

2. Semester Projekt: Gadstrup Rosenplanteskole



Klasse id: (rf23da2c1-1c)

Lucas Valls-Sjøstrand: luv001@edu.zealand.dk

Rasmus Avndal Hansen: rah001@edu.zealand.dk

Silas Stilling-Jørgensen: sis002@edu.zealand.dk

Tan Ti Khuu: tak002@edu.zealand.dk

Zimon Thais Jacobsen: zimo0008@edu.zealand.dk

Tegn med mellemrum: 69.280

Link til github: <https://github.com/devexcsgo/Eksamensprojekt.git>

Link til website: <http://gproser.silasstilling.dk/>

Indholdsfortegnelse

Forberedelse	1
Indledning	1
Afgrænsning	1
Metoder, Teknikker og Værktøjer	1
Virksomhedsanalyse	2
Cost Benefit (se bilag 1) (Hovedansvar: Rasmus)	2
Risikoanalyse (Hovedansvar: Silas, Rasmus)	2
Business Model Canvas (se bilag 2) (Hovedansvar: Lucas, Rasmus & Zimon)	3
Pestel Analyse (se bilag 3) (Hovedansvar: Rasmus, Silas & Zimon)	4
SWOT (se bilag 4) (Hovedansvar: Lucas, Rasmus, Silas & Tan)	5
TOWS (se bilag 5) (Hovedansvar: Rasmus, Tan & Zimon)	6
Inception Deck (Hovedansvar: Lucas, Rasmus & Silas)	7
Delkonklusion	10
Analyse og design:	11
User Stories (Hovedansvar: Lucas, Tan & Zimon)	11
Product Backlog (Hovedansvar: Rasmus)	16
Business Process Modeling (se bilag 8) (Hovedansvar: Rasmus)	16
Domænemodel (se bilag 7) (Hovedansvar: Silas & Tan)	16
Enhanced Entity Relationship Diagram (Se bilag 9) (Hovedansvar: Tan & Zimon)	18
Klassediagram (se bilag 10) (Hovedansvar: Lucas & Tan)	18
Software Klassediagram (se bilag 11) (Hovedansvar: Silas)	18
System Sekvens Diagram (se bilag 15) (Hovedansvar: Zimon)	19
Test Cases (se bilag 16) (Hovedansvar: Lucas, Tan & Zimon)	19
Skitser/Mockups (se bilag 17) (Hovedansvar: Alle)	20
Review (se bilag 18)	20
Sprint:	21
SCRUM (Hovedansvar: Rasmus & Tan)	21
Sprint 1 (Hovedansvar: Rasmus)	21
Sprint 2 (Hovedansvar: Tan)	23
Sprint 3 (Hovedansvar: Ingen)	24
Afslutning af rapporten	24
Razor Pages og Kode: (Hovedansvar: Se navn under specifikt emne)	24
Database (Hovedansvar: Silas & Tan)	31
Refleksion (Hovedansvar: Alle)	33
Konklusion (Hovedansvar: Alle)	35

Kildeliste.....	36
Bilag.....	37
Logbog.....	52

Forberedelse

Indledning

Dette projekt har som mål at udvikle en webapplikation for Gadstrup Rosenplanteskole, en lille dansk virksomhed specialiseret i produktion og salg af haveroser. Gadstrup Rosenplanteskole bliver grundlagt i 1974 af ægteparret Sonja og Carsten, som starter med at producere 10.000 roser om året. Siden da oplever virksomheden gradvis vækst og når en produktionsrekord på 130.000 roser i 1989. I dag producerer de omkring 60.000 roser årligt, fordelt på over 300 forskellige sorter. Virksomheden opererer primært på to store marker i Gadstrup og et lejet drivhus samt en rosenmark på Ærø.

Projektets hovedfokus er at udvikle, designe og implementere et IT-system ved hjælp af den agile udviklingsmetode SCRUM, C#-programmering og relationelle databaser.

Problemformuleringen lyder: *"Hvordan kan den agile udviklingsmetode SCRUM, C#-programmering og relationelle databaser benyttes til at udvikle, designe og implementere et mindre IT-system?"*

Den eksisterende hjemmeside for Gadstrup Rosenplanteskole er forældet og mangler funktionalitet, hvilket giver anledning til behovet for en opdatering, der kan forbedre brugeroplevelsen og effektiviteten af forretningsprocesserne.

Afgrænsning

Projektet er afgrænset til at udvikle en hjemmeside uden at inkludere et online handelssystem, da dette ville ændre virksomhedens forretningsmodel væsentligt. Fokusområderne er derimod at skabe et forbedret brugerinterface, inkludere produktfiltrering og tilføje administrativ funktionalitet som CRUD-operationer (Create, Read, Update, Delete). Vi vælger også at udelukke en mobilapp, søgefunktion og statistikside, selvom disse elementer bliver omtalt i designafsnittet.

Metoder, Teknikker og Værktøjer¹

For at strukturere projektet effektivt anvender vi både Vandfaldsmetoden og SCRUM. Vandfaldsmetoden bliver brugt i de indledende faser såsom forberedelse, virksomhedsanalysen og designfasen, mens SCRUM bliver anvendt under selve udviklingen gennem forskellige sprints. Projektet bliver planlagt ved hjælp af et Gantt-kort i Project Libre og teknologier som C#, HTML, CSS, Bootstrap, SQL og jQuery bliver brugt til at implementere løsningen. Razor Pages i Visual Studio bliver anvendt til at binde det hele sammen.

I vores virksomheds- og forretningsanalyse bruger vi modeller som Business Model Canvas, SWOT, TOWS og PESTEL for at opnå en dybere forståelse af Gadstrup Rosenplanteskole og identificere deres behov og udfordringer. Dette fundament danner grundlaget for at designe og udvikle en løsning, der ikke blot forbedrer virksomhedens online tilstedeværelse, men også optimerer deres interne processer og kundeservice.

Denne rapport dækker hele projektforsløbet, herunder analyser, design, implementering, test og konklusioner, og reflekterer over de metoder og teknikker, der bliver anvendt for at nå de opstillede mål.

¹ [4 ugers semester projekt.docx](#)

Virksomhedsanalyse

Cost Benefit (se bilag 1) (Hovedansvar: Rasmus)²

Vi har lavet en investeringskalkule for den nye hjemmeside. Her tog vi udgangspunkt i en 10-årig periode. Det er umuligt at kende den nøjagtige levetid for hjemmesiden, derfor er den 10-årig periode et estimat. Vi beregnede omkostningerne ved investeringen ved at gange antallet af timer med vores timeløn. Derefter lagde vi de kontante kapacitetsomkostninger (KKO) såsom overarbejde, indholdsproduktion og domæneomkostninger til. Indbetalingerne hvert år repræsenterer den nye indtjening fra de nye kunder, som hjemmesiden tiltrækker. Desuden er der større chance for mere salg, da hjemmesiden er mere brugervenlig og ser mere professionel ud. Vi anvendte en kalkulationsrente på 6% som vores diskonteringsrente. Da virksomheden ikke skal optage store lånebeløb, er en rente på 6% meget realistisk. Den er endda sat lidt højt, for at være på den sikre side. Samlet set viser vores analyse, at investeringen er rentabel med en gevinst på 10.803 kr. Man kan ikke regne med gevinsten, da de fleste tal er baseret på estimater som er umulige at kende. Det er ikke særlig mange penge virksomheden tjener. Faktisk kunne de højst sandsynligt tjene flere penge ved at investere i noget andet. Men virksomheden laver ikke denne investering for at tjene penge, men hovedsageligt for at kunne konkurrere med konkurrenterne og virke mere tiltrækkende over for kunder.

Risikoanalyse (Hovedansvar: Silas, Rasmus)

Den følgende risikoanalyse er udarbejdet som en del af projektplanlægningen for udviklingen af en webapplikation til Gadstrup Rosenplanteskole. Formålet med denne risikoanalyse er at identificere potentielle trusler og udfordringer, der kan påvirke projektets succes, samt at udvikle passende strategier til risikohåndtering. Risikoanalysen er opdelt i fire hovedkategorier: tekniske risici, procesrisici, forretningsrisici og eksterne risici. Hver kategori indeholder specifikke risikofaktorer, der er relevante for projektet.

Tekniske Risici

Sikkerhedsproblemer: Utilstrækkelig sikkerhed kan udsætte systemet for sårbarhed og potentielle angreb, hvilket kan resultere i datatab eller uautoriseret adgang til følsomme oplysninger.

Ydelsesproblemer: Stor datamængde eller ineffektiv kode kan føre til langsom ydelse eller nedbrud af systemet under belastning.

Proces Risici

Ressourcemangel: Mangel på tilstrækkelige ressourcer som tid, budget eller ekspertise kan forsinke projektet eller påvirke kvaliteten af det endelige produkt.

Manglende klarhed om projektets mål: Risikoen for at drifte væk fra projektets mål kan resultere i tidsspilde og produktion af irrelevante elementer.

Uenigheder i gruppen: det er normalt, at alle ikke har den samme holdning.

Dårlig kommunikation: Utilstrækkelig kommunikation internt i teamet eller med kunden kan føre til misforståelser, fejl og forsinkelser i projektet.

² [Gadstrup Rosenplanteskole Investeringskalkule.xlsx](#)

Tidsmangel: Utilstrækkelig tid til at fuldføre opgaver kan påvirke kvaliteten af det endelige produkt og øge stressniveauet i teamet.

Forretnings Risici

Ændrede forretningsbehov: Ændringer i Gadstrup Rosenplanteskolens forretningsmæssige krav eller strategi kan kræve tilpasning eller omstrukturering af webappen.

Eksterne Risici

Lovgivningsmæssige ændringer: Ændringer i lovgivningen relateret til datasikkerhed kan kræve ændringer i webappens funktionalitet eller drift.

Business Model Canvas (se bilag 2) (Hovedansvar: Lucas, Rasmus & Zimon)

Gadstrup Rosenplanteskole er en lokal virksomhed, der specialiserer sig i produktion og salg af roser. Grundlagt i 1974, har Gadstrup Rosenplanteskole etableret sig som en pålidelig leverandør af et bredt udvalg af kvalitetsroser til både private- og erhvervskunder.

Gadstrup Rosenplanteskole har etableret nøglepartnere som Coop, Meny, Silvan, Løvbjerg og SuperBrugsen, samt med adskillige andre planteskoler i hele Danmark. Disse partnerskaber sikrer en bred distribution af deres roser og udvider deres markedstilgang.

Virksomhedens nøgleaktiviteter omfatter ikke kun salg af roser, men også dybdegående viden om dyrkning og pleje af forskellige rosetyper. Dette inkluderer timingen for plantning og blomstring samt bekæmpelse af sygdomme og skadedyr, hvilket bidrager til at sikre kvaliteten af deres produkter.

Gadstrup Rosenplanteskole tilbyder en række Value Propositions til deres kunder, herunder et bredt udvalg af kvalitetsroser, personlig service fra en lokal virksomhed med kendskab til deres stamkunder, og adgang til Den Åbne Rosenmark, hvor kunder kan opleve og købe sjældne eller specielle roser samt få rådgivning og vejledning.

Virksomheden opretholder tætte Customer Relationships ved at imødekomme kundernes forventninger om et stort udvalg af roser. Derudover fremmer det også mund-til-mund-reklame blandt lokale kunder, som ofte er gentagne købere og vigtige ambassadører for virksomheden.

Kunderne spænder fra blomsterentusiaster til romantiske partnere og folk, der søger roser til specielle begivenheder. Den Åbne Rosenmark tilbyder også en oplevelse for familier og par, hvilket udvider virksomhedens kundebase og tiltrækker både forbrugere (B2C) og andre planteskoler (B2B).

Virksomhedens primære ressourcer inkluderer produktionen af ca. 300 forskellige rose typer samt den fysiske placering af rosengården på Tyrsegårdsvej 6 i Gadstrup.

Gadstrup Rosenplanteskole når deres kunder gennem en hjemmeside på internettet samt direkte salg til virksomheder og kunder. Mund-til-mund reklame spiller også en væsentlig rolle i at øge virksomhedens synlighed.

De primære omkostninger for virksomheden omfatter grundstammer til at dyrke roser, pesticider til beskyttelse mod insekter, lønninger til medarbejdere, brændstof, udgifter til vedligeholdelse samt ejendomsskat.

Virksomhedens indtægter genereres primært gennem salg af roser til både forbrugere og virksomheder. Betalingsmetoder inkluderer kontanter og mobilepay.

Pestel Analyse (se bilag 3) (Hovedansvar: Rasmus, Silas & Zimon)

På baggrund af vores PESTEL analyse har vi identificeret flere aspekter, der kan påvirke Gadstrup Rosenplanteskoles forretningsmiljø.

Inden for det politiske område er statens lovgivning om virksomhedsdrift og miljøbeskyttelse af afgørende betydning. Ligeledes påvirker reguleringer vedrørende import og eksport af planter, såsom grundstammer, virksomhedens drift og markedsadgang.

Økonomiske faktorer, såsom rentesatser, inflation og arbejdskraftmangel, har betydelig indvirkning på virksomhedens økonomi. En lav rente kan øge forbrugernes købekraft og dermed potentielt stimulere efterspørgslen efter roser. Samtidig kan arbejdskraftmangel påvirke priserne på roser. Desuden kan inflationen påvirke folks købekraft, hvilket er meget problematisk for virksomheden da roser ikke et fysisk behov og hurtigt ville blive nedprioriteret af kunderne.

Sociale normer spiller en afgørende rolle i salget af roser, da det blandt andet er almindeligt at give roser som en symbolsk handling i romantiske sammenhænge eller lignende begivenheder. Ændringer i forbrugernes præferencer inden for blomster og havearbejde samt demografiske forhold, såsom aldersfordeling, påvirker også efterspørgslen efter roser.

På det teknologiske plan kan effektiv digital markedsføring via en velfungerende hjemmeside være afgørende for at nå ud til kunder og forbedre salget. Gadstrup Rosenplanteskole kan forbedre sig ved at investere i teknologi, der optimerer deres online tilstedeværelse og kundekommunikation.

Miljømæssige faktorer, såsom klimatiske forhold og CO₂-udledning fra transport, påvirker dyrkningen og distributionen af roser. Desuden kan fremtidig lovgivning om brug af pesticider have indvirkning på Gadstrup Rosenplanteskoles drift og praksis, da kun godkendte pesticider må anvendes ifølge loven.

Juridisk set er lovgivningen vedrørende anvendelse af pesticider og forskelle mellem økologisk og konventionelt landbrug også relevante for Gadstrup Rosenplanteskoles forretningsdrift. Det er vigtigt for virksomheden at være opmærksom på disse faktorer og tilpasse sig for at opretholde konkurrenceevne og bæredygtighed.

SWOT (se bilag 4) (Hovedansvar: Lucas, Rasmus, Silas & Tan)

Vi har udført en analyse af Gadstrup Rosenplanteskole ved hjælp af SWOT-metoden, hvor vi har integreret PESTEL-analysens observationer i vores vurdering af muligheder og trusler (**Opportunities**)/(**Threats**). Resultatet af denne SWOT-model fremhæver tydeligt Gadstrup Rosenplanteskoles styrker (**Strengths**), som inkluderer et imponerende sortiment på omkring 300 arter samt den omhyggelige dyrkning, der sikrer roser af høj kvalitet.

De ansatte besidder en betydelig viden om roser, hvilket bidrager til en forbedret kundeoplevelse, især blandt de lokale kunder, som de har et stærkt forhold til. Derudover tilbyder de en informativ hjemmeside, som er en kilde til omfattende oplysninger om deres roser. Endelig har Gadstrup Rosenplanteskole indgået salgsaftaler med flere virksomheder og andre planteskoler, hvilket styrker deres forretningsmæssige netværk og muligheder.

Planteskolen har en række svagheder (**Weaknesses**). Det gode er, at vi har identificeret disse svagheder, så nu kan planteskolen fokusere på dem. Alle svagheder kan rettes, hvilket er en mulighed for at omdanne dem til styrker. En del af deres svagheder stammer fra deres manglende evne til at markedsføre sig selv. Derfor er de primært begrænset til lokale kunder og mund-til-mund-reklame. Dette øger risikoen for at blive udkonkurreret af større blomsterfirmaer. Den sidste svaghed vi identificerede er, at de kun sælger roser. Det er selvfølgelig stadig en fordel at specialisere sig, men det betyder ikke nødvendigvis, at man behøver at begrænse sit sortiment.

Der er flere eksterne muligheder (**Opportunities**), som virksomheden kan udnytte til at forbedre sin position og opnå vækst. Virksomheden har mulighed for at udnytte sin lokale tilstedeværelse til at tiltrække lokale kunder, der foretrækker at handle lokalt og støtte lokale virksomheder frem for de større. De lokale kunder, som har haft gode oplevelser ved at handle hos dem, kan dele deres oplevelser og dermed give gratis reklame gennem mund til mund. Virksomheden kan udnytte begivenheder som fx bryllupper og Valentinsdag ved at markedsføre specielle produkter, såsom rosenhjerter til bryllupper eller buketter til Valentinsdag. Desuden kan en sund økonomi øge kundernes villighed til at købe kvalitetsroser til forskellige begivenheder.

Udover dette står deres Rosengård over for flere eksterne trusler (**Threats**). Virksomheden kan opleve konkurrence fra andre (og muligvis større) firmaer, der sælger roser, hvilket kan påvirke deres indtjening og mulighed for at miste kunder. Desuden kan klimaændringer have stor indflydelse på dyrkningen af roser, fx kraftigt regnvejr og ændringer i lovgivningen om pesticider kan skabe yderligere udfordringer. Økonomiske udfordringer påvirker også kundernes sandsynlighed for at købe roser, da kundernes disponible indkomst har stor indflydelse på virksomhedens salg og indtjening. Endelig kan sæsonbaseret efterspørgsel resultere i perioder med lav efterspørgsel, hvilket kan påvirke virksomhedens omsætning.

Gennem vores SWOT-analyse har vi lokaliseret en række positive og negative aspekter ved planteskolen. Disse aspekter kan kombineres for at danne strategier til, hvordan vi kan optimere virksomheden. Dette har vi gjort i vores TOWS-analyse.

TOWS (se bilag 5) (Hovedansvar: Rasmus, Tan & Zimon)

På baggrund af TOWS-Analysen for Gadstrup Rosenplanteskole har vi identificeret interne styrker og svagheder samt eksterne muligheder og trusler, som virksomheden står over for.

SO Strategier:

Vi fandt frem til en SO-strategi hvor Gadstrup Roseplanteskolen opretter profiler på de sociale medier, fordi dette kan bruges til reklamering og er en slags mund til mund reklamering fra at kunder engagerer på disse profilers posts. En del af det de allerede gør som deres SO-strategi er at have kvalitets roser og mange arter af roser, hvilket gør at der er gode muligheder til specielle begivenheder, såsom Bryllupper, Valentins Dag, Fødselsdage osv. Da danskerne har et højt rådighedsbeløb, kan de godt betale det ekstra for et kvalitetsprodukt som roser er.

