

# Kravspesifikasjon

## Parkeringsplass reservasjons-app

Gruppe 12

# Innholdsfortegnelse

<b>1</b>	<b>Problemstilling</b>	<b>4</b>
<b>2</b>	<b>Løsningen</b>	<b>4</b>
<b>3</b>	<b>Begrensninger og antagelser</b>	<b>4</b>
<b>4</b>	<b>Ansvarsområde og Avhengigheter</b>	<b>4</b>
<b>5</b>	<b>Funksjoner i applikasjonen</b>	<b>4</b>
<b>6</b>	<b>Begreper og definisjoner</b>	<b>5</b>
<b>7</b>	<b>Brukerklasser</b>	<b>5</b>
7.1	Bruker . . . . .	5
7.2	Bedrift . . . . .	6
7.3	Administrator . . . . .	6
<b>8</b>	<b>Personaer</b>	<b>6</b>
8.1	Jørgen Moe . . . . .	6
8.2	Siggurd Olson . . . . .	7
8.3	Karina Gausrud . . . . .	7
8.4	Karin . . . . .	8
8.5	Jørn Berg . . . . .	9
<b>9</b>	<b>Funksjonelle krav</b>	<b>9</b>
9.1	Kontooprettelse . . . . .	9
9.2	Innlogging . . . . .	10
9.3	Verifiseringskode . . . . .	10
9.4	Presentasjon av parkeringsplasser (Feed) . . . . .	11
9.5	Leie parkeringsplass . . . . .	11
9.6	Tilbakemelding på innlegg (Anmeldelser) . . . . .	11
9.7	Opprette parkeringsplasser . . . . .	12
9.8	Leiebetingelser . . . . .	12
9.9	Betaling . . . . .	13
9.10	Administrering . . . . .	13
9.11	Persistent data . . . . .	14
<b>10</b>	<b>Ikke funksjonelle krav</b>	<b>14</b>
10.1	Tilgjengelighet . . . . .	14
10.2	Sikkerhetskopi(Backup) . . . . .	15
10.3	Dependency på tredjepart . . . . .	15
10.4	Legal and licensing issues . . . . .	15
10.5	Nettverkstopologi . . . . .	15
10.6	Preformance / response time . . . . .	15
10.7	Personvern . . . . .	15
10.8	Kvalitet . . . . .	15
10.9	Sikkerhet . . . . .	15
10.10	Supportability . . . . .	15
10.11	Testability . . . . .	15
10.12	Interface . . . . .	15
<b>11</b>	<b>Prototype</b>	<b>15</b>

11.1	Sammendrag . . . . .	15
11.2	Innstallasjon . . . . .	15
11.3	Bruksmanual . . . . .	16
11.4	Implementerte krav og tester . . . . .	16
11.5	Avhengigheter . . . . .	16
11.6	Begrensninger . . . . .	16
<b>12</b>	<b>Diagrammer</b>	<b>18</b>
12.1	Sekvensdiagram - Leie en parkeringplass. . . . .	18
12.2	Dataflytdiagram - Tilbakemelding . . . . .	19
12.3	Sekvensdiagram - Opprette reservasjon . . . . .	19
12.4	Tilstandsdiagram - Registrere parkeringsplass . . . . .	20
12.5	Aktivitetsdiagram - Leie en parkeringsplass . . . . .	21
12.6	Sekvensdiagram - Leie ut en parkeringsplass . . . . .	22
<b>13</b>	<b>Estimering</b>	<b>22</b>
13.1	Enheter . . . . .	22
13.2	Estimeringstabell - Funksjonelle krav . . . . .	23
13.3	Estimeringstabell - Ikke funksjonelle krav . . . . .	23

# 1 Problemstilling

Ofte finner man ikke parkeringsplass mens man er i farta. Det er tett på senterene, gateparkering er privat eller ulovlig og man har ofte dårlig tid. Gjerne etter litt leting finner man en parkering 500 meter unna, og når man kommer dit er det selvfølgelig fullt. Hvordan kan man unngå å miste så mye tid, og i tillegg finne en parkeringsplass?

## 2 Løsningen

Tjenesten baserer seg på delingsøkonomi og enheten som deles er parkeringsplasser. Ideen er at bedrifter og enkeltpersoner kan registrere en eller flere parkeringsplasser de eier for utleie til enkeltpersoner. Dermed kan disse utleierene tjene penger passivt, samt gjøre det enklere for privatpersoner å forsørge parkering. Parkeringsplassen de legger ut vil være merket med bredde, lengde og høyde, samt om den er reservert for el-biler eller handicappede. På denne måten kan alle som bruker tjenesten få et godt innblikk i parkerings-bildet.

