

Årsprojekt

Gruppe 4

2A

Adam Shumiye Ziebe

Dina Johanne Sode Watson

Oliver Boots

Signe Sjørslev Hansen

**Organisation:
Kvintus Kollegiet**

4. Juni 2023

Københavns ErhvervsAkademi



Vejledere:
Bo Hansen
Dan Madsen
Malene Hasse
Kevin Lindemark Holm
Chalie Demasi
Tahseen Uddin
Mike Kandi
Michael Brandt

kea
COPENHAGEN SCHOOL OF DESIGN
AND TECHNOLOGY

Anslag
Med mellemrum:

60.262

1. Resumé

Udgangspunktet for udarbejdelsen af projektet er ønsket om at bringe Kvintus Kollegiets nuværende bookingsystem til vaskekælderen fra en fysisk løsning til en webbaseret.

Projektgruppen ønskede at undersøge hvilke udfordringer, der er ved det nuværende system, samt hvordan dette kan afhjælpes med en webbaseret løsning. Desuden ønskede gruppen at undersøge, hvordan man kan nedbringe mængden af vasketider, der går til spilde grundet det nuværende system.

Projektgruppen foretog kvalitative og kvantitative undersøgelser på kollegiets beboeres behov og ønsker for en sådan løsning. Herefter udarbejdede gruppen den løsning, der beskrives i rapporten. Funktionaliteten af websitet er baseret på ønsker fra beboerne, samt de features der tidsmæssigt for projektet har været ladesiggørligt.

Konklusionen på projektet er, at projektet på mange punkter kunne imødekomme ambitionen om at løse de givne problemstillinger.

2. Indholdsfortegnelse

1. Resumé	3
2. Indholdsfortegnelse	4
3. Indledning	6
4. Problemformulering	7
5. Indledende undersøgelse	8
5.1 Ideudvikling	8
5.2 Research	9
5.3 Brugerundersøgelse / Design Thinking	10
5.3.1 Brugerundersøgelse.....	10
5.3.2 - Design Thinking:	14
5.3.3 - Design:	15
5.4 Persona og User stories	16
Persona 1	17
Persona 2	19
Persona 3	21
Persona 4	23
Persona 5	25
5.5 User stories	27
6. Kravspecifikation og accepttest metode	32
6.1 Rationale for prioritering af krav	35
7. Analyse	36
7.1 Arkitektur af pathfinding	36
7.2 Den endelige arkitektur	38
7.3 Teori af relevante elektroniske/programmering/netværk/server arkitekturblokke	40
8. Løsningsdesign	44
8.1 Hjemmeside Struktur	44
8.2 Kode-beskrivelse.....	47
9. Test af løsning	59
9.1 Udførelsel af brugertest.....	59
9.2 Udførelsel af Accepttest.....	59
10. Praktisk projektplanlægning og ledelse.....	62
10.1 WBS	62
10.2 Gantt.....	63
10.3 Kanban / Scrum.....	64
10.4 Projektanalyse	65
10.5 Virksomhedsanalyse	67

10.5.1 - Laloux' farver.....	67
10.5.2 - Interessentanalyse	69
11. Konklusion	70
12. Projektforløbet.....	71
13. Perspektivering.....	72
14. Litteraturliste	73
15. Bilag	75

3. Indledning

Flere gange årligt flytter nyoptagede studerende til større danske byer for at forfølge drømmen om uddannelse. (Engmann T.S., 2020) For mange af disse tilflytttere er livet på kollegie en realitet. En hverdag som studerende byder på logistiske og tidsmæssige udfordringer, når man ønsker at opretholde studie, socialt liv og - for nogle - et studiejob. Huslige pligter som indkøb, tøjvask og rengøring skal derfor optimalt set forløbe ukompliceret.

Et gruppemedlem fra projektgruppen oplevede en logistisk problematik ved booking af vasketider på sit kollegie. Når kollegiets beboere booker vasketider, skal de fysisk stå foran en stor tavle for at få overblik, se ledige tider og booke en tid. Ligeledes skal beboerne stå foran tavlen for at ændre eller aflyse en tid. Dette gav anledning til et irritationsmoment for gruppemedlemmet, da vasketider krævede meget energi og tid at planlægge.

Her sås en mulighed for at udvikle et system, der ville kunne lette de upraktiske aspekter ved dette. Ved at finde frem til beboernes ønsker og behov ved et webbaseret system, fandt gruppen frem til en vision for en løsning, der kan lette processen med tøjvask for kollegiet beboere.

4. Problemformulering

Som student på kollegie, kan det være en irriterende proces at få vasket tøj. Uden en måde at se status på vaskemaskinerne, skal der bruges ekstra tid og viljekraft på at gå ned og tjekke dem manuelt. Begge af disse ressourcer er ikke noget studenter altid har nok af.

Vi ville forsøge at kigge i dybden på problemerne, der kan opstå i et kompetitivt tøjvask-miljø og lave en IT-løsning, der gavner både administrationen og beboerne.

Ud fra det, fokuserer vi på følgende spørgsmål:

1. *Hvilke udfordringer er der med det nuværende system, og hvordan påvirker det beboernes dagligdag?*
2. *Hvordan kan man skabe en digital løsning der afhjælper udfordringerne med det nuværende system?*
3. *Under hvilke rammer kan man sørge for at formindske vasketider?*

5. Indledende undersøgelse

Projektets businesscase ses i **bilag 1**.

5.1 Ideudvikling

I projektets idéudviklingsfase gjorde vi os tanker om hvilke features, vi kunne forestille os, at en slutbruger kunne ønske af løsningen. Idéen til optimering af vaskekælderens booking-system kom fra et gruppemedlem, så på den baggrund er idéudviklingen farvet af de udsagn, vedkommende har givet om problematikken. Projektet er på den måde atypisk, da et gruppemedlem har direkte indvirkning på projektet, men samtidig er påvirket af slutproduktet, hvis løsningen implementeres.

Med den viden der blev givet om det nuværende system, gav det anledning til brainstorming omkring hvilke problematikker, en slutbruger kunne have, samt hvordan disse problematikker kan løses. Det tog afsæt i personlig erfaring med andre vaskekældre, samt i den vaskekælder projektet drejer sig om. I processen sammenlignede projektgruppen med andre online booking-systemer, vi har kendskab til. Dette indebar løsninger fra andre vaskekældre, men også booking-systemer fra andre erhverv, såsom frisørsaloner.

I det nuværende system er der både problematiske og funktionelle elementer, og projektgruppen har forsøgt at medbringe de funktionelle i den nye løsning. Den mest åbenlyse problematik ved det nuværende system er begrænsningen af kun at kunne tilgå det fysisk. Et funktionelt element er, at det er muligt hurtigt at ændre på sin tid, hvis man står foran tavlen. Derudover kan vedkommende, som booker en tid, identificeres ved hjælp af værelsesnummer, hvilket også er praktisk. Efter en brainstorming omkring hvilke elementer fra den gamle løsning, der skulle forkastes eller medbringes, udviklede projektgruppen idéen til den løsning, der til slut blev udarbejdet.

I overvejelserne tog projektgruppen også højde for, at projektet har en begrænset tidshorisont, og der derfor kun er kapacitet til at løse en begrænset mængde problematikker. Omfanget af løsningen skal passe til projektets scope, og den afsatte tidsperiode er derfor essentielt at tage med i sin begrænsning.

5.2 Research

I research-fasen gentog projektgruppen processen med at søge inspiration i lignende løsninger. Der var mulighed for at trække på individuelle erfaringer med diverse booking-systemer. Som nævnt ovenfor inkluderede disse booking-systemer til andre vaskekældre og frisørsaloner. Eksemplerne havde hver især elementer, som kunne være relevant for den løsning, projektgruppen havde ambitioner om at udvikle. Dette gav i sammenfatning inspiration til hvilke elementer, der kunne være attraktive for denne løsning.

Fordi Kvintus Kollegiet er et lille kollegie, som ikke producerer data om sine faciliteter til omverdenen, var det kun muligt at foretage research i den forstand, at vi foretog spørgeskemaundersøgelser og interviews med beboerne. Projektet er derfor atypisk, da der ikke er mulighed for at foretage anden direkte research på målgruppen end den, man som projektgruppe selv opsøger. Ved at foretage denne type undersøgelser kan man danne sig et overblik over, hvad den specifikke målgruppe kunne ønske af den løsning, man gerne vil udvikle.

Spørgeskemaundersøgelsen og interviewet med beboerne vil blive gennemgået og fremlagt yderligere i senere afsnit, hvor resultaterne og resultaterne også vil blive præsenteret. De resultater der blev fundet frem til, understøttede i høj grad den vision som projektgruppen, som for det meste, allerede havde fundet frem til gennem vores egen research og idéudvikling.

5.3 Brugerundersøgelse / Design Thinking

5.3.1 Brugerundersøgelse

Den komplette transskribering af interviewet ses i **bilag 2**. Den samlede oversigt over diagrammer udarbejdet i forbindelse med spørgeskemaundersøgelsen ses i **bilag 3**.

Der blev i forbindelse med brugerundersøgelsen til projektet foretaget en kvantitativ analyse, som bestod af en spørgeskemaundersøgelse, samt en kvalitativ brugerundersøgelse i form af et interview med en beboer. I forbindelse med dette interview indgik også spørgsmål, som omhandler kollegiets kultur og struktur. Dette interview kan også findes i transskriberingen, men uddybes ikke i forbindelse med brugerundersøgelsen for vaskekælderen.

Kvantitativ analyse - spørgeskemaundersøgelse

Beskrivelse

Spørgeskemaundersøgelsen gav besvarelser fra en lille stikprøve af kollegiets beboere. Før undersøgelsen blev beboerne stillet spørgsmål vedrørende alder, køn og hvor længe vedkommende har boet på kollegiet. Herefter blev der spurgt ind til graden af tilfredshed med det nuværende system, samt hvordan vedkommende planlægger sine vasketider. Der blev også spurgt ind til i hvor høj grad beboerne har tillid til, at de andre beboere på kollegiet overholder deres vasketider. Til slut blev et mock-up af et website vist, hvorefter beboeren blev spurgt ind til graden af lyst til at benytte et lignende system, samt hvilke forbedringer websitet eventuelt kunne få.

Resultat og findings

I de modtagne besvarelser var 40% kvinder, 10% non-binære og 50% mænd. I aldersgruppen var størstedelen mellem 25-30 år, mens mindretallet var mellem 18-24 år. Alle beboerne har boet på kollegiet over 1 år, og af det kan man udlede, at beboerne formentlig har erfaring med det nuværende system for vaskekælderen.

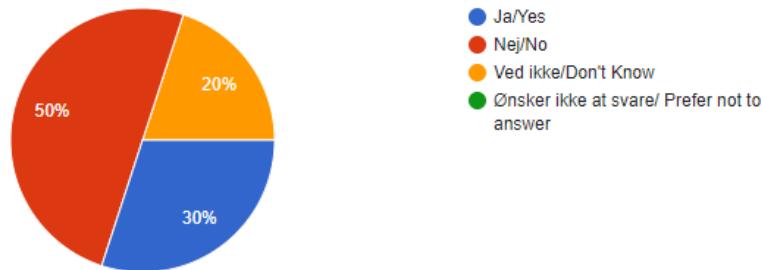
I en sektion af undersøgelsen forsøgte vi også at klarlægge i hvor høj grad, beboerne har tillid til hinanden i forhold til, hvorvidt de andre beboere overholder sine bookinger. Der var også spørgsmål angående vaskevaner.

Halvdelen har ikke tillid til, at de andre beboere overholder deres vasketider. Flertallet benytter desuden ofte vasketider, de ikke selv har booket. Den generelle tendens er, at beboerne planlægger deres vasketider et par dage forinden.

Jeg stoler på, at de andre beboere overholder deres vasketider, så de ikke går til spilde. / I trust the other residents use their time slots so they don't go to waste.

 Copy

10 responses



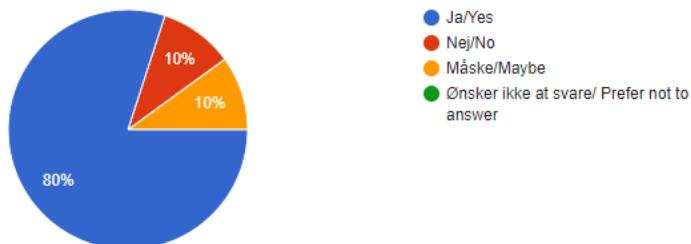
[Figur 1 - Spørgsmål 1]

I den del af spørgeskemaundersøgelsen der skulle undersøge tilfredsheden med det gamle system, samt interessen for det nye, var resultatet overvejende, at tilfredsheden med det nuværende system var middel-godt, mens interessen for at kunne tilgå systemet online og benytte et lignende system med mock-uppen var høj. 80% ønskede at benytte det nye system.

Ville du være interesseret i at benytte et lignende website til booking?/Would you be interested in using a similar website for booking?

 Copy

10 responses



[Figur 2 - Spørgsmål 2]

Resultatet af undersøgelsen gav os en dybere vished om, at interessen for en løsning som den, der har været i udvikling under projektperioden, er interessant for beboerne på kollegiet. På trods af en middel tilfredshed med det nuværende system, gav flertallet udtryk for, at de ønsker en alternativ måde at booke vasketider på.

Det gav desuden anledning til at skabe klarhed omkring hvilke funktionaliteter, beboerne ønsker systemet skal have. Dette omhandlede blandt andet layout og muligheden for at kunne aflyse sine bookede tider. Det var muligt at efterleve en høj andel af disse ønsker.

Kvalitativ analyse - Interview

Beskrivelse

For at supplere spørgeskemaundersøgelsen og samtidig få en mere dybdegående forståelse af situationen med det nuværende system, blev et interview med en beboer afholdt. Beboeren er en mand på 25 år, som har boet på kollegiet i snart 2,5 år. Han blev stillet spørgsmål, som er lignende med dem fra spørgeskemaet.

Resultat og findings

Beboeren gav indblik i sin egen oplevelse med systemet, samt den opfattelse af systemet, de andre beboere udtrykker. Han fortalte, at systemet på flere parametre er upraktisk. Det skyldes, at det er en besværlig proces at booke, fordi man fysisk skal flytte sig ned foran booking-tavlen i kælderen. Desuden kan andre beboere fejlagtigt komme til at efterlade brikker i booking-felter. Det resulterer i fejlvisning af reelle bookinger, der benyttes.

“...én ting er, at der ofte er gamle vaskebrikker, der sidder dernede, da det sidder fysisk på væggen. Så, der kan sagtens være nogen, der har bestilt en vasketid sidste måned, som så stadig blokerer den samme vasketid hver eneste måned..”

- Beboer, Kvintus Kollegiet

Han fortæller, at han kender beboere på 4. sal, der på baggrund af den lange tur ned i kælderen finder det uoverskueligt, og derfor ikke booker tider.

“..En anden ting er, at for dem der bor på for eksempel 4. Sal, der hører jeg, at de snakker om, at de gider heller ikke booke vasketider fordi, at det er en lang tur at gå ned i kælderen for at vurdere, om man kan få lov at vaske eller ej så gå op og hente sine ting igen, så det bliver en hel process i sig selv..”

- Beboer, Kvintus Kollegiet

Det er også en oplevelse, han selv har. Beboeren beskrev, at han ofte selv helt undgår at booke vasketider, fordi systemet er for besværligt, som det er på nuværende tidspunkt.

“..Jeg synes faktisk, at det er så besværligt, at jeg ofte undgår at booke vasketider, og i stedet bare går ned og for at tjekke, hvorvidt der er nogle ledige i virkeligheden.. Og fordi der er det system, der er lige nu, så er jeg nødt til at gå derved 10 minutter efter en vasketid er startet for at bekraefte, at der ikke er dukket nogen op til vasketiden, og så vaske der. Jeg er nødt til at time, hvornår jeg dukker op til vasketider på trods af, at jeg ikke booker en tid..”

- Beboer, Kvintus Kollegiet

Han beskrev også, at det er systemet, der giver anledning til den adfærd. Han følte ikke, at det er besværet værd at booke tider, og der er derfor ingen struktur i hans vasketider. Han fortalte, at et online bookingsystem ville gøre det mere overskueligt.

“.. Det ville bare gøre det mere overskueligt. Det ville gøre det mindre til noget, man skal planlægge sin dag omkring måske, og så være noget, der kan blive planlagt omkring ens dag..”

- Beboer, Kvintus Kollegiet

Resultat og findings - sammenlagt

Med den viden understøtter det de besvarelser, vi modtog i forbindelse med spørgeskemaet. Resultaterne stemmer overens i den forstand, at et online bookingsystem er noget, beboerne ville se som en forbedring i forhold til det nuværende system. På den baggrund vurderes det, at projektet formentlig ville blive modtaget med positivitet fra beboerne.

5.3.2 - Design Thinking:

Vi kan kun kalde vores projekt en succes, hvis det aktivt bliver brugt, samt at det opfylder brugerens behov og forventninger. Såfremt der er elementer i løsningen der ikke fungerer optimalt, skal vi gøre alt hvad vi kan for at sikre at produktet bevæger sig omkring disse problematikker.

Design thinking er grundlagt at se tingene fra brugerens synspunkt. Ved at analysere og forstå brugerens behov og adfærd, kan man udrette et bedre konkret kort over, hvad der er nødvendigt og hvad brugeren har af forventninger til produktet.

Det er klart, at produktet skal være forståeligt for brugeren, hvor elementerne nавигerer brugeren fra den ene interaktion til den anden, uden at brugeren stopper op og tænker “hvorfor?”. (Johnson A., 2023), (Interaction Design Foundation, 2022)

Med vores brugerundersøgelse har vi forsøgt at indsamle konkrete data for, hvad en bruger af det nuværende booking system ønsker, at det skal være.

Til brugerundersøgelse lavede vi mock-ups, og spurgte ind til designets “appeal”. Vi bad dem også uddybe deres følelser og tanker.

Design Thinking er en konstant og iterativ metode, der påvirker alle dele af designet, men ikke overtager fuld kontrol over visionen.

Ud fra det data vi havde indsamlet, lavede vi personaer der skal forestille sig så autentisk som muligt, de personligheder der oftest ses på kollegiet.

