

ObliGentle: Produktrapport

Svendeprøve ved
Kenneth Løvgren
&
Rasmus Ladefoged Wolffram

Aarhus Tech

Sune Koch Rønnow
sune@kochroennow.dk

8. december 2023

Titelblad

Deltagere	Sune Koch Rønnow
Projektnavn	ObliGentle
Skole	Aarhus Tech
	Hasselager Allé 2, 8260 Viby J
Projektperiode	13. november 2023 - 15. december 2023
Afleveringsdato	8. december 2023
Vejleder	Kenneth Løvgren & Rasmus Ladefoged Wolffram

Underskrifter

Sune Koch Rønnow	Dato
------------------	------

Kenneth Løvgren	Dato
-----------------	------

Rasmus Ladefoged Wolffram	Dato
---------------------------	------

Indhold

1	Læsevejledning	3
2	Kravsspecifikation	4
2.0.1	Funktionalitet	4
2.0.2	Begrænsninger	4
2.0.3	Testkonditioner	5
3	Brugervejledning	6
3.1	Flowchart	6
3.2	Mockup	8
4	Teknisk dokumentation	13
4.1	Testede API routes	13
A	API routes	I
A.1	evaluering af fejlet GET /status	II

Kapitel 1

Læsevejledning

I forbindelse med svendepøven er vi blevet pålagt at aflevere 2 rapporter, en om projektets proces og en om dets produkt. Dette er produktrapporten og fokuserer på gennemgang af brugen af projektet.

Brugervejledningen er skrevet, således at den ikke kræver nogen særlig teknisk viden. Den tekniske dokumentation forudsætter en teknisk viden svarende til en færdiguddannet datateknikker.

Kravspecifikationen er skrevet med henblik på at være læselig for alle, men da den inkluderer fagudtryk, er den fulde forståelse er muligvis ikke tilgængelig for alle.

Kapitel 2

Kravsspecifikation

ObliGentle er en opgave-organisering-/kalenderapplikation, som har til formål at hjælpe brugeren med at organisere deres hverdag ved at give brugeren en positiv oplevelse med, hvad brugeren faktisk når og mindre skyldfølelse med hvad brugeren ikke når. ObliGentle forsøger at opnå dette ved behandle opgaver forskelligt og deler¹ op i 3 slags opgaver. Aftaler: Opgaver som skal være færdige til eller før et bestemt tidspunkt, Projekter: selvvalgte opgaver med fokus på fremskridt i stedet for færdiggørelse, 3) Tjanser: Gentagende opgaver.

2.0.1 Funktionalitet

ObliGentle er i sin essens 3 klasser/lister, som hver rummer opgaver. De forskellige klasser/lister har forskellige regler. Endvidere har de også følgende funktioner:

- uulighed for at tilføje nye opgaver
- mulighed for at vise udvalgte opgaver
- *crossplatform accessibility*
- mulighed for at let at kunne tilføje ekstra funktioner til programmet²
- mulighed for brugerne kan bruge *two factor authentication*

2.0.2 Begrænsninger

ObliGentle bliver ikke tilgængeligt på *IOS*-enheder.

Selvom ObliGentle vil være tilgængeligt online, vil de juridiske sider i forhold til GDPR og databeskyttelse ikke være en prioritet at færdiggør i en sådan grad, at det frit kan afbenyttes.

ObliGentle er ikke baseret på forskning, men på personlige præferencer.

Sidst men ikke mindst bliver ObliGentle kun en prototype og ikke alle funktioner vil være færdigudviklet.

ObliGentle designes til at kunne videreudvikles til at kunne overkomme alle de begrænsninger, men det kan ikke nås under svendeprøven.

¹I hvert fald i første omgang

²Altså programmet er skrevet modulært; skal ikke forstås som om brugere selv kan ændre programmet

2.0.3 Testkonditioner

- API'en skal kunne tilgås
- Skal virke crossplatform i både en *browser* og på en *tablet*
- Skal kunne gemme profiler
- skal kunne klare ændringer fra forskellige profiler
- Skal kunne oprette nye brugere
- Skal kunne live opdatere data
- Skal kunne håndtere, at brugere forsøger at ændre det samme objekt fra forskellige enheder på samme tid
- Kan ikke logge ind med den forkerte *two factor authentication* kode
- Virker på forskellige enheder på forskellige netværk

Kapitel 3

Brugervejledning

I skrivende stund kører ikke, som den skal. Dette er noget, som jeg regner med at få udbedret inden svendepøven, men lige nu virker, så virker login og registreringssiden.