WO Strategier:

Vi identificerede flere WO-strategiske muligheder for virksomheden. Først er der potentialet for at lancere en ny hjemmeside, hvilket ikke kun vil adressere de eksisterende svagheder ved den nuværende side, men også udnytte mulighederne for at forbedre synligheden og tiltrække nye kunder. Ved at udvide sortimentet til at inkludere salg af haveredskaber og andre relaterede produkter, kan virksomheden tackle deres begrænsning i produktudvalget og åbne op for nye kundesegmenter.

Samtidig er der potentiale i at implementere en digital og lokal markedsføringsstrategi, mens forbedringer af den online tilstedeværelse og markedsføring også er afgørende, da det vil øge synligheden og tiltrække flere kunder.

Til sidst kan virksomheden opfordre deres kunder til at dele deres positive oplevelser på de sociale medier, hvilket effektivt kan adressere manglende opmærksomhed og samtidig muliggøre gratis mund-til-mund reklame.

ST Strategier:

For at sikre en stabil og fremtidssikret salgsstrategi, er det afgørende for virksomheden at fokusere på at etablere robuste salgsaftaler. Ved at sørge for disse aftaler, kan virksomheden levere roser til butikkerne, som de normalt ikke ville have adgang til, hvilket styrker virksomhedens position på markedet og sikrer en konkurrencemæssig fordel. Desuden bør virksomheden satse på at dyrke roserne økologisk og bæredygtigt, hvilket ikke blot tilføjer en ekstra dimension af kvalitet til produkterne, men også appellerer til den stigende efterspørgsel på bæredygtige varer. Sidst men ikke mindst, ved at have stærke salgsaftaler på plads, kan virksomheden begrænse de økonomiske risici og konsekvenser, der måtte opstå som følge af økonomiske udfordringer i Danmark, og dermed sikre en mere robust forretningsmodel på B2B-markedet.

WT Strategier:

I WT fokuseres der meget på at sikre virksomheden kan blive ved med at holde deres salg højt. Dette er bl.a. ved at lave markedsføring, da det er lettest at skaffe nye kunder igennem markedsføring, mens man er en lille virksomhed.

En anden strategi vi identificeret, var at de kunne begynde at sælge andre produkter såsom haveredskaber, som kunder kunne bruge til at pleje deres egne roser. Dette ville give lidt mere diversitet i hvad de sælger, men er stadig fokuseret imod deres niche som er roser.

En af de andre trusler var at klimaet/vejret kan påvirke dyrkningen af roser, og da virksomheden kun sælger roser, er det vigtigt og sørge for produktionen ikke bliver påvirket af tilfældige faktorer som vejret. Dette fandt vi en mulig løsning på, ved at investere i andre dyrkningsmetoder, såsom at skaffe flere drivhuse.

Dog skal man også huske på at hvis man kun havde alt produktion i drivhuse, ville stedet hurtigt miste sin charme, som også tiltrækker kunder så det er om at finde den rette balance.

Den sidste udfordring, som vi har håndteret, var at kundernes præferencer potentielt kunne ændre sig med tiden,

dette løses ved at være på forkant med at lave nogle markedsundersøgelser for at sikre os at vores kvalitetsprodukt opfylder kundernes aktuelle ønsker.

Inception Deck (Hovedansvar: Lucas, Rasmus & Silas)

1. Why Are We Here?

For at lave en hjemmeside for Gadstrup Rosenplanteskolen, fordi deres nuværende hjemmeside er meget begrænset i dens funktioner til at f.eks. mangel af sortering af kategorier af deres Roser.

2. Create an Elevator Pitch³

Til Gadstrup Rosenplanteskolen

Som mangler hjemmeside

Gproser.dk

Er en ny, moderne hjemmeside

Der lader dig se alle de roser som er til salg i butikken

I modsætning til andre, kan du på vores hjemmeside sortere roser efter duft rating, farve samt se mange flere oplysninger om hvilken rose der er den perfekte til dig.

Vores produkt er mere end bare en hjemmeside, oplev den revolutionerende måde at drive din forretning på.

³ [4 ugers semester projekt.docx](#)

3. Design a Product Box (se bilag 6)



4. Create a Not list.

IN SCOPE	OUT OF SCOPE
Ny forbedret hjemmeside	Hjemmesiden vil ikke inkludere onlinehandel
Større sortiment (tilbehør til roser)	En mobilapp til onlinehandel
Gøre det nemmere at se hvor man skal klikke på hjemmesiden for hvilke roser de har.	
UNRESOLVED	
Et månedligt abonnement.	
Markedsføring	

5. Meet Your Neighbors⁴

I projektet bliver vores egen gruppe ikke direkte påvirket af projektet, til gengæld har vi stor indflydelse på projektet. Derfor er vi grå eminence. Produktejeren bliver påvirket meget af projektet samt har stor indflydelse på det. Det vil sige, at han er en Ressourceperson. Til sidst har vi kunderne af planteskolen, de bliver påvirket meget af projektet, men har en lille indflydelse på det. Det gør medlemmer til gidsler.

6. Show the Solution

Domainmodel (se bilag 7)

7. Ask What Keeps Us Up at Night

Vi har allerede lavet en risikoanalyse (se evt. Bilag.), her kom vi ind på disse punkter:

- Sikkerhedsproblemer
- Ydelsesproblemer
- Ressourcemangel
- Manglende klarhed om projektets mål
- Dårlig kommunikation
- Tidsmangel
- Uenigheder i gruppen
- Ændrede forretningsbehov
- Lovgivningsmæssige ændringer

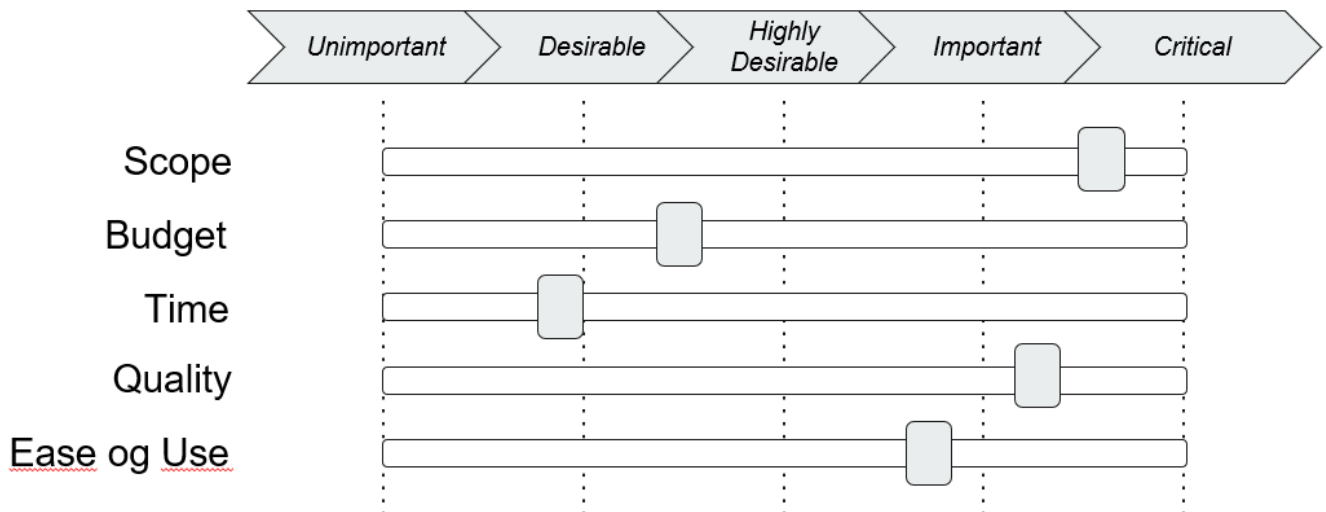
8. Size It Up

Dette projekt vil tage cirka halvanden måned at gennemføre. Opgaven går ud på at forbedre deres hjemmeside ved at udvikle et mere effektivt filteringsystem, samt andre funktioner der ville gøre bruger oplevelsen bedre. De har allerede oplysninger om roserne (navn, farver, type og duftstyrke).

⁴ [4 ugers semester projekt.docx](#)

9. Be Clear on What's Going to Give⁵

Trade-off Sliders



Scope, kvalitet og brugervenlighed er de tre vigtigste faktorer for projektet. Alle tre punkter retter sig mod kundeoplevelsen, hvilket er det vigtigste mål med projektet. Tid og budget er naturligvis også vigtige, men hovedmålet med projektet er ikke at tjene penge, men at sikre en fremragende kundeoplevelse. Derfor prioriterer vi høj kvalitet, omfattende scope og en nem og intuitiv brugeroplevelse, så vi kan opfylde og overgå kundernes forventninger.

10. Show What It's Going to Take

Projektet kommer til at forløbe på 45 dage, hvor meget det kommer til at koste og eventuelle gevinster kan ses i vores cost-benefit analyse. Teamet som er nødvendigt for at løse denne opgave, er 5 software udviklere som har erfaring i at arbejde med både vandfaldsmodellen og SCRUM.

Delkonklusion

Først og fremmest kan vi konkludere at der er et klart behov for en opdatering af Gadstrup Rosenplanteskoles hjemmeside, da den eksisterende hjemmeside er forældet. Den nye webapplikation skal ikke blot være en visuel opgradering, men også fokusere på funktionalitet og brugervenlighed.

Det er afgørende at den nye webapplikation har et brugervenligt interface for brugere og kunder. Funktionalitet såsom produktfiltrering, søgefunktioner og muligheden for at tilføje favoritter vil forbedre kundeoplevelsen og gøre det lettere for brugerne at finde de ønskede produkter.

Desuden er der behov for en admin-bruger der har adgang til CRUD-funktionalitet. Dette giver admin-brugeren mulighed for nemt at tilføje, fjerne og redigere produkter. Dette vil bidrage til at holde webapplikationen opdateret og relevant.

Derudover skal der være en integration af eksisterende forretningsinformation. Gadstrup Rosenplanteskole besidder allerede en betydelig mængde information om deres roser, såsom forskellige kategorier, priser og tilgængelighed. Den nye webapplikation bør integrere denne eksisterende information på en struktureret og brugervenlig måde.

⁵ <https://github.com/kyanny/trade-off-sliders-template>

Samlet set vil den nye webapplikation for Gadstrup Rosenplanteskole ikke kun forbedre deres online tilstedeværelse, men også bidrage til at optimere deres forretningsprocesser og kundeoplevelser. Med de rigtige tiltag og implementeringer kan webapplikationen blive en værdifuld ressource for både virksomheden og dens kunder.

Her har vi en liste over ting, som skal være med på hjemmesiden:

- Database
- Login/bruger system
- Admin med rettigheder for at tilføje/fjerne produkter
- Bruger med mulighed for at vælge favoritter, for at se tilgængelighed og information
- Mulighed for kunder at filtrere produkter efter kategorier, som rosentype, farve, størrelse og duft level. (Gadstrup Roseplanteskolen har allerede information for disse ting, så det der mangler, er et system til at tag denne information i brug)
- Prisen på de forskellige roser
- Optimere designet via css/bootstrap i Razor Pages, så layoutet ser bedre ud
- Favoritfunktion
- Statistik Services

Analyse og design:

User Stories (Hovedansvar: Lucas, Tan & Zimon)

Titel: Loginfunktion	Prioritet: 3	Estimering: S
User Story: Som bruger vil jeg gerne kunne logge ind, så jeg kan have adgang til de relevante funktioner for min logintype		
Funktionelle Krav: <ul style="list-style-type: none"> • Brugeren skal kunne logge ind med et gyldigt brugernavn og en adgangskode • Der skal være et login system • Systemet skal validere brugeroplysninger og give adgang til deres respektive rettigheder • Hvis brugernavnet/kodeordet er ugyldige, skal systemet vise en fejlmeddelelse og forhindre adgang • Brugeren skal blive omdirigeret til den relevante side på hjemmesiden efter vellykket login • Brugeren skal have mulighed for at logge ud • Der skal være mulighed for at nulstille sin adgangskode, ved hjælp af en "glemt adgangskode" knap. (f.eks. ved at sende et link til brugerens e-mailadresse) Non Funktionelle Krav: <ul style="list-style-type: none"> • Systemet skal anvende kryptering for at beskytte brugernes adgangskoder og oplysninger (PasswordHasher) • Når der er en fejl under loginprocessen, skal beskederne være klare, så du nemt kan rette det • Adgangskoden skal være gemt sikkert, så ingen kan stjæle den • Hvis nogen prøver at gætte din adgangskode mange gange, bliver din konto midlertidigt låst for at beskytte den • Det skal være muligt at tilgå log ind-siden fra forskellige enheder og browsere 		

Titel: Administrator funktioner	Prioritet: 2	Estimering: L
User Story: Som ejer ønsker vi nogle administratorfunktioner som CRUD, så vi kan håndtere roser, nyheder og andre begivenheder		
Funktionelle Krav: Create: <ul style="list-style-type: none"> Administrator skal have adgang til en formular til oprettelse af nye opslag. Disse opslag skal indeholde relevante oplysninger såsom navn og beskrivelse Administrator skal have adgang til at kunne oprette nye produkter. Ved oprettelsen skal der angives relevante oplysninger som navn, beskrivelse, kategori, pris og billeder Systemet skal validere de indtastede oplysninger for at sikre at de er indtastet korrekt i deres respektive felter En administrator skal kunne oprette en ny administrator Read: <ul style="list-style-type: none"> Administrator skal have adgang til en side som viser oversigten over alle eksisterende opslag Administrator skal kunne filtrere og søge efter specifikke opslag Update: <ul style="list-style-type: none"> Administrator skal have mulighed for at redigere enhver opslag Administrator skal have mulighed for at ændre og opdatere oplysninger som er nødvendigt Delete: <ul style="list-style-type: none"> Administrator skal have mulighed for at slette enhver opslag Administrator skal have en bekræftelse pop-op besked, for at undgå sletning af forkert opslag Non Funktionelle Krav: <ul style="list-style-type: none"> Almindelige kunder skal ikke have administratorrettigheder Det skal kun være muligt for administrator der kan have rettigheder til CRUD Det skal være let og forståeligt at oprette/opdatere/slette tingene på siden Systemet skal kunne håndtere en stigende mængde af CRUD-operationer uden væsentlig forsinkelse eller nedetid Når administrator indtaster oplysninger, skal det som bliver indtastet matche de tilsvarende datatyper. (f.eks. Pris skal indeholde numerisk værdi) 		

Titel: Historik over ændringer	Prioritet: 9	Estimering: M
User story: Som administrator, vil jeg gerne have en historik over ændringer foretaget af Administrator		
Funktionelle Krav: <ul style="list-style-type: none"> Der skal være en sektion for historik over ændringer Administratoren skal kunne se en liste over alle tidligere ændringer der er foretaget af andre administratorer Historikken over ændringer skal indeholde oplysninger om, hvilke dele af hjemmesiden der er sket ændringer på, hvornår de blev foretaget og hvem (navnet hentes på deres ID) der foretog ændringen Der skal være klare oplysninger om hver enkelt ændring, detaljer om de specifikke felter eller sider, der er blevet ændret Det er kun administrator som skal have adgang til at se denne side Non Funktionelle Krav: <ul style="list-style-type: none"> Alle ændringer foretaget af administratorer i systemet skal logges og registreres, for at muliggøre overvågning og revision Brugergrænsefladen til historikken over ændringer skal være nem at navigere rundt i, så man nemt kan ændre ting 		

Titel: Filtrering	Prioritet: 4	Estimering: M
User Story: Som kunde vil jeg gerne kunne søge efter specifikke kategorier og filtrere resultaterne: Såsom rosetype, farve og størrelse		
Funktionelle Krav: <ul style="list-style-type: none"> • Det skal være muligt som kunde at vælge mellem flere forskellige kategorier • Det skal kunne være muligt at bruge et filtreringssøgesystem • Der skal være en dropdown menu, hvor man skal kunne vælge at sortere efter en eller flere kategorier • Når kategorier er valgt, skal søgeresultaterne opdateres øjeblikkeligt for kun at vise roser, der opfylder de valgte kriterier • Systemet skal reagere indenfor 1 sekund på brugerens valg af kategorier uden behov for at genindlæse hele siden • Brugeren skal have mulighed for at fjerne eller ændre valgte kategorier, så søgeresultaterne justeres efter behov • Systemet skal vise antallet af resultater, der matcher de valgte kategorier, for at give brugeren et overblik over deres behov Non Funktionelle Krav: <ul style="list-style-type: none"> • Det er muligt at bruge flere filetere på en gang. F.eks. farve og størrelse • Filtreringsmulighederne skal være intuitive og nemme at bruge for alle • Systemet skal kunne håndtere filtrering af søgeresultater hurtigt og effektivt, selv når der er store mængder af data • Filtreringsfunktionerne skal være kompatible med forskellige browsere og enheder, for at sikre at folk på mobiler eller bærbar/stationære har adgang til siden 		

Titel: Søgssystem	Prioritet: 5	Estimering: M
User story: Som kunde, vil jeg kunne søge efter specifikke roser så jeg kan hurtigt finde det jeg leder efter		
Funktionelle Krav: <ul style="list-style-type: none"> • Søgfeltet skal være synligt og tilgængeligt på alle sider af hjemmesiden • Efter succesfuld søgning skal der komme et billede af rosen samt en beskrivelse af produktet med relevante informationer • Efter man har skrevet 3 eller flere bogstaver, skal den komme med en autocomplete dropdown som matcher med første bogstaver, så man hurtigt kan finde den rose man ønsker • Hvis der kommer flere resultater ude fra sine søgeord, skal det være let og navigere imellem dem og der skal være et billede ved siden af navnet på dem Non Funktionelle Krav: <ul style="list-style-type: none"> • Søgsystemet skal være hurtigt og effektivt med præcise søgeresultater, selv når der er store mængder af data • Søgsystemet skal være let at bruge og intuitivt for at sikre en god brugeroplevelse • Søgsystemet skal være i stand til at håndtere en stigende mængde af søgninger og søgeresultater 		

Titel: Design af hjemmesiden	Prioritet: 7	Estimering: L
User story: Som kunde vil jeg, gerne kunne have et overblik over hvad GProser.dk tilbyder		
Funktionelle Krav: <ul style="list-style-type: none"> Der skal være en side om os, hvor der står en beskrivelse om vores virksomhed og vores historie. Dette inkluderer åbningstider og dage Der skal være en side som viser de roser vi gror. Denne side skal indeholde prisliste og andre nyttige informationer Der skal være en nyheds side hvor vi kan annoncere ting På bunden af forsiden, skal der være kontakt info, for dem der gerne vil kontakte os, til f.eks. firmaer som gerne vil sælge vores roser Der skal være en side hvor der er galleri af diverse roser med mulighed for at klikke på billede for at se yderligere detaljer om rosen Der skal implementeres et søgefelt på hjemmesiden, så brugerne nemt kan søge efter specifikke roser og se deres oplysninger Non Funktionelle Krav: <ul style="list-style-type: none"> Design af siden skal være let og brugervenlig for at sikre at brugeren, nemt kan se og læse på hjemmesiden (GastaltLovene, Heuristic og KISS) Hjemmesiden skal være responsiv på forskellige enheder og skærmstørrelser Designet af hjemmesiden skal være optimeret for at man hurtig kan load side, for at minimere ventetiden 		