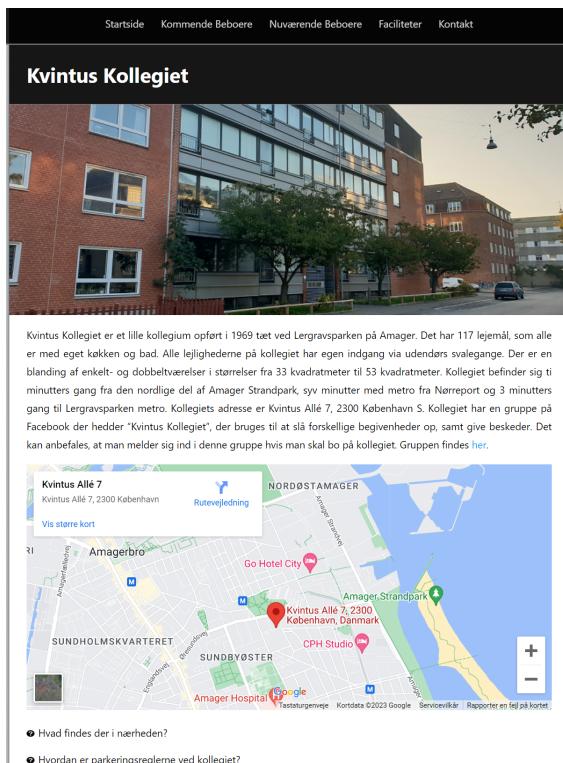
5.3.3 - Design:

Vi anbefaler at se figur 3 nedenunder, eller tilgå Kvintus Kollegiet hjemmeside:

<https://www.kvintus.dk/>

For ikke at forvirre brugeren af kollegiet, har vi lagt vægt på at fastholde de essentielle design-elementer af kollegiet hjemmeside til booking systemet. Kvintus Kollegiet's hjemmeside er en simpel side med få animationer og distraheringer. De vigtigste informationer kan tilgås hurtigt fra toppen af siden, og at scrollle igennem siden får ingen kritik. Dette gør det nemmere for os at adapterer til vores side, men der er få kerne-elementer vi kan lægge anker til og farverne leder os igennem alle "shades of grey". Hvor vi ønsker at forbedre produktet i forhold til hjemmesiden, er med afstanden mellem elementerne på siden.

Uanset layout eller design, kan vi med sikkerhed sige at vi vil bibeholde sidens gode og stabile brugeroplevelse. For et billede af Kvintuss hjemmeside, venligst kig på figur 3.



[Figur 3 – Kvintus' Hjemmeside]

5.4 Persona og User stories

Personærer er livlige og detaljeret beskrevet fiktive repræsentanter af den ideelle kunde eller bruger af produktet. Personærer laves for at bedre forstå og imødekomme kundernes behov, og for at sikre at alle har samme ide og vision om hvordan brugerne faktisk ser ud, så der arbejdes imod samme mål. Personerne bruges også til user stories. (Relevans, 2023)

Persona 1

Navn:

Marius Møller

Alder:

21 år

Job:

Studerende



(iStock, 2023, [Link](#))

Påklædning:

Marius går oftest afslappet klædt. Han foretrækker jeans og t-shirts eller skjorter. Han har altid en praktisk rygsæk med sig til at bære sine studiebøger og bærbare computer.

Personlige og professionelle ambitioner:

Marius ønsker at færdiggøre sin bachelorgrad i datalogi med gode karakterer og derefter fortsætte med en kandidatgrad. Han drømmer om at arbejde som softwareudvikler.

Problemer og udfordringer:

Marius har sommetider svært ved at finde en balance mellem studier og fritid. Han kan blive overvældet af deadlines og kravene fra sit studie.

Marius dører også med at få ordentlige vaskerutiner, da det som regel først er noget han tænker på, når alt hans tøj er brugt.

Interesser/Hobby:

Marius er passioneret omkring computere og teknologi. Han nyder at arbejde med kodning og eksperimentere med forskellige programmeringssprog. Han er også en ivrig gamer og nyder at tilbringe tid med sine venner i kollegiets fællesområde, hvor de hænger ud og gamer sammen.

Adfærd på sociale medier:

Marius er aktiv på sociale medier og bruger primært platforme som Twitter og Reddit til at følge med i de nyeste trends inden for teknologi og softwareudvikling. Han kan lide at deltage i faglige diskussioner og dele relevante artikler og nyheder inden for sit felt.

IT-kompetencer:

Marius har gode IT-kompetencer grundet sin uddannelse.

Persona 2

Navn:

Anja Person,

Alder:

24 år, født og opvokset i Sverige, men rejst til København for at bo og studere.

Job:

Uddanner sig til fysioterapeut og underviser i et yogacenter på Islands Brygge.

Påklædning:

Går ofte i træningstøj, som fx joggingbukser eller tights, samt løse crewneck eller hættetrøjer. Hun er glad for sneakers, og hendes yndlingsmærke er Nike.

Personlige og professionelle ambitioner:

På det personlige plan vil Anja gerne have en sund krop og sjæl så hun så længe som muligt har det godt. Ud over det vil hun rigtigt gerne hjælpe andre til at komme i balance med deres krop og have en sund tilværelse.

Interesser/Hobby:

Anja går meget op i yoga som hun underviser i, samt at træne i fitness, eller at løbe ture. Hun bruger meget tid på Instagram og følger diverse trænings-influenceres, holistiske profiler og mærker som Planet Nusa.



(iStock, 2023, [link](#))

Problemer og udfordringer:

Grundet Anjas meget aktive livsstil har hun brug for ofte at vaske sit tøj, og da hun også går op i sit udseende og hvordan andre ser hende, foretrækker hun rent tøj. Samtidig har hun en travl kalender med job og uddannelse, og har derfor svært ved at finde tid til at ordne sit vasketøj.

Adfærd på sociale medier:

Hun deler sin dagligdag, spisevaner og træningsrutiner flittigt med hendes instagram-følgere og på Tik Tok. Hun havde engang en Facebook, men den er slettet nu.

IT-kompetencer:

Anja bruger sin computer til noter på studiet, samt streaming når hun en sjælden gang imellem sidder stille.

Persona 3

Navn:

Sandra Søgaard

Alder:

20

Job:

Studerende, datamatiker

**Påklædning:**

Går i afslappet tøj, som hun har til vane at bruge og hendes typiske outfit inkluderer jeans, t-shirts og hættetrøje. Hun ser ikke nødvendigheden i at udskifte sin garderobe ofte, men når det bliver nødvendigt, prøver hun at finde tøj i samme stil.

Foto genereret af

AI:www.aiimagegenerator.org

Personlige og professionelle ambitioner:

Sandra ønsker at færdiggøre studiet, så hun kan få et job, hvor hun kan sidde ved en computer et sted, hvor hun ikke skal forholde sig til særlig mange mennesker.

Problemer og udfordringer:

Det er ofte svært at overskue alt det man skal, så det ender ofte med at Sandra ser film eller serier i stedet for at ordne de lektier, opvask eller vasketøj, hun burde. Det meste bliver ordnet i sidste øjeblik, når der ikke er nogen anden mulighed.

Interesser/Hobby:

Sandra elsker at slappe af til serier og film. Derudover er hun glad for at spille og designe små spil.

Adfærd på sociale medier:

Sandra har konto på flere forskellige sociale medier, blandt andet på Twitter og Facebook.

Hun poster ikke selv særlig meget, men er glad for at kunne følge med i hvad andre laver.

IT-kompetencer:

Sandras it-kompetencer er middel til gode, grundet hendes interesse, og hun selv har prøvet at få forskellige opsætninger til at virke til hendes forskellige spil.

Persona 4

Navn:

Marlene Lassen

Alder:

23 år

Job:

Studerer fysik på Københavns Universitet. Har et studiejob på uddannelsesinstitutionens tilhørende biblioteket.

**Påklædning:**

Om vinteren foretrækker hun cardigans og jeans. Strikker selv sine uldsweatre og cardigans, så går derfor meget op i, at de plejes ordentligt. Om sommeren ses hun i toppe og shorts i lyse farver. Er bevidst om de klimaafttryk, som tøjindustrien sætter, og går derfor op i at passe på sine ting, så hun økonomisk og samvittighedsmæssigt er tilfreds.

Foto genereret af AI: <https://openai.com/dall-e-2>

Personlige og professionelle ambitioner:

Har ambitioner om at blive forsker efter endt uddannelse. Hun har på nuværende tidspunkt ikke en fast partner, og med hendes høje ambitioner inden for faglig udvikling og videnstilegning er der ikke udsigter til det.

Problemer og udfordringer:

Stort set al tid i hverdagen bliver brugt på arbejde og studie, og den resterende tid ønsker hun at bruge på familie og venner. Det kan være svært at få overskud og tid til huslige pligter i en travl hverdag. Hun bor på Kvintus Kollegiet, som er et gammelt kollegie med forældede metoder til booking af faciliteter, og det giver irritationsmomenter, når det er svært at danne sig overblik og planlægge ud i fremtiden.

Interesser og hobbyer:

Når hun ikke opsøger ny viden gennem læsning og informative videoer på internettet, nyder hun at tilbringe tid sammen med venner og familie.

Adfærd på sociale medier:

Bruger sociale medier på alment niveau, som indebærer at poste opslag ved særlige lejligheder, men ikke så ofte som personer, der opretholder et jævnt flow af opslag. Benytter dog sociale medier til at holde sig opdateret på andre menneskers opslag. Er derfor ikke en aktiver ‘poster’, men en aktiv tilskuer.

IT-kompetencer:

Har basale IT-kompetencer på niveau med andre fra hendes generation. Herunder hører brug af smart-devices i alle formater. Derudover besidder hun IT-kompetencer inden for de programmer, der bliver brugt på uddannelsen.

Persona 5

Navn:

Mikkel Jacobsen

Alder:

26

Job:

Studerer til ingeniør på Københavns Universitet.

Arbejder i SATS som personlig træner ved siden af sit studie for at tjene ekstra penge ved siden af sin SU.



Foto genereret af AI: <https://openai.com/dall-e-2>

Påklædning:

Til hverdag er han klædt casual-pænt, så han opretholder et stilrigt image. Det kunne blandt andet være habit bukser eller jeans sammensat med turtlenecks eller t-shirts. I sin fritid går han meget i idræstøj på baggrund af sit arbejde og sin personlige træning.

Personlige og professionelle ambitioner:

Stræber efter at blive en dygtig ingeniør, da han både brænder for sit fag og for at tjene penge. Han har en personlig ambition om at være i fysisk god form, samt forbedre sine egne resultater i fitnesscenteret. Han drømmer om en dag at stifte familie og starte sin egen virksomhed.

Problemer og udfordringer:

I en travl hverdag med studie og træning er det svært at finde tid til at udføre huslige pligter, og derfor halter lejligheden ofte, når det kommer til rengøring og tøjvask. Opvasken og vasketøjet når ofte at ligge for længe, og han føler sig derfor bagud. Det er en hæmsko, for han har ikke lyst til spontant at invitere venner og familie over, da han ikke har lyst til at byde folk ind i sit rod. Han ønsker sig en hverdag med bedre overblik og struktur, hvor det er

lettere at udføre huslige pligter samtidig med, at hans hverdag kan køre i samme tempo som nu.

Interesser og hobbyer:

Han er meget passioneret indenfor personlig træning, både for sig selv og for sine klienter.

Derudover nyder han at bruge tid sammen med sine venner. De tager ofte til sociale arrangementer, både på og uden for studiet. Da han ikke har en fast partner, nyder han at tage i byen med sine venner, hvor han møder nye mennesker og socialiserer.

Adfærd på sociale medier:

Han poster ofte billeder og videoer af sig selv og fra sin træning. Han håber at kunne motiverer andre til at få en sundere livsstil og en bedre træningsrutine. Han følger ligeledes andre med samme interesser som han selv, og gennem dette opnår han også en større motivation for at træne. Ud over træningsrelateret indhold følger han også inspirerende mennesker, der er nytænkende og fremme inden for ingeniørfaget.

IT-kompetencer:

Han besidder IT-kompetencer på lige fod med andre fra hans generation. Dette indebærer smart-devices og anden elektronik. Der er i den almene brug af teknologi ingen direkte begrænsninger for hans viden, og han besidder også rimelige tekniske færdigheder grundet sit studie.

5.5 User stories

Persona	Story nr:
Marius Møller	1
Behov	Prioritet:
Jeg kommer først i tanke om at skulle vaske, når jeg ikke har mere tøj tilbage. Derfor ønsker jeg at jeg i løsningen kan reservere fast vasketid, med en påmindelse der bliver sendt, så jeg ikke glemmer tiden.	2
Resultat	Størrelse/Effort:
Da der skal bruges sms til reset af password og id, kan jeg bruge nummeret til at få påmindelser om min vasketid, evt et selvvalgt tidsinterval før vaskens start. Når jeg reservere dagen, kan jeg få en knap hvor jeg kan få den til automatisk at reservere hver, eller hver anden uge.	3

Persona	Story nr:
Anja Person	2
Behov	Prioritet:
<p>Jeg ønsker en nem måde hvor jeg kan planlægge og være sikker på at jeg kan vaske tøj på når jeg har brug for det på mit kollegie, så jeg ikke går forgæves og løber tør for rent tøj.</p> <p>Helst en løsning hvor jeg bruger minimalt tid på at få overblik og reservere maskinerne.</p>	1
Resultat	Størrelse/Effort:
Ved at lave en online løsning kan jeg få en løsning, hvor jeg hurtigt og nemt kan få overblik over maskiner, hvornår der er ledige vasketider og reservere en vasketid.	4

Persona	Story nr:
Sandra Søgaard	3
Behov	Prioritet:
Jeg ønsker en funktion hvor man kan reservere en tid med fast interval, og man hurtigt kunne få overblik over hvilke perioder hvor maskinerne var mindst brugte, så man kunne undgå for mange mennesker. Der skal være en påmindelse så man ikke glemmer hvornår man har en vasketid.	2
Resultat	Størrelse/Effort:
Da sms funktion alligevel skal bruges til en anden funktion, kan man også bruge sms til påmindelsen, så man kun skal lave en knap, så man reservere en tid hver anden uge eller en gang om måneden.	3

Persona	Story nr: 4
<p>Som studerende har jeg en travl hverdag. Jeg skal få mit program til at gå op både med studie, fritidsjob og socialt samvær med venner og familie. Det kan være svært at finde tid til at opfylde alle huslige pligter, når der foregår så mange andre ting i mit liv.</p> <p>Faciliteterne på mit kollegie er gammeldags, og man skal fysisk ned til tingene for at kunne booke dem. Dette gælder for vaskerum såvel som fitness-lokalet. Det er upraktisk og svært at danne sig overblik over, og så kræver det i øvrigt, at man kan planlægge længere ud i fremtiden.</p>	
Behov	Prioritet: 2
<p>Som bruger ønsker jeg hurtigt og let at kunne danne mig et overblik over ledige vasketiden, uden at skulle flytte mig fysisk ned til faciliteten på mit kollegie. Det skal være let tilgængeligt at kunne booke en tid, så jeg ikke behøver at planlægge mine tøjvaske længe i forvejen.</p>	
Resultat	Størrelse/Effort: 3
<p>Med et lettere tilgængeligt booking-system kan jeg booke mine vasketider uden at skulle bruge tid på at gå ned i kælderen til den nuværende booking-tavle, som kun kan tilgås fysisk. Jeg kan derfor lettere planlægge en huslig pligt i min travle hverdag, og det fjerner irritationen ved ikke at have overblik over tiderne på farten.</p>	

Persona	Story nr: 5
<p>Som studerende er min dagligdag tætpakket med aktiviteter. Jeg har meget at se til, når jeg både skal følge med på mit studie, passe mit arbejde og deltage i sociale aktiviteter med familie og venner. Mine weekender er ofte tætpakkede med aktiviteter, og jeg er sjældent hjemme før sent i hverdagen. Det er svært at få tid til opvask og vasketøj. Jeg har ofte ikke overskud til at gå ned i kælderen for at booke en vasketid, når jeg kommer sent hjem.</p>	
Behov	<p>Prioritet:</p> <p>Som bruger ønsker jeg at kunne booke en vasketid let og hurtigt, uden at skulle tænke på at gå ned i kælderen og stå foran booking-tavlen. Ønsket til et booking-system er et velfungerende system med godt overblik i simpelt design.</p>
Resultat	<p>Størrelse/Effort:</p> <p>3</p> <p>Således at jeg kan fjerne det irritationsmoment, hvor jeg glemmer at booke en tid, og derfor ikke har mere rent tøj, når jeg skal bruge det. Det ville fjerne byrden af at skulle vaske på skæve tidspunkter, hvor det passer mig dårligt. Med et elektronisk booking-system kan jeg bedre planlægge mine vasketider, og vaske mit tøj når det passer mig.</p>

6. Kravspecifikation og accepttest metode

ID: 1	Krav: Websitet skal give mulighed for at foretage en booking af 2 vaskemaskiner	Prioritet: 1
Kategori:	Accepttest: Såfremt websitet visuelt kan vise booking af time-slot og databasen har gemt bookingen, accepteres kravet	Passed/failed/tested

ID: 2	Krav: Websitet skal give mulighed for at logge ind med en brugerprofil, samt nulstille password	Prioritet: 1
Kategori:	Accepttest: Såfremt websitet visuelt og korrekt i overensstemmelse med databasen kan foretage et valideret login og nulstilling af password, accepteres kravet.	Passed/failed/tested

ID: 3	Krav: Websitet skal kunne fremvise en administrator-side, der giver mulighed for redigering af oplysninger	Prioritet: 1
Kategori:	Accepttest: Såfremt websitet kan fremvise en dedikeret administrator-side, der giver mulighed for modificering af databasens oplysninger, er kravet godkendt.	Passed/failed/tested

ID: 4	Krav: Løsningen skal kunne give en reminder om en booket tid.	Prioritet: 2
Kategori:	Accepttest: Såfremt løsningen kan give en form for feedback/påmindelse til brugeren om, at vedkommende har en tid x dag på x tidspunkt, er kravet godkendt.	Passed/failed/tested

ID: 5	Krav: Websitet skal kunne vise en bekræftelses-side, når en booking skal foretages og er foretaget.	Prioritet: 1
Kategori:	Accepttest: Såfremt websitet kan fremvise en før-bekræftelse og en efter-bekræftelse, som er i overensstemmelse med de korrekte data fra databasen, vurderes kravet som godkendt.	Passed/failed/tested

ID: 6	Krav: Websitet skal være simpelt og let-navigerbart.	Prioritet: 1
Kategori:	Accepttest: Såfremt en bruger let og uden forvirring kan benytte websitet, anses kravet for godkendt.	Passed/failed/tested

ID: 7	Krav: Websitet skal være mobile-ready, og kunne tilgås i flere formater.	Prioritet: 2
Kategori:	Accepttest: Såfremt websitet kan tilgås i flere formater, herunder smartphones og pc'er, anses kravet som værende godkendt.	Passed/failed/tested

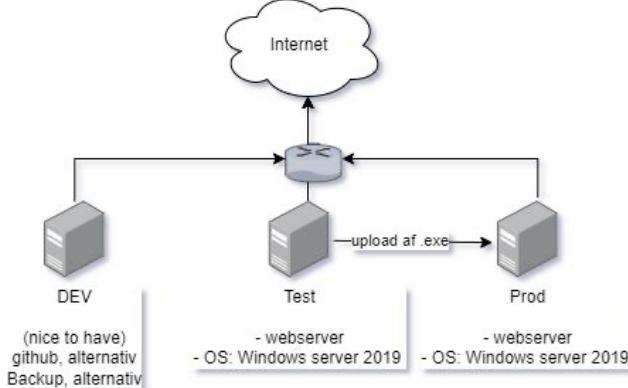
6.1 Rationale for prioritering af krav

De ovenstående krav er udarbejdet på baggrund af, at de er essentielle for funktionaliteten af systemet, og for løsningen af de problematikker, der er observeret. Kravene med førsteprioritet er søjlerne i systemets evne til med fordel at kunne afløse det gamle system. De kan derfor ikke undværes for funktionaliteten. Projektgruppen vurderede, at funktionalitet for systemet vægtes højere end design. Det skyldes også den begrænsede tidshorisont for projektet. I et projektforløb, der løber over en længere tidsperiode, er der i højere grad mulighed for at fokusere på sekundære krav. Dette har ikke været tilfældet for vores projekt. På den baggrund er de ovenstående krav prioriteret i den rækkefølge, de er.

