte

Maj 2020



|  |  |
| --- | --- |
| **Datamatiker 2. Semester, Hold E, Gruppe 666** | |
| **Alexander Pihl**  *Email:* [cph-as509@cphbusiness.dk](mailto:cph-as509@cphbusiness.dk)  *Github:* <https://github.com/AlexanderPihl> | **Morten Rahbek**  *Email: [cph-mr462@cphbusiness.dk](mailto:cph-mr462@cphbusiness.dk)*  *Github:* [*https://github.com/amazingh0rse*](https://github.com/amazingh0rse) |
| **Mick Larsen**  *Email: [cph-ml616@cphbusiness.dk](mailto:cph-ml616@cphbusiness.dk)*  *Github:* [*https://github.com/MivleDK*](https://github.com/MivleDK) | **Per Kringelbach**  *Email:* [cph-pk171@cphbusiness.dk](mailto:cph-pk171@cphbusiness.dk)  *Github:* <https://github.com/cph-pk> |
| **Jean-Poul Leth-Møller**  *Email: [cph-jl360@cphbusiness.dk](mailto:cph-jl360@cphbusiness.dk)*  *Github:* [*https://github.com/Jean-Poul*](https://github.com/Jean-Poul) | |

Indhold

[0. Projektformaliteter 3](#_Toc40899100)

[1. Indledning (Evt indrag problemformulering her) 4](#_Toc40899101)

[2. Baggrund 4](#_Toc40899102)

[3. Teknologivalg 4](#_Toc40899103)

[4. Krav 4](#_Toc40899104)

[5. Overordnet virksomhedsbeskrivelse 4](#_Toc40899105)

[5.1 Arbejdsgange der skal IT-støttes 4](#_Toc40899106)

[6 SCRUM Userstories 4](#_Toc40899107)

[7 Diagrammer 4](#_Toc40899108)

[7.1 ER Diagram 5](#_Toc40899109)

[7.2 Navigationsdiagram 5](#_Toc40899110)

[8 Særlige forhold 5](#_Toc40899111)

[9 Udvalgte kodeeksempler 6](#_Toc40899112)

[10 Status på implementation 6](#_Toc40899113)

[11 Test (Husk word fil fra Jon her og indrag v-model) 6](#_Toc40899114)

[12 Process 6](#_Toc40899115)

[12.1 Arbejdsprocessen faktuelt 7](#_Toc40899116)

[12.2 Arbejdsprocessen reflekteret 7](#_Toc40899117)

[13 Konklusion 7](#_Toc40899118)

[14 Perspektivering 7](#_Toc40899119)

# 0. Projektformaliteter

Produktets demovideo: a

Produktets website: a

Projektets kildekode: <https://github.com/Jean-Poul/Eksamensprojekt_Fog>

Projektets JavaDoc: a

# 1. Indledning (Evt indrag problemformulering her)

*Kort intro til hvad dette projekt omhandler. Formålet med indledning er at sætte en fagfælle i stand til at forstå resten af rapporten. For jer som studerende er en “fagfælle” en anden datamatiker studerende på 2. semester der er på samme niveau, men som ikke kender opgaven.*

# 2. Baggrund

*En kort beskrivelse af den virksomhed som skal bruge systemet*

*Hvilke krav kunden har til systemet forklaret i brede termer, f.eks. “kunden skal kunne bestille en cupcake, hvor man kan vælge både bund og top.” i modsætning til “der skal være en drop down menu med scrollbar i højre side med en liste over hvilke bunde der er og hvad de koster”.*

# 3. Teknologivalg

*En kort beskrivelse af hvilke teknologier der er brugt (jdbc, mysql, …). Her er det ikke meningen at I skal beskrive disse teknologier, men man skal sige hvilke der er brugt sådan at den der skal overtage projektet ved hvilken software der skal bruges. Der skal versions numre på (Netbeans 8.2, ikke blot “Netbeans”).*

# 4. Krav

*Der er to dele i dette afsnit:*

*- Hvad er firmaets håb med dette system*

*- scrum userstories (product backlog)*

# 5. Overordnet virksomhedsbeskrivelse

Beskriv hav firmat Fog forventer og ønske rat få ud af dette nye IT-system

## 5.1 Arbejdsgange der skal IT-støttes

Dels skal afsnittet beskrive de overordnede arbejdsgange før og efter IT systemet.