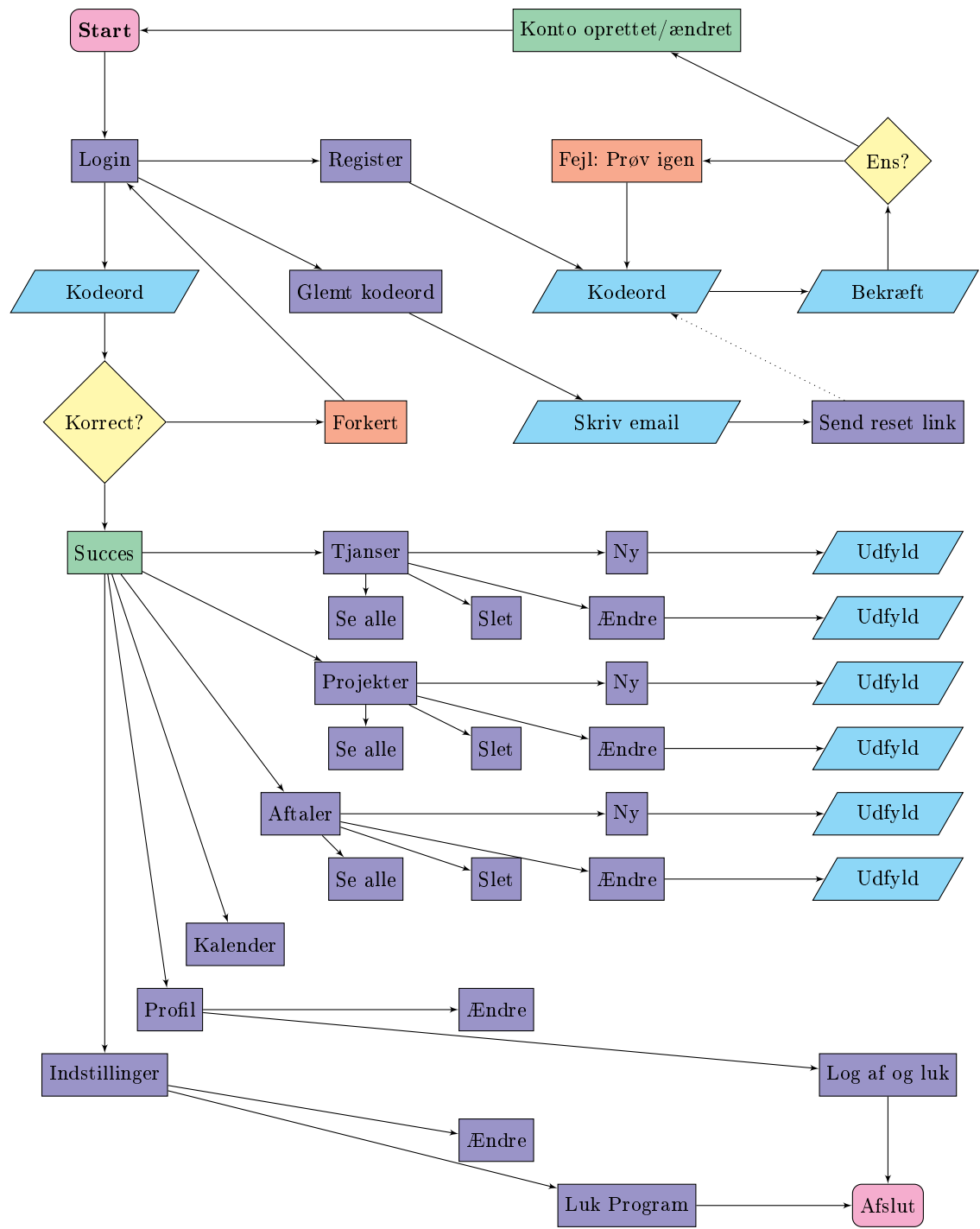
Det er desværre ikke meningsfuld at beskrive, hvordan ikke fungerede del af et program ikke virker.

Derfor har jeg inkluderet flowchart'et og mock'et fra godkendelsesproceduren, således det er muligt at se hensigten. Håber på at få programmet til at virke og lave noget dokumentation til den ved samme lejlighed.