Titel: Oprettelse af bruger	Prioritet: 2	Estimering: S
User story: Som kunde, vil jeg gerne kunne oprette en bruger på hjemmesiden, for at kunne modtage eksklusive tilbud og nyheder om roser		
Funktionelle Krav: <ul style="list-style-type: none"> Der skal være en side med en formular, der giver brugeren for at indtaste deres oplysninger og oprette en bruger Brugere SKAL opfylde visse felter såsom, brugernavn, e-mail og adgangskode Brugeren skal kunne ændre deres adgangskode Systemet skal validere de indtastede oplysninger, f.eks. ved at tjekke om e-mail er gyldig og om adgangskoden opfylder sikkerhedskravene Efter oprettelsen af brugerprofil, skal systemet sende en bekræftelses e-mail til den angivne e-mailadresse for at verificere brugerens identitet Efter registreringen, vil brugeren blive bedt om at logge ind på systemet Brugeren (undtagen Admins) skal have evnen til at slette eller lukke deres profil Non Funktionelle Krav: <ul style="list-style-type: none"> Oprettelse af brugernavnet må kun indeholde bogstaver og tal. Symboler såsom: Spørgsmålstegn, Udråbstegn Underscore, Bindestreg osv. Vil give en fejlmeddelelse ved oprettelsen Ved valg af adgangskoden skal koden minimum være 8 tegn, skal indeholde: 1 stort bogstav, 1 tal og 1 symbol Brugeroprettelsesprocessen skal være responsiv og hurtig, med en maksimal ventetid på få sekunder under normale belastningsforhold Der kan ikke oprettes en profil med en e-mail som allerede er brugt Det skal være muligt at registrere sig for forskellige enheder og browsere, da folk bruger/har forskellige enheder og behov 		

Titel: Online handel	Prioritet: 6	Estimering: L
User story: Som en kunde vil jeg gerne kunne bestille roser online og hente dem ved Gadstrup Rosenplanteskolen		
Funktionelle Krav: <ul style="list-style-type: none"> • Der skal være en kurv, hvor man kan se de roser man har tænkt sig at bestille indtil videre • Vælge dato og tidspunkt for afhentning • Man skal have evnen til at betale for roserne i kurven online • Efter man har bestilt produktet, skal brugeren modtage en kvittering via e-mail med oplysninger om ordren afhentningsdetaljer Non Funktionelle Krav: <ul style="list-style-type: none"> • Der skal være god sikkerhed for at beskytte brugerens betalingsoplysninger • Det skal være nemt at navigere og betale • Systemet skal kunne håndtere at der kommer mange ordrer uden at påvirke ydeevnen 		

Titel: Favoritliste	Prioritet: 10	Estimering: L
User story: Som medlem, vil jeg gerne kunne tilføje roser til min favoritliste, så jeg nemt kan finde tilbage til dem på et andet tidspunkt		
Funktionelle Krav: <ul style="list-style-type: none"> • Der skal være en favorit knap/stjerne, som man kan klikke på der gør rosen til favorit • Der skal være en side hvor man kan se alle roser man har valgt som favorit, og se om rosen er tilgængelig • Det skal være muligt at bruge den samme knap man bruger til at tilføje til favoritter, til at fjerne dem igen • Det skal være tydeligt ved hjælp af farve på ikonet om den er i sine favoritter eller ikke • Det skal være muligt at sortere sin favoritliste, ved hjælp af filtreringssystemet fra vores user story med titlen "Kategorier/Filtrering" Non Funktionelle Krav: <ul style="list-style-type: none"> • Det skal være let og intuitivt at navigere, så man nemt kan tilføje/fjerne roser fra listen • Man skal kunne tilføje og fjerne fra listen hurtigt, selv med store mængder af data • Listen skal kun være synlig til den individuelle bruger 		

Titel: Statistik services	Prioritet: 8	Estimering: M
User story: Som Administrator vil jeg gerne have en liste med statistikker over køb/favoritter		
Funktionelle Krav: <ul style="list-style-type: none"> • Siden har en liste over alle roser der er blev købt på hjemmesiden over en specificere periode som kun er synlig for en administrator • Der er en liste, som kun er synlig for administrator, som viser hvad det dyrest online order er, hvad det billigste online order er, og hvad der i gennemsnittet er blevet betalt per online order i løbet af en specificeret periode • Administratoren skal kunne se en detaljeret liste over alle køb, antal købte roser, totale indtægter og købstidspunkter • Administratoren skal have adgang til en liste over de mest populære roser, baseret på antallet af tilføjelser til favoritlister • Administratoren skal kunne vælge forskellige parametre (f.eks. Dato, kategori, farve) til at sortere og filtrere data til specificeret statistikker Non Funktionelle Krav: <ul style="list-style-type: none"> • Statistiksidens layout og brugergrænseflade skal være nem at navigere rundt i for administrator • Systemet skal kunne genere statistikker hurtigt, selv med store mængder af data • Adgang for statistikkerne skal kun være tilgængelig for folk med admin rettigheder 		

Product Backlog (Hovedansvar: Rasmus)

Titel	Prioritet
Oprettelse af bruger	1
Administrator funktioner	2
Loginfunktion	3
Kategorier/Filtrering	4
Søgesystem	5
Online handel	6
Design af hjemmesiden	7
Statistik services	8
Historik over ændringer	9
Favoritliste	10

Business Process Modeling (se bilag 8) (Hovedansvar: Rasmus)⁶

I vores BPM ser vi hvordan bestillingen af en ordre forgår fysisk. Det er her vi ser hvad der sker efter en kunde, har trykket på ordre knappen. Der er to forskellige versioner. Den ene er når en privatperson opretter en ordre, og den anden er til virksomheder. Forskellen er at virksomheder har muligheden for levering, hvorimod private kunder ikke har. Starten af vores BPM er nok ikke nødvendig, men den giver alligevel et godt indblik i vores login- og ordre procedurer.

Domænemodel (se bilag 7) (Hovedansvar: Silas & Tan)

I udviklingen af vores webapplikation har vi udformet en simpel men struktureret domænemodel for at repræsentere de centrale elementer og deres relationer i vores system. Kernen i vores domænemodel er User-klassen, som agerer som en central aktør i alle systeminteraktioner. Herfra udspringer både administrative handlinger og brugerinitierede aktiviteter.

User-Klassen fungerer som grundlaget for alle brugere, da de alle er repræsenteret som instanser af User-klassen. En vigtig egenskab ved User-klassen er dens rollebaserede adgangskontrol. Oprindeligt var admin og User separate klasser hvor admin arvede fra User-klassen i stedet for roller. Men vi har valgt at simplificere det, ved at have roller bestemt af en 'string' i User-klassen. På den måde slipper vi for at have en næsten tom admin klasse, desuden er det også en god beslutning sikkerhedsmæssigt, da filer ved navn admin i programmet er mere sårbare over for hacker angreb. En bruger kan have en rolle som admin, der giver dem særlige beføjelser til at administrere produkter i systemet. Derudover kan brugere også have roller som almindelig bruger.

User har 0 eller flere relationer produkter, men der skal være minimum en User med rollen admin, for at kunne tilføje/ændre produkter. User har 0 eller flere ordre, men der skal være en User for at lave en ordre.

Product-Klassen repræsenterer de produkter, som Gadstrup Rosenplanteskole tilbyder. Et produkt har en række attributter såsom navn, kategori, farve, pris og tilgængelighed. En vigtig relation her er, at kun en admin-bruger kan oprette eller redigere produkter.

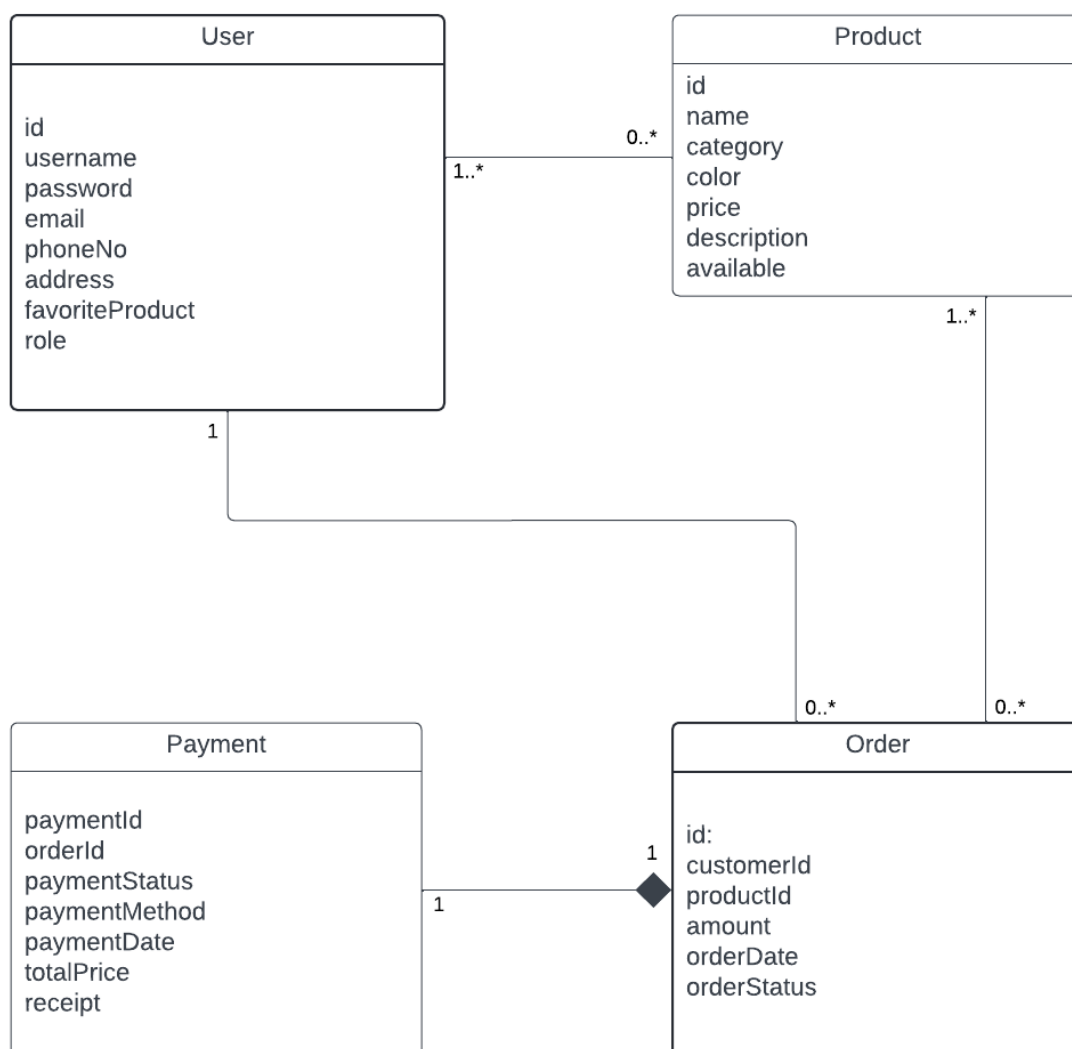
⁶ Se vedhæftet bilag i excel ark: Gadstrup_Rosenplanteskole_BPM.bpmn

Når en bruger ønsker at erhverve et produkt, oprettes en ordre i Order-Klassen. En ordre er forbundet til både brugeren, der placerede ordren, og det produkt, der blev købt. Det er også vigtigt at bemærke, at en ordre har en status, der afspejler dens status, såsom 'under behandling' eller 'fuldført'.

Når en ordre er afsluttet, genereres automatisk en betaling i Payment-Klassen. En betaling er altid knyttet til en ordre og indeholder oplysninger som betalingsmetode, betalingsstatus, beløb og dato for betalingen. Derfor har Order og Payment altid en 1 til 1 relation.

Denne domænemodel giver en klar struktur for vores system, der letter både udviklingen af softwaren og forståelsen af, hvordan brugerne interagerer med systemet. Ved at definere klart afgrænsede klasser og deres relationer kan vi sikre, at vores applikation fungerer korrekt og lever op til Gadstrup Rosenplanteskole forventninger.

Domænemodel



Enhanced Entity Relationship Diagram (Se bilag 9) (Hovedansvar: Tan & Zimon)

Vores EER-diagram (Enhanced Entity-Relationship Diagram) viser strukturen for Gadstrup Rosenplanteskoles database. Ved at analysere attributterne og relationerne i domænemodellen og oversætte dem til tabeller, kolonner og relationer har vi udvidet det til at indeholde flere oplysninger end et normalt ER-diagram.

Første Normalform opfyldes ved at fjerne alle de mange til mange relationer. Dette har vi gjort ved at lave 2 yderligere tabeller som hedder 'UserProduct' og 'OrderProduct'. Dette gør at begge mange til mange relationer forsvinder og derefter bliver det 1 til nul/mange.

Anden Normalform opfyldes ved at sikre at alle attributter i entiteten bruger HELE nøglen. Det er aktuelt i de entiteter der fremkommer, når man har sikret 1. NF

Tredje Normalform udvider 2NF. Det er det vi har gjort ved at lave en 'Address' tabel hvor byen er afhængigt af postnummeret som Primary key, da bynavnet ikke er en information som er afhængigt af user id.

Klassediagram (se bilag 10) (Hovedansvar: Lucas & Tan)

Vores klassediagram viser klasserne i systemet og deres strukturer af de objekt-orienteret systemer for Gadstrup Rosenplanteskolens hjemmeside.

For eksempel ved at analysere hvordan klassernes relationer er, kan vi identificere at klassen Payment er en komposition af Order, fordi Payment kan ikke eksistere uden at en Order eksisterer.

Vi har tænkt os at en User skal kunne tilføje/fjerne et produkt som favorit, som gør at der er en relation mellem User og Produkt-klassen. En bruger skal også have mulighed for at logge ind og ud.

En User har også altid en UserType, og det er fordi en user er enten en kunde eller en admin, hvilket har betydning for hvilke yderlige funktioner en bruger vil have på hjemmesiden. Mens en rolle som admin kan blive brugt af flere User.

Order har en del operationer der gør forskellige ting. PlaceOrder() opretter Payment hvor en kunde vil betale for deres bestilling, hvor CalculateTotal() er den samlet udregnet pris for produkterne i kurven, hvor prisen for de enkle produkter kommer fra getOrderDetail() og calculateTax() er hvor meget at den samlet udregnet pris er for moms.

Software Klassediagram (se bilag 11) (Hovedansvar: Silas)

Dette softwareklassediagram repræsenterer vores softwarearkitektur, fordi vi ønskede et diagram, der ligger så tæt som muligt på koden, før vi startede med at skrive den.

Den største forskel mellem de to diagrammer er vores repository-klasse, som indeholder den logik, som hele koden vil bygge på. CRUD (Create, Read, Update, Delete) funktioner skal kunne anvendes på alle klasser fra vores normale klassediagram undtagen UserType. En smart måde at opnå dette på er ved brug af generics. Dette gør, at alle klasserne kan hente deres CRUD-funktioner fra repository, og vi undgår at skulle skrive den samme funktion i alle klasserne.

Relationerne mellem User, Product, Order, og Payment forbliver dog de samme, som de er i vores normale klassediagram.

Design Sekvens Diagram (se bilag 12, 13, 14 og 15) (DSD) (Hovedansvar: Rasmus & Zimon)

Designsekvensdiagrammerne er udarbejdet ved hjælp af reverse engineering. Dette valgte vi at gøre, fordi vi begyndte at lave dem efter, at vi havde programmeret en stor del af systemet. Disse diagrammer vil være meget nyttige, hvis der er andre, som skal arbejde videre med systemet efter os, da det gør det nemmere at forstå, hvordan det hele er bundet sammen.

Login-funktionen starter med, at en bruger logger ind. Derefter bliver attributterne (Username og Password) verificeret i databasen. Herefter kan der ske en af to ting: Enten er verificeringen ikke succesfuld og returnerer derfor en fejlmeddelelse, eller verificeringen er succesfuld, og brugeren bliver returneret til forsiden som logget ind. Se Bilag (12)

Til sidst er der en logud-funktion. Her logger brugeren ud og bliver returneret til forsiden som logget ud.

Vores filterfunktion foregår over to omgange. Først vælger brugeren enten en rosetype, en farve eller begge dele på én gang. Når brugeren har valgt sine filtre og trykker "filtrer," går processen i gang. Den starter med en liste med alle roser. Derefter filtreres listen efter brugerens valgte rosetype (hvis der er en). Man ender med en ny liste sorteret efter rosetype. Den nye liste bliver så igen sorteret efter rosefarve (ud fra hvad brugeren har valgt). Til sidst står brugeren med en liste af roser, som matcher brugerens ønsker. Se (bilag 13)

Favoritfunktionen består af to funktioner: en "add to favorites" og en "remove from favorites". "Add"-funktionen foregår ved, at brugeren tilføjer et produkt, og produktet med det valgte ID bliver tilføjet til brugerens favoritliste.

"Remove"-funktionen foregår på samme måde. En bruger vælger at fjerne et produkt, og produktet med det tilhørende ID bliver fjernet fra brugerens favoritliste. Se (bilag 14)

System Sekvens Diagram (se bilag 15) (Hovedansvar: Zimon)

Systemsekvensdiagrammet (SSD) viser, hvordan man kan oprette, opdatere og slette produkter i vores system. For at oprette et produkt skal man give navn, pris, farve og kategori (type). Systemet returnerer de samme attributter, men giver automatisk også et unikt ID med. For at opdatere et produkt skal man angive ID'et på det produkt, man gerne vil ændre, og derefter kan man ændre navn, pris m.m. I dette eksempel bliver der dog givet en beskrivelse med. Man kan opdatere så mange ting ad gangen, som man vil. Det sidste, som diagrammet viser, er slettefunktionen, som er meget simpel: man angiver ID'et på det produkt, man gerne vil fjerne, og så fjernes det fra systemet.

Test Cases (se bilag 16) (Hovedansvar: Lucas, Tan & Zimon)⁷

Vi har lavet testcases for nogen af vores user stories. Dette skal sikre, at koden fungerer som forventet og reducerer risikoen for fejl. Dette giver os mulighed for at identificere og rette fejl tidligt i processen, hvilket kan spare os tid og ressourcer på lang sigt. Disse testcases er alle beskrevet meget nøjagtigt for at mindske risikoen for at testen af systemet ikke er tilstrækkelig. Et eksempel på dette er vores test af filtreringssystemet hvor vi prøver at filtrere efter farve først, så type og derefter farve og type samtidig. Dog skal man altid huske på at selvom man har været grundig vil der altid være en chance for at der er en kombination af ting man muligvis har overset, det handler om at finde den rette mængde tests som passer til hvor risikovillig man er i forhold til om der er fejl i systemet.

