

FOG Carport

Datamatikeruddannelsen

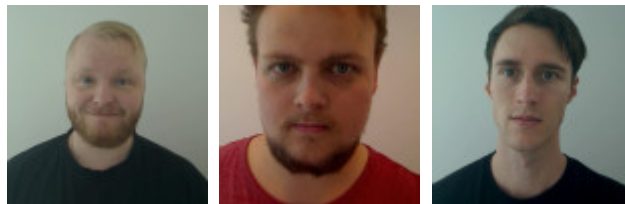
Gruppe 1, hold 20dat2e

Gruppemedlemmer:

Alex Richardt Gottorp Wagner (GH: A-R-G Wagner)
(cph-aw116@cphbusiness.dk)

Joakim Stensnæs (GH: JoakimKSS)
(cph-js437@cphbusiness.dk)

Lasse Emil Støvring Larsen (GH: Lforlasse)
(cph-ll330@cphbusiness.dk)



Copenhagen Business Academy
26. maj 2020

Abstract

This rapport contains a detailed description of how we created the requested product from FOG-Carport and the work-flow in which our group delivered this. It's a system for both customer and employee, wich takes a specific customers carport-request, configures it and sends it to a database. The employee can select a single request and the system will generate a list of part, a visual of the carport and a price-calculation as a salesprice.

Links

- **GitHub:** <https://github.com/Lforlasse/FOG-Carport>
- **JavaDocs:** [INDSÆT LINK](#)
- **Demo-video:** [INDSÆT LINK](#)
- **DigitalOcean:** [INDSÆT LINK](#)

Indhold

1	Indledning	4
1.1	Baggrund	4
1.2	Teknologivalg	5
2	Krav	6
2.1	Projekt krav	6
2.1.1	Produktkrav	6
2.1.2	Proceskrav	6
2.1.3	Dokumentationskrav	7
3	Diagrammer, Alex	8
3.1	ER diagram	8
3.1.1	Sortiment	9
3.1.2	Komponenter	9
3.1.3	Tagelementer	10
3.1.4	Dele	10
3.1.5	Lagerstatus og priser	10
3.1.6	Konfigurationer	11
3.1.7	konfigurationsdel	11
3.1.8	konfigurationsdel	11
3.1.9	Dimensioner og tillæg	12
3.1.10	Validering	12
4	Særlige forhold	13
5	Udvalgte kode eksempler	14
6	Status på implementation, Lasse	15
6.1	Projektets fremtid	15
7	Test, Alex	18
7.1	Forklaring	18
7.1.1	Siden c (calcSideC())	19
7.1.2	Siden b (calcRoofHeight())	19
7.1.3	Konvertering mellem datatyper (toInt())	19

8 Arbejdsproces, Lasse	20
8.1 Arbejdsprocesen faktuel	20
8.1.1 Mødestruktur	20
8.2 SPRINT kronologisk	23
8.2.1 Scrumreview- og Sprintmøde, Torsdag d. 23/04 og Fredag d. 24/04	23
8.2.2 Scrumreview- og Sprintmøde, Torsdag d. 30/04 og Fredag d. 1/05	24
8.3 Arbejdsprocessen reflekteret, Lasse	25
8.3.1 SCRUM som værktøj	25
9 Konklusion	27
10 Appendix	28
10.1 LOG, Lasse	28
10.1.1 Scrum møde 1, Torsdag d. 16/04-20	28
10.1.2 Scrum møde 2, Fredag d. 17/04-20	29
10.1.3 Scrum møde 3, Mandag d. 20/04-20	29
10.1.4 Scrum møde 4, Tirsdag d. 21/04-20	29
10.1.5 Scrum møde 5, Onsdag d. 22/04-20	29
10.1.6 Scrum møde 6, Torsdag d. 23/04-20	30
10.1.7 Scrum møde 7, Fredag d. 24/04-20	30
10.1.8 Scrum møde 8, Lørdag d. 25/04-20	31
10.1.9 Scrum møde 9, Mandag d. 27/04-20	31
10.1.10 Scrum møde 9, Tirsdag d. 28/04-20	31
10.1.11 Scrum møde 10, Onsdag d. 29/04-20	32
10.1.12 Scrum møde 11, Torsdag d. 30/04-20	32
10.1.13 Scrum møde 12, Fredag d. 1/05-20	32
10.1.14 Scrum møde 13, Mandag d. 4/05-20	33
10.1.15 Scrum møde 14, Tirsdag d. 5/05-20	33
10.1.16 Scrum møde 15, Onsdag d. 6/05-20	33
10.1.17 Scrum møde 16, Torsdag d. 7/05-20	34
10.1.18 Scrum møde 17, Fredag d. 8/05-20	34
10.1.19 Scrum møde 18, Tirsdag d. 12/05-20	34
10.1.20 Scrum møde 19, Onsdag d. 13/05-20	34
10.1.21 Scrum møde 20, Torsdag d. 14/05-20	35
10.1.22 Scrum møde 21, Fredag d. 15/05-20	35

Kapitel 1

Indledning

Denne rapport beskriver gruppens arbejdsproces samt produktet som FOG-carport har bestilt. Det omhandler hvorledes en kunde kan specificere dimensionerne for en karport de ønsker bygget. Systemet tager imod denne anmodning, sender den til en database, og når sælger derefter vælger konfigurationen der fra, sendes den igennem en konfigurator, som skaber et visuelt billede med SVG, en stykliste af komponenter og dele hertil, samt en pris-regulator. Sælger kan herved generere det perfekte tilbud til den enkelte kunde.

1.1 Baggrund

FOG Trælast er en virksomhed der tilbyder og formidler professionelle gør-det-selv løsninger, og som forhandler al materiale, transport m.m. til sådanne konstruktioner.

FOG Trælast tilbyder kunder at købe karporte som de selv har givet mål på. Denne gør-det-selv løsning kræver et system på deres hjemmeside, hvori formidlingen af kundens dimensioneringer kan indgå i deres database. Kunden skal have mulighed for at vælge dimensioner, materiale ud fra FOG's sortiment, samt mulighed for at vælge hvilken type tag der skal anvendes, og hældningen på dette.

FOG ønsker ligeledes mulighed for at kunne overskue de specifikke anmodninger på karporte, og derved et system til sælgeren som kan redigere i både kundens anmodning hvis der er ændringer, samt holde styr på og redigere anmodningerne generelt.

1.2 Teknologivalg

- **IDE:** IntelliJ IDEA 2019.3.3
- **Database:** MySQL 8.0.18 (2019-10-14)
- **Mock-ups:** Adobe XD
- **API:** apache-tomcat-8.5.53
- **API:** JDBC
- **Hosting:** DigitalOcean Webserver
- **Server:** Virtual Ubuntu 19.10 x64
- **Graphics:** SVG
- **Rapport:** Latex Overleaf

Kapitel 2

Krav

2.1 Projekt krav

- Der skal genereres et produkt (produktet)
- Der skal være en dokumenteret arbejdsproces (processen)
- Der skal afleveres en rapport og der skal præsenteres et projekt (dokumentationen).

2.1.1 Produktkrav

Der skal genereres og testes (deleligt) et system som møder både FOG's og FOGs kunders behov.

FOG ønsker mulighed for at en kunde kan indsende en anmodning med specifikationer af de dimensioner kunden ønsker karporten skal bygges efter i højde, længde og bredde. Der ud over skal kunden kunne til- og fravælge tag med hældning og kunne angive det antal grader hældningen skal falde med.