3.1 Flowchart

Flowchart'et bærer er væsentligt færdigudviklet for login delen, som også er mest ekspansiv. For de 3 opgave menuer har jeg fokuseret på processer og funktioner fremfor de enkelte knapper, da de sidder endnu ikke færdigudviklet og jeg ikke vil sætte mig for fast på det helt specifikke forløb¹. De 3 resterende menuer, for "*Kalender*", "*Profil*" og "*Indstillinger*", er mindre beskrevet, fordi jeg endnu mindre grad ønsker at ligge mig fast.

¹Se dog mock-up for en forklaring af, hvad de menuer kunne se ud og indeholde af specifikke knapper: section 3.2



3.2 Mockup

Jeg har valgt at inkludere 6 skærme i denne mock-up. Kun 3 er en del af kernen opgaven og derfor har jeg fokuseret på dem. Ville har dog inkluderet de 3 ekstra faner, Kalender-, Profil- og Indstillingsfanen, fordi jeg på den ene side håber at kunne nå dem og for at vise, at der tænkt yderligere menuer ind. Jeg vil dog koncentrere mig tid på de 3 vigtigste faner.

Billedet af en tablet er fundet her². Farverne har jeg valgt en tertiære farve, som jeg kunne lide og brugte *colorhex.com* til at finde dens triadiske farveskema³. Jeg valgte et triadisk farveskema, fordi det både er farverigt og udtryksfuld, men stadig harmoniserende⁴. Det bliver dog ikke endelige farvevalg, da dette specifikke triadiske farveskema ikke er så harmoniserende.⁵

Jeg har ikke lavet et flowchart, men i stedet for at forsøgt at forklare det her.



Her viser programmet de top 5 vigtigste Tjanser. Tjanser er tilbagevendende og har beskrivelse, en nuværende eller kommende tilstand, et interval og en prioritet.

- **Lav ny** lader dig oprette en tjans.
- **Se alle** viser alle nuværende tjanser og ikke kun de 5 højest prioriterede.
- **Kommende** viser alle kommende tjanser
- Tjanserne er de vigtigste interaktioner og der kan man trykke flere forskellige steder på knappen:

²<https://purepng.com/photo/25263/electronics-android-tablet>

³<https://www.colorhexa.com/ff99ff>

⁴<https://multimediedesigneren.dk/farveteori-og-farvehjul/>

⁵Bortset fra centsors version af programmet, som ikke har mulighed for at skifte farveskemaet

- **I midten** viser beskrivelse af tjansen og mulighed for at kunne ændre tjansens indstillinger.
- **Venstre side:** sætter tjansen til kommende for dens intervallet og øger dens prioritet.
- **Højre side:** sætter tjansen til kommende for dens intervallet og dens prioritet til, hvad den var da den oprettes.
- Når knapperne bliver smalle nok, forsvinder først den venstre knap og derefter den højre. I de ville kunne tilgås ved at trykke **i midten**.



Her vises de 5 aktive projekter med den seneste aktivitet. Projekter har blot en beskrivelse, aktiv eller passiv tilstand; og en fremskridtstræller.

- **Ny:** Opretter en nyt projekt.
- **Aktive** viser alle aktive projekter og ikke kun de 5 med seneste aktivitet.
- **Passive** viser alle passive projekter.
- Projekter er de vigtigste interaktioner og der kan man trykke flere forskellige steder på knappen:
 - **I midten** viser beskrivelse af projektet, dets fremskridt, mulighed for at ændre dens indstillinger eller nulstille dens fremskridt.
 - **Venstre side:** sætter projektet til passiv.
 - **Højre side:** sætter projektets en højere.
 - Når knapperne bliver smalle nok, forsvinder først den venstre knap og derefter den højre. I de ville kunne tilgås ved at trykke **i midten**.



Her vises de 5 næste relevante alle. Aftaler har en beskrivelse, relevans-status, tidspunkt og indstillinger for påmindelser.