⁷ [Testcase for User Stories af GProser.xlsx](#)

Testcases er vedlagt som bilag i en Excel-fil og har flere faner for de forskellige user stories.

Vi har fået oprettet testcases for følgende områder: Loginfunktion, administratorfunktion, filtrering & online handel.

Vi havde oprindeligt gjort så alle kunne oprette en bruger, men af sikkerhedsmæssige årsager (f.eks. så bots ikke kan auto oprette brugere i systemet) har vi begrænset funktionen "Create User" for en almindelig bruger, og kun gjort det muligt for en admin kan oprette en bruger.

Brugere til test:

Brugernavn: admin

Kode: abe

Brugernavn: per

Kode: 1234

[Skitser/Mockups \(se bilag 17\) \(Hovedansvar: Alle\)](#)

Vi fik lavet en skitse af vores forside, som satte et godt udgangspunkt for designet af alle siderne. Der er en header og footer som kommer til at gå igen på alle siderne. Derudover har vi et baggrundsbillede, som også vil gå igen. Vi fandt også et bud på vores farve tema. Vi fik også lavede et udkast til designet af vores filtreringsfunktion.

Refleksion: Det endelige resultat af vores hjemmeside, endte med at se en smule anderledes ud fra vores mockups.

[Review \(se bilag 18\)](#)

Vi har aftalt at lave et review med en anden gruppe (Jonas, Mathias, Oliver, Tobias & Victor) som vi vil vedhæfte som bilag.

Notater fra review af vores projekt udført af den anden gruppe

- Mangler sidetal
- Risikoanalyse om forretningen kan være nede i PESTEL
- Product backlog oprettelse af bruger skal have højere prioritet
- Der skal være ikke databaser i vores DSD
- CRUD i SSD update skal opdatere id, name, price, description
- Business Processing Model: måske dele opret bruger funktion som admin og kunde
- Domænemodel i inception deck

Sprint:

SCRUM (Hovedansvar: Rasmus & Tan)⁸

Sprint 1: Vi holdt daglige stand-up møder hver dag. Rasmus var SCRUM master og var ansvarlig for at registrere, hvor meget der blev udført dagen før, i sprint backlog burndown chart.

I Sprint 2 arbejdede vi på at forbedre vores daglige stand-up møder og uddelegere opgaverne så vi ikke kom til at lave det samme. Her prøvede Tan at være SCRUM master. Vi mødtes alle online til møderne og anvendte de tre spørgsmål for hver daglige stand-up.

1. Hvad gjorde vi i går?
2. Hvad vil du gøre i dag?
3. Er der nogen hindringer på din vej?

Sprint 1 (Hovedansvar: Rasmus)⁹

Vi startede med at planlægge og estimere vores sprint 1. det gjorde vi ved at bryde vores userstories ned i tasks, og estimere de tasks. Sådan så vores sprint backlog ud:

1. Klasser
2. Database
3. EF Core
4. Opret Pages
5. Services/Repository
6. Tilføj CRUD til pages
7. Oprette bruger system
8. Login system
9. Roller
10. Design af hjemmesiden

Vi startede sprint 1 med at oprette tabeller.

Vi har oprettet tabeller via SQL Query's før vi oprettede de andre klasser, da vi skulle bruge EFCore Powertool til at lave Generated klasser.

Vi oprettede SQL Query's for det til vores database¹⁰

Derefter har vi lavet klasserne og skrevet koden i hver klasse.

Vi lavede det meste af sprint 1 sammen, så vi alle havde et overblik over hvor tingene lå i hver klasse og hvad der hang sammen med hinanden. Vi gjorde brug af Visualstudios liveshare funktion, så vi kunne være med allesammen. Vi havde estimeret de fleste ting til og tage langt længere end de egentlig gjorde hvilket resulteret i at vi endte med og tage hul på en masse andre ting. En anden ting som vi nok burde blive bedre til, er at starte dagen med at planlægge hvad vi vil gøre, da det egentlig var hvad vi følte gav mening i momentet vi tog os til og ikke i en planlagt

⁸ <https://blivprojektleder.dk/daily-standup/>

⁹ Se logbog (bilag 21)

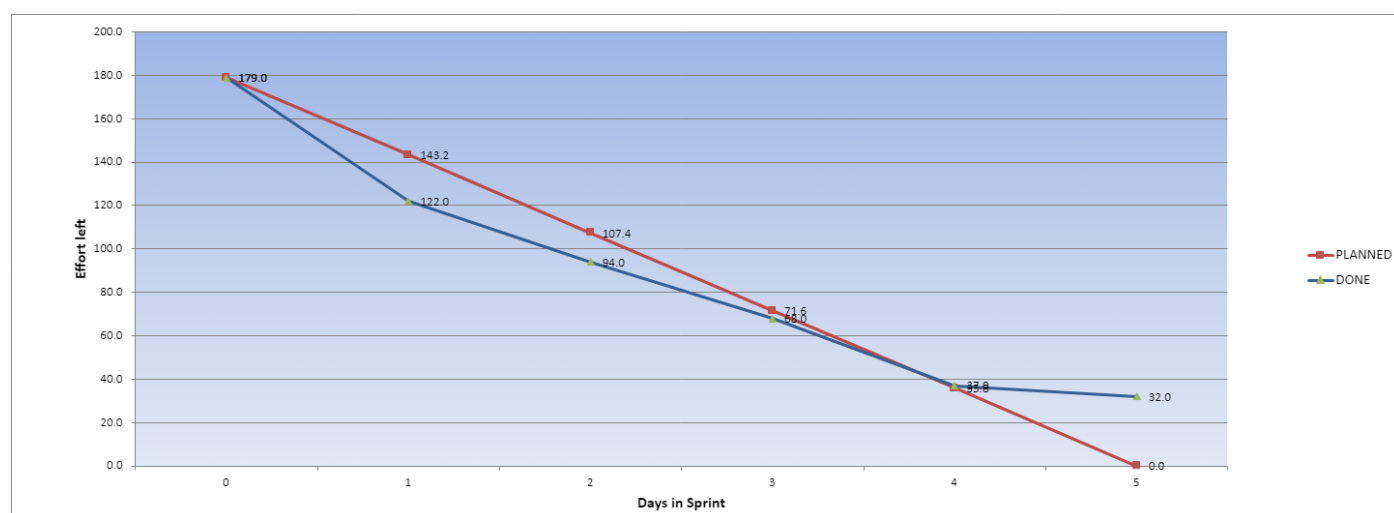
¹⁰ <https://github.com/perslaursen/OOProg22>

rækkefølge. Dog var sprintet stadig succesfuldt og vi kom længere end vi havde regnet med hvilket er positivt da det betyder, vi nu kan prøve på at lave nogle mere komplekse funktioner til hjemmesiden. De ekstra ting vi nåede, var at vi kan tilføje type/farve til vores roseprodukter som vi opretter, og at man kan tilføje billeder til et roseobjekt. Desuden fik vi også offentliggjort hjemmesiden.

I design delen kunne vi alle individuelt arbejde lidt med html/css, for at finde det bedst og mest optimale design taget ud fra vores mockups.

Vores leverancer for dette sprint er følgende: Man kan gå ind på en side over roser og se de roser som er oprettet af en admin. Udover dette fik vi også implementeret et login system, hvor en admin vil kunne se CRUD-funktionerne i Rose-siden, disse funktioner er ikke synlige for en almindelig bruger. Udover dette er hjemmesiden i sin helhed en leverance, man kunne vise til kunden. Vi havde lavet lidt basalt design som stadig havde mangler, men det er stadig en hjemmeside man kunne navigere rundt i.

Vi fulgte vores burn down chart efter planen i starten, vi var endda lidt foran de første par dage. Til sidst i sprintet gik det dog ikke lige så godt som planlagt. Grunden til det var at vi glemte at Udføre vores Testcases og Review som skulle være med i sprintet. Da vi opdagede fejlen, rykkede vi Udførelse af Testcases & Review over til vores Sprint 2



11

¹¹ Se bilag 19

Sprint 2 (Hovedansvar: Tan)¹²

Vi har overført de opgaver, der ikke blev færdiggjort i sprint 1, til sprint 2. Vi kunne også se hvad vores velocity var for vores sprint1. Det målte vi i timer og derfor har vi et estimat i timer for hvad vi kunne nå i sprint 2. Vores velocity var på 147 timer, derfor planlagde vi efter det i stedet for de 125.¹³

1. Testcases for sprint 1
2. Review for sprint 1
3. Data i view product
4. Data i All product
5. Tilføjelser til CRUD
6. Image Upload funktion
7. Filtrerings system
8. Design af hjemmesiden
9. Rydde op i koden
10. Udførelse af Testcases
11. Review for sprint 2

Vi startede med at få lavet de resterende ting fra sprint 1, da det var det vigtigste. Derefter fik vi lavede nogle små ting på kodedelen.

Vores 2 største tasks i andet sprint var Image Upload funktion og Filtreringssystemet. Modsat vores første sprint uddelegerede vi så opgaverne så 2-3 personer satte sig til at lave det ene, og en anden gruppe det andet.

Imageupload skal forstås som en tilføjelse til CRUD, da man nu kan angive et billede når man laver et nyt produkt og hvis man vil redigere et eksisterende produkt.

Udførelses af testcases blev gjort da vi endelig var færdige med sprint 2 for at se om alle funktionerne virkede. Vi fandt en fejl der gjorde at man kunne logge ind med enhver password for et brugernavn som var oprettet. Dette problem blev heldigvis løst.

Vores leverancer for vores sprint 2 er, at hjemmesiden nu kan oprette en rose hvor man kan uploade et billede af rosen og man kan redigere billedet, ved at uploade et nyt billede som erstatter det gamle.

Vi kan nu filtrere vores roser via rosefarver og –typer på rosesiden, så man kan vælge sin rose efter sit eget behov. Design af hjemmesiden og oprydning i koden gjorde vi alle individuel undervejs.

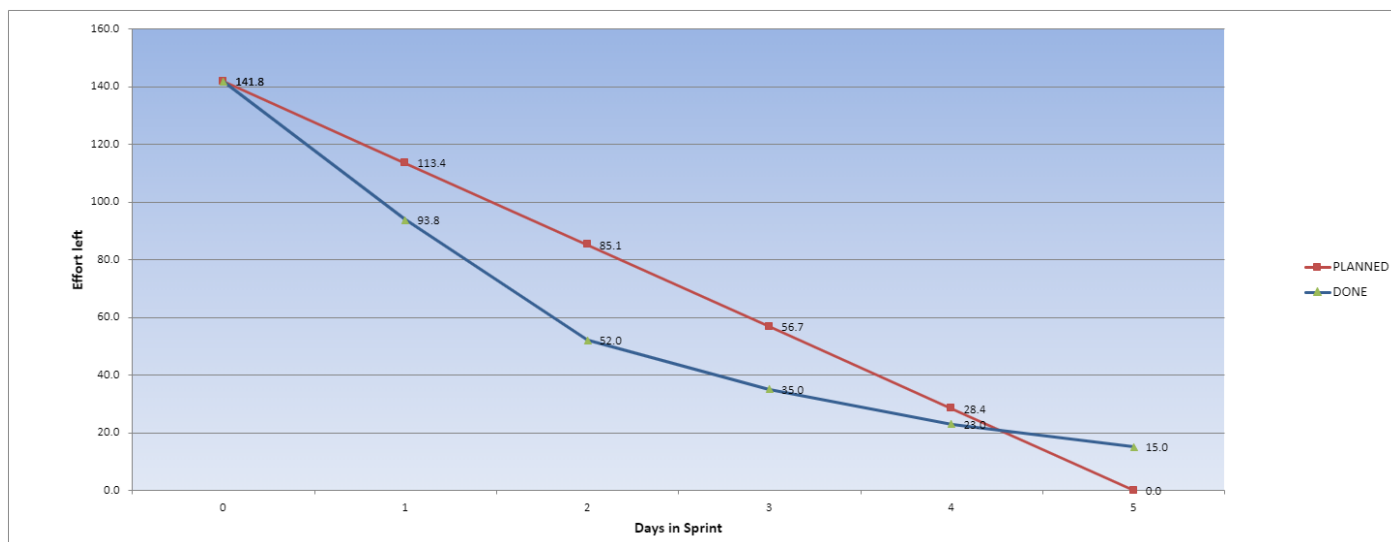
Sprint 2 gik også bedre end forventet i starten. På vores 4. og 5. dag af sprintet valgte vi at fokusere mere på rapporten, da vi mente at det vigtigste allerede var implementeret, derfor blev vi ikke helt færdige med vores Udførelses af Testcases og Review. Det er også derfor den blå graf i vores burndown chart bliver mere vandret til sidst.

Vi fik selvfølgelig færdiggjort de ting som vi ikke nåede, dog først efter sprintet.

Vores velocity for sprint 2 var 124,8. Dette er ikke så meget, men igen lavede vi på rapporten, som ikke talt med.

¹² Se logbog (bilag 21)

¹³ Jonathan Rasmusson The Agile Samurai (side 20) Pragmatic Programmers, LLC, 2010



14

Sprint 3 (Hovedansvar: Ingen)

Vi nåede ikke sprint 3 da vi ikke havde taget højde for at dele af sprint 1 og sprint 2 lå i weekenderne.

Vi havde planlagt disse dele i sprint3: Favoritfunktion, statistik services og søgefunktion

Afslutning af rapporten

Razor Pages og Kode: (Hovedansvar: Se navn under specifikt emne)¹⁵

Emne: Admin System (Hovedansvar: Rasmus & Silas)

Placering: /Pages/LogIn/LogInPage.cshtml.cs

Kodebeskrivelse:

Denne kode sørger for at når en admin logger ind, at de har admin rollen.

Dettes gøres ved at når en bruger logger ind, så bliver en if-statement kørt som tjekker om deres brugernavn er admin, og vis det er tilfældet skal denne bruger have admin rollen.

Derefter bliver deres rolle gemt i 'ClaimsIdentity' og til sidst returneret.

Teoretisk hvis en bruger selv skal kunne oprette en profil, kunne de navngive sig selv admin og have adgang til alle admin specifikke funktioner. I vores nuværende version af websitet der det dog kun muligt for admins at oprette nye brugere.

¹⁴ Se bilag 19

¹⁵ Alle screen-shots af koden, kommer fra vores egen kode i projektet (se vedhæftet Visual Studio solution)

```

51 private ClaimsPrincipal BuildClaimsPrincipal(User user)
52 {
53     // Build Claims
54     List<Claim> claims = new List<Claim> {
55         new Claim(ClaimTypes.Name, user.UserName)
56     };
57
58     if (user.UserName == "admin")
59         claims.Add(new Claim(ClaimTypes.Role, "admin"));
60
61     // Create claims-based identity
62     ClaimsIdentity claimsIdentity = new ClaimsIdentity(
63         claims,
64         CookieAuthenticationDefaults.AuthenticationScheme);
65
66     // Create and return principal
67     return new ClaimsPrincipal(claimsIdentity);
68 }
69

```

Emne (Hovedansvar: Lucas, Tan & Zimon): Links og Admin specifikke funktioner.

Placering: /Pages/Shared/_Layout.cshtml

Kodebeskrivelse:

Koden her er en del af Headeren og er links til de andre sider på hjemmesiden. Desuden er det sidste link inde i noget kode som gør at kun en admin har adgang til det.

Linksene bliver først specificeret med `<li class="nav-item">` som fortæller at det der skal stå, skal være inde i en knap, og bliver påvirket af `<style>` som står tidligere i koden (billede 2). Den næste linje fortæller at det er et link og hvor brugeren bliver ført til, og teksten for linket.

Det sidste link er sat ind i et if-statement som tjekker om man er logget ind på en profil som har rollen "admin", hvis bruger er logget ind som admin, vil dette link stå i Header som adminen så kan bruge.

Denne kode som tjekker om man er logget ind som admin bliver brugt andre steder til hjemmesiden for admin specifikke funktioner så som knapper til at oprette og slette roser.

```

52 <div class="navbar-collapse collapse d-sm-inline-flex justify-content-between">
53 <ul class="navbar-nav flex-grow-1">
54 <li class="nav-item">
55 <a class="nav-link text-light" asp-area="" asp-page="/About">About</a>
56 </li>
57 <li class="nav-item">
58 <a class="nav-link text-light" asp-area="" asp-page="/Products/All">Roser</a>
59 </li>
60 <li class="nav-item">
61 <a class="nav-link text-light" asp-area="" asp-page="/Privacy">Privacy</a>
62 </li>
63 <li class="nav-item">
64 <a class="nav-link text-light" asp-area="" asp-page="/FAQ">FAQ</a>
65 </li>
66 @if
67 // Er kun synligt når man er logget ind med en profil med rollen Admin.
68 if (Context.User.Claims.FirstOrDefault(c => c.Type == ClaimTypes.Role)?.Value == "admin")
69 {
70 <li class="nav-item">
71 <a class="nav-link text-light" asp-area="" asp-page="/Admin/All">Admin</a>
72 </li>
73 }
74 }
75 </ul>
76
14 <style>
15 .navbar-brand {
16 font-size: 30px; /* Ændre til ønsket størrelse */
17 }
18
19 .nav-link {
20 font-size: 24px; /* Ændre til ønsket størrelse */
21 }
22
23 .header-text {
24 color: white; /* Gør teksten hvid */
25 font-weight: bold; /* Gør teksten fed */
26 font-size: 20px; /* Ændr til ønsket størrelse */
27 }
28
29 .nav-item {
30 display: inline-block;
31 vertical-align: middle;
32 }
33
34 .nav-link {
35 font-size: 20px;
36 white-space: nowrap;
37 display: inline-block;
38 vertical-align: middle;
39 }
40 </style>

```

Emne (Hovedansvar: Lucas, Tan & Zimon): Filteringssystem

Kodebeskrivelse: Vi implementerede et filter baseret på rosernes farver og typer. For mere information om, hvordan farver og typer fungerer, se klasserne `RoseColor` og `RoseType` under mapperne `/Models/Generated`.

Denne kode opretter en liste over roser, der matcher de valgte farver og typer fra filtreringen. Da det er muligt at vælge flere farver og typer samtidigt, er `colorId` og `typeld` implementeret som arrays. Dette gør det muligt at have en liste af valgte farve- og type-id'er for at finde roser, der passer til en eller flere af de valgte kriterier.