# 6 SCRUM Userstories

*Dette afsnit skal beskrive de user-stories der er aftalt med product-owner. Det er vigtigt at I har en håndfuld userstories som er lavet fuldt ud, dvs:*

*der er beskrevet how-to-demo*

*der er brudt ned i tasks*

*der er lavet et estimat*

*Den fulde produkt backlog kan ligge som appendix.*

# 7 Diagrammer

## 7.1 ER Diagram

*Det interesante ved denne domæne og database er at den langt hen af vejen er grundlaget for resten at systemet. Tabeller og relationer siger noget om hvad systemet arbejder med, ikke hvordan. Så det er godt sted at starte.*

*Som led i beskrivelsen af ER diagram skal man have følgende med:*

*Diagram over hele modellen. Det er vigtigt at få plads til alle tabeller og alle relationer. Det kan så betyde at man ikke kan få plads til alle attributter på de enkelte tabeller. Dem kan man slå op i databasen, så det er ikke så vigtigt*

*Hvis nogle af tabellerne ikke er på 3. normal form vil det være almindeligt at nævne det, og forklare hvorfor det er gjort (tidspres eller anden overvejelse).*

*Hvis der anvendes 1-1 relationer kan man beskrive hvorfor man ikke blot har en tabel.*

*Hvis nogle tabeller implementerer en mange-mange relation vil det være normalt at beskrive det.*

*Hvis der er flere veje at nå fra et sted til et andet vil det nemt gøre det svært at holde databasen konsistent. Hvis I har gjort det alligevel så skal I forklare hvorfor.*

*Hvis der er tabeller hvor man benytter andet end et automatisk generetet ID som nøgle skal man forklare det.*

*Der er interessant at beskrive hvilke overvejelser der ligger til grund for de konkrete valg der er i ER modellen (fremmednøgler, constraints, triggers, osv)*

## 7.2 Navigationsdiagram

*Det som brugeren oplever er en række websider hvor man kan indtaste oplysninger gå videre til andre sider. I større systemer kan det være svært at bevare overblikket over hvilke sider der er, og hvordan man kommer fra den ene til den anden. Navigationsdiagrammet er beregnet på at vise dette på en mere overskuelig måde. Som led i beskrivelsen af navigationsdiagrammet skal følgende med:*

*Oversigts diagrammet. Hvis det bliver for stort må man dele det op. Men det er vigtigt at der er et overordnet diagram. Hvis man har benyttet sig af en “fælles navigations bar” i toppen af alle sider skal man forklare det. Hvis nogle sider kun kan nåes af nogle brugere (dem der har konto, dem der er logget ind, dem der arbejder i butikken,…), så skal det fremgå.*

*Navne på jsp sider skal fremgå, og hvilke servlet der bringer en fra den ene side til den næste.*

# 8 Særlige forhold

*Dette afsnit bruges til at beskrive særlige forhold der benyttes i programmet. Det kan f.eks. være:*

*Hvilke informationer gemmes i session*

*Hvordan håndterer man exceptions.*

*Hvordan man på har valgt at lave brugerindput validering*

*Hvordan man har valgt at lave sikkerhed i forbindelse med login*

*Hvilke brugertyper der er valgt i databasen, og hvodan de er brugt i jdbc*

*… andre elementer – i Fog projektet kan det være:*

*Tegning*

*Stykliste beregner*

*Husk: det er bedre med 2 linjers dokumentation end ingen.*

# 9 Udvalgte kodeeksempler

*Det er ikke sikkert at censor (eller eksaminator) finder alle jeres guldkorn i selve koden. Derfor er det en god ide at vælge særlige kode stumper ud og vise dem i rapporten.*

*De eksempler der er givet uder “særlige forhold” afsnittet kan man godt tage og illustrere med kode direkte i rapporten.*

*Det kommer til at virke særligt overbevisende hvis den kode man vælger ud indgår som led i et af sekvensdiagrammerne.*

*Der er mange af jer der vil skrive jeres ting i word eller googledocs. Vær opmærksom på hvordan i formaterer jeres kode. Man vælger ofte en lidt mindre font, en der er “monospaced” (alle bogstaver optager samme bredde). Der er også nogle der sætter små skærmbilleder fra Netbeans ind. Det er OK, men så husk at vælge et tema fra netbeans med hvid baggrund og mørke/farvede bogstaver da nogle censorer skriver rapporten ud på blækprintere som ikke gengiver lyse bogstaver på sort baggrund særligt godt.*

# 10 Status på implementation

*Dette afsnit skal liste hvor langt man er nået med implementationen. Typiske ting man kan have sprunget over er:*

*Man har ikke nået at lave alle de jsp sider man har med i navigationsdiagrammet.*

*Man har ikke nået at lave alle CRUD metoderne til alle tabellerne*

*Man har ikke fået stylet sine sider*

*Man har fundet en fejl “i sidste øjeblik”, men har ikke haft tid til at rette det. - F.eks. at man har brugt session forkert sådan at man på en af siderne kan komme ind uden at være logget in.*