Der skal være et overordnet overblik over disse anmodninger, som den ansatte kan hente og redigere i. Sælger skal også kunne arbejde på en enkelt konstruktion, og der skal genereres en stykliste, et visuelt billede samt være en prisberegner, som sælger kan arbejde i.

2.1.2 Proceskrav

Projektet skal forløbe efter SCRUM modellen - opdelt i diverse Sprints. Vi beslutter os selv internt for hvordan rollen som SCRUM-master varetages. De tilknyttede lærere vil agere som product-owner i de foreliggende SPRINT review- og tekniske møder, som afholdes torsdage og fredage.

2.1.3 Dokumentationskrav

Der er krav om dokumentation i forhold til både kodning og rapport. I kodningen bruger vi Javadocs, som er linket under LINKS, øverst i rapporten. Rapporten er specificeret efter [en rapportskabelon](#), med linket:
<https://datsoftlyngby.github.io/dat2sem2020Spring/projekt/RapportSkabelonFinal.html>

Kapitel 3

Diagrammer, Alex

3.1 ER diagram

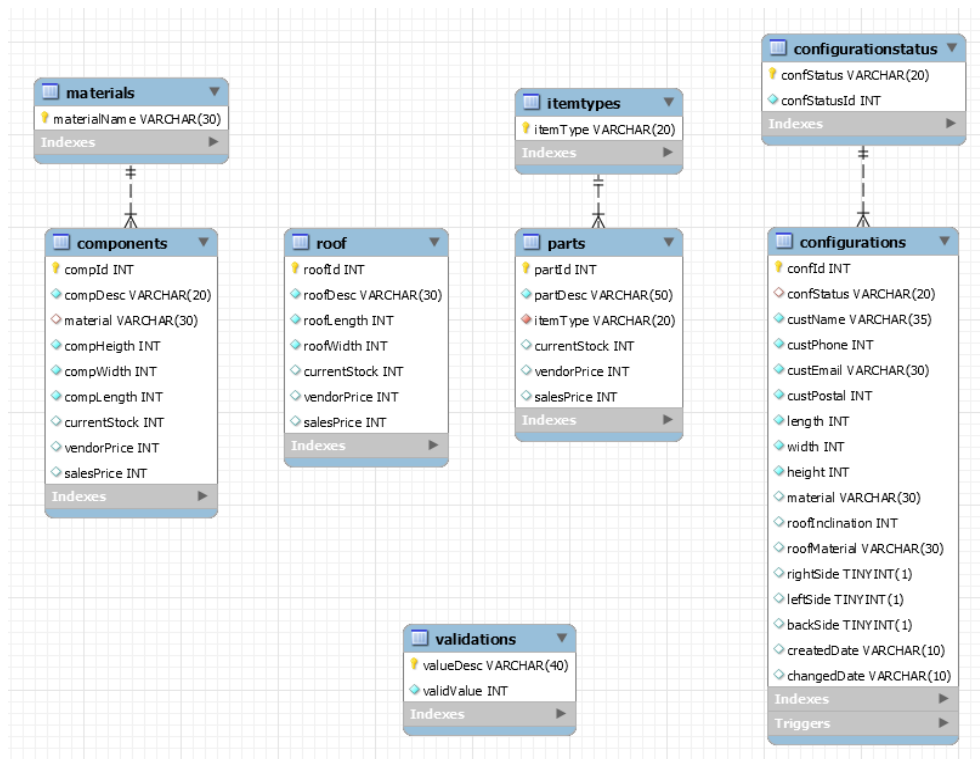


Figure 3.1: ER diagram

ER diagram forklaret

Vi har lavet en database, hvis funktion er at opbevare FOG's varesortiment og FOG's kunders carport konfigurationer. Databasen kan derved opdeles i to dele, sortiment delen og konfigurationsdelen. Vi vil i dette afsnit forklare intentionen for disse dele og hvad de indeholder. Der eksisterer en ekstra del som vi kalder validering, men den er ikke blevet aktiveret på nuværende tidspunkt.

3.1.1 Sortiment

Vores sortiment indeholder tre dele, komponentdelen som består af de forskellige brædder, stolper og andre byggematerialer som bruges til at konstruere en carport, tagdelen som består af forskellige typer tagelementer og til sidst, "deledelen" som indeholder de metalelementer som sætter komponenterne sammen. De tre kolonner (currentStock), (vendorPrice) og (salesPrice) vil blive forklaret i punkt 3.1.5.

3.1.2 Komponenter

compId	compDesc	material	compHeight	compWidth	compLength	currentStock	vendorPrice	salesPrice
2000	Stolpe	Trykimprægneret	10	10	400	74	250	475
2001	Stolpe	Egetræ	9	9	400	56	400	750
2002	Rem	Trykimprægneret	5	19	400	45	135	205
2003	Rem	Egetræ	5	18	400	4	145	215

Figur 3.2: Komponent tabel

- **compId:** Komponentens varenummer, dette er unikt og kan bruges til at identificerer en specifik komponent.
- **compDesc:** Komponentens beskrivelse, her angives komponentens type i forhold til hvad den bruges til i carporten.
- **compMaterial:** Komponentens materiale bestemmer træsorten på carporten, men hvilke 'komponentsæt' som skal benyttes til de forskellige konfigurationer. Denne kolonne har en relation til (materials) tabellen, så der ikke findes komponenter som har et ukendt materiale.
- **compHeight:** Komponentens højde dimension i centimeter.
- **compWidth:** Komponentens bredde dimension i centimeter.
- **compLength:** Komponentens længde i centimeter. Denne er sat til DEFAULT 400, da det er den ubehandlede længde på komponenten hos FOG(i vores projekt).

3.1.3 Tagelementer

roofId	roofDesc	roofLength	roofWidth	currentStock	vendorPrice	salesPrice
4000	PLASTMO	200	120	71	75	130
4001	Betontagsten, sort	42	33	224	25	45

Figur 3.3: Tagelement tabel

- **roofId:** Tagelementets varenummer, dette er unikt og kan bruges til at identificerer et specifikt tagelement.
- **roofDesc:** Tagelementets beskrivelse, her angives tagelementets type.
- **roofLength:** Tagelementets højde dimension i centimeter.
- **roofWidth:** Tagelementets bredde dimension i centimeter.

3.1.4 Dele

partId	partDesc	itemType	currentStock	vendorPrice	salesPrice
6001	Skruer 4,5 x 120mm 200stk	Pakke	120	15	45
6007	Bræddebolt 10 x 120mm	Sæt	128	10	19
6013	Bræddeplade 200 x 150 x 5mm	Styk	264	15	28

Figur 3.4: Dele tabel

- **partId:** Delens varenummer, dette er unikt og kan bruges til at identificerer en specifik del.
- **partDesc:** Delens beskrivelse, her angives beskrivende navne med hvilken slags del det er, størrelsen og antallet hvis det er en pakke.
- **itemType:** Delens enhedstype, her angives om delen fås i pakker med mange enkeltelementer, er et sæt af mindre dele eller bare eksisterer stykvis. Denne kolonne har en relation til (itemTypes) tabellen, så der ikke tilføjes enhedstyper som ikke findes.

3.1.5 Lagerstatus og priser

- **currentStock:** Lagerstatus på de individuelle varenumre.
- **vendorPrice:** Prisen på et varenummer, som FOG betaler for at få varen i kroner.
- **salesPrice:** Prisen på et varenummer, som FOG forventer at sælge for i kroner.