Derudover er der to metoder, der returnerer en `IEnumerable` af objekter, som repræsenterer alle de tilgængelige farver og typer for roserne.

```

5
6  // Forklaring: Denne kode returnere listen af Roser der passer til de valgte filter.
7  // 'Int'-ene bliver brugt som Arrays.
8  2 references
   List<Product> MatchingFilters(int[] colorIds, int[] typeIdIds);
9  2 references
   IEnumerable<RoseColor> GetColors();
10 2 references
   IEnumerable<RoseType> GetTypes();
11 }

```

Inde i vores GPRoser\Pages\Products\All.cshtml.cs er der kode, der understøtter filtreringssystemet. Først er der public int[] arrays under [BindProperty], hvor (SupportGets = true) muliggør binding af integers fra en HTTP GET-anmodning. Derudover er der to lister (ColorSelectList og TypeSelectList), som giver programmet en oversigt over de mulige farver og typer, man kan vælge.

Inde i OnGet()-metoden udføres filtreringen ved først at oprette en liste over de valgte farver og typers id'er og navne. if-sætningen tjekker, om der er mindst én farve eller type valgt og at de ikke er null. Hvis dette er tilfældet, gemmer den en liste kaldet Data. Data bruger listegenereringen MatchingFilters(int[] colorIds, int[] typeIdIds) fra IProductDataService, hvor listen SelectedColorIds sendes som colorIds og listen SelectedTypeIds sendes som typeIdIds. På denne måde genereres en liste over de valgte roser.

Vores else-statement aktiveres kun, hvis ingen farver og typer er blevet valgt. I dette tilfælde vil Data-listen indeholde alle roserne i databasen.

```

16
17 // Under dette er kode som støtter filtreringssystemet
18 [BindProperty(SupportsGet = true)]
19 3 references
   public int[] SelectedColorIds { get; set; }
20 [BindProperty(SupportsGet = true)]
21 3 references
   public int[] SelectedTypeIds { get; set; }
22
23 2 references
   public SelectList ColorSelectList { get; set; }
24 2 references
   public SelectList TypeSelectList { get; set; }
25
26 1 reference
   public override void OnGet()
27 {
28     ColorSelectList = new SelectList(_dataService.GetColors(), "Id", "Name");
29     TypeSelectList = new SelectList(_dataService.GetTypes(), "Id", "Name");
30
31     if ((SelectedColorIds != null && SelectedColorIds.Any()) || (SelectedTypeIds != null && SelectedTypeIds.Any()))
32     {
33         Data = _dataService.MatchingFilters(SelectedColorIds, SelectedTypeIds);
34     }
35     else
36     {
37         Data = _dataService.GetAll().ToList();
38     }

```

For at programmet kan hente roser fra Databasen rigtig, har vi en baseklasse som bruger lambda udtrykket Return context.Set<Product>().Include(c => c.Orders).Include(c => c.Color).Include(c => c.Type); hvilket sørger for at koden inkludere rosernes relaterede data med en enkel forespørgsel ved hjælp af ".Include".

```

17 4 references
18 protected override IQueryable<Product> GetAllWithIncludes(DbContext context)
19 {
20     return context.Set<Product>()
21         .Include(c => c.Orders)
22         .Include(c => c.Color)
23         .Include(c => c.Type);
24 }

```

Inde i EFCProductDataService oprettes listen af roser, der passer til de valg, der er foretaget i filteret. var query gemmer de fundne data og anvender GetAllWithIncludes-metoden til at hente alle roserne sammen med deres tilknyttede data.

I vores if-statement sikrer colorIds != null, at betingelsen ikke udføres, hvis ingen farver er blevet valgt til filtreringen, mens colorIds.Any() sikrer, at alle valgte farver behandles, uanset om det er en eller flere.

query = query.Where(p => colorIds.Contains(p.ColorId.Value)) filtrerer roserne, så kun de roser, der matcher en af de valgte farver, inkluderes.

Den næste if-statement gør det samme, men for typer i stedet for farver.

```

25
26 // Brugte Lambda udtryk til filteringsystem
27 2 references
28 public List<Product> MatchingFilters(int[] colorIds, int[] typeIds)
29 {
30     var query = GetAllWithIncludes(_context);
31
32     if (colorIds != null && colorIds.Any())
33     {
34         query = query.Where(p => colorIds.Contains(p.ColorId.Value));
35     }
36
37     if (typeIds != null && typeIds.Any())
38     {
39         query = query.Where(p => typeIds.Contains(p.TypeId.Value));
40     }
41
42     return query.ToList();
43 }
44
45 2 references
46 public IEnumerable<RoseColor> GetColors()
47 {
48     return _context.RoseColors.ToList();
49 }
50
51 2 references
52 public IEnumerable<RoseType> GetTypes()
53 {
54     return _context.RoseTypes.ToList();
55 }

```

Emne (Hovedansvar: Rasmus & Silas): ImageFile upload funktion¹⁶

Kodebskrivelse : For at lave en funktion til at uploade billeder af roser, var vi nødt til først at tilføje følgende properties til **Product.cs** klassen: **public string ImagePath { get; set; }** og **public IFormFile ImageFile { get; set; }**. Derefter tilføjede vi ImagePath til Update-metoden i Product Extended.

```
[Table("Product")]
21 references
public partial class Product
{
    [Key]
    15 references
    public int Id { get; set; }

    [Required]
    15 references
    public string Name { get; set; }

    12 references
    public double Price { get; set; }

    12 references
    public string ImagePath { get; set; }

    [NotMapped]
    12 references
    public IFormFile ImageFile { get; set; }
}
```

```
2 references
public void Update(Product tOther)
{
    Name = tOther.Name;
    Price = tOther.Price;
    ImagePath = tOther.ImagePath;
    ColorId = tOther.ColorId;
    TypeId = tOther.TypeId;
}
```

Efter dette tilføjede vi en kolonne til SQL-databasen for ImagePath. Upload af billeder sker i vores CreateModel for Products i OnPostAsync-metoden.

```
if (Data.ImageFile != null)
{
    var uploadsFolder = Path.Combine(_webHostEnvironment.WebRootPath, "uploads");
    var uniqueFileName = Guid.NewGuid().ToString() + "_" + Data.ImageFile.FileName;
    var filePath = Path.Combine(uploadsFolder, uniqueFileName);
    using (var fileStream = new FileStream(filePath, FileMode.Create))
    {
        await Data.ImageFile.CopyToAsync(fileStream);
    }
    Data.ImagePath = "/uploads/" + uniqueFileName;
}
```

Først kontrollerer vi, om der er en billedfil med `if (Data.ImageFile != null)`. Denne betingelse sikrer, at der kun udføres upload, hvis der er valgt en fil.

Derefter opsættes mappen, hvori billederne skal gemmes:

```
var uploadsFolder = Path.Combine(_webHostEnvironment.WebRootPath, "uploads");
```

Denne linje kombinerer webroden (`_webHostEnvironment.WebRootPath`) med en undermappe kaldet "uploads" for at bestemme stien, hvor billedet skal gemmes.

For at undgå navnekonflikter genereres et unikt filnavn ved at kombinere en GUID med det oprindelige filnavn:

```
var uniqueFileName = Guid.NewGuid().ToString() + "_" + Data.ImageFile.FileName;
```

Så kombinerer vi uploadmappen med det unikke filnavn i denne linje.

```
var filePath = Path.Combine(uploadsFolder, uniqueFileName);
```

Filen skal derefter kopieres til serveren.

```
using (var fileStream = new FileStream(filePath, FileMode.Create))
```

¹⁶ <https://www.youtube.com/watch?v=K61wZruCJl8&t=695s>


```
{  
    await Data.ImageFile.CopyToAsync(fileStream);  
}
```

Her åbnes en filstream til den bestemte sti, og filen kopieres asynkront fra det midlertidige lagringssted til serverens filsystem.

Til sidst opdateres dataobjektet med stien til det uploadede billede, så denne information kan gemmes i databasen eller bruges i applikationen:

```
Data.ImagePath = "/uploads/" + uniqueFileName;
```

Derudover skulle vi også lige tilføje funktionen til Create.cshtml

```
<div class="form-group mb-3">  
    <label asp-for="@Model.Data.ImageFile" class="control-label"></label>  
    <input asp-for="@Model.Data.ImageFile" type="file" class="form-control" />  
    <span asp-validation-for="@Model.Data.ImageFile" class="text-danger"></span>  
</div>
```

Label

Denne linje genererer en etikette for filuploadfeltet.

Input felt

Input feltet tillader brugeren at vælge en fil fra deres computer. Asp-for binder inputfeltet til ImageFile egenskaben i modellen. Type=file angiver, at dette input er et filuploadfelt.

Validation

Denne linje tilføjer et valideringsfelt, der viser eventuelle valideringsfejl for ImageFile-egenskaben. Klassen text-danger anvender Bootstrap-styling for at vise fejlmeddelelser i rød tekst.

Denne User tabel er lavet, så vi kan oprette en bruger med et password. Hver user har et unikt id og hvis man vil tilføje flere users, vil de blive tilføjet i databasen udefra sidste tilføjet id (1, 1)

```
CREATE TABLE [dbo].[User] (  
    [Id] INT IDENTITY (1, 1) NOT NULL,  
    [UserName] NVARCHAR (MAX) NOT NULL,  
    [Password] NVARCHAR (MAX) NOT NULL,  
    PRIMARY KEY CLUSTERED ([Id] ASC)  
);
```

Product tabellen er lavet til at gemme oplysninger om de produkter, som Gadstrup Rosenplanteskole sælger

Hvert produkt har et unikt ID, der automatisk forøges for når et nyt bliver tilføjet, hvilket sikrer at hvert produkt kan identificeres entydigt. Produktets navn gemmes i kolonnen 'Name', denne kolonne kan ikke være en null-værdi, da hvert produkt skal have et navn.

Prisen på hvert produkt gemmes i kolonnen 'Price', som også er en nødvendig værdi og derfor ikke tillader null-værdier. For at kunne vise billeder af produkterne indeholder tabellen en kolonne 'ImagePath', der gemmer stien til produktbilledet. Denne kolonne er valgfri, hvilket betyder, at et produkt kan eksistere uden et billede.

Tabellen indeholder også to 'Foreign keys': 'colorId' og 'typeId'. Disse nøgler refererer til henholdsvis 'RoseColor' og 'RoseType' tabellerne. Ved at bruge 'Foreign Keys' sikres referentiel integritet, hvilket betyder, at enhver id-værdi skal matche en eksisterende id-værdi i deres tilsvarende tabeller. Dette sikrer, at kun de rigtige farver og typer kan gives til produkterne.

Sammenfattende giver Product tabellen en god struktur til at gemme alle nødvendige oplysninger om produkterne, samtidig med at den opretholder dataintegritet gennem brugen af 'Foreign Keys' og nødvendige felter. Denne tabel er essentiel for at kunne oprette og vedligeholde produktkataloget på Gadstrup Rosenplanteskoles webshop.

```
CREATE TABLE [dbo].[Product] (  
    [Id] INT IDENTITY (1, 1) NOT NULL,  
    [Name] NVARCHAR (MAX) NOT NULL,  
    [Price] FLOAT (53) NOT NULL,  
    [ImagePath] NVARCHAR (MAX) NULL,  
    [colorId] INT NULL,  
    [typeId] INT NULL,  
    PRIMARY KEY CLUSTERED ([Id] ASC),  
    CONSTRAINT [FK_Product_RoseColor] FOREIGN KEY ([colorId]) REFERENCES [dbo].[RoseColor] ([Id]),  
    CONSTRAINT [FK_Product_RoseType] FOREIGN KEY ([typeId]) REFERENCES [dbo].[RoseType] ([Id])  
);
```

RoseColor og RoseType Tabel:

Hver tabel har et Id og et Navn, hvor hvert Id er 'Foreign Key' i Product Tabellen.

Disse tabeller er oprettet så vi kan bruge dem, når vi skal hente information til Product tabellen til f.eks. oprettelse eller opdatering af et produkt. Id'et fra enten RoseColor eller RoseType har alle nødvendige informationer for at lave et produkt. Desuden bruges RoseColor og RoseType også i filteringsystemet, hvor man kan se alle navnene ved at hente deres ID fra databasen.

¹⁷ Se vedhæftet Visual Studio solution

```
CREATE TABLE [dbo].[RoseColor] (
    [Id] INT NOT NULL,
    [Name] NVARCHAR (40) NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC)
);
```

```
CREATE TABLE [dbo].[RoseType] (
    [Id] INT NOT NULL,
    [Name] NVARCHAR (40) NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC)
);
```

Vi har tilføjet Customer tabellen til videreudvikling, men har ikke tilføjet nogle funktioner til den endnu i Razor Pages. Denne tabel er designet til at gemme oplysninger om kunder, inklusive deres navn, e-mail og adresse.

```
CREATE TABLE [dbo].[Customer] (
    [Id] INT IDENTITY (1, 1) NOT NULL,
    [Name] NVARCHAR (MAX) NOT NULL,
    [Email] NVARCHAR (MAX) NOT NULL,
    [Address] NVARCHAR (MAX) NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC)
);
```

Vi har ikke rigtigt brug for Order tabellen, da vi ikke har tilføjet nogle funktioner til den endnu i Razor Pages. Denne tabel er beregnet til at gemme ordrer, hvor hver ordre indeholder en henvisning til en kunde og et produkt samt mængden af produktet i ordren.

```
CREATE TABLE [dbo].[Order] (
    [Id] INT IDENTITY (1, 1) NOT NULL,
    [CustomerId] INT NOT NULL,
    [ProductId] INT NOT NULL,
    [Amount] INT NOT NULL,
    PRIMARY KEY CLUSTERED ([Id] ASC),
    CONSTRAINT [FK_Order_Customer_CustomerId] FOREIGN KEY ([CustomerId]) REFERENCES [dbo].[Customer] ([Id]),
    CONSTRAINT [FK_Order_Product_ProductId] FOREIGN KEY ([ProductId]) REFERENCES [dbo].[Product] ([Id])
);
```

Samlet oversigt:

Tabellerne indeholder følgende oplysninger: produkter, brugere, rosetyper, rosefarver, kunder og ordrer.

Relationerne mellem disse tabeller sikrer, at data kan hentes, manipuleres effektivt, især når der oprettes eller opdateres produkter, og når de filtreres efter rosetyper og -farver.

Selvom Customer og Order tabellerne, er der til fremtidig brug, er deres funktionalitet ikke implementeret endnu i Razor Pages.

Refleksion (Hovedansvar: Alle)

Introduktion:

Dette projekt har omfattet udviklingen af en webapplikation for Gadstrup Rosenplanteskole. Formålet var at modernisere deres eksisterende hjemmeside og forbedre brugervenligheden ved at tilføje nye funktioner som produktfiltrering og billedupload. Vi arbejdede i flere sprints og anvendte både vandfaldsmodellen og SCRUM-metoden for at sikre en struktureret og fleksibel tilgang til udviklingen.

Beskrivelse:

Under projektets forløb startede vi med en grundig planlægning og estimering af vores opgaver, hvorefter vi delte dem op i mindre, håndterbare tasks. I sprint 1 fokuserede vi på at oprette databasen og implementere grundlæggende funktioner som CRUD, login- og brugeroprettelsessystemer. I sprint 2 fortsatte vi med at forbedre og tilføje nye funktioner såsom billedupload og filtreringssystemet.

Vi holdt daglige stand-up møder for at sikre, at alle var opdaterede og for at identificere eventuelle hindringer. Vores arbejde blev dokumenteret og evalueret løbende, hvilket hjalp os med at justere og optimere vores processer.

Analyse:

Hvad gik godt?

Projektet blev gennemført med succes, og vi nåede mange af de planlagte mål inden for tidsrammen. Vores samarbejde og kommunikation fungerede godt, hvilket var afgørende for projektets fremdrift. Vi var i stand til at implementere de fleste af de ønskede funktioner, og vi lærte at håndtere både individuelle og gruppeopgaver effektivt.

Hvad gik ikke som forventet?

En af udfordringerne var at estimere den nødvendige tid korrekt. Vi undervurderede nogle opgaver og overestimerede andre, hvilket påvirkede vores sprintplanlægning. Derudover mødte vi tekniske udfordringer, især med integrationen af billedupload-funktionen og filtreringssystemet, som tog længere tid at implementere end forventet. Ydermere fandt vi også 2 dage før vores afleveringsfrist en fejl i koden der gør det muligt at logge ind selvom man skriver adgangskoden forkert. Dette problem blev heldigvis opdaget og løst takket være vores test cases.

Faglige indsigter:

Hvad har vi lært?

- Vigtigheden af detaljeret planlægning og nøjagtig tidsestimering.
- Effektiv anvendelse af SCRUM-metoden til at håndtere komplekse projekter.
- Håndtering af tekniske udfordringer, især i forbindelse med databaser og CRUD-operationer.

Personlige indsigter:

- Betydningen af god kommunikation og samarbejde i et team.
- Hvordan man håndterer stress og pres under stramme deadlines.
- Vigtigheden af løbende at evaluere og justere arbejdsmetoder for at opnå de bedste resultater.

Fremtidige handlinger:

Fremover vil vi fokusere på mere præcise tidsestimater og bedre planlægning af opgaver. Vi vil også styrke vores kommunikationsstrategier for at sikre, at alle teammedlemmer er fuldt opdaterede og klar over deres ansvar.

Forbedringer til fremtidige projekter

- Implementering af mere detaljerede testcases tidligt i projektet for at identificere og rette fejl hurtigt.
- Bedre fordeling af opgaver for at undgå overbelastning af enkelte teammedlemmer.
- Tidlig involvering af alle interessenter for at sikre, at alle krav og forventninger er klart definerede fra starten.

Projektet har givet os værdifulde erfaringer både teknisk og organisatorisk. Selvom vi stødte på udfordringer undervejs, formåede vi at levere en forbedret webapplikation til Gadstrup Rosenplanteskole, der opfylder de fleste af deres behov. Vi har lært at arbejde effektivt som et team og at håndtere de udfordringer, der opstår i løbet af et projekt. Disse erfaringer vil være uvurderlige i fremtidige projekter.

Konklusion (Hovedansvar: Alle)

Udviklingen af en webapplikation for Gadstrup Rosenplanteskole har været udfordrende proces, hvor vi har kombineret forskellige metoder og teknologier for at levere en funktionel og brugervenlig løsning. Projektet startede med en grundig forretningsanalyse og planlægning ved hjælp af modeller som PESTEL, SWOT, og TOWS, hvilket gav os en solid forståelse af virksomhedens behov og mål.

Vores afgrænsning af projektet sikrede, at vi fokuserede på de vigtigste funktioner som produktfiltrering og billedupload, mens vi undgik at inkludere onlinehandel og mobilapplikationer. Ved at anvende både Vandfaldsmetoden og SCRUM kunne vi strukturere vores arbejde effektivt og tilpasse os løbende udfordringer.

Gennem hele projektet har vi lagt vægt på at levere høj kvalitet og brugervenlighed. Vi har oprettet en ny hjemmeside med et forbedret design, der gør det nemmere for kunder at finde og vælge roser baseret på forskellige kriterier som farve og type. Vores implementering af CRUD-funktionalitet og admin-roller sikrer, at virksomheden nemt kan vedligeholde og opdatere deres produktkatalog.