*tests der fejler på afleveringstidspunkt*

*… andre mangler*

# 11 Test (Husk word fil fra Jon her og indrag v-model)

*Der skal være lavet test. Du kan dokumentere tests ved at beskrive i tabel form:*

*Hvilke klasser er testet*

*Hvilke metoder der er testet*

*Dækningsgrad af dine tests for de valgte metoder og klasser*

|  |  |
| --- | --- |
| **Gruppenavn** |  |
|  |  |
| **Spørgsmål** | **Teststrategi** |
| Virker vores beregner? | Unit testing: White-box |
| Passer antallet af stolper? | Unit testing: White-box |
| Passer antallet af spær? | Unit testing: White-box |
| Andre beregninger? | Unit testing: White-box (Alle relevante metoder I beregner) |
| SVG-motor |  |
| Kan jeg tegne et rektangel med en vis størrelse? Virker vores metode? | Unit testing: White-box |
| Kan jeg tegne en pil som f.eks. er 300px lang? | Unit testing: White-box |
| Datamappere |  |
| Er der overhovedet hul igennem til databasen? | Integrationstest: Black box |
| Kan jeg indsætte en ny bruger? | N/A |
| Kan jeg hente et produkt / liste af produkter? | Integrationstest: Black box |
| Bliver en ordres status gemt i databasen når man opdaterer den på jsp siden? | Integrationstest: Black box |
| Hvad med vores brugergrænseflade? |  |
| Kan en kunde finde ud af at bestille en carport? Hvordan afgør vi det? | Testet på potentielle kunder |
| Kan de ansatte hos Fog finde ud af at bruge systemet? | Testet på ”Ikke-softwarekyndige”. |
| Hvordan håndterer vi input validering? | jQuery på front-end. Diverse validering på back-end. |
| Generelt |  |
| Er vores kode skrevet, så vi automatisk kan teste den? | Ingen selenium tests, men god struktur og base for unit- og integrationstests. |
| Hvordan sikrer vi os en ensartet kodekvalitet? | God dialog og daglig præsentation af gårsdagens arbejde. |
| Hvordan sikrer vi os at vi ikke tjekker fejlagtig kode ind på vores master- eller developerbranch i GitHub? | Ved altid at bruge *git pull* inden et commit og den daglige dialog. |

*Desuden kan du beskrive hvordan i systematisk har arbejdet med at teste koden før den er blevet gjort til en del af master branch.*

# 12 Process

Der skal være et afsnit hvor I beskriver jeres arbejsprocess i projekt perioden. Der skal dels være et faktuelt afsnit og et reflektions afsnit.

## 12.1 Arbejdsprocessen faktuelt

Dette afsnit skal beskrive:

Hvilke sprints der var, og hvilke user stories der blev arbejdet med.

Hvem der var scrum master i hvilke dele af projekt perioden. Giv gerne nogle eksempler på hvad scrum master gjorde i udvalgte sprint.

Et eksempel på et af PO-møderne, hvad der var planlagt fra jeres side, og hvordan det gik.

Hvordan i afholdt jeres daglige standup møder.

Hvornår i holdt retrospectives.

## 12.2 Arbejdsprocessen reflekteret

Dette afsnit skal beskrive jeres overvejelser over hvilke dele der har fungeret godt og hvilke dele der måske er faldet lidt på gulvet. I kan f.eks. beskrive:

Om scrummaster rollen fungerede, hvilke problemer I så i den, og hvad I gjorde for at rette op på det.

Hvad der var de væsentligste emner på jeres retrospektiv møder

Om I havde problemer med at nedbryde user stories i tasks

Om I var spot-on med jeres estimeringer

Om der var problemer med vejledningen og PO møderne

Hvor langt inde i processen I fandt en rytme der var produktiv

Andre elementer der har at gøre med at forsøge at arbejde i et scrum team

13 Konklusion

En konklusion er en sammenfatning af de resultater, man har fundet i forbindelse med udarbejdelse af rapporten. Dette skal dog ikke forstås som et resumé af, hvad man har skrevet, men som et svar på de spørgsmål man stillede i sin problemformulering. Det er meget vigtigt, at der bliver svaret på de spørgsmål, der blev stillet i problemformuleringen, men konklusionen skal kun indeholde de vigtigste pointer og påstande. Der må ikke inddrages nyt stof i en konklusion.

14 Perspektivering

Ønsker man at inddrage andre perspektiver, kan man lave et afsnit efter konklusionen, der hedder ”Perspektivering”. Dette afsnit er ikke obligatorisk, men kan bruges til at beskrive, hvilket videre arbejde rapporten danner grundlag for, hvilke syn der kunne være interessante at inddrage i andre sammenhænge, hvilke paralleller der kan drages til andre områder, osv. Perspektiveringen er en diskussion af konsekvenserne af konklusionen i et større perspektiv.