7. Analyse

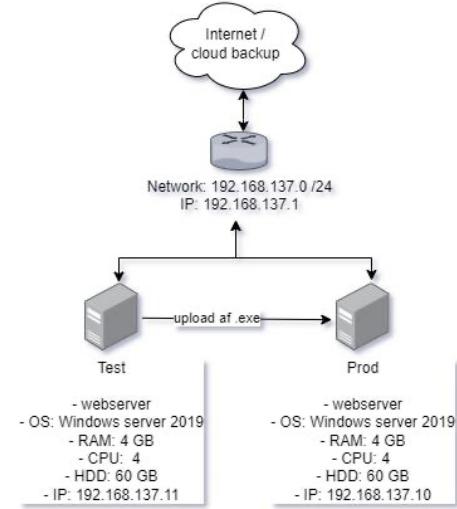
7.1 Arkitektur af pathfinding

Server



[Figur 4]

Diagram rev. 1



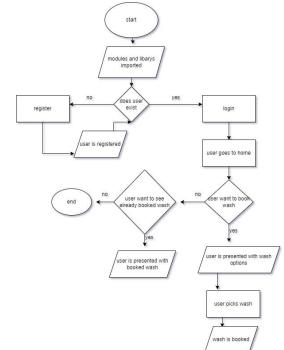
[Figur 5]

diagram rev. 2

Vores infrastruktur har ikke ændret sig meget. Da det hele tiden har været vores tanke at hoste siden på en webserver. Denne server hoster vi via VMWare da det er en nem og billig måde, mens man er i udviklingsfasen, at teste ens løsning. Skal den på et tidspunkt implementeres, er det forholdsvis nemt at flytte til cloud, eller over på en server, alt efter kundens ønske og økonomi. Når alt er klar til at implementere løsningen, har man brug for mere end bare 1 virtuel maskine. På grund af tidspress har vi ikke et Development environment, som man normalt ville have når man skal lave patches og nye funktioner. Har man udviklet noget, skal det testes i et miljø der minder mest muligt om produktionsmiljøet, hvor det så bliver implementeret, når det er færdigt og testet. Derfor har vi en test og en produktions server med.

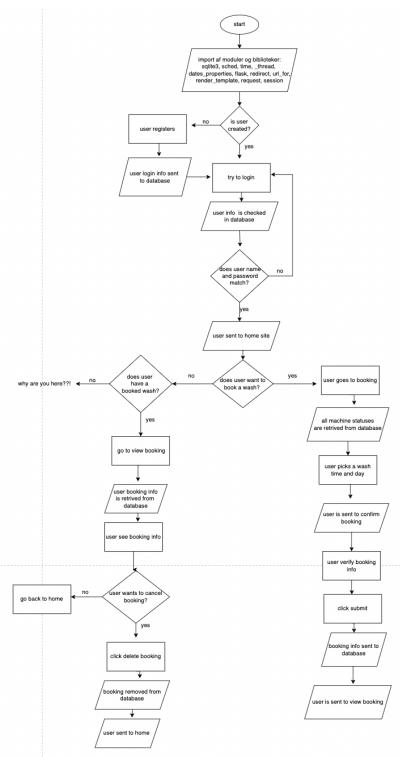
Programmering:

Dette er vores første udkast til programmerings flowchart. Vi var fra starten ret sikre på, hvilke funktioner siden skulle have. Den viser grundideen, hvad vi gerne ville have siden skulle kunne. Man skulle kunne registrere sig som bruger. Logge ind og booke en vasketid. Efterhånden som arbejdet skred frem, fik vi udbygget denne flowchart, så vi fik alle funktioner med.



[Figur 5]

Dette er næste revision af flowchartet, som det endte med at være efter vi havde arbejdet med koden. Den har grundlæggende de samme funktioner, der er dog tilføjet hvornår den gemmer og henter noget fra databasen.



[Figur 6]

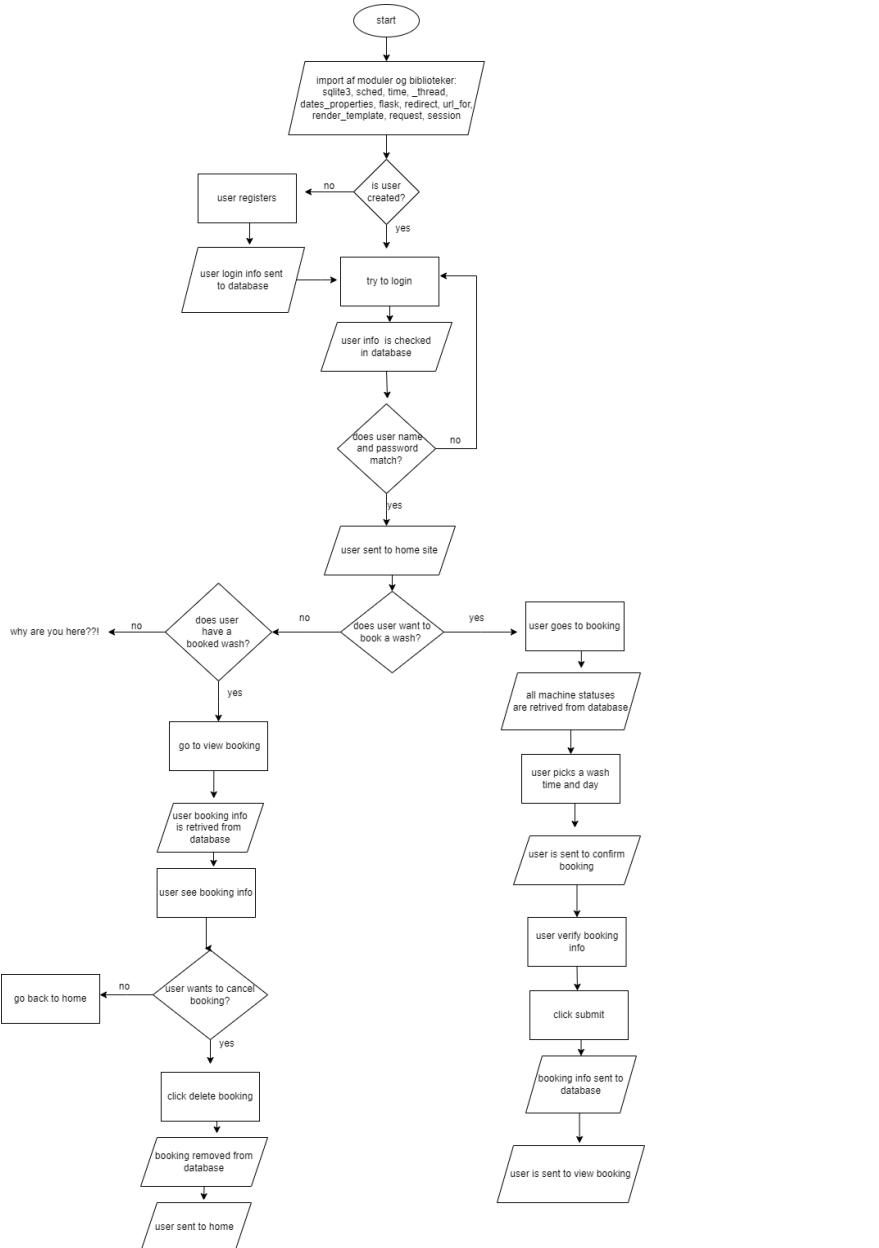
7.2 Den endelige arkitektur

Server

Den endelige arkitektur adskiller sig ikke væsentligt fra sidste revision af diagrammet. Dette skyldes, at vi havde problemer med at få scriptet til konfiguration af serveren, til at virke. Vi kunne godt få en VM startet og installere det vi skulle, men vi kunne ikke få scriptet til at sætte en statisk IP og hente vores filer til API'en inden afleveringsfristen for rapporten.

Programmering

Herunder ses det endelige flowchart. Den endelige arkitektur er ikke markant anderledes fra den først planlagte. Ændringerne ses i højre søjle af flowchartet. For at illustrere ændringerne, er disse fremhævet.



[Figur 7]

7.3 Teori af relevante elektroniske/programmering/netværk/server arkitekturblokke

Server

Som arkitektoniske byggeblokke har vi valgt at bruge en virtuel maskine i VMware. Som router bruger vi den, computeren har adgang til. Vi nåede ikke helt at blive færdige med begge scripts, som var planen vi skulle bruge.

Blok navn: Virtuel maskine

Vi har valgt at bruge VMWare. En virtuel maskine er noget software, som gør det muligt at låne hardware ressourcer, så det virker som om man har flere maskiner.

Det har nogle fordele, da det ikke kun er almindelige PC man kan lave virtuelt, men også servere og netværk. Dette kan spare mange penge, som man ikke behøver at bruge på hardware.

Vi har valgt at bruge VMWare fordi vi har brugt den i undervisningen, den er forholdsvis hurtig og så er den billigere end at have en cloud løsning.

Vi har to maskiner, en test og en produktion. De ligner hinanden, men det er produktions maskinen brugeren ser. Imellem siden og databasen er der brugt SQL, mellem serveren og brugeren er det en HTTP/TCP forbindelse.

Blok navn: Router

Vi har valgt at bruge router som vores pc i forvejen er forbundet til, da det er nemt og hurtigt. Routeren er en essentiel del af et netværk, da den opdeler det i mindre netværk. Det er en fordel for mængden af trafik samt sikkerheden. Det er en fordel for mængden af trafik fordi, at man til tider er nødt til at lave en broadcast, og her er routeren et filter, så broadcasten bliver inden for netværket, og det er også det der er med til at øge sikkerheden ved mindre netværk.

Blok navn: scripts

Scripts er små filer, med kommandoer til ens maskine, som man kan få til at køre, både manuelt og automatisk. Det at de kan køre automatisk er en fordel, da man kan automatisere en masse opgaver. For eksempel som det ene script vi har lavet med backup. I stedet for at vi skal bruge tid på at logge ind, sætte en backup i gang og vente på at den bliver færdig. Kan vi få maskinen til at gøre det automatisk en gang om dagen, uden at skulle tænke mere over det.

Der findes forskellige scripting sprog. Hvis scriptet skal køres automatisk, må man fortælle maskinen hvilket sprog skriptet er skrevet i. Derfor vil man starte scriptet med at skrive sproget sådan her #!/bin/sh, hvis det er et powershell script.

Programmering

I denne del af arkitekturbeskrivelsen ville der fremgå fire essentielle funktioner fra programmet bagved websitet. Disse funktioner er et udpluk, da der er mange essentielle funktioner. Det fulde overblik over koden ses i den vedhæftede zip-fil, samt i det givne link til github.

Funktion: Update_machines()

Overordnet er update_machines() en funktion, der skal gennemføre den del af programmet, hvor der oprettes en booking. Funktionen tager 7 arguments: ‘maskine1’, ‘maskine2’, ‘username’, ‘wash_day’, ‘timeslot’, ‘sms_reminder’ og ‘id’. I toppen skabes en forbindelse til SQLite-databasen ‘database.db’. En ‘select’-query køres for databasen, hvor dataen for en specifik ‘user’ hentes. Dette lægges i en variabel kaldet “result”. Hvis værdierne for result[0][4] er lig med 0, køres to update-queries. Den ene opdaterer en tabel for machines. Den anden opdaterer en tabel for users, hvor variablen ‘has_a_booking’ sættes til 1. Ændringerne committes til databasen, og forbindelsen lukkes. Hvis værdien af results[0][4] ikke er 0, printes “You already have a booking”, og der re-directes til booking-siden.

Funktion: Register()

Overordnet er register() en funktion, som er den del af programmet, der gør registrering af brugere muligt. Øverst tjekker funktionen, om der er en POST-anmodning ved brug af 'request.method'. Hvis dette er tilfældet, har brugeren trykket på 'register' knappen. I denne POST-anmodning er der noget data, som er tilknyttet den bruger, der ønsker at registrere sig. Funktionen tjekker, at alle felter er udfyldt, da ingen kanstå tomme. Hvis de ikke er udfyldt korrekt, returneres en fejlmeddeelse. En HTTP-header sender brugeren tilbage til registreringssiden. Der tjekkes også, om brugernavnet er mere end 3 tegn langt. Hvis det er tilfældet, returneres igen en fejlmeddeelse, og brugeren sendes tilbage til registreringssiden. Den samme procedure gør sig gældende for password-felterne: Hvis de ikke matcher, sendes brugeren tilbage. Funktionen kalder 'check_if_user_exist(username)' for at tjekke, om brugeren findes i forvejen. Hvis det er tilfældet, sendes brugeren tilbage til login-siden. Hvis alle valideringer er vellykkede, kaldes 'register_user_to_db', som registrerer brugeren. Hvis der ikke er nogen POST-anmodning, har brugeren ikke indsendt formularen, og så returnerer funktionen en HTML-skabelon kaldet "register.html".

Funktion: view_bookings()

Overordnet er view_bookings() en funktion, som er den del af programmet, der skal fortælle brugeren den tid, vedkommende har booket. Øverst tjekkes om om brugerens ‘username’ er gemt i sessions-variablen. Hvis det er tilfældet betyder det, at brugeren er logget ind. Hvis der er en POST-anmodning, tjekkes det om feltet ‘delete_booking’ er ‘yes’. Hvis det er tilfældet, opdateres username forskellige steder, og brugeren tages til ‘home’-siden. ‘Username’ hentes fra session, og en anden funktion kaldet “view_booking(username) kaldes, som henter data for brugeren. Hvis der er booking-oplysninger på brugeren, hentes disse. Hvis dataen for ‘machine 1 and 2’ er 1, indsættes “Machine 1 og 2” som tekst. Hvis dataen for ‘machine 3 and 4’ er 1, indsættes “Machine 3 and 4”. Er ingen af felterne 1, er felterne tomme.

Vaskedagen omformateres ved at rydde op de strings den har. Hvis tallet/bookingnummeret er mindre end 113, indsættes “This Week” foran vaskedagen. Hvis bookingnummeret er større, indsættes “Next Week” foran vaskedagen. Tidspunktet omformateres også ved fjernelse af nogle tegn. I slutningen returneres en HTML-template kaldet “view_booking.html”. Booking-oplysningerne sendes som argumenter til templetten og vises på websitet.

Hvis brugeren ikke er logget ind - Altså, hvis ‘username’ ikke er i session - omdirigeres brugeren til login-siden med HTTP-headeren “Refresh”.

8. Løsningsdesign

8.1 Hjemmeside Struktur

Visuel repræsentation:

Booking siden skal laves som en ugentlig kalender, hvor man ser den nuværende uge og ugen fremad. Eftersom man kun kan booke på ugen man er i og ugen efter, behøver man ikke en normal kalender. Men i stedet kan man sætte det op som en tabel, hvor dagene for en uge bliver vist, med tiderne derved. Hver dag har to sæt af maskiner. Fordi man booker 2 af gangen, ses det visuelt med X eller O, om de er optaget eller fri hver især. Når et tidspunkt er fuldt booket, bliver feltet grå og ikke klikbart. Resten er felterne kan man ved at trykke på dem, gå til bekræftelses skærmen.

Et mock-up af denne beskrivelse kan ses ved **figur 8**

View							
	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
7-9	X - X	O - X	X - X	O - X	X - X	O - X	O - X
8-9	X - X	O - O	X - X	O - X	O - O	O - X	X - O
11-13	X - X	X - O	O - X	O - O	X - X	O - O	O - O
13-15	X - X	O - X	O - O	X - O	O - O	X - O	O - X
15-17	X - X	X - O	X - O	X - O	X - X	O - O	O - X
17-19	X - X	O - X	O - O	X - O	O - O	X - X	X - X
19-21	X - X	X - O	O - O	X - O	X - X	X - X	X - X
21-23	X - X	X - O	X - O	O - X	X - X	O - X	O - X

X = Optaget, O = Ledig

[Figur 8]

Bekræftelses-skærm:

Her kommer man ind efter at have klikket på et felt i tabellen. Man kan her se dato og tidspunkt for det felt man har valgt, samt ens eget rumnummer.

Formålet med denne side er at vælge hvilket af de to sæt af maskiner man ønsker at booke.

Efter man har bekræftet sit valg, videresender den efterspørgslen til serveren. Hvor den kigger i databaser for at tjekke at alt er i orden. Såsom at der ikke allerede er én der har booket i det tidsrum, at man er logget ind og har de rigtige oplysninger. Og at det kan gemmes korrekt. Når disse opgaver er gennemført, sender den brugeren videre til en ny side, der giver OK.

Administrations side:

Som administrator, kan det være nødvendigt at ændre på kontoejere som enten er glemt eller bliver misbrugt. Derfor skal administratoren have deres egen personlige side, hvor de har adgang til back-end. Her kan du ændre direkte på dataen i databaserne og ændre på kontiene direkte.

FLASK:

Hjemmesiden er bygget op omkring Flask. Flask holder siden sammen og sender data frem og tilbage mellem Jinja og back-end databasen.

Python sørger for disse features:

- GET/POST Requests
- SQL
- URL management

HTML:

Hjemmesiden er opdelt i fire kernekomponenter; Booking side, Bekræftelses side, Kontoadministration og Login-Side

- Siderne skal være optimeret til brug på smartphones i forskellige skærmstørrelser.
- Elementerne skal være lette at se og farverne skal passe til materialet.
- Hav de vigtigste interaktive elementer nemt tilgængelige på skærmen.

Figma:

For at få en idé om hvordan vores side kunne potentielt se ud og hvordan der integreres med den, brugte vi siden Figma til at designe en interaktiv prototype af hjemmesiden. På denne platform er det muligt at kreere og linke siderne sammen hurtigt. på bilag 4 kan man se formen for hvordan vi har kodet siderne.