- **Ny:** Opretter en ny aftale..
- **Se alle** viser alle kommende aftaler og ikke kun de 5 næste.
- **Tidligere** viser alle tidligere aftaler samt kommende ikke-relevante aftaler.
- Aftaler er de vigtigste interaktioner og der kan man trykke flere forskellige steder på knappen:
 - **I midten** viser beskrivelse af aftalen og får mulighed for at ændre dens indstillinger-
 - **Venstre side:** mulighed for at ændre dens tidspunkt eller påmindelse.
 - **Højre side:** sætte aftalen til ikke-relevante.
 - Når knapperne bliver smalle nok, forsvinder først den venstre knap og derefter den højre. I de ville kunne tilgås ved at trykke i **midten**.



Viser en, endnu ikke desginet, kalender, hvor man kan se ens aktivitet. De forskellige opgaver vises lidt forskellige. Der ville også skulle være en række styringsknapper.

Aftales tidspunkter vises samt hvornår aftaler ikke længere relevante, om det så skyldes om man færdiggjorde dem før tid eller de blot oprendte, gøres der ikke forskel på.

For projekter vises der, hver gang man har trykket fremskridt for et projekt.

For tjanser vises der hver gang en sættes til kommende, uanset om den har ændret prioritet.



Her vises, en endnu udesignet, profilfane, hvor det vil være muligt at få et overblik over ens profil og ændre den.



Her vises, en endnu udesignet, indstillingsfane, hvor det vil være muligt at styre en række ting fra. Eksempelvis antallet af tjanser, aftaler og projekter eller lignende ting.

Kapitel 4

Teknisk dokumentation

4.1 Få programmeret til at køre

Indtil videre er det nødvendigt bruge `uvicorn main:app -reload` fra `./API/-`mappen for at få backenden til at køre og bruge `npm run web` eller `npm run webnb`¹ fra `./UI/-`mappen for at få fronten til at køre.

Backenden virker som den skal, men frontenden fejler desværre på en centralskærm.

Jeg håber på at udbedre dette og lave en ny vejledning

4.2 Testkonditioner

Det er muligt at logge ind, lave en bruger og anmode om at få en reset password token fra fronten i skrivende tidspunkt.

Det er ikke imponerende, men det betyder, at de fleste frontend test ikke giver mening. Jeg har dog testet alle end points.

4.3 Testede API routes

Disse routes er testet fra localhost:

¹hvis man ikke ønsker applikationen selv skal åbne sig i en ny browser

[HTTP request method] / [route]	funktion	Test 8/12
GET /	Root	success!
GET /authenticated-route	Authenticated Route	success!
GET /status	Getstatus	Fejlet!
POST /auth/jwt/login	Auth:Jwt.Login	success!
POST /auth/jwt/logout	Auth:Jwt.Logout	success!
POST /auth/register	Register:Register	success!
POST /auth/forgot-password	Reset:Forgot Password	success!
POST /auth/reset-password	Reset:Reset Password	success!
POST /auth/request-verify-token	Verify:Request-Token	success!
POST /auth/verify	Verify:Verify	success!
GET /users/me	Users:Current User	success!
PATCH /users/me	Users:Patch Current User	success!
GET /users/id	Users:User	success!
DELETE /users/id	Users:Delete User	success!
PATCH /users/id	Users:Patch User	success!
GET /appointments/	Routetosomething	success!
POST /appointments/	Routetocreation	success!
GET /appointments/id	Routetoonothing	success!
PUT /appointments/id	Routetoupdate	success!
DELETE /appointments/id	Routetooblivion	success!

[HTTP request metoden] [/route]	funktion	Test 8/12
GET /chores/	Routetosomething	succes!
POST /chores/	Routetocreation	succes!
GET /chores/id	Routetoonething	succes!
PUT /chores/id	Routetoupdate	succes!
DELETE /chores/id	Routetooblivion	succes!
GET /projects/	Routetosomething	succes!
POST /projects/	Routetocreation	succes!
GET /projects/id	Routetoonething	succes!
PUT /projects/id	Routetoupdate	succes!
DELETE /projects/id	Routetooblivion	succes!
GET /tasks/	Routetoeverything	succes!
POST /tasks/	Routetocreation	succes!
GET /tasks/id	Routetoonething	succes!
PUT /tasks/id	Routetoupdate	succes!
DELETE /tasks/id	Routetooblivion	succes!

4.3.1 evaluering af fejlet GET /status

Route fejlede, da den ved en fejl var sat til at kræve autorisering. Det er ikke meningen, så derfor tæller det selvfølgelig som en fejl, selvom det er rettet nu.