3.1.6 Konfigurationer

Når en kunde besøger FOG's hjemmeside, bygger en carport og bestille en anmodning om et tilbud, så gemmes denne konfiguration i databasen. For overskuelighedens skyld opdeles bestillingerne, her i forklaringen, i relevante kategorier.

3.1.7 konfigurationsdel

confId	confStatus	createdDate	changedDate
224466	Afsluttet	2020-05-24	2020-05-24
224467	Behandles	2020-05-24	2020-05-24

Figur 3.5: Konfigurationsdel tabel

- **confId:** Konfigurationens identifikations nummer. Dette er unikt og kan bruges til at identificerer en specifik bestilling.
- **confStatus:** Bestillingens status viser hvor i processen en bestilling befinder sig, fra den bliver oprettet, til den bliver betalt og afsluttet. Denne kolonne har en relation til (configurationStatus). Dette er for at holde styr på trinene i bestillingsprocessen.
- **createdDate:** Dette er et autogenerated timestamp som viser hvornår kunden oprettede bestillingen.
- **changedDate:** Dette er et autoopdaterende felt som angiver hvornår den seneste ændring til bestillingen har fundet sted.

3.1.8 konfigurationsdel

custName	custPhone	custEmail	custPostal
Abbott	44884488	abbott@bot.com	5020
Costello	22662266	costello@ost.com	2550

Figur 3.6: Kundedel tabel

- **custName:** Navnet på kunden.
- **custPhone:** Telefonnummeret på kunden.
- **custEmail:** E-mail adressen på kunden.
- **custPostal:** Postnummeret på kunden. Dette er relevant i forhold til kundens leveringsomkostninger.

3.1.9 Dimensioner og tillæg

length	width	height	material	rightSide	leftSide	backSide	roofInclination	roofMaterial
300	580	225	Egetræ	1	1	1	0	PLASTMO
300	450	225	Trykimprægneret	0	1	1	0	Betontagsten, sort

Figur 3.7: Dimensioner og tillæg tabel

- **length:** Længden på carporten i centimeter.
- **width:** Bredden på carporten i centimeter.
- **height:** Højden på carporten uden tag i centimeter.
- **material:** Materiale typen på carporten.
- **rightSide:** Her indikeres om der skal en væg på siden af carporten.
- **leftSide:** Her indikeres om der skal en væg på siden af carporten.
- **backSide:** Her indikeres om der skal en væg på bagsiden af carporten.
- **roogInclination:** Her indikeres om taget på carporten skal have rejsning, målt i grader.
- **roofMaterial:** Tagbelægningstypen.

3.1.10 Validering

Vi havde en ambition om at indføre validering på mange af carporten, kundens og materialernes størrelser og forhold, således at når en konfiguration overtrådte disse valideringskriterier, så vil det genererer en advarsel eller notits hos sælger.

valueDesc	validValue
Carspace height	185
Carspace length	300
Carspace width	250

Figur 3.8: Validering tabel

- **valueDesc:** Hvis vi ønsker et udfald på en validering, eksempelvis i forhold til højde, så skrives det her.
- **validValue:** Her indikeres grænseværdien for hvornår valideringen skal træde i kraft.

Kapitel 4

Særlige forhold

SÆT KODEEKSEMPLER HER, FØLG ENKELT ANMODNING

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 hvordan 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.

Kapitel 5

Udvalgte kode eksempler

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

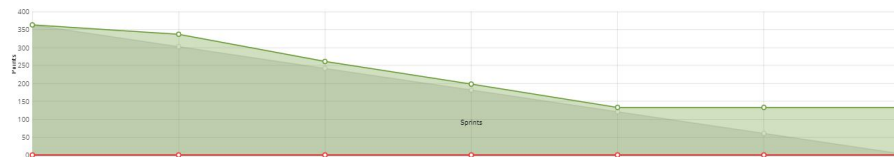
De eksempler der er givet under “særlige forhold” afsnittet kan man godt tage og illustrere med kode direkte i rapporten.

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 skærbilleder 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

Kapitel 6

Status på implementation, Lasse



Figur 6.1: Sprint forløb

Som det kan ses af figur 6.1 er vi som udgangspunkt nået i mål med vores plan for projektet, som sluttede efter vores 4. sprint forløb. PO kan modtage anmodninger fra en kunde, med specifikke dimensioner til en karport, samt valg af materialer. Konfigurationen er automatiseret, så den skaber en stykliste, plantegning og prisberegner anmodningen ud fra de nødvendige materialer samt dele til disse dimensioner.

Anmodningen ligger i deres database, og kan derfra behandles af sælger, som også kan ændre status for hvor i behandlingsprocessen anmodningen er nået til.

6.1 Projektets fremtid

Projektet har flere funktionaliteter planlagt for fremtiden. Efter aftale med PO, gav det bedst mening for projektet, at vi kunne få en simpel karport anmodning sendt hele vejen igennem system, inden vi tilføjede flere valgmuligheder. Havde forløbet været forlænget ville PO have haft mulighed for at vælge i mellem flere forskellige userstories.

På figur 6.2 (næste side), kan disse muligheder ses og herefter bliver de uddybet.

Votes		User Stories	Status	Points
<input type="checkbox"/>	▲ 0	Important #65 Carport konfigurator: Skur	New	34
<input type="checkbox"/>	▲ 0	Important #4 Carport konfigurator: Materiale valg	New	5
<input type="checkbox"/>	▲ 0	secondary #69 Bestil carport: Leveringsomkostninger	New	5
<input type="checkbox"/>	▲ 0	extra #43 Carport konfigurator: Advarsler	New	21
<input type="checkbox"/>	▲ 0	extra #67 Ansæt funktionalitet: Tilbud	New	21
<input type="checkbox"/>	▲ 0	extra #68 Ansæt funktionalitet: Planlagt tilbud	New	13
<input type="checkbox"/>	▲ 0	extra #146 JavaScript til visualisering af plantegning for kunde i anmodning	New	34

(a) Userstories
(b) Estimerede point

Figur 6.2: Figurer for ekstra userstories

Figur 6.2 viser fremtidsplanerne for projektet. Disse er ikke færdigt arbejdet, og ville som selvfølge undergå en revurdering inden disse skulle fremlægges for PO i et sprintmøde.

- **Nr 65, Carport konfigurator: Skur:** Denne userstory skal give kunden mulighed for at tilvælge et skur til karporten. Efter en revurdering af selve userstoryen, skal der skrives endnu en userstory, som beskriver kundens mulighed for at vælge skurets dimensioner, og en beregner som kan fortælle kunden hvorvidt skuret bliver for stort.

Dette ville kræve tilføjelser og ændringer i både Databasen, funktionslaget og Front-end. Derfor ville denne userstory blive en del større end estimeret.

- **Nr 4, Carport konfigurator: Materiale valg:** Ideen bag denne userstory bunder ud i, at vi havde ideen om at lave system, så man kunne vælge at konfigurere sin egen carport eller vælge færdiglavede konfigurationer.

Funktionaliteten til at vælge materialer og træsort er implementeret, så kunden kan vælge fra et sortiment, når han konfigurerer en specifik karport. Derfor kan denne userstory kan ses, som et levn fra fortiden.

- **Nr 69, Bestil Carport: Leveringsomkostninger:** Som en del af omkostningerne for kunden, er transport til hele landet en del af prisen. Derfor kunne dette tilvælges som en funktionalitet til hjemmesiden, der ville kunne udregne leveringsomkostningerne ud fra kundens indtastede postnummer. Den skulle også implementeres på ordrebeskrivelsen. Denne pris ville indgå i den totale pris for karporten.