Database:

På figur 9 og 10, er en tidlig version af indholdet af databasefilen.

DB 1: Booking

Machine 1/2	Machine 3/4	Who? (Room number of account)	Day (DATETIME)	SMS-Enabled Notification?
TRUE	FALSE	420	2023-05-21 7:00:00	TRUE
FALSE	TRUE	391	2023-05-21 7:00:00	TRUE
FALSE	TRUE	69	2023-05-21 9:00:00	FALSE
TRUE	FALSE	69	2023-05-21 9:00:00	TRUE
FALSE	TRUE	21	2023-05-21 11:00:00	TRUE
TRUE	FALSE	33	2023-05-21 11:00:00	TRUE
TRUE	FALSE	9	2023-05-21 13:00:00	FALSE
FALSE	TRUE	3	2023-05-21 13:00:00	FALSE
TRUE	FALSE	64	2023-05-21 15:00:00	TRUE
FALSE	TRUE	76	2023-05-21 15:00:00	TRUE
FALSE	TRUE	33	2023-05-21 17:00:00	TRUE
TRUE	FALSE	89	2023-05-21 17:00:00	TRUE
TRUE	FALSE	41	2023-05-21 19:00:00	FALSE
FALSE	TRUE	90	2023-05-21 19:00:00	FALSE
FALSE	FALSE	75	2023-05-21 21:00:00	FALSE
FALSE	TRUE	35	2023-05-21 21:00:00	FALSE

[Figur 9]

DB 2: Accounts

Phone Number	Room Number	Password	Has a booking?
+45 64 20 12 84	420	HelloPassword123	TRUE
+45 31 35 84 23	21	KuteCitty96	TRUE
+45 28 31 74 12	54	Scorblion!32	FALSE

[Figur 10]

8.2 Kode-beskrivelse

Update_machines funktionen fra app.py

Funktionen startes på linje 159-160 ved figur 11, med at skabe forbindelse til databasen ‘database.db’. På linje 163-164 sendes første query afsted, hvor vi spørger efter indholdet i tabellen ‘users’. Linje 163 ses en variabel med den kommando, vi gerne vil give databasen. Linjen efter, spørges cursoren om at skrive/sende kommandoen til databasen. Linje 165 fås/hentes svaret, somlægges i variablen result.

Efter at vi har fået resultatet, sættes en betingelse op, der ser om brugeren i forvejen har en booking. Har brugeren en booking, vil koden gå til 'else'-sætningen linje 214. 'else'-sætningen vil lukke forbindelsen til databasen, og give en kort besked, hvor den minder vedkommende om at man har en vasketid, samt at brugeren vil blive redirected til siden ‘view_booking’, så vedkommende kan se, hvornår vasketiden er.

Har brugeren ikke en booking i forvejen, vil vedkommende komme ind i ‘if’- sætningen linje 167. Når vi er kommet ind i 'if'-sætningen, laves på linje 169-171 en query mere til databasen. Denne query laves, for at opdatere databasens tabel ‘mashine_booking’, med oplysninger om hvilke maskiner, der er booket af hvem og hvilken dag. På linje 173-175 laver vi igen en query til databasen, denne gang for at opdatere tabellen ‘users’. På den måde styres, at de kun kan have 1 booking ad gangen. Efter vi har opdateret den sidste tabel, lukkes forbindelsen til databasen.

```

156 # Updates the database with the booking information.
157 def update_machines(machine1, machine2, username, wash_day, timeslot, smsReminder, id):
158     print('print 1') #####!!!!!!!
159     con = sqlite3.connect('database.db')
160     cur = con.cursor()
161     print('print 1.5')
162     #query = 'SELECT * FROM users WHERE id=?'
163     #query2 = 'SELECT * FROM users WHERE username=?'
164     cur.execute(query, (username,))
165     result = cur.fetchall()
166     print(result)
167     if result[0][4] == 0:
168         print('print 2') #####!!!!!!!
169         query2 = 'UPDATE machine_booking SET machine_1_2=?, machine_3_4=?, username=?, wash_day=?, timeslot=?, sms_enabled=? WHERE id=?'
170         cur.execute(query2, (machine1, machine2, username, wash_day, timeslot, smsReminder, id))
171         con.commit()
172         print('print 3') #####!!!!!!!
173         query3 = 'UPDATE users SET has_a_booking=? WHERE username=?'
174         cur.execute(query3, (1, username))
175         con.commit()
176         con.close()
177         print('print 4') #####!!!!!!!
178         if smsReminder == 1:
179             pass # ----- sms sat på pause -----
180             #time = '2023-05-29 14:53:00'
181             # sms function
182             def send_sms():
183                 phone_number = result[0][3]
184                 print('print 5') #####!!!!!!!
185
186                 account_sid = 'AC890b2e93b27ca68060de4a7c02d93'
187                 auth_token = 'AuthToken';
188                 client = Client(account_sid, auth_token)

```

[Figur 11]

```
191     message = client.messages.create(
192         from_='+13157401145',
193         body=f'Hej {username}. Husk din vasketid {wash_day}. ', 
194         to=f'{phone_number}'
195     )
196     print(message.sid)
197
198     _thread.exit()
199
200 scheduler = sched.scheduler(time_module.time, time_module.sleep)
201 t = time_module.strptime(wash_day, '%Y-%m-%d %H:%M:%S')
202 t = time_module.mktime(t)
203 scheduler.enterabs(t, 1, send_sms, ())
204 print('print 6') #####
205 _thread.start_new_thread(scheduler.run())
206 """
207
208 else:
209     # make booking without reminder
210     # redirect to my booking
211     print('print 2') #####
212     return 'dont forget your time!!!', {"Refresh": "3; url=/view_booking"}
213
214 else:
215     print('print 8') #####
216     con.close()
217     return 'you already have a booking !!!', {"Refresh": "3; url=/view_booking"}
218     # show booking not allowed.
```

[Figur 12]

Beskrivelse af register_user_to_db(), @app.route('register') fra app.py og tilhørende html.

Funktionens formål er at registrere en bruger i databasen med rumnummer, kodeord og telefonnummer. Funktionen ses under på figur (13)

```
12 def register_user_to_db(username, password, phone_number):
13     con = sqlite3.connect('database.db')
14     cur = con.cursor()
15     cur.execute('INSERT INTO users(username,password, phone_number, has_a_booking) values (?,?,?,?)', (username, password, phone_number, 0))
16     con.commit()
17     con.close()
```

[Figur 13]

Funktionen oprettes på linje 12, og tager imod 3 argumenter; ‘username’, ‘password’ og ‘phone_number’. På linje 13 oprettes en variabel kaldet ‘con’, som via funktionen ‘connect’ fra moduliet ‘sqlite3’ opretter forbindelse til databasen kaldet ‘database.db’. På linje 14 oprettes en variabel kaldet ‘cur’ som er cursorobjektet, og bruges til at interagere med med databasen. På linje 15 bliver en sql-string kørt af execute funktionen.

I tabellen ‘users’ indsættes fire ting: et username, et password, et telefonnummer og om de har en booking eller ej. Values er sat til spørgsmålstege, så navnene på variablerne ikke afsløres i

sql-stringen. ‘Execute funktion’ får vores sql-string samt en tupel med de ting, vi vil sætte ind som values.

På linje 16 og 17 gemmes det, vi har tilføjet til databasen og forbindelsen lukkes.

Her ses html for ‘register.html’ på figur(14)

```
Flask_klatvask > templates > register.html > ...
1  {% extends "base.html" %} 
2
3  {% block content %}
4  <div class="container_login_top">
5    <p style="font-size: 40px; color: #whitesmoke; text-shadow: 0px 4px 4px #rgba(0, 0, 0, 0.25); text-align: center; ">Register</p>
6    <br>
7    <div class="container">
8      <form action="/register" method="post">
9
10        <div class="mb-3">
11          <label for="Username" class="form-label">Room number</label>
12          <br>
13          <p style="font-size: 10px; font-style: italic;">*Numbers only</p>
14          <input type="text" class="form-control" id="Username" name="username">
15        </div>
16        <div class="mb-3">
17          <label for="Password" class="form-label">Password</label>
18          <br>
19          <p style="font-size: 10px; font-style: italic;">*We recommend having at least one special symbol and a mix of upper and lowercase letters</p>
20          <input type="password" class="form-control" id="Password" name="password">
21        </div>
22        <div class="mb-3">
23          <label for="Repeat_Password" class="form-label">Repeat password</label>
24          <br>
25          <input type="password" class="form-control" id="Repeat_Password" name="repeat_password">
26        </div>
27        <div class="mb-3">
28          <label for="Phone_number" class="form-label">Phone number</label>
29          <p style="font-size: 10px; font-style: italic;">*Please include country numbers with a +</p>
30          <input type="text" class="form-control" id="Phone_number" name="phone_number">
31        </div>
32      <br>
33      <button type="submit" class="btn btn-primary">Register</button>
34    </form>
35    <br>
36    <a href="/login" class="btn btn-primary btn-sm">Back</a>
37  </div>
38  {% endblock %}
```

[Figur 14]

Det vigtige her er, at der er en form, som starter på linje 9 og lukker på linje 35. Den har ‘method’ sat til ‘post’, så den kan sende information.

Formen har fire input felter: et til rumnummer på linje 15, to til password på linje 21 og 26 og et til telefonnummer på linje 31. I bunden af formen på linje 34 er en knap med ‘type submit’. Trykkes den knap, sendes informationen fra inputfelterne til det specificerede sted, som angives med ‘action’ oppe i formen på linje 9.

```

203     @app.route('/register', methods=["POST", "GET"])
204     def register():
205         if request.method == 'POST':
206             username = request.form['username']
207             password = request.form['password']
208             repeat_password = request.form['repeat_password']
209             phone_number = request.form['phone_number']
210             if not username or not password or not repeat_password or not phone_number:
211                 return 'please fill out everything', {"Refresh": "3; url=/register"}
212             if password != repeat_password:
213                 return 'password dont match', {"Refresh": "3; url=/register"}
214
215             if check_if_user_exist(username):
216                 return '<h1>User already exists</h1>', {"Refresh": "3; url=/login"}
217
218             else:
219                 register_user_to_db(username, password, phone_number)
220                 return redirect(url_for('index'))
221
222         else:
223             return render_template('register.html')

```

[Figur 15]

Går man ind på ‘@app.route(‘/rigster’)', kaldes funktionen ‘register’ som ses på linje 204. Fra linje 206 til 209 tages værdierne sendt fra ‘register formen’, og puttes i variabler. På linje 210 tjekkes om et af input felterne ikke er blevet udfyldt. På linje 212 og 215 tjekkes om de to passwords matcher, samt om der allerede findes en bruger i databasen. Hvis der gør, bliver man ledt til login. Hvis der ikke allerede findes en bruger med det indtastede navn, bliver funktionen ‘register_user_to_db’ kørt med de indtastede værdier, og en bruger bliver oprettet.

Beskrivelse af @app.route('login') og funktionen check_user() fra app.py filen samt tilhørende html

Login-funktionen har til formål at give en adgang til siden, hvis man allerede har oprettet en bruger tidligere.

```

1  [% extends "base.html" %]
2
3  {% block content %}
4  <div class="container_top1">
5    |   <h1> Kvintus Kollegiet
6    |   <p style="font-size: 27px; color: teal; text-align: center; ">Login</p></div>
7    |   <br>
8
9
10 <div class="container">
11   <form action="/login" method="post">
12     <div class="mb-3">
13       |       <label for="Username" class="form-label">Room number</label>
14       |       <br>
15       |       <input type="text" class="form-control" id="Username" name="username">
16     </div>
17     <div class="mb-3">
18       |       <label for="Password" class="form-label">Password</label>
19       |       <br>
20       |       <input type="password" class="form-control" id="Password" name="password">
21     </div>
22     <button type="submit" class="btn-primary-login">Login</button>
23   </form>
24   <br/>
25   <p>Don't have an account? <a href="/register" class="btn-primary-login">Create account</a></p>
26 </div>
27
28 {%- endblock %}
29

```

[Figur 16]

På figur (16) ovenfor ses html delen af login-siden. På linje 11 starter en form med ‘method post’ som sender det, der bliver skrevet i inputfelterne på henholdsvis line 15 og 20 til ‘login’ funktionen via formens ‘action’. Dette sker, når knappen af ‘type submit’ trykkes, som set på linje 22.

```

22  # Check if user and password match
23  def check_user(username, password):
24      con = sqlite3.connect('database.db')
25      cur = con.cursor()
26      cur.execute('Select username,password FROM users WHERE username=? and password=?', (username, password))
27
28      result = cur.fetchone()
29      if result:
30          return True
31      else:
32          return False

```

[Figur 17]

På figur (17) ovenfor er ‘check_user’ funktionen, som tager imod et brugernavn og et password. Så kigger den i databasen, om der eksisterer en user med det givne password. Hvis ja returneres ‘True’ ellers returneres ‘False’.

På figur (18) på ‘app.route(‘/login’)’ ses vores ‘login’ funktion. På linje 302 er der en condition, som tjekker om ‘request.method’ er ‘post’. Hvis den condition er ‘True’, oprettes variabler til username og password. I variablen ‘username’ kommer informationen fra inputfeltet med navnet username fra ‘login html’, og det samme gøres med ‘password’.

```

300  @app.route('/login', methods=["POST", "GET"])
301  def login():
302      if request.method == 'POST':
303          username = request.form['username']
304          password = request.form['password']
305          print(check_user(username, password))
306
307      if check_user(username, password):
308          session['username'] = username
309
310          return redirect(url_for('home'))
311      else:
312          return 'The account information provided did not match our database.', {"Refresh": "3; url=/login"}
313
314      else:
315          return redirect(url_for('index'))

```

[Figur 18]

På linje 307 er der en condition, der ved hjælp af den fornævnte funktion ‘check_user’, ser, om de indtastede informationer stemmer overens med det, der ligger i databasen. Hvis den returnerer ‘True’, bliver ‘username’ gemt i ‘session objektet’, der er en slags dictionary og ses på linje 308. Man bliver herefter sendt til ‘home’-siden, da login er sucessfuldt. Det ses på linje 310.

Hvis ‘check_user’ er ‘False’, bliver man efter 3 sekunder sendt tilbage til login siden via refresh. Det ses på linje 312.

Beskrivelse af funktionen status_machines() fra app.py filen

Funktionens formål er at få en status over hvilke sæt af maskiner, der enten er helt fri, halvt bookede eller fuldt bookede. Funktionen oprettes på linje 81, som ses på figur (19)

Der oprettes 5 tomme lister: en liste til hver maskinstatus.

```

81  def status_machines():
82      alle_maskiner = []
83      fuld_booked = []
84      alle_ledige = []
85      machine_1_2_fri = []
86      machine_3_4_fri = []
87      con = sqlite3.connect('database.db')
88      cur = con.cursor()
89      cur.execute('SELECT * FROM machine_booking')
90      result = cur.fetchall()
91      for row in result:
92          if row:
93              alle_maskiner.append(row)
94
95          if row[1] == 1 and row[2] == 1:
96              fuld_booked.append(row)
97
98          elif row[1] == 0 and row[2] == 0:
99              alle_ledige.append(row)
100
101         elif row[1] == 1 and row[2] == 0:
102             machine_3_4_fri.append(row)
103
104         elif row[1] == 0 and row[2] == 1:
105             machine_1_2_fri.append(row)
106
107     return alle_maskiner, fuld_booked, alle_ledige, machine_1_2_fri, machine_3_4_fri

```

[Figur 19]

Der oprettes forbindelse til databasen og køres en ‘select all’ fra vores ‘machine_booking’-tabel. Fetchall’ funktionen bruges til at tage det, vi får fra databasen, og komme det i en liste.

En liste af tupels her. På linje 91 itereres over listen med et ‘for-loop’. På linje 92 ses en condition, der tjekker om der er en row. Hvis ja, kommer den i listen ‘alle_maskiner’ med ‘append’ funktionen. Fra linje 95 til 105 kommes rows i de passende lister alt efter, om de har alle maskiner ledige, eller nogle er halvt eller helt bookede,

På linje 107 returneres listerne, således at de kan tilgås med forskellige funktionskald, som fx ‘status_machines()[0]’ for at få listen med alle maskinerne kaldet ‘alle_maskiner’.

Beskrivelse af booking() funktionen fra app.py samt tilhørende html

Funktionen har 2 formål: At give et visuelt overblik over, hvilke vaskemasinesæt som er optaget, og at videresende den valgte information til ‘confirm_booking’ funktionen og siden.

```

335     @app.route('/booking', methods=["POST","GET"])
336     def booking():
337         if 'username' in session:
338             if request.method=="POST":
339                 button_id=request.form["calender_button"]
340
341             if request.form["calender_button"] == str(button_id):
342                 button_data=dates_properties.date_data(button_id)
343                 print("ID: ",button_data.id, "Date: ", button_data.date, "Timeslot: ", button_data.timeslot)
344                 time_data=[button_data.id],[button_data.date],[button_data.timeslot]
345                 return redirect(url_for("confirm_booking", time_data=time_data, id=button_id))
346             # ----- prøver noget -----
347
348             status_machines()
349             temp_list = []
350             for i in range(len(status_machines()[0])+1):
351                 #for i in range(225):
352
353
354             for item in status_machines()[0]:
355                 if item[0] == i:
356                     print('item: ', item )
357                     if item[1:3] == (1,1) or item[1:3] == (1,0) or item[1:3] == (0,1):
358
359                         temp_list.append(item[0:1])
360
361             temp_list = str(temp_list)
362             temp_list = temp_list.replace('[(,)].replace((),)', '').replace('((,)).replace((),)', '').replace(' ', '')
363             print(temp_list)
364             return render_template('booking2.html', temp_list=temp_list)
365
366
367         else:
368             return 'please log in!', {"Refresh": "3; url=/login"}
369

```

[Figur 20]

‘Booking()’-funktionen ses overfor på figur 21.

På linje 337 er der en condition, som ser om ‘username’ er i session. Hvis det ikke er, bliver man bedt om at logge ind, og sendes til ‘login’-siden, som ses på linje 367. På linje 348 oprettes en tom liste kaldet ‘temp list’. Den kommer til at indeholde id fra alle de maskiner, der er bookede. På 349 ses et for-loop, som kører lige så mange gange, som der er items i ‘status_machines()[0]’ listen. På linje 354 ses for-loop, som løber over listen ‘status_machines()[0]’.

På linje 355 er der en condition, der tjekker om item på plads nul i tuplen, som er vaskemaskinens id, er det samme som 'i' fra 'for-loopet- på linje 349.

På linje 357 tjekkes om, 'item[1:3]' er det samme som (1,1), (1,0) eller (0,1).

Hvis det er det samme, bliver ‘item[0]’ puttet i listen kaldet ‘temp_list’ med ‘append’-funktionen. På figur (21) er et udsnit af vores tabel ‘machine_booking’ for at give et billede af, hvad der ligger i ‘status_machines’-listerne.