Risikoanalysen hjalp os med at identificere og håndtere potentielle trusler, mens vores kontinuerlige review-processer sikrede, at vi kunne forbedre og justere vores arbejde undervejs. Selvom vi stødte på tekniske udfordringer, især med integration af billedupload og filtreringssystemer, formåede vi at overkomme disse og levere en funktionel løsning inden for tidsrammen.

Desuden fandt vi også to dage før vores afleveringsfrist en kritisk fejl i koden, der gjorde det muligt at logge ind, selvom man skrev adgangskoden forkert. Hvilket understreger vigtigheden af grundige tests og reviews gennem hele udviklingsprocessen. Denne fejl kunne være blevet opdaget og rettet tidligere ved hjælp af mere omfattende test cases, hvilket viser nødvendigheden af detaljeret testplanlægning og udførelse fra projektets begyndelse.

I refleksionsafsnittet har vi vurderet vores arbejdsmetoder, identificeret vores styrker og svagheder, og lagt en plan for fremtidige projekter. Vi har lært vigtige lektioner om tidsestimering, planlægning og samarbejde, som vil være uvurderlige i fremtidige projekter.

Samlet set har vi med succes leveret en ny og forbedret webapplikation til Gadstrup Rosenplanteskole, der ikke blot opfylder deres nuværende behov, men også giver dem mulighed for at udvide og tilpasse sig i fremtiden. Projektet har styrket vores færdigheder inden for softwareudvikling og projektledelse.

Kildeliste

Gadstrup Rosenplanteskolens hjemmeside: <https://gproser.dk/>

Investeringskalkule: [Gadstrup Rosenplanteskole Investeringskalkule.xlsx](#)

Inception Deck Section 5. Taget fra vores 4 ugers semester projekt [4 ugers semester projekt.docx](#)

User Story: Non Funktionelle Krav har brugt lidt inspiration fra vores [4 ugers semester projekt.docx](#) & "Online Examination System"

Domænemodel, Klassemodel og ER Diagram: https://lucid.app/lucidchart/3dd059d3-c0b6-499f-a680-1cbbe86f5770/edit?invitationId=inv_46957be0-b16c-421e-9ec5-457734af5e75

Software klassediagram: https://lucid.app/lucidchart/1ca4021f-0bbc-4d52-8313-84a911336395/edit?viewport_loc=-2131%2C-1200%2C4217%2C1701%2C0_0&invitationId=inv_3ed2eb6f-b467-42b1-826f-4a9f7859da64

Design Sekvens Diagram (DSD):

AdministratorFunktioner: https://lucid.app/lucidchart/59e5d017-d25f-4289-93c4-b0003746e4ca/edit?viewport_loc=-312%2C-570%2C2503%2C1119%2C0_0&invitationId=inv_f8c0e3e3-413c-4a98-a039-d0d09ad455dc

Favoritliste: https://lucid.app/lucidchart/8c2b7ba0-9bdc-4f75-85cd-6c5c472425d6/edit?viewport_loc=-10%2C-11%2C1555%2C695%2C0_0&invitationId=inv_e93db246-f1d2-47be-8cfb-b04647cd682a

Login: https://lucid.app/lucidchart/aa538401-f427-4c4a-a773-f9c2dd83c88c/edit?viewport_loc=-779%2C-381%2C3745%2C1834%2C0_0&invitationId=inv_67ad2e45-1429-4b5d-a216-3fbccb3510e3

System Sekvens Diagram (SSD):

https://lucid.app/lucidchart/42bdcb6e-f1d3-4252-8c30-6c41842cc9ff/edit?view_items=V4Nzzamua_oB&invitationId=inv_5f9afc31-9f10-42bb-a750-a9185806b691

DSD for login og filter:

https://lucid.app/lucidchart/aa538401-f427-4c4a-a773-f9c2dd83c88c/edit?viewport_loc=1496%2C-123%2C2742%2C1343%2C0_0&invitationId=inv_67ad2e45-1429-4b5d-a216-3fbccb3510e3

Testcases: [Testcase for User Stories af GProser.xlsx](#)

Sprint 1: [Gadstrup Rosenplanteskole sprint 1.xlsx](#)

Sprint 2 [Gadstrup Rosenplanteskole sprint 2.xlsx](#)

Razorpages: <https://github.com/perslaursen/OOProg22> Kode inspiration brugt fra RazorPagesOpgaver2sem.docx i OOProg22-main (WebShopRPv5)

Mockups: [2. Semester Mock-Ups](#)

Image file upload funktion: <https://www.youtube.com/watch?v=K61wZruCJl8&t=695s>

Bilag

Bilag 1: Cost benefit excel ark

Gadstrup Rosenplanteskole

Forkalkulation for en time	
kr.	
Timeløn for en medarbejder	160,00
Antal Medarbejdere	5,00
Timeløn for alle medarbejdere	800,00

Investering	
Antal timer (Estimat)	250
Ekstra salg pga. nye kunder	30.000,00
Kontante kapacitetsomkostninger, kr.	10.000
Investering, kr.	200.000
Scrapværdi, kr.	0
Levetid, år	10
Kalkulationsrente, % p.a.	6%

År	Investering og scrapværdi	Indbetaling	Netto- betalingsstrøm
	kr.	kr.	kr.
0	-210.000		-210.000
1		30.000	30.000
2		30.000	30.000
3		30.000	30.000

4		30.000	30.000
5		30.000	30.000
6		30.000	30.000
7		30.000	30.000
8		30.000	30.000
9		30.000	30.000
10		30.000	30.000
I alt			90.000

År	Investering og scrapværdi	Indbetaling	Netto- betalingsstrøm	Faktor	Nutidsværdi
	kr.	kr.	kr.		kr.
0	-210.000		-210.000	1,000000	-210.000
1		30.000	30.000	0,943396	28.302
2		30.000	30.000	0,889996	26.700
3		30.000	30.000	0,839619	25.189
4		30.000	30.000	0,792094	23.763
5		30.000	30.000	0,747258	22.418
6		30.000	30.000	0,704961	21.149
7		30.000	30.000	0,665057	19.952
8		30.000	30.000	0,627412	18.822
9		30.000	30.000	0,591898	17.757
10		30.000	30.000	0,558395	16.752
I alt			90.000		10.803

Business Model Canvas bilag

Key partners - Sælger roser til Coop, Meny, Silvan & Superbrugsen, Løvbjerg Sælger til mange andre planteskoler i hele Danmark	Key activities - Viden om hvornår forskellige typer roser kan plantes og hvornår de så blomstrer. - Viden om sygdomme og pest for roser. -Salg af Roser	Value propositions - Bredt udvalg af kvalitets roser. - Hyggelig lokal virksomhed som kender stamkunder. - Den åbne Rosenmark (Den 15/7-31/10 kl. 9-18) - Tilbyde sjældne/specielle roser som ikke er let tilgængelige andre steder - Rådgivning og vejledning om de specielle roser	Customer relationships - Kunderne forventer at der er et stort udvalg af roser de kan købe fra. - Som et mindre firma er mund til mund vigtigt Kunderne er ofte lokale og er gentagende kunder. Det er også de kunder som laver m2m. Channels - En hjemmeside på internettet. - Roserne bliver købt direkte hos virksomheden (fysisk eller telefonisk) - mund til mund mellem kunderne for at nå ud til flere (reklamerings) De laver ikke selv den store markedsføring, de laver små ting som at hænge sedler op i supermarkeder og andre gratis steder.	Customer segments - Folk som elsker blomster, specielt roser. - Folk som vil købe roser for en romantisk partner. - Kunder som vil købe roser til specielle begivenheder - Den åbne rosenmark tilbyder aktivitet for blomsterentusiaster , familier og par. - Åben Rosenmark: Lokale kunder kan møde op og plukke/købe roser (Se 60.000 haveroser i blomst i ca. 300 sorter.) Der er både b2c og c2c
Cost structure - Grundstammer til at gro roserne. - Pesticider til at beskytte roserne fra insekter. - Løn - Benzin til køreredskaber til at plante roserne og vedligeholde. - Ejendomsskat			Revenue streams - Sælge roser til kunder (B2C) - Sælge roser til forretninger (B2B) - Betalings metoder: Kontanter, mobilepay.	

Pestel Analyse bilag

Politics	Economic	Social
<p>Staten kan lave love som påvirker hvordan man må drive en virksomhed. Bæredygtighed, miljøbeskyttelse</p> <p>Afgifter på pesticider, kan påvirke virksomhedens omkostninger og indtægter</p> <p>Lovgivning og reguleringer vedrørende import og eksport af planter (Grundstammer)</p>	<p>Renten har stor betydning for vores virksomhed.</p> <p>Hvis vi kigger på Maslows behovspyramide, er Roser er ikke et fysisk behov, hvilket betyder at roser ikke er en førsteprioritet for forbrugerne. Derfor med en lav rente får forbrugerne i større grad råd til roser. Roser ligger ikke i en bestemt del af pyramiden, det kommer an på den enkelte forbruger. Dog ligger den i de 3 sidste behov.</p> <p>Mangel på arbejdskraft kan sætte priserne på roser op (udbud efterspørgsel)</p> <p>Den disponible indkomst har stor betydning for os, da forbrugernes rådighedsbeløb har stor indflydelse på om de gider at købe roser, og om de har råd til det.</p> <p>Andre virksomheder der også sælger Roser, gør at de bliver nødt til at konkurrere om de samme kunder.</p> <p>Inflationen kan påvirke folks købekraft</p>	<p>Det anses som en socialnorm at give roser til folk man elsker, specielt når det på en mere romantisk måde eller i forbindelse med begivenheder som bryllupper og Valentins dag.</p> <p>Ændringer i forbrugernes præferencer inden for blomster og havearbejde</p> <p>De demografiske forhold, som aldersfordeling, har stor betydning for salget af roser, da ældre har en større tendens til at købe roser. I Danmark har vi rigtig mange ældre, hvilket er positivt for rosevirksomheder</p>
Technological	Environment	Legal
<p>Rose branchen er ikke plaget af teknologisk konkurrence, virksomhederne skal have en god hjemmeside med en database og god funktionalitet. Her kan Gadstrup Rosenplanteskole forbedre sig.</p> <p>Man kan lave digital markedsføring via en hjemmeside for at nå ud til kunder og forbedre salget</p>	<p>Klimaet har stor indflydelse på dyrkningen af roser. Gadstrup planteskole har både roser udendørs, men også i drivhuse. Derfor påvirker vejret kun dem udenfor. Langt de fleste roser står også udenfor. Det tager over et år for rosen af vokse færdig, derfor kommer rosen igennem vinterperioden med frost. Frost kan dræbe roser, derfor hypper de roserne med en maskine. Maskinen får jord omkring roserne, så de er beskyttet</p>	<p>Fremtidige love for brug af pesticider kan have indflydelse på forretningen.</p> <p>Der må kun anvendes pesticider, der er godkendt af Miljøstyrelsen.</p>

	<p>mod frost. Dog er der stadig roser som dør til frost hvert år. Hvis det regner virkelig meget, kan der same sig vandpytter på marken, som også kan dræbe nogle af roserne.</p> <p>Da Gadstrup Planteskole sælger til virksomheder landet over, skal roserne transporteres langt hvilket udleder CO2.</p>	
--	---	--

Bilag 4

SWOT

Strengths (Intern)	Weaknesses (Intern)
<ol style="list-style-type: none"> 1. Stort udvalg af roser. Ca. 300 arter. 2. Kvalitetsprodukter som er dyrket med omhu. 3. Loyale kunder 4. Erfarenhed/Viden. De ansatte har viden om roser, som giver god kundeservice og rådgivning. 5. GProser.dk er en internethjemmeside. 6. Har salgsaftaler med diverse virksomheder og andre planteskoler. 	<ol style="list-style-type: none"> 1. Vi har ikke ressourcer kapital til at kunne reklamere om os selv. 2. Forældet hjemmeside 3. GProser.dk er lille hjemmeside, som ikke mange kender til. 4. Sælger kun roser
Opportunities (Ekstern)	Threats (Ekstern)
<ol style="list-style-type: none"> 1. Lokalt firma, så lokale der bor i området, kan oftere være interesseret i at købe end fra et stort fjernt firma. 2. Kundernes positive oplevelser af køb på rosenmarken, kan give reklamering igennem mund til mund. 3. Roser er anset som en normal del af visse begivenheder. 4. Danskerne har gennemsnitligt et højt rådighedsbeløb. 	<ol style="list-style-type: none"> 1. Blive udkonkurreret af andre store firmaer som sælger roser. 2. Klimaforandringer/vejret kan påvirke dyrkningen af roserne negativt. 3. Lovgivningen om brugen af pesticider kan være en udfordring 4. Økonomiske udfordringer og ændringer i kundernes præferencer 5. Sæsonbaseret efterspørgsel. Der kan måske være perioder med lav efterspørgsel.

<p>T.O.W.S. Analyse: Gadstrup Rosenplanteskole</p>	<p>Strengths (Intern):</p> <ol style="list-style-type: none"> 1. Stort udvalg af roser 2. Kvalitetsprodukter 3. Loyale kunder 4. Erfarenhed/Viden. 5. De har en hjemmeside 6. Har mange salgsaftaler med virksomheder og planteskoler. 	<p>Weaknesses (Intern):</p> <ol style="list-style-type: none"> 1. Lille kapital til markedsføring 2. Forældet hjemmeside 3. GProser.dk er lille hjemmeside, som ikke mange kender til. 4. Sælger kun roser
<p>Opportunities (Extern):</p> <ol style="list-style-type: none"> 1. Lokalt firma, lokale kunder. 2. Reklamering igennem mund til mund. 3. Begivenheder salg. f.eks. bryllupper, valentinsdag, mm. 4. Danskerne har god økonomi. 	<p>SO Strategier:</p> <ol style="list-style-type: none"> 1. Da danskerne har et højt rådighedsbeløb, kan de godt betale det ekstra for et kvalitetsprodukt (S2, O4) 2. Markedsføring af roser til begivenheder (bryllupper, fødselsdage og jubilæer) (S1, S2, O3) 3. Social medie profiler for reklamering (S3, O2) 	<p>WO Strategier:</p> <ol style="list-style-type: none"> 1. Lave en ny hjemmeside (W2, O1, O2). 2. Sælge haveredskaber og andre relateret ting. (W4, O4). 3. Digital/Lokal markedsføring, til at promovere nye produkter. (W2, O4) 4. Forbedre online tilstedeværelse og markedsføring (W1, O1) 5. Opfordre kunder til at dele deres positive oplevelser på sociale medier for at få mere opmærksomhed og tiltrække nye kunder (W2, O2)
<p>Threats (Extern):</p> <ol style="list-style-type: none"> 1. Blive udkonkurreret af store firmaer som sælger roser. 2. Klimaforandringer/vejret kan påvirke dyrkningen af roserne negativt. 3. Lovgivningen om brugen af pesticider kan være en udfordring 4. Økonomiske udfordringer og ændringer i kundernes præferencer <ol style="list-style-type: none"> 1. Sæsonbaseret efterspørgsel. Der kan måske være perioder med lav efterspørgsel 	<p>ST Strategier:</p> <ol style="list-style-type: none"> 1. Sørge for salgsaftaler, giver butikkerne roser som de normalt ikke vil have (S1, T1). 2. At have roserne gro økologisk/bæredygtigt er en til type kvalitet (S2, T3) 3. Have robuste salgsaftaler kan begrænse konsekvenser, der kan opstå hvis der kommer økonomiske udfordringer i Danmark (b2b) (S6, T4) 	<p>WT Strategier:</p> <ol style="list-style-type: none"> 1. Markedsføring, det er nemmere at skaffe nye kunder når man er en lille virksomhed. Det ville så give et større kapital (W1, T1) 2. Salg af andre produkter ville give mere diversitet i firmaets indkomst (W4, T2, T5). 3. Investerer i alternative dyrkningsmetoder/teknologier for at modvirke klimaforandringer (flere drivhuse) (W2, T2) 4. Lave markedsundersøgelser for at overvåge kundernes præferencer. Så vi kan tilpasse produktet. (W4, T4) 5. Fokuserer på at finde alternative metoder til dyrkning af roser, som overholder alle lovmæssige krav (W1, T3)