- **Carport konfigurator: Advarsler:** Hvis kunden valgte funktionaliteter som kunne være i strid mod lovgivningen af konstruktionen, skulle han modtage en advarsel om dette.

Ligeledes hvis kunden gav et modstridigt input til skurets og karportens dimensioner, skulle han modtage en advarsel. Sidst skulle det noteres i anmodningen, hvis den var blevet gennemført trods advarsler.

- **Nr 67, Ansat funktionalitet: Tilbud:** Hjemmesiden skal vise tilbud på diverse carporte i perioder. Dette kræver at siden er udstyret med færdiglavede karporte, som er en userstory for en anden gang.

Den ansatte kan sætte en karport i 'tilbud' status. Prisen kan reguleres med en procentvis på den totale pris. Hjemmesiden skal genkende varen som værende på tilbud. Selve tilbuddet kan ikke overstige procent eller være negativ.

- **Nr 68, Ansat funktionalitet: Planlagt tilbud:** Det ville indebære at den ansatte kunne sætte en karport som tilbud på et specifikt tidspunkt, at han kunne vælge perioden tilbuddet skal være aktivt, og at siden vil advare ham hvis der allerede er et planlagt tilbud for den valgte karport.

- **Nr 146, JavaScript til visualisering af plantegning for kunde i anmodning:** Her er tanken, at kunden kan se en visualisering af de mål han indtaster til en karport. Tegningen skal ændre sig samtidigt med at han aktivt ændrer dimensionerne på i konfiguratoren.

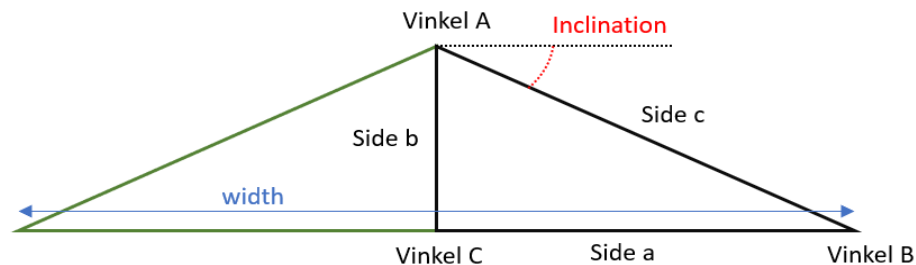
Kapitel 7

Test, Alex

Vi har ikke lavet mange tests i IntelliJ. Vi har i stedet benyttet os af at udvikle metoderne i et separat program, for så at implementere dem direkte i klasserne når metoderne er klar, tilpasse vores variabel-navne til vores projekt og så teste om de virker ved at køre programmet. Dette var let og fantastisk hurtigt, men ikke 'best practice' og kan give problemer i større, mere sofistikerede projekter, så vi ønsker at implementere unit testing mere i fremtiden.

Vi har dog valgt at køre et par unit tests på vores Roof klasse, da denne indeholder nogle ret ligetil, matematiske metoder.

7.1 Forklaring



Figur 7.1

Som det ses på figur 7.1 ovenfor, så indeholder et tag mange mål. Når en kunde bygger en carport med tagrejsning, så vælger de kun hældningsgraden, hvilket vi kalder inclination. Denne grad, sammen med carportens bredde, som vi her kalder width, indføres i visse metoder som giver os de resterende mål som vi har brug for til vores stykliste.

7.1.1 Siden c (calcSideC())

Siden c eller hypotenusen i vores trekant er det første vi udregner. Udover at give et godt udgangspunkt for senere at finde siden b, tagets højde, så er det en vigtig dimension, både for vores komponenter som bruges til taget eller ligger langs taget, men også direkte for vores tagbelægning.

Der er foretaget test af maksimum og minimum værdierne for inclinationen, samt en tilfældig inclination's værdi indenfor disse.

7.1.2 Siden b (calcRoofHeight())

Siden b fungerer som tagets højde, som går fra carportens top til tagets top. Denne dimension er vigtig for vores lodrette midterspær og for beklædningen af tagets lodrette sider.

Der er foretaget test af maksimum og minimum værdierne for side c, samt en tilfældig side c værdi indenfor disse.

7.1.3 Konvertering mellem datatyper (toInt())

Vores datatyper for tal i vores projekt er altid integers, men mange af det Math. funktioner som java har, kræver double værdier (da der er behov for decimaler). Dette betyder at vi i vores to forrige metoder har været nød til at konvertere vores integers til double værdier, hvilket er let nok, men når disse værdier skal genintroduces til programmet løber man ind i nogle problemer. Derfor har vi lavet en metode som tager imod en double værdi og returner den som nærmeste integer.

Der er foretaget tests af værdier som rundes ned og af værdier som rundes op.

Kapitel 8

Arbejdsproces, Lasse

8.1 Arbejdsprocesen faktisk

Gruppe 1, er en gruppe bestående af 3 medlemmer. Vi mistede vores fjerde medlem dagen efter projektets start, og en af de resterende tre, manglede sin arbejdsstation i 2 uger. Men vi har formået er nå vores målsætninger.

8.1.1 Mødestruktur

Dagligt SCRUM møde

Der afholdes et dagligt SCRUM-møde internt i gruppen, hvor weekender er fritaget medmindre andet er aftalt. Mødetiden er klokken 10 om formiddagen, og denne tid kan også rykkes efter intern aftale. Disse møder færdiggøres først når en dagsorden er gennemarbejdet og debatteret.

Til mødet skal der, som mindste krav, være gennemgået en dagsorden som indeholder hvad der var aftalt op til mødets begyndelse samt hvad der arbejdes videre på til mødet dagen efter.

Der er udarbejdet en skabelon over en fast dagsorden, se figur 8.1. De vigtigste pointer bliver der lavet referat på i form af en ”¹LOG”. Der skal være skrevet LOG til alle arbejdsdage, og disse skrives ind i rapportens bilag under LOG.

Dagligt møde:
(forventet tid 5-15 minutter):

Dirigent:
Referent:

Punkt 1 – Opsummering

Hvert medlem gennemgår:

- Jeg har arbejdet på userstory/task...
- Jeg har nået/ikke nået det jeg ville...
- Jeg har haft succes/problemer med...

Punkt 2 – Opgaven fortsat

Hvert medlem gennemgår:

- Jeg skal arbejde på userstory/task...
- Jeg har brug for...
- Jeg forventer at nå...

Punkt 3 – Næste møde

- Vores næste møde er...

Figur 8.1: Daglige mødeskabelon

¹LOG - Bilag: side X.

Hver dag skal der vælges en referent og en dirigent. Vi valgte fra start, at uddele disse roller som en fast titel.

Referenten har været Lasse, som har ført LOG under hele forløbet og til alle møder. Dirigenten har været Alex, som med struktur og en konstruktiv tilgang har ledet møderne dagligt og ugentligt. Joakim blev tildelt rollen som agerende SCRUM-Master under hele projektet, og sørgede for vi holdte overblikket. På denne måde kunne vi overholde vores tidsplan og nemt kunne revurdere arbejdsvilkårene ved at alle havde et individuelt ansvar for det fælles mål.

Ugentligt SPRINT retrospective møde

Der afholdes et ugentligt SPRINT-møde internt i gruppen efter Review- og Sprintmøder med ²PO, hvor tidslinjen og arbejdsbyrden debatteres. Under mødet sidder alle medlemmer med ³Backloggen åben i [taiga.io](https://tree.taiga.io/).