[Figur 21]

Tables (3)		id	machine_1_2	machine_3_4	username	wash_day	timeslot	sms_enabled
>	users	1	1	0	111	['Monday']	'7-9'	0
>	sqlite_sequence	2	0	1	420	['Monday']	'7-9'	0
>	machine_booking	3	0	0	0	0	0	0
		4	0	0	0	0	0	0

På linje 361 laves ‘temp_list’ om til en string. På linje 362 bliver alle unødvendige tegn fjernet fra listen, med ‘replace’-funktionen. På linje 364 retunes booking2.html, samt ‘temp_list’ via ‘render_template’.

[Figur 22]

På figur (22) ovenfor ses et udsnit af booking2.html. Da den er lang, vises kun det essentielle. På linje 5 tages ‘temp_list’, som blev sendt som en string af tal med kommaer imellem, og splittes ved hvert komma med ‘split’-funktionen. Det er nu en liste, og ikke en string. På linje 6 ses et for-loop, som kører så mange gange, som der er ting i listen via ‘jinja’-filter for length. På linje 7 inde i for-loopet bliver indholdet af ‘temp_list’ konverteret fra string til int via funktionen ‘__getitem__()’ og jinjas filter ‘| int’. Liste-elementerne er nu int, og kan bruges senere.

På linje 17 ses en tabel, på linje 18 er en table-row med table-headers inden i for hvert timeinterval. På linje 31 er en for-loop, der kører 16 gange, og laver en form med ‘method post’. Inde i den form, er der en knap kaldet ‘calender_button’, der har en value med værdien af “i” fra for loopet. Inde i knappen er der også en jinja-condition, der tjekker om ‘i’ fra for-loopet findes i listen ‘temp_list’. Hvis den gør, bliver knappen rød via style tag, og sat til ikke at kunne trykkes på med ‘disabled’. Hvis “i” ikke findes i listen, går jinja-condition til ‘else’, og knappen bliver farvet grøn med et andet style tag. I alt genereres 224 knapper på denne side: 112 knapper for den ene uge, og 112 knapper for den anden.

Hvis en grøn knap trykkes, sendes value til ‘booking’ funktionen med metoden ‘post’ fra den form, knappen ligger i. På linje 340 i figur 20-booking funktion-figuren ses variablen ‘button_id’ som får værdien af value fra knappen det blev trykket med name ‘calender_button’. På linje 342 ses en condition der tjekker om den value den fik er en string. På linje 343 ses variablen ‘button_data’ som via funktionen ‘date_data’ fra ‘date_properties’ får tildelt de korrekte informationer omkring dag og tidsinterval. På linje 345 kommer informationerne i en liste kaldet time_data. Til sidst bliver vi via redirect sendt til ‘confirm_booking’ med listen ‘time_data’ og id’et fra den valgte vaskemaskine.

8.3 Konfiguration af valgte servere

IP's

Network IP Adresse: 192.168.137.0/24

Subnet Mask: 255.255.255.0

Default gateway: 192.168.137.1

Available hosts: 254

IP Reservations:

(.1 –.9 Network Devices)

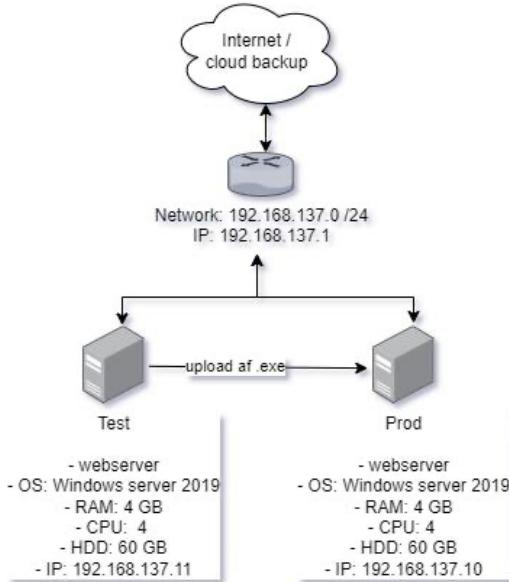
.1 – Firewall (Default Gateway) Assigned to the inside or LAN Interface of the Device

.2 – .9 can be used for routers and the IDS/IPS

.10 –.19 Servers

.20 –.29 Printers

.30 –254 Hosts



[Figur 23]

Vi har valgt, for at gøre det simpelt og spare ressourcer, at bygge serverne som 2 virtuelle maskiner bygget i VMWare. De er bygget på det samme virtuelle netværk. Begge servere er bygget som webservere, den ene som et testmiljø for nye funktioner og updates af siden, den anden server er til at hoste siden. Serverne får statiske IP adresser, for at være sikker på at de bliver ved med at kunne findes.

På grund af netværkets lille størrelse, har vi valgt den simple stjerne topologi for vores netværk.

Da vi i undervisningen meget har brugt Windows, valgte vi at vores servere skulle bruge Windows Server 2019 som operativsystem. For at være sikker på at der ikke bliver for meget rod med vaske reserveringer, har vi valgt at lave et script som laver backup en gang om dagen. Hvis en server skulle gå ned, eller hvis vi skulle få en anden kunde, har vi også lavet et script til at konfigurere serverne. På denne måde kan vi hurtigere få en server op at køre.

Der er også lavet en logbog, som beskriver ca. hvor lang tid hver opgave, med opsætning af server, vil tage. Det er god viden at have, ikke kun til fakturering af opgaven, men også til hvis der skulle være en server nede, så man ved hvor lang tid det tager at få en ny på at køre.

Loggen kan ses i bilag 5

9. Test af løsning

9.1 Udførelsel af brugertest

En brugertest af websitet blev afholdt med en beboer fra kollegiet. Brugeren beskriver, at det overordnede indtryk af websitet er, at den er logisk, samt at den er overskueligt at navigere rundt på. Brugeren kommenterede på nogle af de designmæssige beslutninger, der er blevet taget, og kom med forslag til ændring af disse for funktionaliteten. Brugeren kommenterede også på den information, der bliver givet ved booking-bekræftelse, og satte spørgsmålstegn ved nødvendigheden af den. Dette påviser, at selvom man som udvikler har en idé om, hvad brugeren gerne vil have, gør en jævnlig revurdering af dette gavn for det færdige produkt.

9.2 Udførelsel af Accepttest

Som produktet nærmede sig sin endelige version, begyndte vi testfasen. Testen gjordes ved at have forskellige typer af enheder, i dette tilfælde en PC, en tablet og en smartphone til at tilgå siden og se om elementerne fungerer korrekt under diverse skærmstørrelser.

Vi har selvfølgelig vores kravspecifikation at fokuserer på, men vi sørger også for at tjekke for andre mulige bugs eller glitches undervejs. Der anbefaler at have kravspecifikationen inden for rækkevidde.

1. Test:

- Video af vores test kan findes her: <https://www.youtube.com/watch?v=hALYv7PcKz8>

Vi startede vores første test med at opsætte Flask på en normal computer, og derefter forbinde den til internettet. Via det lokale WiFi, har enhederne adgang til til siden.

Der blev lavet brugere og der blev nавигeret uden problemer gennem siderne der var opsat.

Der blev prøvet at booke en tid på vores unikke kalender, og det lykkedes. Efter man har valgt et felt, skulle man vælge maskine, før en rigtig bookning kunne gennemføres.

Her fandt vi en større fejltagelse med vores program. Efter at der var foretaget en anden bookning på samme tidspunkt, men med en andet sæt af maskiner, forsvandt den tidlige bookning fra brugerens bookings-oversigt. Vi fandt ud af at ID til positionen på kalender-

oversigten, var det ID som også blev booket på databasen. Tildels var det hvad vi designede, men hvis hver tidsslot har to tider, der hver har et ID inde i databasen, skulle der være 224 bookninger. Men vi havde kun 112. Derfor, når der blev booket på et tidsslot, så overskrev den nye bookning, den tidligere bookning. Selvom det ikke var med de samme maskiner. Det gjorde at vi måtte stoppe vores test for at fikse denne fundamentale fejl.

2. Test:

- Video af vores test kan findes her:

<https://www.youtube.com/watch?v=e6UXqEKOCCs>

Efter fejlen blev rettet, genskabte vi vores test miljø og fokuserede på at tjekke op med vores kravspecifikation.

Under testen gik vi igennem login, konto-registrering, password-reset, booking, booking status og sletning af booking.

Som ses i videoen, kan man lave en ny bruger så snart man tilgår hjemmesiden og man kan derefter logge ind med den. Hvis der allerede er en bruger med samme rumnummer som blev indtastet, bliver brugeren bedt om at registrere et unikt nummer. Dette nummer ville ikke være højere end antallet af værelser tilgængelig på kollegiet. Når man har logget ind, kan man vælge at lave et nyt password. For at kunne dette, skal man have adgang til kontoen og kende det gamle password. Hvis ikke, så er det kun administratoren der kan slette brugeren og derefter lave et nyt password.

Så vi kan derfor sige at **krav 2 er : Opfyldt**

Administratoren har et unikt login og er sat som admin i databasen. Hjemmesiden tjekker admin privilegier, og giver adgang til ekstra funktioner på "Hjemmeskærmen". På dette tidspunkt hvor rapporten skrives, kan de slette en konto i databasen. Det er forudset at admin skal kunne redigere flere variabler med siden, som ville være der til en 1.0 udgivelse.

Vi kan derfor sige at **krav 3 er: Opfyldt**

Hjemmesiden har fra start været designet med mobile enheder som fokus. Næsten alle elementer er centreret, og hvad der kan interageres med, følger konventionen af andre sider på internettet.

Brugerens oplevelse, som vises i brugertesten, giver et bedre overblik over krav 6's succes.

Vi foretog en booking på én enhed, og derefter opdaterede siden på en anden enhed. Den booking der lige var blevet foretaget, var nu ikke længere muligt at trykke på. Da brugeren, der havde bookningen, gik ind og slettede den, var bookings tiden tilgængelig for alle.

Vi kan derfor sige at **krav 1 er: Opfyldt**

Inden bookingen blev foretaget, kommer man ind på en statusside. Her kan man se hvilket rum nummer man har, hvilket maskiner der er booket. hvornår de er booket til med både dag og timetal. Når man trykker bekræft-knappen og godkender bookningen, bliver bookningen booket og brugeren sendes derefter til "view_booking", hvor de kan se status på deres bookning også.

Vi har sagt booking meget i det her afsnit, men vi kan også sige at **krav 5 er: Opfyldt**

Den del vi ikke kunne teste denne gang var SMS-funktionen.

Det var meningen at en SMS ville blive sendt til ens telefon, tæt på tiden hvor man havde reserveret. Men på grund af komplikationer med koden, nåede denne del ikke at blive færdig.

Derfor kan vi sige at **krav 4 er: Ikke Opfyldt**

10. Praktisk projektplanlægning og ledelse

10.1 WBS

WBS kan ses i **bilag 6**.

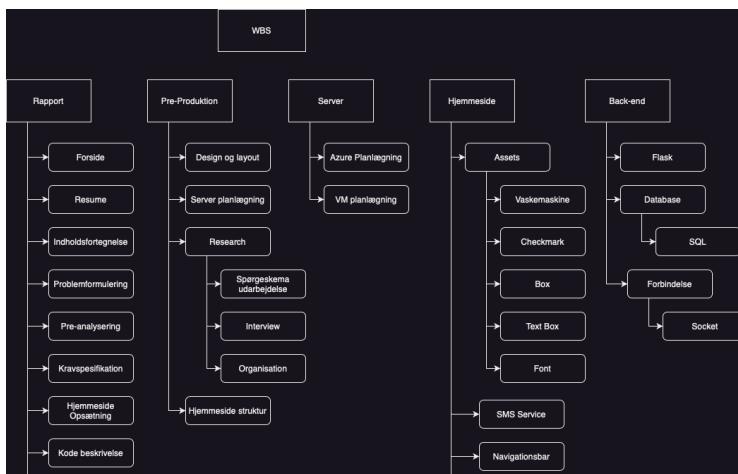
Under pre-produktionsfasen fastlagde vi problemet, skabte en potentiel løsning og udførte en generel plan for projektets scope og tidslinje. Efter dette udbyggede vi en mere konkret plan for enkelte dele af projektet, så vi havde et bedre overblik over alle potentialer.

Vi startede med en brainstorm. Det vi blev enige om, blev skrevet ind i vores officielle WBS flowchart, og derefter overført og tilpasset til et Scrum i Trello.

Punkterne til WBS'en har vi sammensat til fem hovedkategorier: **Rapport, Pre-produktion, Hjemmeside, Backend** og **Server**. Disse kategorier er blevet udvalgt efter deres estimerede tyngde på projektets tidsplan og arbejdsbyrde.

Som set i den tidlige version af WBS'en i figur 24, er der stor forskel mellem kategorierne i forhold til deres aktiviteter.

Dette er tilfældet med den erfaring, teamet har. Da teamet mangler erfaring med professionel brug af server og cloud-teknologi, er det komplette overblik ikke nødvendigvis fastsat i de tidlige stadier af projektet. I bilaget ses vores progression af WBS'en over tid i takt med, at scopet defineres bedre. Den fulde WBS kan findes på bilag 6.



[Figur 24 WBS, Tidlig ver.]

10.2 Gantt

Gantt-diagrammet for dette projekt kan findes i bilag 7.

Gantt-diagrammet, opfundet af Henry Gantt i 1910'erne, bruges som et ledelsesværktøj til at estimere og producere en tidslinje for længden af faserne, der tilhører projektet. (Olsen, N.V. og Lassen, S.M., 2021, s. 145 - 148)

Hvert projekt kræver deres egen tidslinje, da omstændighederne for hvert projekt er unikke og kan have indflydelse på estimeringen.

Det er dog vigtigt at påpege Gantt-diagrammets fokus på de overliggende opgaver og faser, hvor de mindre “tasks” fritages for at holde det overskueligt og konkret. Delegeringen af mindre opgaver og deres tidsestimat er pålagt Scrum og Sprints.

I vi har valgt at opdele faserne således i diagrammet:

- Rapport
- Pre-produktion
- Front-end
- Back-end
- Server
- Test

Vi ville med disse kategorier forklare deres omfang og vigtighed for projektet.

Rapport:

Som den videnskabelig repræsentation af projektet, er rapporten delen som sendes videre ud i verden, og derfor skal det indeholde overblikket over alle dele af projektet, samt at være finpudset til at overgive til en 3. aktør uden spørgsmål bliver stillet til dens autenticitet.

Rapporten formes i takt med vores egne opdagelser gennem projektet, derfor skal denne del arbejdes på gennem hele forløbet.

Pre-produktion:

Første del af projektet og vores skattekort til et vidunderligt produkt. I denne fase, der varer den første uge af projektet, udarbejder vi ideerne og vejledningen til produktet, så alle er i

samme sind og forstår skridtene der skal tages for at opnå vores mål. Der er tilslidesat den første uge af projektet til dette.

Front-End/Back-End:

Under denne kategori tilhører størstedelen af produktet. Det spindelvæv, der fastholder de mange dele af bookingsystemet, samt brugerfladen som kunder og brugere integrerer med, skal begge være omhyggeligt konstrueret og indbydende for målgrupperne at bruge. Denne del kan med andre ord siges at være vores produktionsfase og er derfor den del, der kræver mest samarbejde mellem teamets medlemmer. Denne fase er fastlagt til hele uge 22.

Server/Test:

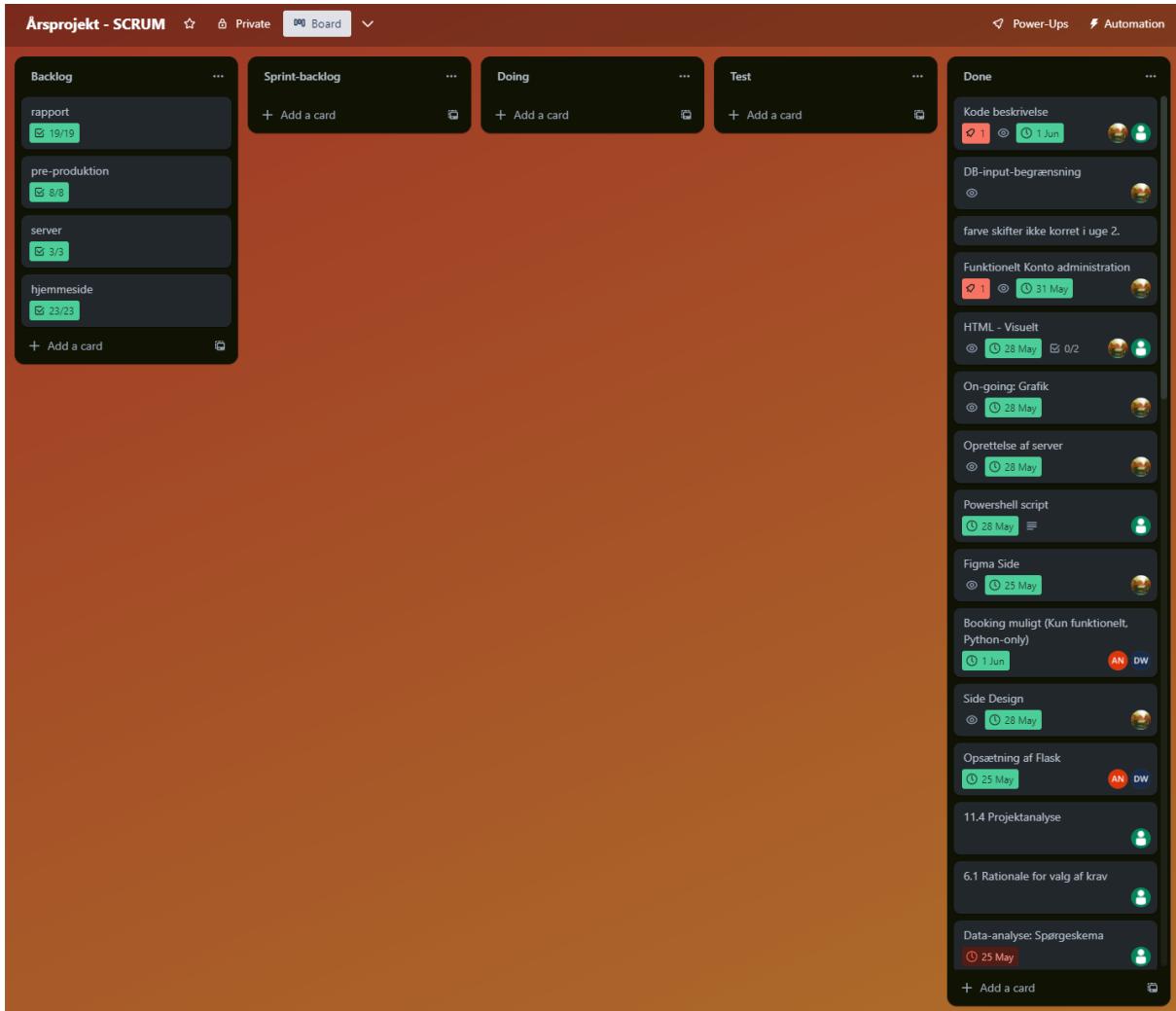
Denne fase af projektet tilhører delen hvor produktet overføres fra at være lokalt til at være på internettet. Opsætningen af dette og budgetteringen af ressourcerne hænger sammen med testfasen, der foregår i samme tidsrum og kræver at alle er “hands-on” med overførslen. Denne fase finder sted i sidste uge af projektet.