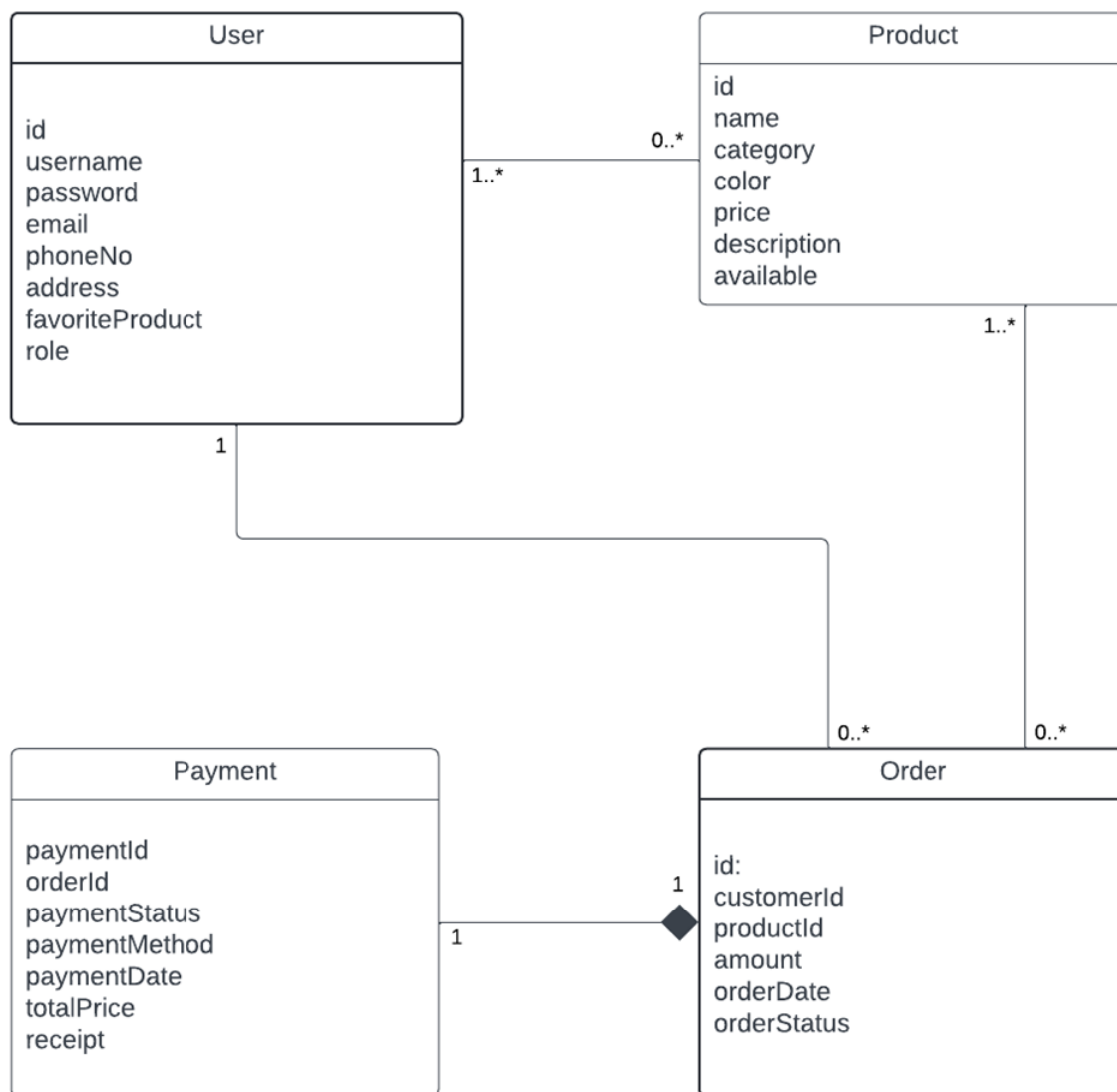
--	--	--

Bilag 6

Product Box

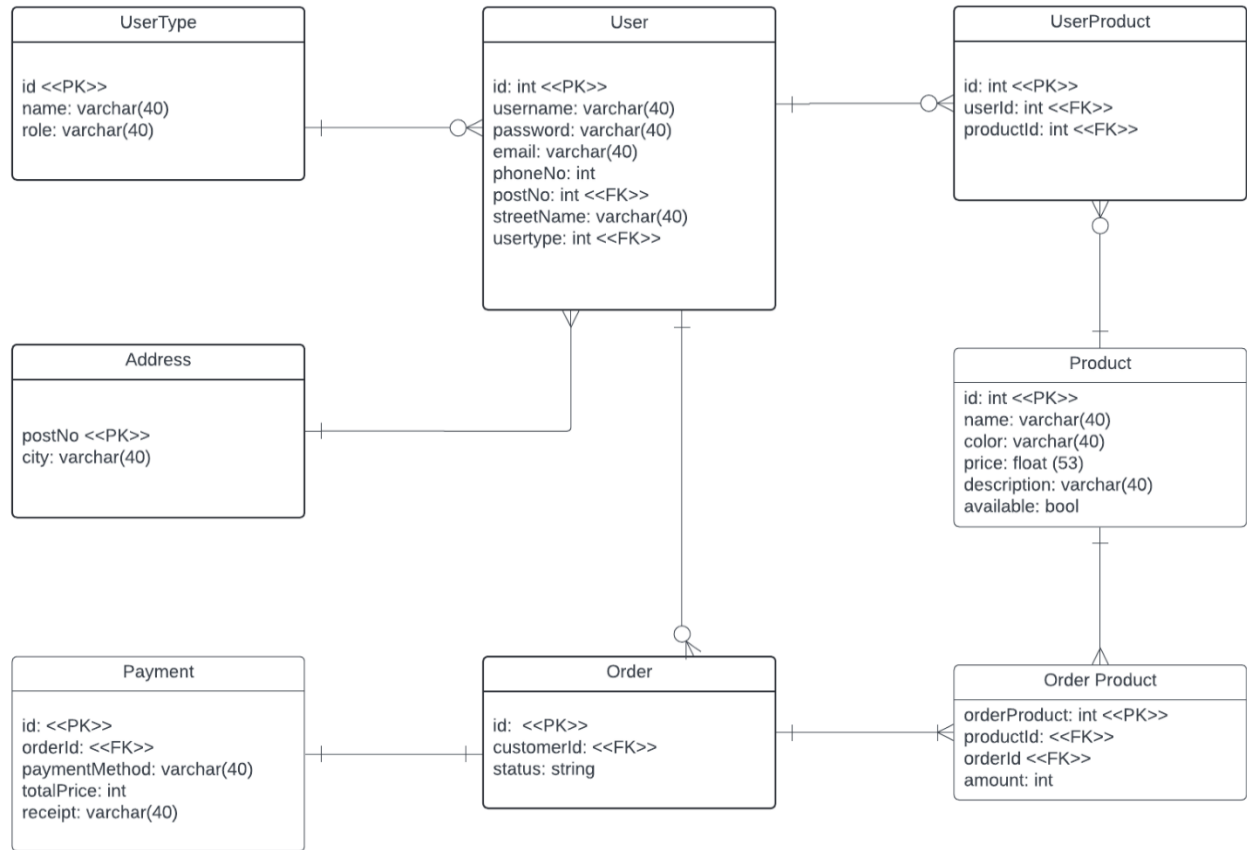


Domænemodel Bilag 7

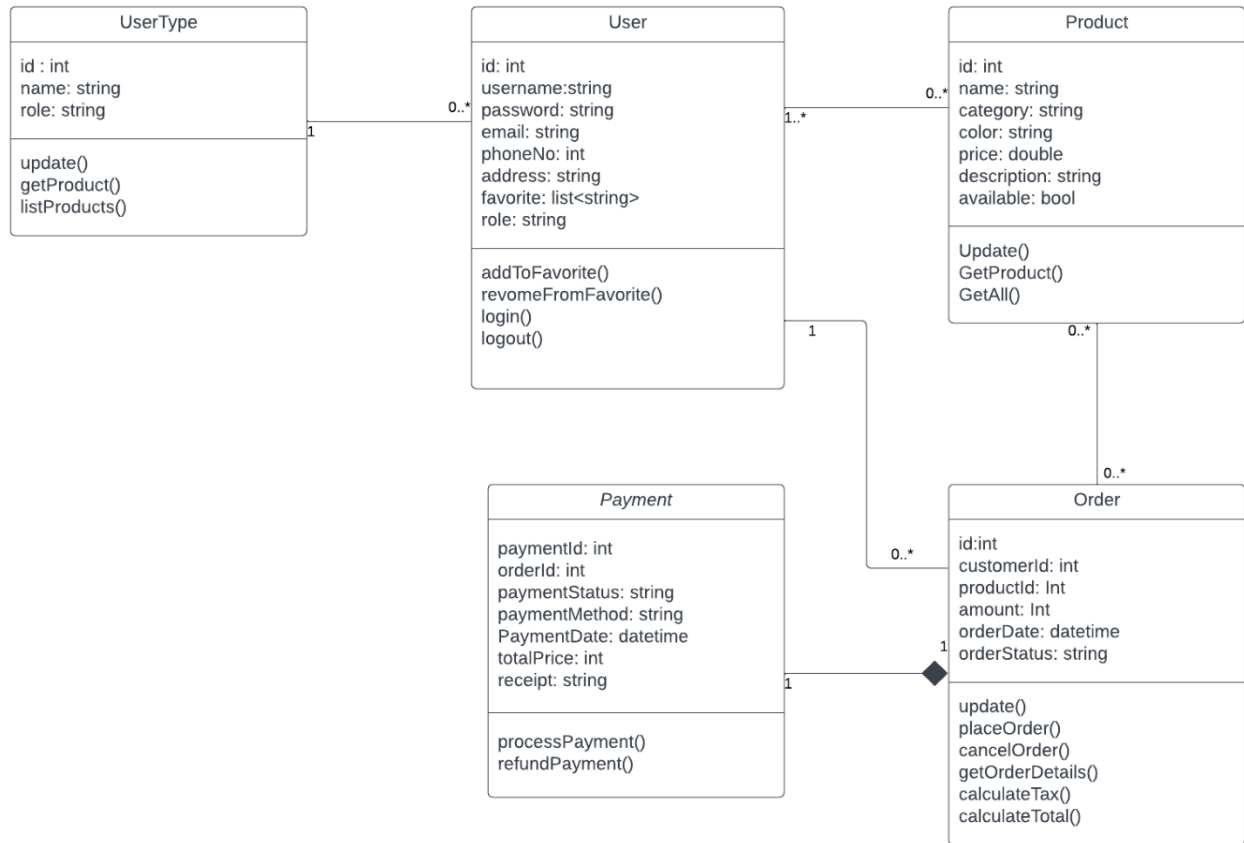


Domænemodel, Klassemodel og ER Diagram: https://lucid.app/lucidchart/3dd059d3-c0b6-499f-a680-1cbb86f5770/edit?invitationId=inv_46957be0-b16c-421e-9ec5-457734af5e75

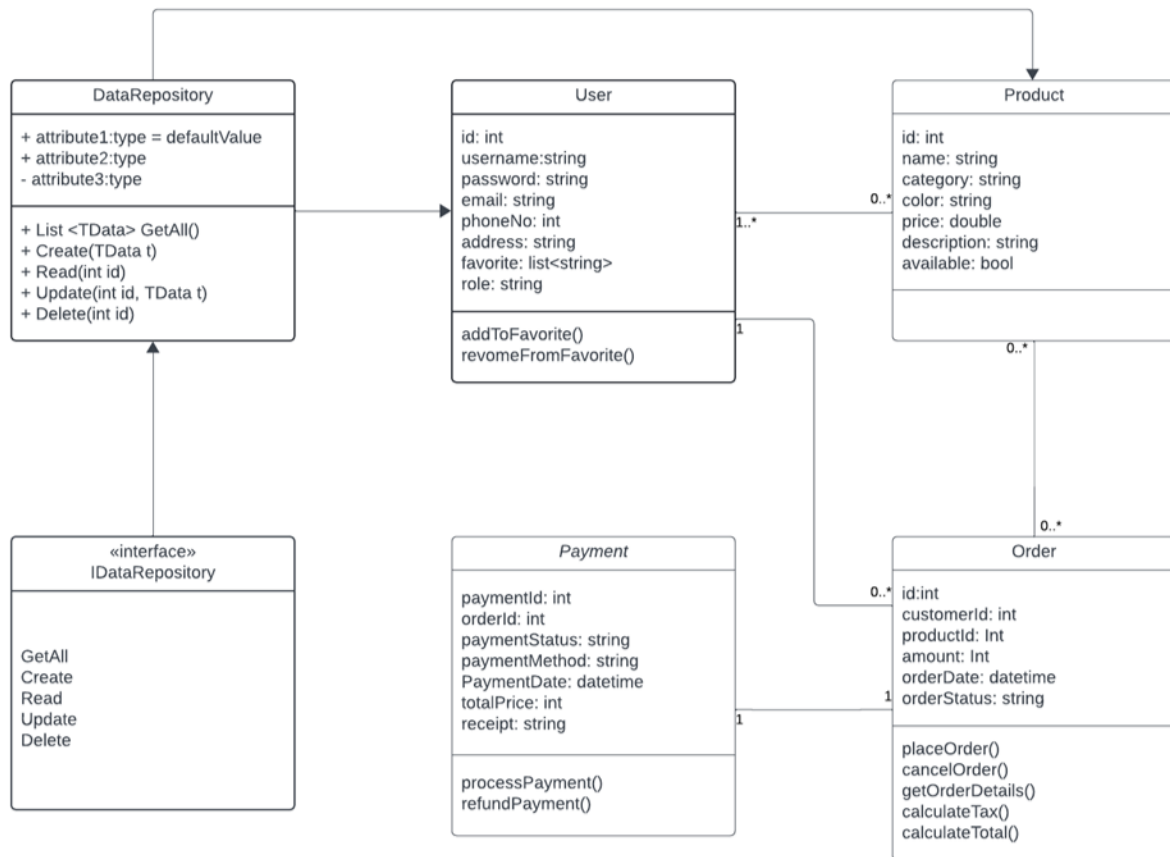
EER Diagram



Klassediagram

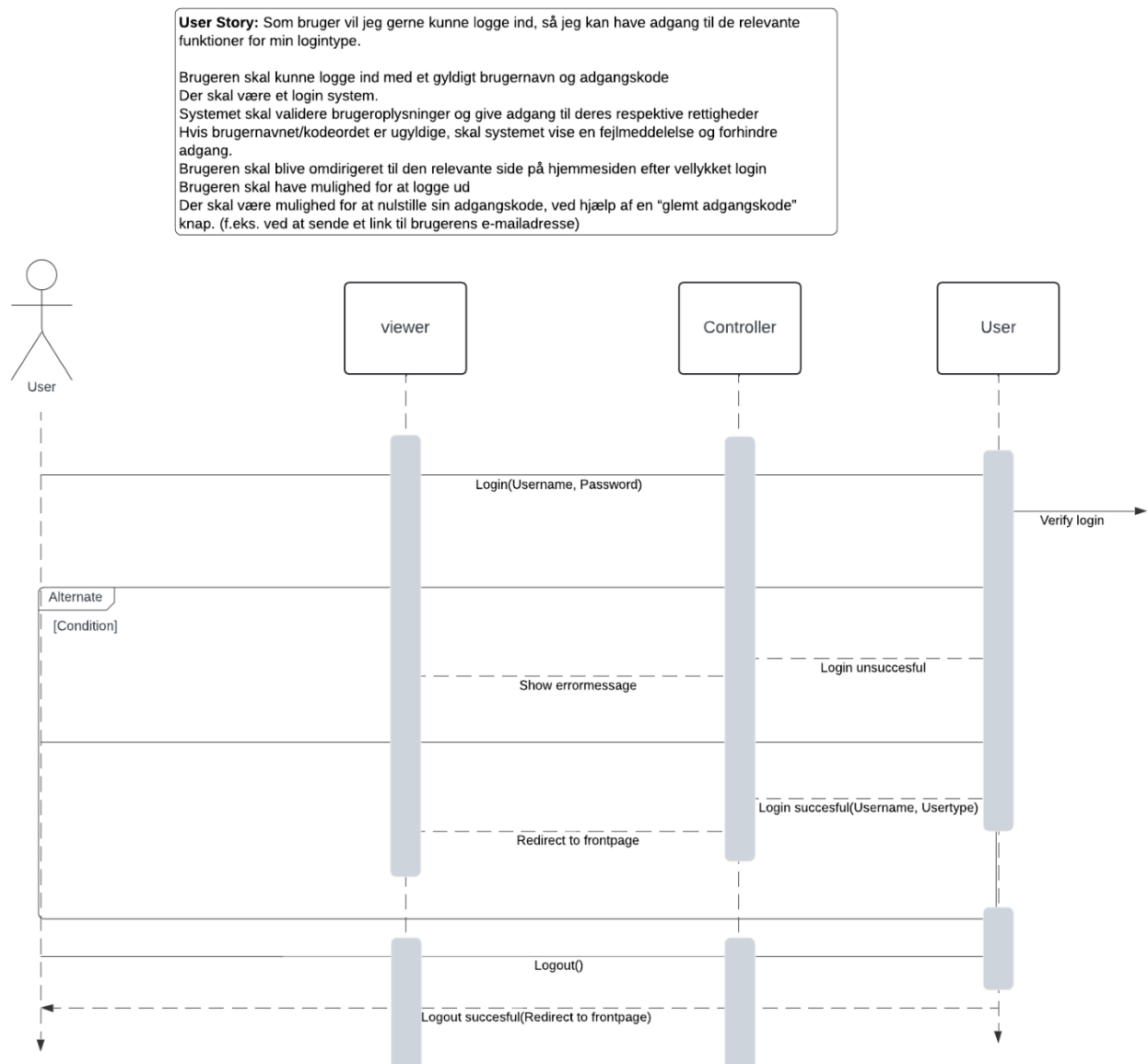


Software Klassediagram



Design Sekvens Diagram

Bilag 12



Bilag 13

User Story: Som kunde vil jeg gerne kunne søge efter specifikke kategorier og filtrere resultaterne: Såsom rosetype, farve og størrelse.

Det skal være muligt som kunde at vælge mellem flere forskellige kategorier.

Det skal kunne være muligt at bruge et filtreringssøgesystem.

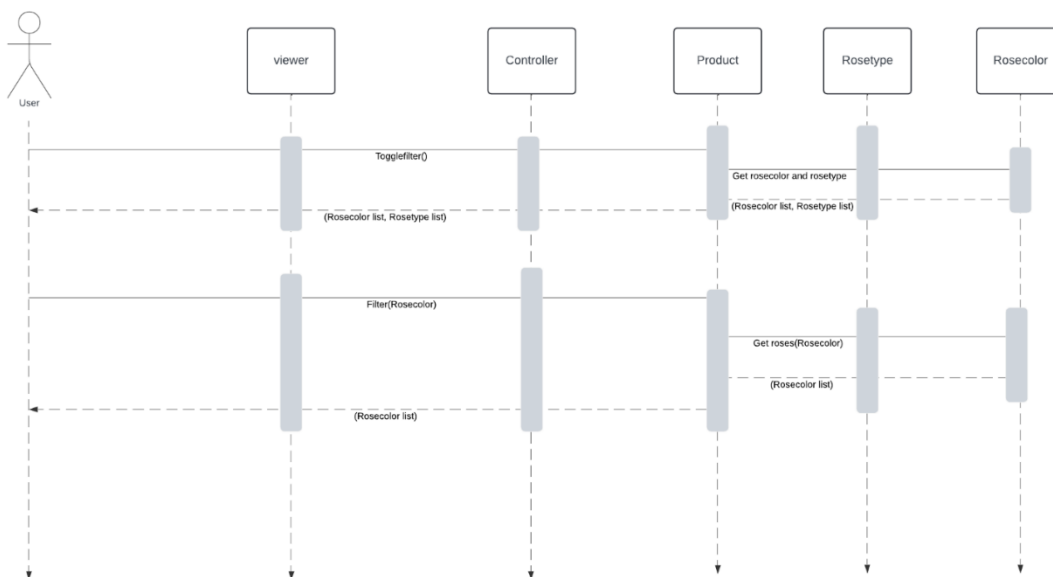
Der skal være en dropdown menu, hvor man skal kunne vælge at sortere efter en eller flere kategorier

Når kategorier er valgt, skal søgeresultaterne opdateres øjeblikkeligt for kun at vise roser, der opfylder de valgte kriterier.

Systemet skal reagere hurtigt på brugerens valg af kategorier uden behov for at genindlæse hele siden.