Her fremlægges problemstillinger, og der revurderes, på baggrund af PO's anvisninger, om der er user-stories som er estimeret højere end forventet og derfor skal om-lægges.

Det er ved brug af vores ugentligt Sprintmøde-skabelon, at vi får planlagt hensigtsmæssigt efter Sprint- og Planningmøderne, se figur 8.2 (næste side).

Processen af arbejdet er tilrettelagt efter Backloggen. Ideen er, at vi får løst fremgangsmetoden for vores arbejdsproces ved hjælp af Scrum- og Sprint-metoderne. Dernæst bygger vi projektet op gennem de modelværktøjer vi kender som: ER-diagram, Navigationsdiagram, ⁴Domænemodel, ⁵klassediagram og ⁶aktivitetsdiagrammer, inden vi påbegynder det konkrete kodningsarbejde.

ER- og Navigationsdiagrammet er beskrevet i deres respektive kapitler, resten modellerne kan findes i bilag.

²PO: Product Owner

³Backlog: <https://tree.taiga.io/project/joakimkss-fog-carport/timeline>

⁴Domænemodel: Bilag: Side X

⁵Klassediagram: Bilag: Side X

⁶Aktivitetsdiagram: Bilag: Side X

Sprint møde:
(forventet tid, 45-90 minutter)

Dirigent:
Referent:

Punkt 1 – Opsummering

Er alle usecases klaret?
Hvis en usecase ikke er klaret, hvorfor og hvad nu?
Er der opstået nogen issues i forrige sprint?
Er disse issues blevet løst? Eventuel uddybning.
Er der opstået nye tasks i løbet af sprintet?
Er disse tasks blevet løst? Eventuel uddybning.

Punkt 2 – Aktivitet

Har vi været aktive nok?
Har arbejdspresset været for stort, for småt eller tilpas?
Skal nogen opgaver omfordeles?

Punkt 3 – Samarbejde

Hvordan går samarbejdet?
Bruger vi for meget, for lidt eller tilpas tid på vores møder og generelle samarbejde?
Er vi gode nok til at kommunikere med hinanden?

Punkt 4 – Ny sprint

Hurtig gennemgang af planlagte userstories og tasks.
Redefinering af uopklarede userstories eller tasks.
Introduktion af identificerede userstories eller tasks.
Gennemgang af arbejdskronologi og uddeling af opgaver/ansvar.

Punkt 5 – Diverse

Yderligere beretninger eller bemærkninger.
Har nogen særlige planer eller aktiviteter i løbet af sprintet?

Figur 8.2: Ugentligt Sprintmøde-skabelon

Som det kan ses af figur 8.2 kommer vi ind på og gennemgår ugens forløb på fornuftig og struktureret vis. Med opsummering og revurdering kan vi overholde tidsplanen i Backloggen efter aftale med PO. Der tages også højde for privatlivet på sidelinjen i skabelonens sidste punkt: Diverse.

Disse møder blev afholdt enten lige efter et møde med PO eller fredag morgen klokken 10, før det daglige SCRUM-møde.

Userstory	Estimerede point	Total point	Sprint forløb
96: Hjemmeside	13		1
1: Karport konfigurator, front-end	13	26	1
36: Bestil Karport konfigureret	21		2
92: Bestil karport: Stykliste	55	76	2
58: Anmodninger: Oversigt	21		3
62: Anmodninger: Status	8		3
91: Bestil karport: Plantegning (funktionalitet og logik)	34	63	3
143: Bestil karport: Plantegning (Front-end)	21		4
2: Karport konfigurator: Vægge og tegrejsning	34		4
49: Ansat funktionalitet: Ændring af materialer	5		4
53: Ansat funktionalitet: Ændring af materialer	5	65	4

Tabel 8.1: Sprint oversigt

8.2 SPRINT kronologisk

Opgaven som SCRUM-master var tildelt Joakim, som håndterede rollen ved altid at sørge for, at der blev skrevet relevante subtasks til de userstories vi arbejdede på hver især. Vi sad altid i plenum og debatterede hvorledes en user-story, samt subtask til disse, bedst kunne løses, så det stemte overens med vores tidsplan. Sprint oversigten kan ses via tabel 8.1.

8.2.1 Scrumreview- og Sprintmøde, Torsdag d. 23/04 og Fredag d. 24/04

Vi blev bekræftet i hvordan vi skulle forholde os til værktøjet og arbejdsprocessen. Vi forstod hvordan det vil sige at planlægge på baggrund af de valg, som bliver taget af PO for projektets retning, og hvordan vi spiller en rolle, når der skal argumenteres for hvilke valg der har en effekt på resultatet.

Vi fremlagde til mødet hvordan vi havde opbygget backloggen. Der blev fremvist et udkast på en domænemodel og et navigationsdiagram, og i samarbejde med PO fik vi sat en retning på disse tanker. Som det fremgår af LOG til disse to ⁷møder, fremlagde vi en brugerflade med placeholder data, som fungerer visuelt og kunne give PO en ide om hvordan der blev navigeret i programmet, samt hvad han kunne beslutte sig til af funktionalitet. PO var tilfreds med de tanker vi havde gjort os, og bad os prioritere 2 userstories: Nr. 36, Bestil karport: Konfigureret og Nr. 92, Bestil karport: Stykliste.

Efter mødet fredag, var der på PO's anvisninger udarbejdet en prioritering af de forskellige userstories, som vi havde fremlagt sammen med visualiseringen af programmet. Derved kunne vi planlægge den konkrete kodning, som havde været skubbet indtil bekræftelsen af projektets retning. Her kunne vi opdatere backloggen og revurdere point-estimationen af userstoriesne.

⁷Bilag: side X

8.2.2 Scrumreview- og Sprintmøde, Torsdag d. 30/04 og Fredag d. 1/05

Til forskel fra sidste møde med PO, havde vi lært hvordan vi skulle forholde os til SCRUM værktøjet. Derfor var der til dette møde en tilfredsstillende fremlæggelse af de userstories vi havde arbejdet på, med en kort demo af den funktionalitet vi havde færdiggjort. PO så sammenkoblingen mellem logikken i koden til styklisten og hvordan det blev hentet fra databasen og vist på front-end, som det fremgår af ⁸LOG.

Vi nåede i mål med de opgaver der var aftalt med PO, dog måtte vi revurdere arbejdsbyrden, da vi havde bidt over lidt for meget at løse på en uges arbejde. Derved lærte vi at vurdere point-estimeringen af userstories korrekt, og kunne herfra argumentere for hvad der var vigtigst at få løst på fornuftig vis.

Endnu engang blev backloggen omstruktureret ud fra vores erfaringer i samarbejde med PO, og vi fandt den rigtige mængde arbejde for en uge i de kommende userstories: Nr. 58, Anmodninger: Oversigt, Nr. 62, Anmodninger: Status og Nr. 91, Bestil karport: Plantegning (funktionalitet og logik).

Sprint forløb efterfølgende

Som det fremgår af de to ovenstående eksempler, blev sprintprocessen langt mere trivielt at arbejde med efter vi havde fået bekræftet hvordan vi håndterede userstories og hvordan vi estimerede dem korrekt. I afsnittet "Arbejdsprocessen reflekteret" er det beskrevet hvilke overvejelser det medbragte i vores arbejdsproces.

Vi havde nu styr på hvordan vi skulle forholde os til værktøjet, og kunne effektivt fremlægge vores argumentation for PO i de kommende møder. Dette resulterede i en flydende arbejdsproces og god forståelse efter det andet sprintmøde og til projektets afslutning.