10.3 Kanban / Scrum

Vores scrum kan findes på følgende link: <https://shorturl.at/jrBN1>

Vores scrum følger den anbefalede opsætning, men vi føler det er vigtigt at påpege måden, vi har valgt at opererer vores Scrum på. Det er som følger:

- Backloggen fyldes op med “cards”, der står for de overordnede kategorier, og inde i dem laves der en checkliste over alle de mindre dele, der følger med under den pågældende kategori. Dette gøres for at holde det pænt visuelt.
- Når der aftales en sprint, laves der nye “cards” ud fra checklisten fra backloggen. Disse bliver herfra overført til sprint-backloggen indtil de er færdiggjort.
- Efter en sprint-periode er aftalt for et punkt sættes der dato på, og et medlem af teamet får tildelt ansvar for det. Det er vigtigt, at den person får færdiggjort den del inden næste sprint, hvor nye “cards” bliver tildelt.



[Figur 25]

10.4 Projektanalyse

Formålet med at lave en risikoanalyse er at klarlægge hvilke potentielle problemer, et projekt kan påvirkes af. Den bruges også til at vurdere omfanget af konsekvensen, hvis hændelsen skulle finde sted. Når de potentielle risici er klarlagt, kan man derudfra lave en strategi for at forebygge hændelsen og afhjælpe konsekvensen. De forskellige risici gives en score fra 1-5 i de to kategorier 'konsekvens' og 'sandsynlighed'. Scoren skal illustrere hvor alvorlig hændelsen potentielt kan være. (Mannaz, 2022)

Risikoanalysen udarbejdes ved at udføre en brainstorm, lave en vurdering og til slut lave en plan for håndtering. Brainstorming udføres for at klarlægge alle potentielle risici. Vurderingen laves af sandsynligheden for, at hændelsen finder sted, og størrelsen på konsekvensen, hvis dette sker. Håndteringen er at lave en plan for hvor enkelt risici, så projektet bedst muligt kanstå imod og fortsætte på trods af udfordringer. (Olsen, N.V. og Lassen, S.M., 2021, s. 95-97)

Hvad kan der ske?	Konsekvens 1-5 (K)	Sandsynlighed 1-5 (S)	Risikopoint (K) x (S)	Forebyggende eller afhjælpende foranstaltning	Ansvarlig
gruppemedlem syg	2	2	4	gøre alt materiale tilgængeligt for alle. og sørge for at alle hved hvad hinanden laver.	
fejl i revisionskontrol af filer.	3	3	9	bruge værktøj, f.eks. github, der kan holde styr på revisioner og som kan opdatere ændringer i filer.	
overse deadlines	2	2	4	fremhæv datoer på board	

undervurdering af hvor lang tid en opgave tager	3	3	9	af sæt mere tid end man forventer at bruge	
undervudere hvor teknisk svært en ønsket funktion er at lave	3	3	9	lav grundig research hav evt. simpelt alternativ.	
fejlvurdering af tid til debugging	3	3	9	hold kode så simpelt så muligt, og afsæt god tid	
nedbrud af gruppemedlems laptop	4	2	8	gem data remote, så alle kan tilgå den, også fra en ny computer.	
cloud service bliver efterladt åben, så konto bliver tømt før eksamen	5	2	10	undersøg hvornår en service koster penge, og hold forbruget til et minimum, intil eksamen	

10.5 Virksomhedsanalyse

10.5.1 - Laloux' farver

I en analyse af Kvintus Kollegiet som organisation er det vigtigt at tage i betragtning, at organisationens hovedformål ikke er at tjene penge. Kollegiet er registreret som en frivillig forening, der ledes af et beboerråd. (Proff.dk, 2023).

Den forretningsmæssige del af organisationen stræber efter, at kollegiet genererer penge nok til at alle udgifter dækkes. Det sker gennem betaling af husleje fra beboerne. Organisationen er også registreret som et boligselskab under navnet "Den selvejende institution Kvintus Kollegiet". (Proff.dk, 2023)

Kollegiet er registreret som en andelsforening, og det ejes af beboerne. Det giver et lidt anderledes vurderingsgrundlag i henhold til Laloux' farver.

Laloux' farver er en teori, som forsøger at analysere organisationers struktur og kultur.

Typerne af organisationer deles op i forskellige farver, som repræsenterer forskellige karaktertræk. Ofte besidder organisationer mere end én farve. (Laloux, F., 2017)

På trods af Kvintus Kollegiets lille størrelse som organisation, kan det deles op i forskellige grupper med hver deres dynamik og kultur. Kollegiet har grupperne: beboerrådet og alle beboere som enhed. Givetvis er beboerrådet en mindre gruppe og beboerne som helhed er en stor gruppe. Der er ofte forskel på, hvordan organisationen ser sig selv, og hvordan den reelt er. Til møder på kollegiet har alle lov til at bringe forslag op, som beboerrådet derefter kan tage stilling til. Det er kun beboerrådet, der kan stemme om endelige beslutninger. Desuden kan der forekomme lukkede punkter, hvor kun beboerrådets input tages i betragtning. (Kvintus Kollegiet, 2023)

I skabelse af forståelsen af organisationen har gruppen talt med en kilde, som er beboer på kollegiet. Det er ikke muligt at finde information om kultur og struktur ellers, og derfor er den mest pålidelige kilde en person, der aktivt oplever denne. Dog bør man derfor tage det forbehold, at beretningerne kan være subjektive og farvede af beboerens egne holdninger og opfattelser på trods af, at der blev efterspurgt neutrale svar.

Kollegiet bestræber sig på at være en grøn organisation, hvor alles mening kan tages op til overvejelse, og det er let at få kontakt til den, der tager beslutningerne. Beboerrådet anser sig selv som en gruppe, der er åben for kontakt med den større gruppe. En beboer fortæller, at der generelt sættes stor fokus på at skabe en kultur, som er imødekommede og inkluderende. Dette gælder i beslutningstagning såvel som til sociale arrangementer, der afholdes af kollegiet. Dog beretter beboeren, at beboerrådet ofte kan være intimiderende, når det kommer til at modtage nye idéer og tiltag fra beboergruppen. (interviewundersøgelse, bilag 3)

“... Så der er nok mange gode forslag, der aldrig kommer op, fordi at folk ikke er interesserede, eller at folk er intimiderede af, at det altid er de samme mennesker, der sidder dernede.... Det der med, at det er meget få mennesker kan også godt betyde, at der er lidt et monopol på beslutningerne på kollegiet på trods af, at det er demokrati i virkeligheden” -

Beboer, Kvintus Kollegiet

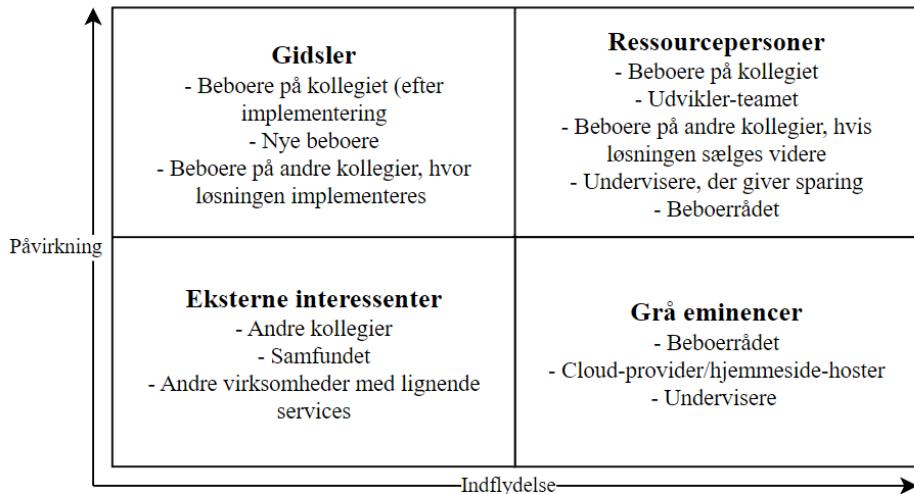
På trods af bred opbakning fra beboergruppen har beboerrådet mulighed for at forkaste et tiltag, de ikke bryder sig om. Det tyder på en hierarkisk struktur, hvilket taler mere ind i en ravfarvet organisation. Beboerrådet sidder med magten til at tage beslutninger, og så længe beboerne ikke udfordre autoriteten ved dette, er det let at være beboer.

På den baggrund er kollegiet et eksempel på en flerfarvet organisation. Generelt taler kulturen og strukturen ind i en grøn organisation, mens der potentielt også kan opstå ravfarvede momenter i beslutningstagningen. Grønne og ravfarvede organisationer ligger to farver fra hinanden, og på mange områder er de modstridende. Derfor bekræfter organisationen laloux' teori om, at organisationer ofte indeholder flere - og ligeledes ofte modstridende - farver på trods af, at det er en samlet helhed.(henvisning til PP fra undervisningen)

10.5.2 - Interessentanalyse

En interessentanalyse har til formål at klarlægge hvilke personer, der kan have interesse i projektet. Dette kan komme til udtryk på forskellige måder, og personerne kan have forskellig indvirkning, kræft og påvirkning på og af projektet. Resultatet af påvirkningen kan både være positiv og negativ. I starten af forløbet afholdes en brainstorming for at finde frem til alle væsentlige personer, der kan have interesse på den ene eller anden facon i projektet. Herefter

sorteres interesserne i kategorier som vist i modellen herunder. Efter sortering udvælges de mest essentielle interesserne, da det ikke er muligt at fokusere på alle. Mål, interesser og succeskriterier for de pågældende interesser diskuteres, og der udarbejdes en beskrivelse af disse. Efter dette gøres der overvejelser omkring hvilke interessekonflikter, projektet måtte have, og hvordan disse løses bedst muligt. Til slut vurderes det samlede overblik, og herefter besluttes, hvorvidt projektet kræver justeringer på baggrund af vurderingen. (Olsen, N.V. og Lassen, S.M., 2021, s. 99-117)



[Figur 26]

11. Konklusion

Konklusionen der kan uddrages af projektet er, at et digitalt bookingsystem potentielt kan afhjælpe nogle af de problematikker, beboerne oplever med det nuværende system. Disse indebærer blandt andet den manglende mulighed for at aflyse tider, samt den manglende evne til at få et hurtigt overblik og booke tid udefra. Beboerne beskrev nogle af disse problematikker, samt påvirkningen af disse igennem de foretagne undersøgelser. Resultaterne blev taget i betragtning, da systemet efterfølgende skulle udarbejdes.

Den digitale løsning kunne skabes ved at lave et website, der opfylder de funktionaliteter, bookingsystemet bør have. Funktionaliteterne indebærer både features fra det gamle system, samt nye features efterspurgt af beboerne. Visionen for projektet var at skabe et webbaseret system, som både efterlever de ønsker, beboerne havde, samt generelt giver en større funktionalitet til systemet. Dette betragtes som værende opfyldt ved udgangen af projektet.

Det har ikke været muligt at foretage en præcis måling af, hvor meget mængden af spilde vasketider kan formindskes. Man kan dog foretage en gisning om, at når muligheden for at aflyse tider gives til beboerne, har de i højere grad mulighed for at nedbringe mængden. Dette er noget, man efter implementering kan måle bedre, og prognosen er foreløbig, at løsningen giver et stort potentiale for at nedbringe spilde vasketider.

12. Projektforløbet

Projektforløbet er generelt forløbet efter forhåbningerne. Gruppen har efter to foregående projektforløb erfaring med at samarbejde, og det har derfor været muligt at tage erfaringer med derfra. Kommunikationen er, som i de foregående forløb, forløbet ukompliceret, og der har i alle aspekter været plads til input fra alle gruppemedlemmer.

En erfaring gjort fra det seneste projekt var, at projektstyring generelt skulle være en mere fremtrædende faktor i processen. Derfor var der i denne projektperiode langt større fokus på styring og forberedelse, inden det tekniske aspekt af projektet havde opstart. Det gav et bedre forløb, hvor struktur og helhed for projektet i højere grad var til at få øje på.

Det er altid svært at sætte tidsestimat på projektets elementer, da komplikationer kan forekomme på steder, man ikke forventer det. Nogle tekniske elementer af projektet tog længere tid end først antaget. Som en erfaring til senere projekter er aspektet med tidsestimat noget, man gang på gang kan prøve at perfektionere.

Det samlede helhedsindtryk af gruppens projektperiode er, at medlemmerne gradvist bliver mere erfarte for hvert forløb. Det kommer til udtryk både i projektstyring, tekniske færdigheder og evnen til at udarbejde et skriftligt produkt. Der er absolut sket en forbedring i alle henseender, når projektperioden sættes i sammenligning med gruppens første projektperiode.

13. Perspektivering

Ved projektstart var det tydeligt, at løsningen var tiltænkt en helt specifik målgruppe - beboerne på Kvintus Kollegiet. På den baggrund er funktionalitet og design tilpasset derefter. I den videre proces med kollegiet er fremtidsudsigten, at løsningen forhåbentlig kan implementeres på en tilfredsstillende måde, og derefter sættes i produktion for beboerne.

Såfremt en succesfuld implementering af løsningen finder sted, ville det være en fordel at undersøge, om løsningen også kunne have relevans for andre kollegier eller boligkomplekser. Inden udviklingen føres videre, kan man med fordel undersøge dybdegående, om der er behov og ønske om dette. En bredere målgruppe for undersøgelser ville ligeledes være at foretrække, så det er muligt at basere sine beslutninger på en større mængde data.

Da løsningen er meget skræddersyet til Kvintus Kollegiet specifikt, ville det formentlig gavne den videre udvikling, hvis man undersøger hvilke behov, de enkelte kunder til løsningen har. Behovene fra andre boligkomplekser og kollegier er måske afvigende fra Kvintus Kollegiets, og dette ville man skulle tilpasse for at sælge løsningen til andre. Derudover ville det være givende for den videre udvikling at undersøge flere features fra andre typer booking-systemer. At foretage den slags research ville give bedre mulighed for at udvide den mængde af features, man udbyder i sin implementering. Andre features ville potentielt også kunne imødekomme ønsker og behov fra andre kunder.

Løsningen kan potentielt forøge kvaliteten af den oplevelse, det er at bo på kollegiet. Det giver kollegiet forbedrede faciliteter, og kan derfor være medvirkende til at gøre Kvintus Kollegiet til et mere attraktivt sted at bo.

14. Litteraturliste

Den selvejd Inst Kvintuskollegiet - Proff.dk (2023) *Den Selvejd Inst Kvintuskollegiet - 74156428 - København S.* Tilgængelig hos: <https://www.proff.dk/firma/den-selvejd-inst-kvintuskollegiet/k%C3%B8benhavn-s/andelsforening/GL9QQSI03T7/> (Tilgået: 16 Maj 2023).

Engmann, T.S. (2020) *18.000 studerende flytter Ved Deres Studiestart på videregående Uddannelser, Danmarks Statistik.* Tilgået at: <https://www.dst.dk/da/Statistik/nyheder-analyser-publ/bagtal/2020/2020-08-28-18000-studerende-flytter-ved-deres-studiestart-paa-videregaaende-uddannelser> (Tilgået: 24 Maj 2023).

Johnson, A. (2023) *Design Thinking: Når IT-Løsninger tager udgangspunkt i brugernes reelle behov*

Kvintus Kollegiets Beboerråd - Proff.dk (2023) *Kvintus Kollegiets Beboerråd - 38311859 - København S.* Tilgængelig hos: <https://www.proff.dk/firma/kvintus-kollegiets-beboerr%C3%A5d/k%C3%B8benhavn-s/medlemsorganisationer/0MT5MBI10PU/> (Tilgået: 16 Maj 2023).

Kvintus Kollegiet (2023) *Kvintus Kollegiet: Et kollegie ved Lergravsparken.* Tilgængelig hos: <http://www.kvintus.dk/> (Tilgået: 16 Maj 2023).

Laloux, F. (2017) 'del 1', in *Fremitidens organisation: En Illustreret Vejledning til meningsfulde former for Samarbejde.* Nørre Snede: Direction Books, side. 12–39.

Olsen, N.V. and Lassen, S.M. (2021) *Projektledelse.* 2. Kbh., DK: Gads Forlag, Side 25 - 256

Relevans (2023) *Sådan Bruger du Persona, Relevans.* Tilgængelig hos: <https://www.relevans.dk/saadan-bruger-du-persona> (Tilgået: 16 Maj 2023).

Risikoanalyse – metodebeskrivelse (2022) Mannaz. Tilgængelig hos:
<https://www.mannaz.com/da/projektmodel/analysefasen/risikoanalyse-metodebeskrivelse/>
(Tilgået: 18 Maj 2023).

15. Bilag

- 1. Bilag - Businesscase**
- 2. Bilag - Interview Transskription**
- 3. Bilag - Spørgeskema diagrammer**
- 4. Bilag - Figma**
- 5. Bilag - Server Log**
- 6. Bilag - WBS**
- 7. Bilag - Gantt**
- 8. Bilag - Work hours**

1. Businesscase - Kvintus Kollegiets booking-system

Forklaring

Formålet med projektet er at lave et attraktivt booking-system til Kvintus Kollegiets vaskekælder. Et web-baseret booking-system kan tilføje til kollegiets portefølje af faciliteter, hvilket der på nuværende tidspunkt sættes fokus på. Omkostningerne og det negative udbytte i forhold til det positive udbytte er små, og på den baggrund anses projektet som en fordelagtig investering i at løfte kollegiets faciliteter til det bedre.

Begrundelser

Projektet kan understøtte Kvintus Kollegiet som organisation, fordi det forbedrer deres faciliteter, da det kan resultere i at kollegiet bliver et mere attraktivt sted at flytte til. Kollegiet har et ønske om at fremvise dets gode faciliteter på den tilhørende hjemmeside, og et online bookingsystem til vaskekælderen ville tilføje til deres portefølje af faciliteter. (Kvintus.dk)

Systemet giver mulighed for en bedre booking-oplevelse, da problematikken med glemte bookinger løses. På nuværende tidspunkt har alle lejemål en brik, som kan låses ind i det gitter, der udgører bookingsystemet. Hvis beboerne ved en fejl glemmer at fjerne deres vaskebrik, kan de uhensigtsmæssigt komme til at booke en tid, de ikke får benyttet. Denne risiko elimineres, da bookingerne i det online system fjernes automatisk efter brug.