Brugeren skal have mulighed for at fjerne eller ændre valgte kategorier, så søgeresultaterne justeres efter behov

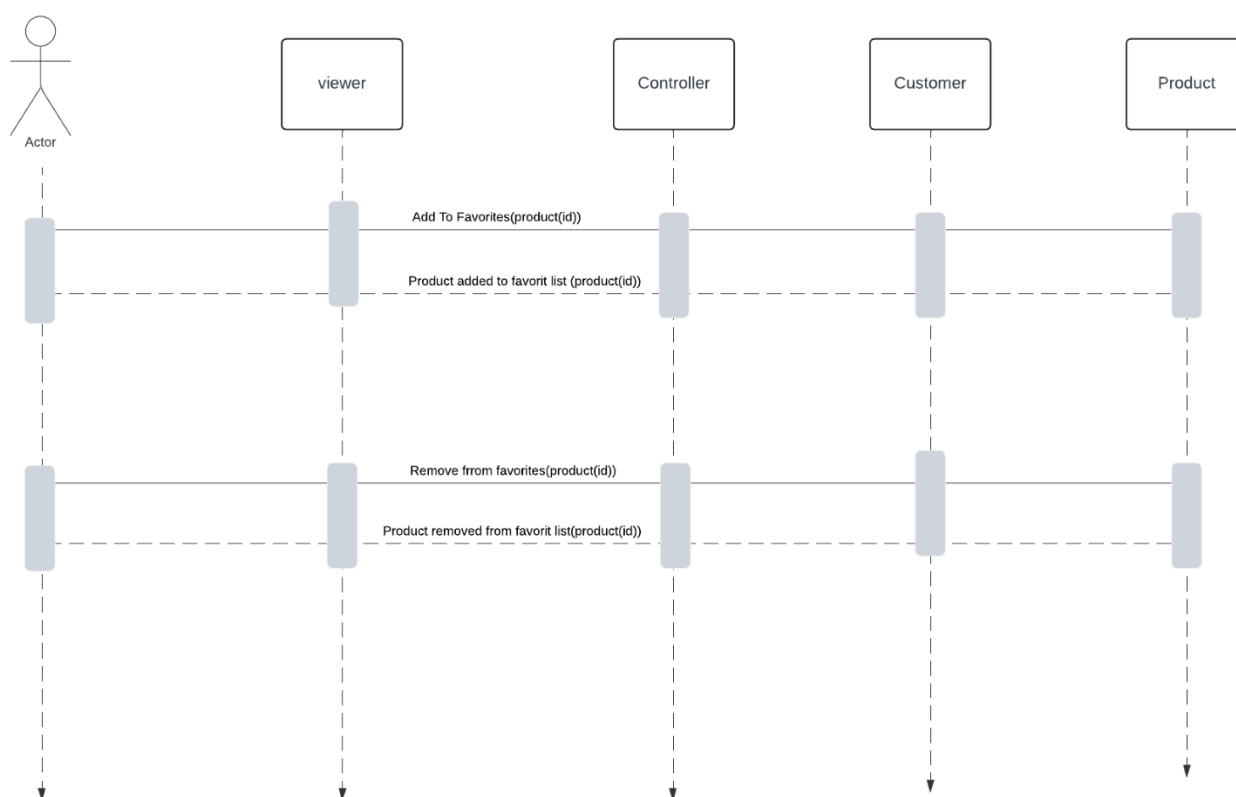
Systemet skal vise antallet af resultater, der matcher de valgte kategorier, for at give brugeren et overblik over deres behov.



Bilag 14

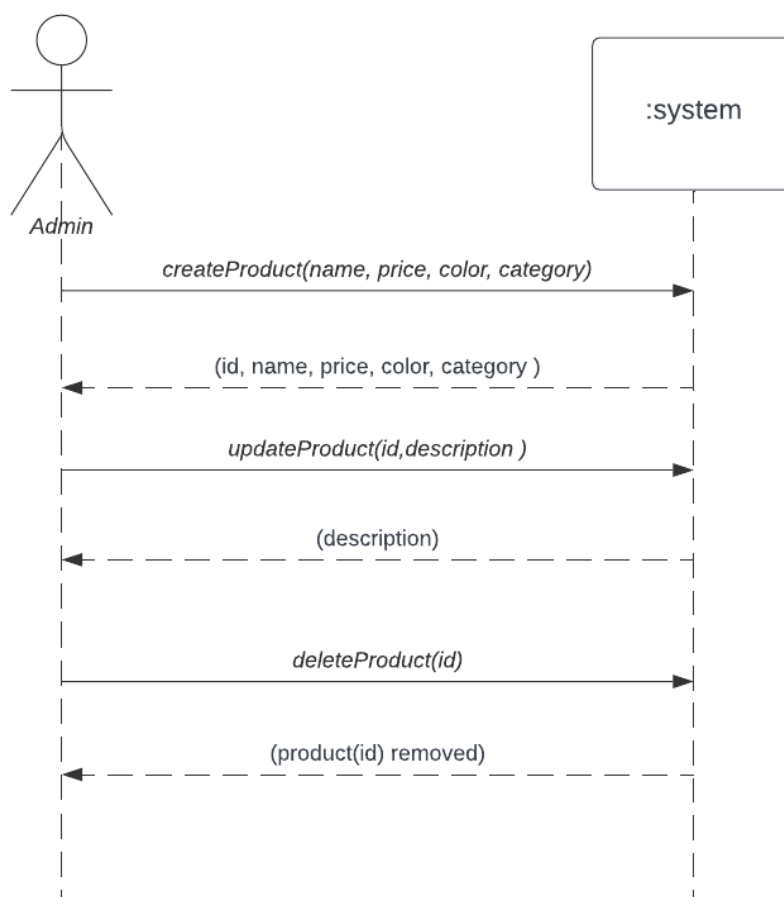
User Story Favoritliste: Som medlem, vil jeg gerne kunne tilføje roser til min favoritliste, så jeg nemt kan finde tilbage til dem på et andet tidspunkt.

- Der skal være en favorit knap/stjerne, som man kan klikke på der gør rosen til favorit
- Der skal være en side hvor man kan se alle roser man har valgt som favorit, og se om rosen er tilgængelig.
- Det skal være muligt at bruge den samme knap man bruger til at tilføje til favoritter, til at fjerne dem igen. Det skal være tydeligt ved hjælp af farve på ikonet om den er i sine favoritter eller ikke.
- Det skal være muligt at sortere sin favoritliste, ved hjælp af filtreringssystemet fra vores user story med titlen "Kategorier/Filtrering"



Bilag 15

User Story Administrator funktioner: Som ejer ønsker vi nogle administratorfunktioner som CRUD, så vi kan håndtere roser, nyheder og andre begivenheder:



Test Cases Bilag 16

Se vedhæftet xlsx fil for Testcases: [Testcase for User Stories af GProser.xlsx](#)

Skitser/Mockups Bilag 17

Se vedhæftet fil Mockups: [2. Semester Mock-Ups](#)

Bilag 18

Review [Feedback review Gproser.xlsx](#)

Bilag 19

Sprint 1: [Gadstup Rosenplanteskole sprint 1.xlsx](#)

Bilag 20

Sprint 2 [Gadstup Rosenplanteskole sprint 2.xlsx](#)

Logbog Bilag 21

Logbog

23/4-2024:

I dag har vi lavet lidt forberedelse og planlægning. Dette indebærer et Gantt kort over hele projektforløbet, som vi har lavet i ProjectLibre. Derudover har vi diskuteret og valgt vores case. Vi kom frem til Gadstrup Rosenplanteskole, som er en lille virksomhed. De har en gammel hjemmeside som trænger til at blive forbedret.

24/4-2024:

I dag færdiggjorde vi vores Gant kort i Projekt Libre. Derudover færdiggjorde vi forberedelsesfasen.

25/4-2024:

I dag har vi arbejdet på Business Model Canvas, Cost Benefit og Risikoanalyse over Roskilde Planteskole, vi blev næsten færdige med alt.

29/4-2024

For at udføre vores Business Process Modeling er vi nødt til at have lavet vores user stories først. Derfor har vi truffet en beslutning om at udskyde denne proces til efter udarbejdelsen af user stories.

Vi har færdgjort afsnittene: Business Model Canvas, PESTEL Analyse

Derudover har vi arbejdet på bilaget til vores SWOT-analyse og TOWS-analyse. Vi mangler dog at skrive om SWOT og TOWS-analysen, samt lave en introduktion til den.

Opdateret gantt kortet i Project libra:

Vi har nu afsluttet

- Om opgaven

- Afgrænsning
- Case
- Metoder/Teknikker/Værktøjer
- Risikoanalyse
- Business Model Canvas
- PESTEL Analyse

Det vil sige at vi er lidt bagud da vi skulle have færdiggjort:

- Cost Benefit
- Business Process Modeling

Opgaver vi er er forud med:

- SWOT – vi er halvvejs færdige med SWOT selvom vi først skulle gå i gang med den i morgen

30/4-2024

I dag har vi færdiggjort vores SWOT og TOWS-analyser og skrevet nogle afsnit om dem. Der mangler dog stadig at blive skrevet lidt mere om disse afsnit.

Vi har sat skabelonen til Inception deck ind og påbegynder denne del i morgen.

1/5-2024

I dag gik vi gang med Inception Deck, vi er næsten færdige med den.

Der er blevet lavet nogle ændringer på SWOT og TOWS i dag.

2/5-2024

I dag har vi lavet en del konklusion samt påbegyndt vores user stories. Desuden har vi lavet et review af opgaven hvor vi har lavet nogle rettelser i virksomhedsanalysedelen. Der opstod uenigheder i gruppen over "Om opgave" delen da flere gruppe-medlemmer er uenige i nogle redegørende tilføjelser der potentielt kan have negativ betydning for resten af opgaven, da det ikke hænger sammen med hvad der er skrevet i vores virksomhedsanalyser. Desuden mangler der at blive rettet grammatik i denne del.

Vi fandt ud af at det ikke var en god ide at lave et review samt have et så stort emne som user stories sat på samme dag. Vi var langt fra færdige med vores user stories, og fortsætter med det i morgen.

3/5-2024

Vi lavede vores review af første del af opgaven, og blev helt færdige med SWOT og TOWS-delen

Vi fortsatte med vores user stories, men er ikke helt færdige endnu

Vi startede også med vores domænemodel men blev uenige i hvad der skulle være i vores model.

6/5-2024

Liste af opgaver vi laver på i dag

Business Process Modeling

- Ikke langt

Userstories

- Vi har lavet alle vores userstories men mangler dog at skrive lidt flere ting om dem.

Product Backlog

- Færdig

Domænemodel

- Vi har lavet en domænemodel og er nu i gang med at skrive et afsnit til den.

ER diagram

- Vi har lavet et ER diagram og mangler at skrive tekst til den.

Klassediagram

Har lavet udkast til klassediagram, mangler feedback.

Project libre

Vi har opdateret planen og er lidt bagud.

7/5-2024

Liste af opgaver vi laver på i dag

User Stories

Domænemodel, Entity Relationship Diagram, Klassediagram

Vi har lavet mange rettelser på vores domænemodel, ER diagram og klassediagram ud fra den feedback vi har fået og efter at have snakket med Victor.

Design Sekvens Diagram

Vi har lavet DSD for vores userstory om Administrator funktioner.

System Sekvens Diagram

8/5-2024

Liste af opgaver vi laver på i dag

Vi tog hen til skolen og fik vejledning af Ivan

Startede med at lave Business Process Modeling (mangler at skrive om det)

Redigerde vores Domænemodel så vores model at korrekt nu (mangler at skrive om den)

Tilpasse vores EER Model (Enhanced Entity Relationship Model) ud fra vores Domænemodel, så de kan implementeres som en database. Samt skrevet om de 3 normalformer

Vi fik lavet klassediagram, men skal have vejledning om vi skal lave en repository klasse med alle CRUD-funktionerne
Vi gik i gang med testcases for 3 user stories: Login, Administrator, Filtreringssystem, mangler at skrive om det.
Vores DSD med Administrator CRUD-funktion skal måske laves til en DSD i stedet for 4, (mangler at skrive om det)
Vi skal lige kigge på SSD igen (mangler at skrive om det)

9/5-2024

I dag har vi fået færdiggjort vores User Stories.

Derudover har vi også færdiggjort Domæne modellen og skrevet et afsnit om den.
Der er også blevet arbejdet videre på Design Sekvens Diagrammet.

Vi lagde mærke til at der var fejl i vores Gantkort i Project libra. Vi havde intastet forkerte datoer på vores sprint planning samt sprint 1. Det vil sige at vi nu er bagud med ca. 2 dage.

Vi har aftalt at lave et review af en anden gruppes projekt, samt få vores egen projekt set til. Som vi så vil mødes op på fredag den 10/5 for at give og modtage feedback

10/5-2024

I dag havde vi aftalt at holde et møde med en anden gruppe for at gennemgå vores Review af hinandens arbejde. Ud fra den feedback vi fik kunne vi så lave rettelser de steder hvor der var fejl og mangler.

Mangler:

- Lave flere DSD og SSD-modeller og skrive om det.
- Skrive om klassediagrammet (mangler at få feedback på om der mangler nogle klasser)
- Læse den sidste del af virksomhedsanalysen (review).
- Tilføjet en user story 'online handel' til vores Testcases. Der blev også tilføjet lidt ekstra til de tidligere.
- Vi rykkede ting som skulle være i vores hjemmeside ned i delkonklusionen

13/5-2024

Tog hen til skolen og lavede vores sprint planning for sprint 1

Fik svar på vores klassediagram af Per

Mangler:

- Færdiggøre: DSD
- Færdiggøre SSD,
- Skrive om Klassediagram
- Lave et Review af sidste del af virksomhedsanalysen
- Skrive om sprint planning

14/5-2024

Vi gik i gang med sprint 1, hvor vi lavede følgende:

- Vi oprettede de basale razorpages, men mangler stadig lidt.
- Oprettede grundlæggende tabeller via SQL Query's fra razorpage 2 semester øvelser (vi tilføjer flere ting senere)
- Brugte EFCore Powertool til at oprette autogenereret klasser.
- Oprettede Pages (Mapper i visual studio)
- Vi færdiggjorde create og delete fra CRUD
- Vi har gemt vores første version af GPRoser razorpages på GitHub.

15/5-2024

- Vi har arbejdet videre på sprint1
- Vi har begyndt Mock-upen til at designe hvordan hjemmesiden skal se ud.
- Lavet design til forsiden (index + wwwroot), hvilket bruger Mock-upen.
- Tilføjet nogle flere klasser som manglede.

16/5-2024

- Silas lavet et logo til hjemmesiden.
- Der blev lavet et Mock-up af hvordan siden hvor roserne (produktet) bliver vist og det tilhørende filtreringsystem, kan se ud i dens layout.
- Der blev arbejdet mere på klasserne og razor pages til hjemmesiden.
 - o Login blev lavet.
 - o Gjorde at kun en Admin kan se 'create'-siden til at oprette nye produkter.
- Vi havde problemer med ikonerne i CRUD, vi fik løst det ved at linke vores html med vores css fil (stylesheet)
- Desuden havde vi nogle problemer med at liveshare i dag, hvor de som det blev delt til ikke kunne redigere i koden.

17/5-2024

- Vi fik lavet at kun Admins kan se og bruge ikonerne for at ændre, slette og oprette roser/produkter.

19/5-2024

Vi besluttede os for at tage en kort dag om søndagen og kører indtil vi ikke lige gad mere. Vi fik også reflekteret lidt over hvordan det første sprint er gået indtil videre og indså at vi nok burde begynde og være mere præcise med vores scrum teknikker.

- Vi arbejdede i at tilføje farver og art til roserne.
- Vi havde ikke taget højde for at sprinterne vi lavet havde kommet til at tælle weekenderne med i gantt kortet, så vi har mindre tid end planlagt.

20/5-2024

- Tilføjet 2 nye tabeller i databasen "RoseType" og "RoseColor"

- Vi tilføjede dem som FK i Product Tabellen så de havde en reference id til RoseType og RoseColor
- Vi fik gjort, så man kan tilføje et billede til roserne når de bliver dannet eller ændret.
- Der blev tilføjet en [Column] til colorId og typeld, så der kommer en dropdown menu, når man vil tilføje/redigere roser
- Vi havde problemer først med at lave colorId og typeld, da vi ville lave dem som enum men vi valgte i stedet at lave to nye tabeller.

21/5-2024

- Vi fik færdiggjort vores Sprint 1 og er begyndt med Sprint 2.
- Gjorte at man kan se hvilken farve og type roserne er.
- Fik lavet og opdateret vores All page, så den viser alt data om roserne korrekt. Dette gælder også for vores view page
- Rettelser i design på alle sider
- Website er blevet published for første gang
- Alle pages gjort mere responsive.

22/5-2024

- Der blev lavet en del på designet af hjemmesiden.
- Tilføjet About page
- Tilføjet FAQ page
- Rettet footer så indholdet står i rigtig rækkefølge
- Tilføjet links i footer, desuden er padding: 40px 0; tilføjet der gør den lidt tykkere
- Fjernet gammel kode der ikke bliver brugt mere.
- Fjernet Forside link da det var der 2 gange i header.
- Ændret alle AddSingleton til AddScoped koden.
- Lavet det meste af filtersystemet, men mangler at ændre systemet hvor man vælger farve/type til en måde der er nem/åbenlyst at vælge flere farver/typer på en gang. (lige nu skal man holde 'Ctrl'-knappen nede)
- Skrev nogen kommentarer ind i koden.
- Vi blev færdig med Kost-Benefit, Business Process Modeling, og punkt 9 i Inception decked.
-

23/5-2024

- Vi rettet vores Gantt kort
- Vi har lavet en Admin side, hvor en Admin kan se alle deres admin funktioner.
- Opdateret css og design på website
- Tilføjet en JavaScript funktion der skjuler toggler synligheden af filtreringssystemet

24/5-2024

- Startede med vores daily sprint meeting
- Vi blev færdig med filtreringssystemet.
- Opdateret css på filtreringssystemet
- Der blev tilføjet en masse kommentere til koden.

- Opdateret css og design på website
- Redigeret tekst og rettet fejl i vores afsnit om Be Clear on What's Going to Give.
- Rettet produkt Box, indeholder nu alle de oplysninger der skal være.
- Startede med at skrive i emnet "Koden" for afslutningsrapporten
- Startede med at skrive i emnet "Database" for afslutningsrapporten
- Lavede lidt mere på designsekves diagrammer og begyndte at skrive noget tekst

25/5-2024

- Lucas arbejdet lidt vide i emnet "Kode" med "Filtrerings systemet".

26/5-2024

- Vi arbejdet videre i emnet "Kode".
- Skrev om sprint 1.

27/5-2024

I dag har vi lavet følgende:

- Rettet fejl i vores sprint modeller.
- Lavet 1 DSD-diagram
- Skrevet om sprint 1 (næsten færdig)
- Skrevet om sprint 2 (ikke færdig)
- Skrevet om Burn down Chart
- Færdiggjort emnet 'Database'

28/5-2024

I dag har vi lavet følgende:

- Løst bug der gjorde det muligt for alle at logge ind selvom de skrev forkert password.
- Lavet DSD færdigt
- Lavet SSD færdigt
- Skrevet tekst til sprint 2
- Skrevet om Burn Down Chart
- Skrevet om kode
- Rykket bilag ned til bilag siden
- Skrevet reflektion
- Skrevet konklusion

29/5-202

Dagen inden aflevering har vi lavet følgende:

- Skrevet hvem der har hovedansvar for hvilken del i Eksamensdokumentet
- Skrevet hvem der har hovedansvar for hvilken del i Razorpages

- Rettet hele teksten igennem flere gange og gennemgået den inden aflevering
- Rettet vores kildeliste og Bilag