⁸Bilag: Side X

8.3 Arbejdsprocessen reflekteret, Lasse

8.3.1 SCRUM som værktøj

Vi håndterede fra start opgaven ved at tage ejerskab for hver vores ansvarområde (vores titler: Dirigent, referent og SCRUM-master), samt at have indblik i hvad vi arbejdede på hver især.

Joakim sørgede for altid at have backloggen opdateret, og førte ordet i de møder vi havde med PO, samt de forskellige sprintmøder intern efterfølgende. Joakims rolle bestod af at opdatere backloggen og de forskellige sprints i forhold til de aftaler der var indgået med PO.

På denne måde havde vi alle en aktiv rolle i beslutningsprocessen, hvilket resulterede i, at alle følte de kunne komme til orde, og at arbejdsflowet var vel-fungerende. Den største 'udfordring' vi så i at en enkelt mand stod med hele ansvaret som scrum-master var, at retningen for opgaven kunne blive farvet af dette. Derfor blev det hurtigt indforstået, at dette ansvar, var en beslutning som skulle tages i plenum - så beslutningerne var fælles.

Det at skrive en god userstory

Det var på forhånd indforstået i gruppen, at der var et fælles ønske om at få planlagt vores arbejdsproces grundigt inden det egentlige kodningsarbejde. Der-ved kunne alle være med fra start og fornemme en retning for projektet. Det betød at vi tog godt imod SCRUM-værktøjet, som er en god måde at tage en ide (ordre) og inddele den i delmål og arbejdsprocesser - altså en planlægning af arbejdet.

Vi var usikre på hvordan vi skulle gribe SCRUM an. Derfor blev vi enige om, at bruge første sprint på at skrive en masse userstories, samt lave en visuel præsentation til PO i form af en demo og forskellige modeller. Visualiseringen skulle være en model som kunne tage imod de userstories PO vælger at priorisere, og dette var den rigtige måde at gribe processen an på. Dog var vi end ikke sikre på hvordan vi skulle estimere point til de enkelte userstories.

Efter første sprintmøde med PO, havde vi også lært hvad det vil sige at skrive en god userstory. Vi havde ikke helt forstået at skrive nogle optimale userstories. De var for store, en episk, og greb om flere opgaver på en gang. Vi måtte genskrive et par af dem for at kunne få valgt de rigtige til den kommende uges sprint.

Først efter andet sprintmøde, hvor vi havde prøvet at håndtere arbejdsbyrden aftalt med PO, kunne vi estimere pointgivningen korrekt. Vi havde gæbt over for meget arbejde for en enkelt uge med kun tre gruppemedlemmer. Vi havde arbejde fra tidlig morgen til sen aften for at løse de valgte userstories, og dette viste sig ikke at være en optimal arbejdsmodel. Derved havde vi fået en fornemmelse for hvor meget en estimering af enkelte opgaver betyder.

Herfra var det ikke svært at håndtere backloggen, og SCRUM som værktøj. Vi havde fået bekræftet hvad det vil sige at skrive en god userstory og hvordan vi skal estimere point hensigtsmæssigt dertil. Arbejdsprocessen var noget nemmere efter at have forstået og debatteret effekten af disse situationer i gruppen, og vi fandt det meget ligetil at være produktive/effektive herfra.

Generel planlægning

Som gruppe har vi fra dag et været struktureret. Første dag gik på at oprette projektet i de forskellige værktøjer: GitHub, taiga.io og Overleaf. Det handlede om at identificere hvad der skulle bruges til at holde overblikket i projektet, og skabe en daglig struktur for hvordan vi skulle håndtere de forskellige værktøjer.

Som det fremgår af ⁹LOG, kan det ses hvordan vi bruger de første par dage på at oprette strukturer for vores arbejdsprocess, og først på tredje dagen igangsætter vi SCRUM som værktøj. Ideen er at de strukturer vi beslutter fra start er gennemgående og behandler bl.a. brugen af scrum og nedskrivning af daglig LOG. Dette viste sig at være en god og konstruktiv måde at håndtere nye værktøjer på, da disse dagligt bliver revurderet til scrum mødet.

Kort om Corona-pandemien

Corona satte sit pres på hele verdenen lige før projektets start. Vi blev hurtigt enige i gruppen om, at det daglige fremmøde var det absolut vigtigste for vores projekts resultat. Derfor aftalte vi en fast mødetid klokken 10 hver morgen til scrum-mødet. Dette sammen med tilgængelighed og en stærk kommunikation gennem programmet Discord, gjorde at vi kunne fastholde en stærk arbejdsmoral og godt teamwork. Sidst men ikke mindst var det vigtigt at få skrevet en daglig LOG af arbejdet, så vi kunne dokumentere vores beslutningstagning.

⁹Bilag, LOG: Scrum møde: 1,2,3.

Kapitel 9

Konklusion

Kapitel 10

Appendix

10.1 LOG, Lasse

Der afholdes et daglig SCRUM-møde, hvor weekender er fritaget medmindre andet er aftalt. Mødetiden er klokken 10 om formiddagen og mødet færdiggøres når en dagsorden er gennemarbejdet og debatteret. Til mødet skal der være gennemgået - som minimum - hvad der var aftalt op til mødets begyndelse samt hvad der arbejdes på til næste møde. Dette føres der referat på, og der bliver skrevet en daglig log af møderne ind i rapporten.

Der afholdes et ugentligt SPRINT-møde, hvor tidslinjen og arbejdsbyrden debatteres. Under mødet sidder alle medlemmer med Backloggen åben i taiga.io. Her fremstilles problemstillinger, og der revurderes om der er user-stories, som er større end forventet og derfor skal om-lægges.

Efter Sprint- og Planningmeetings gennemgås en ugentligt mødeskabelon, som skal gøre arbejdsprocessen let forståelig.

Processen af arbejdet er tilrettelagt efter Backloggen. Ideen er, at vi får løst fremgangsmetoden for vores arbejdsprocess ved hjælp af Scrum- og Sprint-metoderne. Dernæst bygger vi projektet op gennem de modelværktøjer vi kender som: Domænemodel, klassediagram, MED MERE!!!!!!!!!!!!!! inden vi påbegynder det konkrete kodningsarbejde.

Skal vi skrive om vi tester med Unittests inden vi koder??

10.1.1 Scrum møde 1, Torsdag d. 16/04-20

Første Scrum møde var en start af hele processen. Vi oprettede Github repository på Lasses Github: <https://github.com/Lforlasse/FOG-Carport> med regler for commits samt branches til hver enkelt medlem i gruppen.

Der blev oprettet taiga.io på Joakims profil med start af backloggen til fremtidige Sprints- og Scrummøder:

<https://tree.taiga.io/project/joakimkss-fog-carport/timeline>

Sidst blev der skabt en skabelon til både rapportskrivningen i Overleaf samt en fremgangsmetode til de daglige Scrum- og ugentligt Sprintmøder.

10.1.2 Scrum møde 2, Fredag d. 17/04-20

Andet Scrum møde var en procesdiskussion, hvor vi debatterede hvilken retning projektet skulle have. Vi har blandt andet sat estimation på vores user-stories, samt rangerede vigtigheden af hver enkelt user-story. Der er blevet aftalt opgaver til weekenden, og her afholder vi ikke Scrum møder.

10.1.3 Scrum møde 3, Mandag d. 20/04-20

Tredje Scrum møde er første gang vi rigtigt bruger Scrum metoden. Database strukturen er igangsat, og der vurderes om der skal være et opdigtet lagersystem med indkøbspris samt lagerstatus.