Forretningsmuligheder

Gøre noget

Ved at gøre noget, har organisationen muligheden for potentelt at tiltrække flere beboere, da bedre faciliteter gør kollegiet mere attraktivt at bo på. Projektet kan desuden potentelt forøge den positive oplevelse af at bo på kollegiet. Derudover giver implementering af systemet mulighed for at effektivisere bookingen, da et online system fjerner risikoen for, at vaskebrikkerne ikke flyttes til tiden, og derfor bookes uden at blive brugt. Systemet giver

ligeledes mulighed for at aflyse tider, som kan låse op for, at andre har mulighed for at bruge tiden i stedet. Denne tilføjelse til systemet kan potentielt nedbringe mængden af spildte tider betydeligt.

Ved at lave en fuld implementering af bookingsystemet, har organisationen mulighed for at opnå en generel forbedring af faciliteter og kvaliteten af at være beboer, samt nedbringe mængden af glemte vasketider. Det skyldes påmindelsesfunktionen, som giver en elektronisk påmindelse

Gøre et minimum

Ved at gøre et minimum kan organisationen stadig opnå nogle af fordelene ved at implementere løsningen. Dog kunne 'at gøre minimum' også betyde en ikke lige så effektiv reklamering af den nye facilitet på eksempelvis kollegiets website. Det kan resultere i, at potentielle nye beboere ikke finder kollegiet lige så attraktivt, som de måske ville have gjort, hvis der aktivt blev reklameret for løsningen.

Gøre ingenting

Ved at gøre ingenting forbliver kollegiets portfolie af faciliteter, som den er på nuværende tidspunkt. Det betyder, at mulighederne for at booke tid i vaskekælderen forbliver en løsning, hvor man fysisk skal stå foran tavlen for at booke. Det fastholder booking-systemet, ifølge os, i et forældet stadie i forhold til det digitale samfund, vi lever i. Desuden nedbringer kollegiet ikke mængden af glemte vasketider, hvis systemet ikke bliver implementeret, hvilket resulterer i ineffektiv booking.

Ved ikke at implementere systemet går kollegiet glip af en oplagt mulighed for at blive mere attraktivt, samt forbedre oplevelsen af at være nuværende beboer.

Forventet udbytte

Med projektet ønsker vi at nedbringe mængden af glemte vasketider med 80% . Estimatet er lavet på baggrund af, at muligheden for fejlagtigt at booke en tid elimineres, når bookingen

frigøres efter hver vask. Derudover skal elektroniske påmindelser være medhjælpende til, at beboerne ikke glemmer, at de har booket en tid og derfor ikke møder op.

Da beboerne betaler passivt gennem huslejen til kollegiet faciliteter, er det ikke relevant at beregne det økonomiske udbytte. Organisationens formål er desuden ikke at producere overskud, men at balancere udgifter og indtægter.

Udbyttet af projektet er derfor mere målbar i beboertilfredshed og effektivitet af bookingen.

Forventet negativt udbytte

Da det ikke kan forventes, at alle beboere er lige kyndige til IT, kan et muligt negativt udbytte være, at nogle beboerne ikke kan begå sig i systemet og forstå hvordan det nye system skal bruges, eller at de kan finde det forvirrende at navigere i.

Et andet potentiel negativt udbytte kunne være, at beboerne ikke kan lide den nye løsning og hellere foretrækker den allerede eksisterende, analog og fysiske løsning, da det er den løsning de allerede kender og er bekvemme med at bruge.

Et tredje negativt udbytte kunne opstå, hvis IT-systemet er nede og beboerne derfor ikke kan navigere i vasketider, bookinger, mv. og dette kunne derfor give anledning til konflikter og irritationer.

Tidsrammer

Produktionen af selve IT-systemet og dets funktioner forventes at tage omkring fire uger at udvikle og realisere, hvorefter implementeringen nok tager en til to uger.

Det forventes at tage et par måneder at få alle beboere oprettet og indkørt i systemet, da de har forskellige rutiner, ikke tilpasser sig ændringer i samme tempo og løsninger som denne

kræver en omstillingsparathed og dedikation fra brugerne.

Ved at overgå til det nye system vil det ikke være muligt at benytte det gamle.

Omkostninger

Der ville være omkostninger forbundet med at udvikle web-appen og opsætning af den bagvedliggende struktur. Derefter vil der ikke være større udgifter forbundet med drift og vedligeholdelse, andet end server omkostninger, hvis organisationen ikke ønsker at implementere løsningen på den allerede eksisterende hjemmeside.

Hvis der på et senere tidspunkt skulle ønskes yderligere funktioner, ville der være omkostninger forbundet med denne udviklings- og udførselsproces.

Investeringsvurdering

Da organisationen ikke stræber efter at tjene penge, men derimod gerne vil tilbyde deres beboere gode og moderne faciliteter, vil et online bookingsystem passe godt ind i kollegiets nuværende udbud af tjenester og ydelser. Dette vil være et godt supplement til de mange andre ydelser, som kollegiet allerede tilbyder dets beboere og som gør kollegiet til et attraktivt sted at bo og forhåbentligt gøre beboernes hverdag nemmere og mere overskuelig.

Som nævnt i ovenstående afsnit (udbytte, negativt udbytte og omkostninger), vil investeringen være en attraktiv løsning for organisationen, da den negative udbytte er lille, omkostningerne er beskedne og udbyttet stort.

Væsentlige risici

En negativ risiko kunne være som nævnt tidligere, at brugerne simpelthen ikke bryder sig om den nye løsning, og hellere vil have den gamle.

En anden risiko med negativ udbytte kunne være, at organisationen hellere vil have en anden, mere afprøvet IT-løsning end denne, da der er andre udbydere af lignende løsninger.

Hvis udviklingen og implementeringen forløber godt, ville det være muligt at tilbyde samme løsning til lignende organisationer, såsom andre kollegier, andelsforeninger eller boligforeninger med fælles vaskerum, og hermed give projektet en forøget mulighed for at få indpas på markedet.

2. Interview-transskription

Interviewundersøgelse

S = Interviewer // B = beboer

S: Vi er i gang med den del af interviewundersøgelsen, som omhandler vaskekælderen. Og..
Du kan starte med at state; Hvad er dit køn og din alder?

B: Jeg er en mand på 25 år.

S: Og hvor længe har du boet på kollegiet?

B: Jeg har boet på Kvintus Kollegiet i.. Det må være snart.... To, lige over 2,5 års tid.

S: Ja. Og hvordan er din oplevelse af booking-systemet i vaskekælderen, som det er nu?

B: Jeg synes faktisk, at det er så besværligt, at jeg ofte undgår at booke vasketider, og i stedet bare går ned og for at tjekke, hvorvidt der er nogle ledige i virkeligheden.. Og fordi der er det system, der er lige nu, så er jeg nødt til at gå derved 10 minutter efter en vasketid er startet for at bekræfte, at der ikke er dukket nogen op til vasketiden, og så vaske der. Jeg er nødt til at time, hvornår jeg dukker op til vasketider på trods af, at jeg ikke booker en tid.

S: Ja.. Og det leder jo nok lidt videre til; Er du tilfreds med systemet? Og hvorfor/hvorfor ikke?

B: Altså, det er jo ikke et ubrugeligt system. Systemet virker, det er bare ikke praktisk. Det... én ting er, at der ofte er gamle vaskebrikker, der sidder dernede, da det sidder fysisk på væggen. Så, der kan sagtens være nogen, der har bestilt en vasketid sidste måned, som så stadig blokerer den samme vasketid hver eneste måned. Øhm.. En anden ting er, at for dem, der bor på for eksempel 4. Sal, der hører jeg, at de snakker om, at de gider heller ikke booke vasketider fordi, at det er en lang tur at gå ned i kælderen for at vurdere, om man kan få lov at vaske eller ej så gå op og hente sine ting igen, så det bliver en hel process i sig selv. Så jeg vil sige, at det virker, men der er meget plads til forbedring. Det er et oldgammelt system.

S: Hvis man nu skulle forbedre det, hvad ville man i så fald kunne gøre?

B: Øhm... Det ved jeg ikke.. Man kunne hvis man skulle forbedre det, så skulle man gøre det.. Gøre det mere tilgængeligt, og så.. Man kunne jo eventuelt droppe det tidsslotsystem, der er lige nu. Der er jo ikke nogen grund til, at.. at hvis man nu laver et internet/online eller et eller andet i den stil, så er der måske ikke nogen grund til, at det står i de her 7-9, 9-11-

tidsblokke længere. Måske kan det bare være noget med, at man booker x tid op til en maksimum mængde, og så se hvordan det fungerer.

S. Ja.. Du svarede lidt på det før, men bruger du ofte vasketider, du ikke selv booker?

B: Øh.. Ja, jeg bruger næsten kun vasketider, jeg ikke selv booker. Men det forslag jeg havde selv.. Jeg tænker lige over det. Det kan godt være, at det bliver lidt træls med, at så kommer der til at være en halv time, hvor maskinerne ikke bliver brugt imellem og sådan nogle ting. Det ved jeg ikke, om man kan arbejde udenom.

S: Hvor ofte glemmer du dine egne vasketider, eller udebliver fra tider, du har booket?

B: Det sker faktisk ikke særligt tit fordi, at jeg kun booker tider, hvis jeg virkelig ved, at jeg skal nå at have vasket af en eller anden årsag.

S: Tror du, at det system der er nu sådan.... lægger op til, at du har den adfærd omkring det?

B: Ja, altså.. Selvfølgelig, det er.. Jeg tager det meget, meget afslappet omkring det og ikke gør så meget ud af det fordi, at jeg ikke synes det er besværet værd, men øh.. Så det er kun hvis der er en situation, der er besværet værd, hvor jeg skal et eller andet vigtigt, hvor jeg skal bruge noget rent tøj, at jeg faktisk gør det.

S: Ville det være fordelagtigt for dig, hvis booking-systemet kunne tilgås online i stedet for?

B: Ja, ja, selvfølgelig.. Helt sikkert. Selvfølgelig ville det det.

S: Ja.. Og altså.. Hvad tror du, det ville forbedre i din vaskeprocess?

B: Det ville bare gøre det mere overskueligt. Det ville gøre det mindre til noget, man skal planlægge sin dag omkring måske, og så være noget, der kan blive planlagt omkring ens dag.

S: Ja.. Og hvis systemet nu skulle være elektronisk, hvilke features kunne du så godt tænke dig, at systemet havde?

B: Øhm.. Altså.. Jeg synes det kunne være fedt, hvis man kunne garantere, at folk ikke stjæler vasketider ved ligesom at putte det på en NFC-chip, så når man skal bruge maskinen dernede, for ligesom at.. Bekræfte den vasketid, man har. Jeg kunne også godt tænke mig, hvis vi kom ud af det der meget rigide system, der er dernede lige nu, som egentlig ikke behøver at være så.. Så struktureret. Men det er jo nok også smag og behag. Hvad var spørgsmålet igen? Bare for at sikre, at jeg svarer.

S: Hvilke features kunne du godt tænke dig et online booking-system havde?

B: Øh... Det kunne være smart, hvis det kunne integreres til min kalender, eller hvis det skulle være en lidt mindre løsning, kunne det også være smart, hvis jeg kunne trykke på en knap, og så blev det sat ind i min alarm-app. Øh.. Det kunne være fedt at få en eller anden form for notifikation om, at 'nu er din tid startet', og så måske også, hvis.. Det er måske nærmest umuligt.. Men også 'nu er din vask færdig'. Det kunne være super praktisk, så man ikke skulle gå og sætte tiden. Hvis man nu kører 4 maskiner på samme tid, som vi nogen gange gør, så kunne det være rart, hvis man ikke skulle holde 4 ure i gang hele tiden.

S: Tak for din feedback.

B: Selv tak.

Kultur og struktur på Kvintus Kollegiet

Interviewundersøgelse

S = Interviewer // B = Beboer

S: Vi er i gang med at optage den del af interviewundersøgelsen, hvor vi snakker om kultur og struktur på kollegiet. Og du kan lige starte med at state: Hvad er dit køn og din alder?

B: Jeg er en mand på 25 år.

S: Hvor længe har du boet på kollegiet?

B: Jeg har boet på kollegiet mellem 2,5 og 3 år.

S: Hvordan er din oplevelse af kulturen på kollegiet? Man kan sige, at nu har du jo en del erfaring her, når du har boet her så længe.

B: Erfaring i at bo her? Ja, det kan man godt sige, at jeg har. Jeg synes, at der er en rigtig tæt kultur på kollegiet. Der er.. Der er mange aktiviteter, man kan deltage i. Og dem der deltager, er en meget tæt gruppe. Øhm.. Mig selv inklusiv i den gruppe. Jeg vil mene, at der er mange, meget tætte venner, der deltager til rigtig mange ting dernede. Så, det kan godt være, at det bliver for tæt, men der er en venskabelig kultur på kollegiet. Fællesskab.

S: Så i forhold til beboerrådet.. Føler du, at det er let at komme med forslag til beboerrådet?

B: Jeg skal være ærlig og sige, at jeg har ikke deltaget særlig meget ved beboerrådsmøder, fordi det interesserer mig ikke, og det tror jeg.. Eller, det ved jeg, fordi jeg har været der, at det er en holdning, som også holdes af 90% af andre mennesker på kollegiet. Så alene af den grund er det måske svært at komme med forslag, fordi der er meget, meget få mennesker, som faktisk vælger.. Øh.. Det er svært, der er en meget høj 'barrier of entry'. Det er meget

uinteressant, så de fleste folk holder sig fra det. Så der er nok mange gode forslag, der aldrig kommer op, fordi at folk ikke er interesserede, eller folk er intimiderede af, at det altid er de samme mennesker, der sidder dernede.

S: Hvordan føler du, at beboerrådet modtager forslag fra beboerne? Det er lidt en gentagelse af det, du lige har svaret på, men hvordan.. Hvis der er et forslag, hvordan føler du så, at det bliver modtaget?

B: Igen, nu har jeg ikke været til så mange.. Men det der sker er jo, at der bliver stemt på det. Og hvis det er et stort forslag, så skal det også afstemmes til en generalforsamling af en art. Øhm.. Så der er jo lidt bureaucrati, men det er måske meget godt, når det er en vennegruppe, der styrer det, men det kan også betyde.. Det der med, at det er meget få mennesker, kan også godt betyde, at der er lidt et monopol på beslutningerne på kollegiet på trods af, at det er demokrati i virkeligheden. Så.. Ja, så det kan godt være, at det er svært at få noget igennem, men det afhænger af.. Dem der er dernede, er meget praktisk-orienteret, så hvis der er et godt forslag, så skal de nok støtte op omkring det.

S: I forlængelse af det: Gør modtagelsen af idéer, at man føler sig tryg ved at komme til beboerrådet med dem? Er det et ‘safe space’ de giver?

B: Det føler jeg ikke, at jeg kan svare på, når menneskerne dernede også er mine venner, så selvfølgelig er det safe for mig, men for en udefrakommende kan jeg godt forestille mig, at det måske er lidt.. Lidt skræmmende.

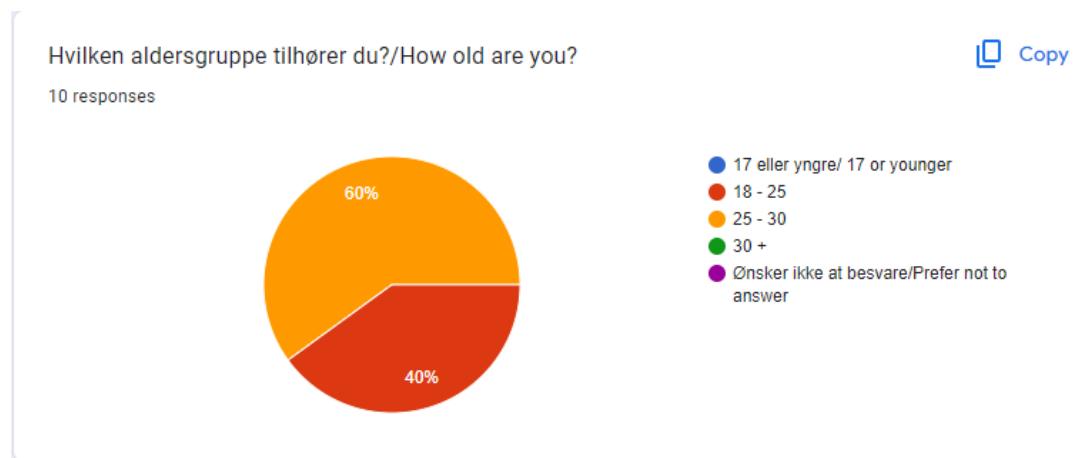
S: Så snakkede du om måden det bliver styret på her. Føler du, at beboerrådet har for meget styring i forhold til de beslutninger, der bliver taget for kollegiet?

B: Nej, det føler jeg ikke. Der er nogle, der skal tage beslutninger, og jeg synes, at den måde det er struktureret på her er den bedste mulighed, der er. Beboerrådet er beboere, som vælger dem.. Dukker op til en generalforsamling, så jeg kan ikke forestille mig en måde, hvorpå man kan få et bedre system til at styre det. Det er bare sådan, at ingen systemer er perfekte, og det er den her imperfekte måde, de har valgt at gå med her.

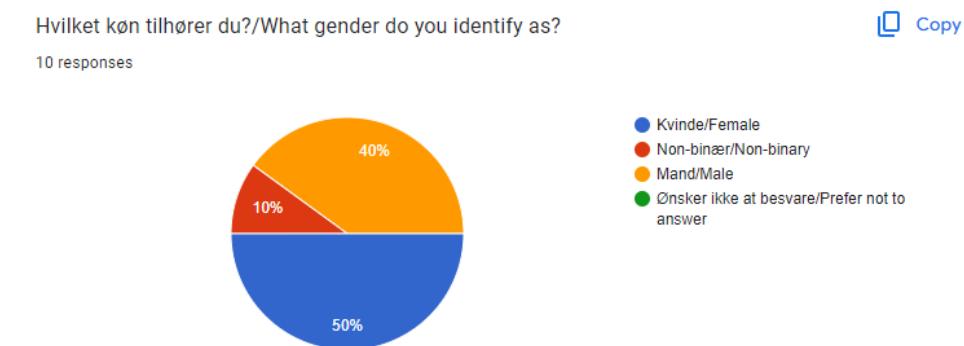
S: Tak for din besvarelse.

B: Selv tak.