Der er oprettet et mavenprojekt ud fra skabelonen. Der skal rettes til, men der er en færdig skabelon som er pushet til GitHub. Der er skrevet en ny User-story til mock-ups og hjemmeside layoutet, og der er skabt mock-ups i HTML.

Vi besluttede, at det ville være en god ide at komme igang med at få lavet nogle domænemodeller, så der kan skabes et overblik af de konkrete opgaver. Sidst er der rettet til i rapportskabelonen.

10.1.4 Scrum møde 4, Tirsdag d. 21/04-20

Databasestrukturen er blevet færdigarbejdet. Der er lavet et udkast til et EER-diagram over databasen, som bliver diskuteret sammen med spørgsmålet om klasse diagram. Ligeledes er navigationen i HTML'en færdig, der er en smule problemer med noget af CSS'en, men det bliver der rettet op på i dag.

Domænemodellen bliver revurderet i dag, og der arbejdes videre på User-story-1 med disse modeller, så vi kan blive klar til at kode.

10.1.5 Scrum møde 5, Onsdag d. 22/04-20

User-tasken med udfyldelse af databasen er 80 procent færdigt. Klasserne i projektet er halvt færdigt. Disse bliver færdige i dag. De sidste navigationelle funktionaliteter er blevet udarbejdet i HTML'en. Vi klargør til første Sprint-møde i morgen.

10.1.6 Scrum møde 6, Torsdag d. 23/04-20

Subtasks i den nye User story er blevet omformuleret en smule. Der vurderes at den skal have nogle færre point i backloggen, og at den er tæt ved færdig. Hjemmesiden er færdigopsat med JSTL, så den forbinder korrekt og henter data.

Logicfacaden i projektet er fyldt med placeholder data ind til vi når til de user-stories.

Klasserne: carport, komponent og order-class er påbegyndt, men viser sig også som en større opgave end forventet. Der er lidt problemer med indførelsen af visse metoder, da vi skal bruge databasemetoderne.

Scrum Review møde 1, Torsdag d. 23/04-20

Første Sprint møde med produkt-owner var en god bekræftelse af, at vi har den rigtige retning med projektet.

Vi præsenterede en brugerflade med placeholder-data, som fungerer visuelt, og som snart er klar med den gældende funktionalitet i sammensplit med databasen. Det er forbindelsen mellem databasen og funktionaliteten som mangler, men vi er stort set i mål.

Grundet vores underbemanding, er den konkrete kodning skubbet en uge i backloggen. Vi ser ikke grund til bekymring for projektet, da vores for-arbejde er rigtig godt.

10.1.7 Scrum møde 7, Fredag d. 24/04-20

SCRUM blev i dag først afholdt efter planningmeeting, som var klokken 13:45.

Backloggen samt Userstories er nu faldet på plads efter vores første planningmeeting. Vi er alle stadig indforstået med at processen kan variere fremadrettet yderligere.

Der skal i dag ses på klasserne og den aktuelle kodning af disse. Vores planlægning heraf er veldokumenteret og vi er forberedt. Der bliver også bygget videre på databasen, så der er en tabel som kan modtage vores carport-konfigurationer, samt en tilknyttet status tabel.

Sprint møde 1, Fredag d. 24/04-20

Der blev i samarbejde med product-owner udarbejdet en prioritering blandt de vigtigste userstories. Der blev sat fokus på 2 userstories: Nr. 36 Bestil carport: Konfigureret og Nr. 92 Bestil carport: Stykliste.

Efter vores genovervejelse mener vi, at vi fik nået de mål til Userstories som var relevante. Det blev accepteret at vi bruger en userstory på at dokumentere vores arbejdsproces i en log-fil til rapporten.

Samarbejdet internt i gruppen fungerer godt, kommunikationen og professionalisme spiller en vigtig rolle, som vi varetager. Vi er blevet enige om at bruge et medlem fra gruppen som "Key-holder- altså en mand, som holder- og skaber overblik over en specifik userstory.

Vi har klarlagt en process for disse og fundet frem til de sub-tasks, som forventes færdige i slutningen af dette sprint 2. gennemgået de to vigtigste med productowner

10.1.8 Scrum møde 8, Lørdag d. 25/04-20

Til de forrige møder har sidste punkt på dagsorden: ”andet”, været fyldt af, at vi manglede en arbejdsstation til et teammedlem. Dette er løst nu og vi har planlagt hvad det kommer til at betyde for det kommende sprint med start på mandag.

10.1.9 Scrum møde 9, Mandag d. 27/04-20

Der er blevet tilføjet persondata til konfiguratoren, så en enkeltperson skal indtaste nogle basale oplysninger når han laver en anmodning om et tilbud, og sælger kan kontakte ham med et tilbud efterfølgende.

I denne uge starter sprint 2, og vi har planlagt at få arbejdet på de 2 vigtigste userstories: Bestil carport: Konfigureret, og Bestil carport: Stykliste, så de kan blive færdige til næste sprint møde med product owner.

10.1.10 Scrum møde 9, Tirsdag d. 28/04-20

Efter mødet i går ramte vi ind i et problem med Github og hvordan den trackede vores filer. Det tog en halv arbejdsdag at få vores arbejdsprocess gennem GH klar igen. Så det kostede os noget tid, men var en nødvendighed for at kunne arbejde hver for sig i denne corona-tid.

Der blev oprettet klasser, samt påbegyndt den konkrete kodning til styklisten. Vi har en god start og sætter 2 mand på dette arbejde. Det skal bygges op korrekt, så vi kan holde farten i arbejdet.

Sidste mand tager fat i arbejdet med DBAcces, og hvordan databasen kan tage imod et objekt. Dette skal klares snarest, så vi kan få et produkt stablet på benene.

10.1.11 Scrum møde 10, Onsdag d. 29/04-20

Funktionlayer og databasen er blevet koblet sammen, så der ligger en konfiguration i databasen samt at der bliver sendt et ID på denne tilbage. Der skal implementeres maks og minimumsværdier i databasen og der skal laves nogle DB-Acces metoder til materialevalgets sammenkobling også.

Strukturen på hvordan styklisten skal bygges op er blevet valgt og påbegyndt. Der skal refactors nogle ting, og når vi kører programmet får vi ikke de ting vi vil, da der mangler noget DB-acces formidling. Dette bliver afklaret snarest muligt. Der er blevet skrevet en del metoder med funktionaliteten, men mangler stadig de sidste.

Der skal i dag arbejdes på flere metoder og på forbindelsen mellem databasen og systemet(styklisten).

Teknisk møde 1, Onsdag d. 29/04-20

Vi fik fremlagt vores systemkoncept tilfredsstillende for Arne, som ikke havde indvendinger. Vi fremlagde ydermere vores diagrammer, og fik afklaret vores spørgsmål om mock-ups.

10.1.12 Scrum møde 11, Torsdag d. 30/04-20

Frontend til anmodningsoversigt, og individuelle anmodningssider er klar til at få indført data på siden fra funktionaliteten i styklisten. Der er en smule refactoring-arbejde til disse sider, men der udover er det i mål.

Styklistens userstory er færdig i logikken og kodning. Der mangler test i form af indsættelse i front-end og databasen. Derefter kan vi afslutte ugens sprint. En ekstra ting er refactoring af navngivningen.

Scrum Review møde 2, Torsdag d. 30/04-20

Vi fremlagde vores funktionalitet og arbejdsproces for productowner tilfredsstillende. Der var ikke de store pointers, og productowner ser frem til i morgen, så han kan se sammenkoblingen mellem logikken i koden til styklisten med databasen og front-end.

10.1.13 Scrum møde 12, Fredag d. 1/05-20

Userstories fra sprint 2, med styklisten og den byggede simple carport er færdige. Logikken og implementationen er i mål. Product owner var godt tilfreds med arbejdet og produktet. Vi fik svar på vores issues dertil, og er klar til at gribe næste sprints opgaver an.

Dagen i dag omhandler forberedelse af materiale til et planlagt møde klokken 12:00. Her afholdes sprint mødet. Efter forberedelsen står den på weekend.

Sprint møde 2, Fredag d. 1/05-20

Ugens userstories er udført og gennemført. Vi er nået i mål med funktionaliteten og implementeringen for dette sprint. Der var kun det ekstra, at vi måtte refactor vores kode.

Vi fik lavet hvad vi havde planlagt, dog revurderer vi, at vi skal bide over mindre ting af gangen til fremtidige sprints. Derfor har vi omstruktureret i samarbejde med productowner, så vi for en mere flexibel og flydende arbejdsproces.

I det kommende sprint har vi planlagt processen, og prioriteret de vigtigste userstories samt subtask deri. Der bliver introduceret SVG i dette sprint. Derudover skal vi tage stilling til sortering i anmodningerne, hvorvidt det skal være på dato eller ordrenummer, men dette er en mindre problemstilling. Derfor kan vi konkludere, at vi er godt på vej med projektet.

10.1.14 Scrum møde 13, Mandag d. 4/05-20

Opstart i dag står på at få en plantegning lavet, og derefter skabe logikken til implementeringen af det. Vi skal også have opdateret logikken og frontend til bestillingers status. Det kræver også en implementering af timestamp.

10.1.15 Scrum møde 14, Tirsdag d. 5/05-20

Der er lavet metoder til at finde den del, som skal tegnes i plantegningen. I dag skal der laves logik til at placere dem korrekt.

Dernæst er der kommet status tekst på anmodninger, denne skal lige rettes lidt til i koden. Front-end er blevet opdateret til dette.

10.1.16 Scrum møde 15, Onsdag d. 6/05-20

Metoderne til logikken om placeringen af de forskellige elementer i plantegning er næsten færdigt. Der mangler de sidste placeElement metoder, og derefter skal vi finpudse metoderne.

Der er blevet refactoret en del af koden. Userstory 62 er færdiggjort og implementeret. Der mangler noget sortering efter tid, og der er lavet et issue ift. change date cellen.

10.1.17 Scrum møde 16, Torsdag d. 7/05-20

De administrative værktøjer er så småt på plads i back- og frontend. Der skal ses på noget Javascript til opdateringen af specifikke felter i database-dataen, så man kan ændre dem som administrativ bruger.

Carportens visuelle implementering skal afklares i dag. Vi skal have lavet SVG-strengen til en fil og bruge den fil i et IMG tag i frontend.

Sprint møde 3, Torsdag d. 7/05-20

Product owner var godt tilfreds efter at have set vores demo til carporten. Vi mangler at få implementeret det i front end, og dette er kriteriet for godkendelsen. Vi har aftalt med productowner, at efter implementationen af den visuelle carport i frontend, kan vi starte på sprint 4.

10.1.18 Scrum møde 17, Fredag d. 8/05-20

Vi har afholdt sprint møde: vi er alle godt tilfredse med samarbejdet i gruppen. Vores gruppedynamik er god og fungerer gnidningsfrit, og vi bliver færdige med de ting vi planlægger. Det næste sprint bliver planlagt og fordelt, og vi regner stærkt med at kunne blive færdige med disse userstories i denne sprint også.

Den mindre userstory med ændring af materialer kan godt blive større end 5 point. Så vi revurderer muligvis arbejdsbyrden på denne.

10.1.19 Scrum møde 18, Tirsdag d. 12/05-20

Vi regner med at være færdige med front-end til userstoriesne i dag. Der ud over bliver logikken til metoderne bag taget også udarbejdet. Implementeringen af Roof-klassen, giver nogle problemer i logikken, som er uforudset, men som vi finder en løsning på. Det giver muligvis en forsinkelse på vores sprint mål.

Vi debatterer en omstrukturering af hvordan kunden vælger mål for højden til tag/carport/total højde, som viser sig at give nogle udfordringer.

10.1.20 Scrum møde 19, Onsdag d. 13/05-20

I dag bliver der arbejdet på styklisten, som skal tage højde for skrå brædder. Der bliver også fundet en løsning til hvordan vi får fat i konfigurationerne i front-end. Der tilføjes også beklædningskomponenter i databasen.

Logikken til udregning af tagets C-linje samt alle de matematiske trekantudregninger er blevet færdig. Styklisten er også opdateret med disse metoder, samt komponent attributter. Front end er klar til disse attributter. Userstory 53 er i mål dog med småfejl klar, der bliver løst. Userstory 49 er delvist implementeret.

10.1.21 Scrum møde 20, Torsdag d. 14/05-20

User story 49 og 53, som handler om ændring af materialer og ændring af priser, er næsten færdig, men mangler blot nogle bugfixes.

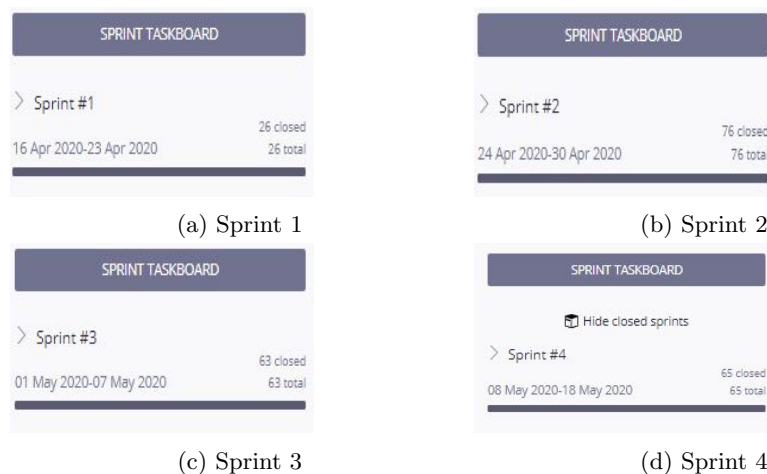
I kodningen vedrørende taget, er der opdaget nogle små ting, som i fremtiden ville kunne forbedre projektet. Men vi anser ikke, at der vil være tid i vores sprint til at rette disse mindre fejl. Der bliver ikke taget højde for visse komponenters størrelse i forhold til hvordan de sammensættes enkelte steder. Ikke alle tagtyper passer til et fladt tag.

Der er skrevet metoder til logikken bag beklædningsbrædder og parts dertil. Der mangler rettelser til dette. Derudover skal der kobles lister til disse metoder, og sidst SVG-kodningen. Det skal indgå i rapporten, at de totale mål for karporten er uden beklædning, og at beklædning blot ligges derpå.

10.1.22 Scrum møde 21, Fredag d. 15/05-20

Der er arbejdet på klasserne med komponenterne og blueprint-delen angående taget, samt roof og parts. Der mangler noget bugfix til addPartBekledning metoden, som ikke giver det rigtige antal skruer på hjemmesiden. Dette bliver fikset i dag.

Der skal refactorers noget kode med lægterne, som ikke tager højde for sin egen bredde, når den måler op til næste lægte.



Figur 10.1: Caption for this figure with two images