3. Spørgeskema diagrammer



Spørgsmål 1

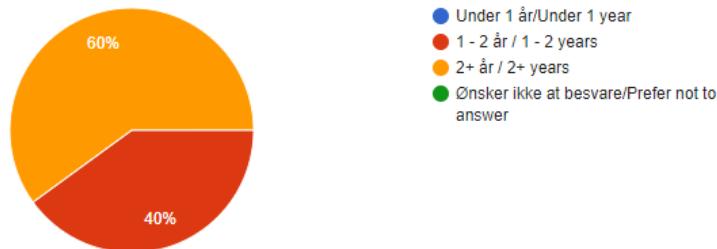


Spørgsmål 2

Hvor længe har du boet på Kvintus Kollegiet?/How long have you been living at Kvintus Kollegiet?

 Copy

10 responses

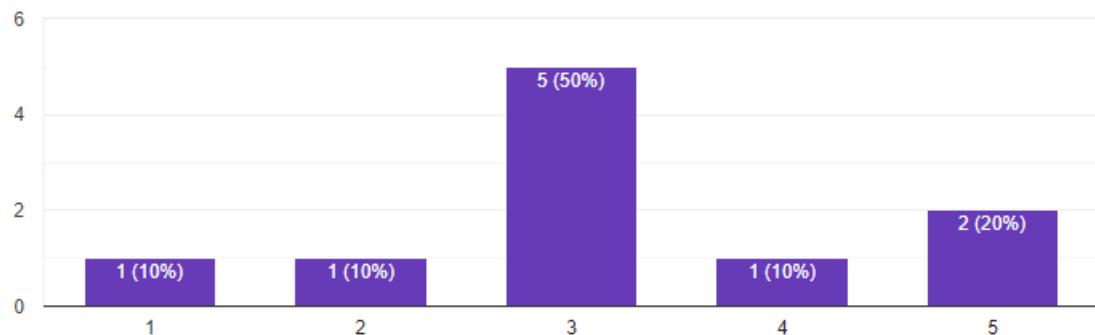


Spørgsmål 3

På en skala fra 1-5, i hvor høj grad føler du, at det er overskueligt at booke en tid i det nuværende system? / On a scale from 1-5, to what extend do you find it manageable to book a time slot in the current system?

 Copy

10 responses

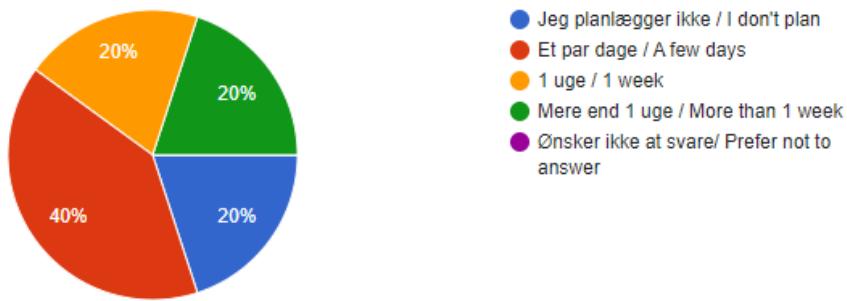


Spørgsmål 4

Hvor lang tid i forvejen planlægger du vasketid? / How far in advance do you plan to do laundry?

Copy

10 responses

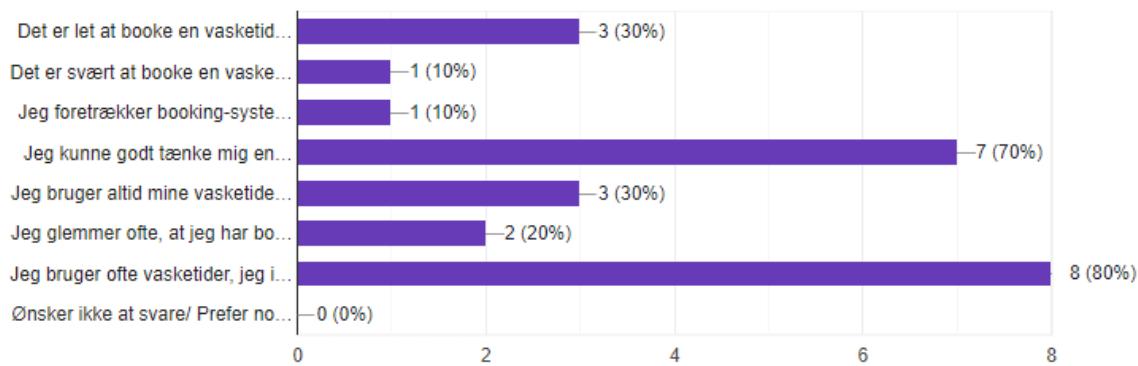


Spørgsmål 5

Hvilke af følgende udsagn er du enig i? / Which of the following statements do you agree on?

Copy

10 responses

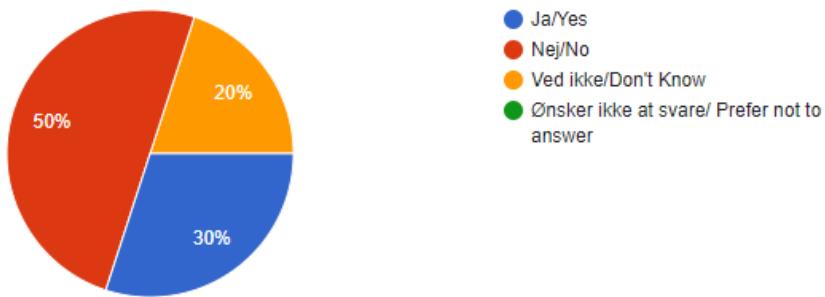


Spørgsmål 6

Jeg stoler på, at de andre beboere overholder deres vasketider, så de ikke går til spilde. / I trust the other residents use their time slots so they don't go to waste.

 Copy

10 responses

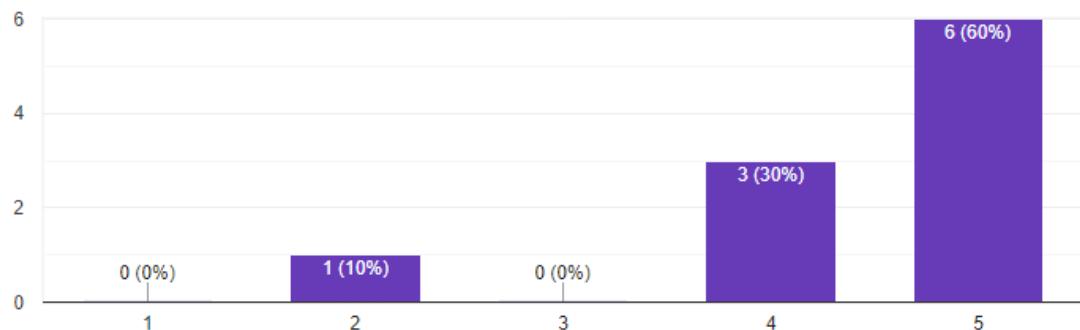


Spørgsmål 7

I hvor høj grad ville det forbedre din booking-oplevelse, hvis det blev muligt at tilgå systemet online? 1 er lavest, 5 er højest / In what degree would it make you booking-experience better, if it was possible to access the system online? 1 is lowest, 5 is highest

 Copy

10 responses

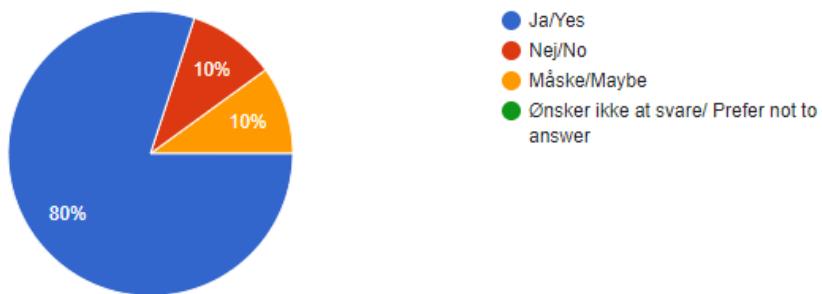


Spørgsmål 8

Ville du være interesseret i at benytte et lignende website til booking?/Would you be interested in using a similar website for booking?

 Copy

10 responses



Spørgsmål 9

Hvad kunne evt. forbedres? / What could be improved?

8 responses

Farver (rød/grøn), som angiver, om tiden er booket. Begrænsninger på, hvor mange gange man kan booke om dagen.

Ikke umiddelbart

Alt efter pris, så har det altid været en interesse på kollegiet. Mange steder har de online booking, men det har vi ikke kunne få af vores leverandør, da det er gratis at vaske her på kollegiet. Såfremt jeres system med login mv. Virker, så er jeg sikker på beboerrådet og Jimmi vil teste det af

Mega smart med online! Især til os der bor længere oppe:))

Jeg ønsker primært muligheden for at kunne aflyse bookede tider, når man opdager at man ikke kan nå hjem. Således andre kan booke dem

Det er lidt kantet og 90'er agtigt. Måske man kunne eminere nogle hjørner? :)

Færre felter, mere overskuelighed

Spørgsmål 10.1

Hvad kunne evt. forbedres? / What could be improved?

8 responses

Ikke umiddelbart

Alt efter pris, så har det altid været en interesse på kollegiet. Mange steder har de online booking, men det har vi ikke kunne få af vores leverandør, da det er gratis at vaske her på kollegiet. Såfremt jeres system med login mv. Virker, så er jeg sikker på beboerrådet og Jimmi vil teste det af

Mega smart med online! Især til os der bor længere oppe:))

Jeg ønsker primært muligheden for at kunne aflyse bookede tider, når man opdager at man ikke kan nå hjem. Således andre kan booke dem

Det er lidt kantet og 90'er agtigt. Måske man kunne eminere nogle hjørner? :)

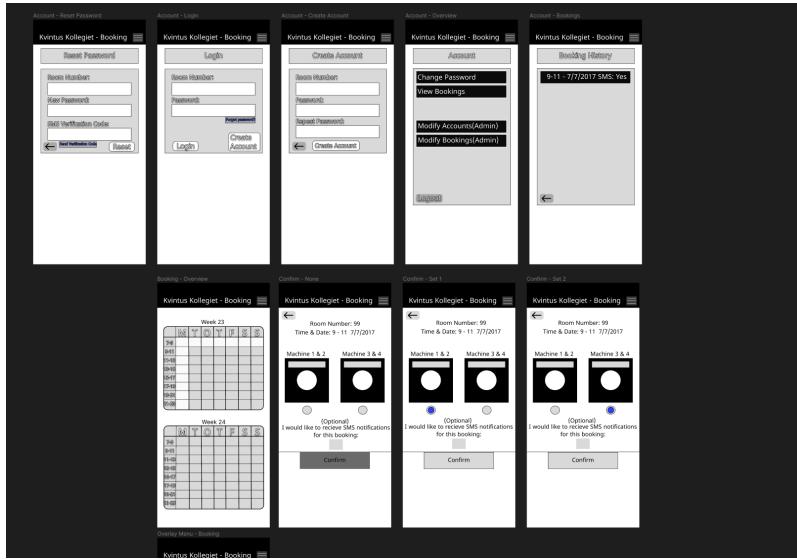
Færre felter, mere overskuelighed

Jeg ville håbe der var mulighed for at annullere sin booking. Ud fra tegningen virker det ikke helt overskueligt hvilke to vaskemaskiner man booker tid til

Spørgsmål 10.2

4. Figma

Se den her: <https://www.figma.com/file/4MIZewmqsv7DkCyixA147U/Booking-Mock-up?type=design&node-id=0%3A1&t=RIr1DF8uUQgjbSJ-1>



5. Server Log

Server Log:

Dato:	Bemærkning:	Tid brugt	udført af:
23.05.2023	<ul style="list-style-type: none"> - skabe overblik til krav i serverfag. (vejledning) - skabe overblik, samt lave diagram over infrastruktur. 	15 min 3t	DW
01.06.2023	<ul style="list-style-type: none"> - ny revision af diagram. - skrive i rapporten 	2t 3t	DW
02.06.2023	<ul style="list-style-type: none"> - lave en VM server - opdatere VM server - lave script 	1 t 30 min 2 t	DW
03.06.2023	<ul style="list-style-type: none"> - lave script - lave implementeringsplan - skrive i rapporten 	3 t 1 min 5 t	DW

Implementeringsplan

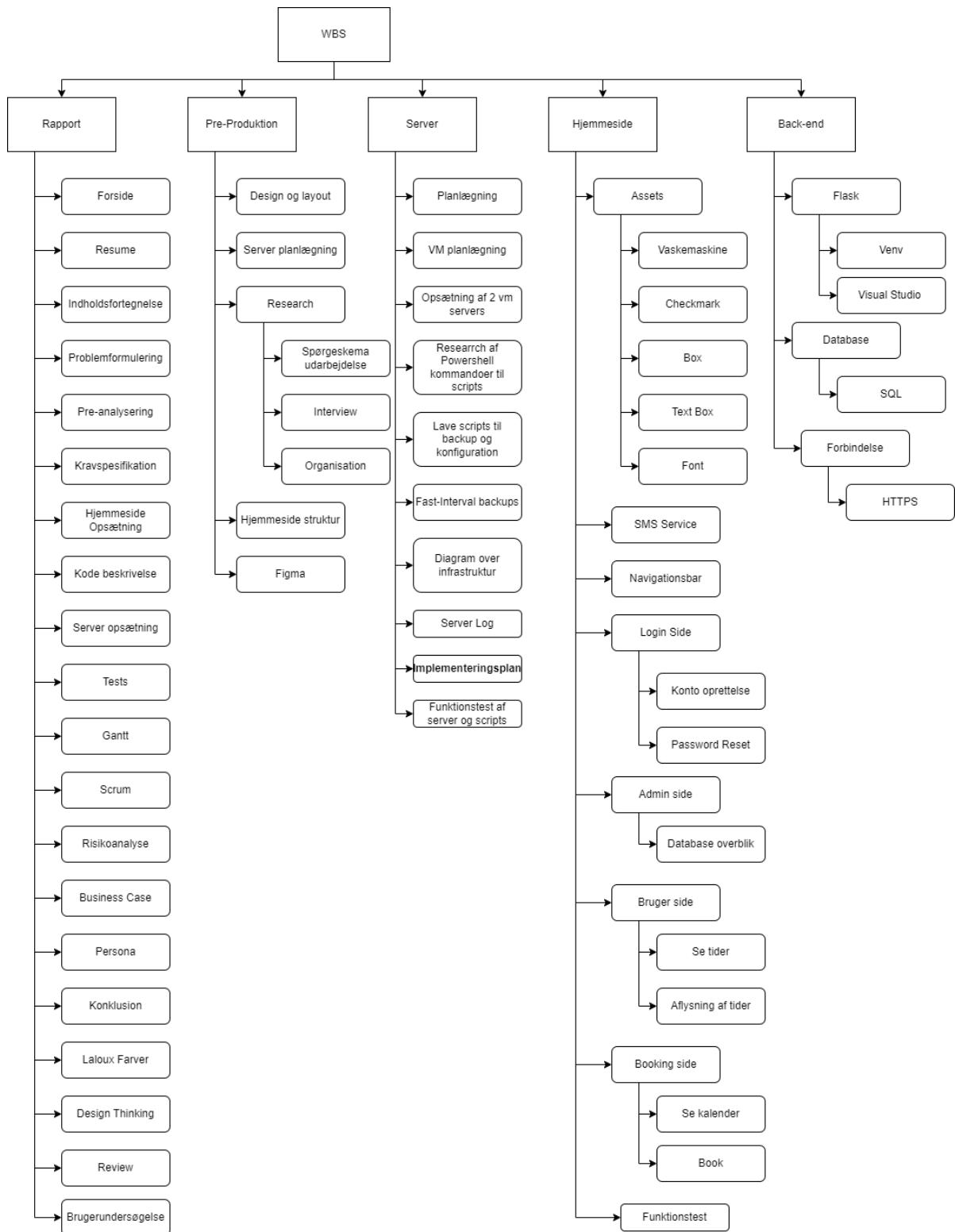
Langt script

Opgaver	Estimeret tid	Faktisk tid	Bemærkninger
Oprette VM	1 time	45 min	
starte konfig script:			
- installation af: · choco · python · sqlite · github · vs code · restart server	7 min 3 min 3 min 4 min 4 min 2 min 1 min 4 min - 28 min	10 min 2 min 2 min 3 min 3 min 2 min 30 sek 3 min - 35 min 30 sek	
opsætning af backup script	10 min	12 min	
Start APIen.	30 sek.	1 min	
	1 time 38 min 30 sek	1 time 33 min 30 sek	

Kort script

Opgaver	Estimeret tid	Faktisk tid	Bemærkninger
Oprette VM	1 time	45 min	
starte konfig script: - sæt statisk IP - hente kopi af APIen - 5 min	1 min 4 min - 5 min	30 sek 2 min - 2,5 min	
opsætning af backup script	10 min	12 min	
Start APIen.	30 sek.	1 min	
	1 time 15 min 30 sek	1 time 3 min	

6. WBS



7. Gantt

Kort ver.

Aktivitet	20	21	22
Rapport			
Pre-Produktion			
Front-end			
Back-end			
Server			
Test			

8. Work Hours

Uge 20							
	May 15	May 16	May 17	May 18	May 19	May 20	May 21
	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Adam	X	X	9 - 14	9 - 14	9 - 14	X	X
Dina	X	X	9 - 14	9 - 14	9 - 14	X	X
Oliver	X	X	9 - 14	9 - 14	9 - 14	11 - 16	X
Signe	X	X	9 - 14	9 - 14	9 - 14	All day	All day

Kristi himmelfart

Uge 21							
	May 22	May 23	May 24	May 25	May 26	May 27	May 28
	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Adam	9 - 14	9 - 14	9 - 14	9 - 14	9 - 14	X	X
Dina	9 - 14	9 - 14	9 - 14	9 - 14	9 - 14	X	X
Oliver	9 - 14	7 - 12	9 - 14	12-14	X	16 - 19	10 - 15
Signe	9 - 14	16-21	16-21	9 - 12	9 - 22	X	16-21

Uge 22							
	May 29	May 30	May 31	June 1	June 2	June 3	Rapport
	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Adam	9 - 14	9 - 14	9 - 14	9 - 14	9 - 14	10 - 16	10 - 16
Dina	9 - 14	9 - 14	9 - 14	9 - 14	9 - 14	10 - 16	10 - 16
Oliver	X	9 - 14	9 - 14	9 - 14	9 - 14	10 - 16	10 - 16
Signe	9 - 14	9 - 14	9 - 14	9 - 14	9 - 14	10 - 16	10 - 16

Anden Pinsedag

Working from home / Holiday	Sick-leave	Extra	Obstructed from completing a full work day

Total Hours

	Allocated
Adam	65
Dina	65
Oliver	65
Signe	65
Total	260