Bruker logger inn som en bedrift eller en vanlig bruker. Brukere kan redigere og legge ut sine registrerte parkeringsplasser, samt leie parkeringsplasser fra andre brukere. Bedrifter kan kun legge ut parkeringsplasser, men har muligheten til å legge til brukere som medlemmer av bedriften. Dette gir de spesialtillatelse til parkeringsplasser som er reservert kun for medlemmer av en spesifikk bedrift. Medlemmer av bedriften vil også kunne legge ut og merke parkeringsplasser for andre bedrifter.

Tjenesten er bygget opp slik at alle brukere har tilgang til en feed som viser en oppstilling av de mest relevante parkeringsplassene, en side for registrering av parkeringsplasser og en profilside. For å kunne reservere en parkeringsplass, må bruker oppgi informasjon om bilen de benytter. På feeden rangeres de tilgjengelige parkeringsplassene etter avstand til bruker eller et spesifikt område som bruker spesifiserer, og vises kun dersom de er passende med bilen. Bruker kan også trykke seg inn på et spesifikt innlegg for å se mer detaljer om parkeringsplassen som leies ut, og kan legge en anmeldelse på parkeringsplasser de har brukt. På denne måten får utleiere tilbakemelding på hva de gjør riktig og hva de gjør galt, som bidrar til å øke engasjementet til å gjøre en god jobb.

Kunde av tjenesten vil kunne opprette administratorbrukere som har spesielle egenskaper for administrering av feeden og brukerne. Dette vil være nødvendig for å minke upassende/ulovelig innhold og øke brukeropplevelsen.

## 3 Begrensninger og antagelser

## 4 Ansvarsområde og Avhengigheter

For at man skal kunne kjøre applikasjonen og testene er det noen ting man trenger:

JavaJDK v11 fasterxml.jackson 2.9.9 maven 3.8.0 webjars vue 2.6.10 javalin 3.11.0 JUnit 5.7.0

## 5 Funksjoner i applikasjonen

Programmet henter inn info om brukere, poster, parkeringsplasser og reservasjoner fra en database(i dette tilfellet json-filer). Det programmet gjør nå er å hente inn data fra controller til javalin som generer front-end for oss. Hver side henter da ut nødvendig info hva hver .json fil slik at nødvendig informasjon som adresse og ledighet til leie vises.

Bruker - Det er mulig å vise oversikt over parkeringsplasser, trykke seg inn til en parkeringsplass, og å legge til en reservasjon, brukeren kan også se oversikt over p-plassene som den har reservert. Det er

mulig å legge ut en ny parkeringsplass som en utleier. Det er mulig å vise bilder av parkeringsplassen via et link system, da opplastning direkte til siden ikke er lagt inn.

Corporation bruker - kan kun legge ut parkeringsplasser. Dette gjør man ved å gå til “My parking spots” hvor “Create new parkingspot” står øverst. For å lage et nytt parkingeringsobjekt må man trykke på denne knappen og fylle ut all nødvendig info(feilmelding dersom ikke alt har noen verdi), og man kan i tillegg velge ved bruk av ikoner dersom det er handikappet eller el-bil plass. Corporation skal kunne få betalt for å leie ut p-plasser.

Admin - Denne brukeren kan suspendere eller slette/banlyse brukere, slette poster og parkeringsplasser. Dette er en bruker-type som i senere iterasjoner kan gi rettigheter til andre brukere (f.eks. modereringsbrukere).

## 6 Begreper og definisjoner

Begrep	Beskrivelse
Applikasjon	Et programvare som benytter datamaskinens ressurser til en oppgave som brukeren ønsker utført
Bruker	Besøkende som samhandler med systemet
Brukerkonto	En som kan kun leie parkeringsplasser
Bedriftskonto	En som kan kun leie ut parkeringsplasser til andre
Administrator	Brukerkonto med administratorrettigheter
Feed	Opplisting av innlegg
Tjenesten	Tjenesten denne dokumentasjonen omhandler
Systemet	Det underliggende systemet til tjenesten
Personas	Er et annet ord for rollefigur, enkeltmenneske, karakterpersonlighet
ID	Identifikator
FK	Funksjonelt Krav (brukes som ID)
IFK	Ikke-Funksjonelt Krav (brukes som ID)
Innlegg	Brukerinnlegg i systemet (post) som består av en eller flere parkeringsplasser
Produktet/ Tjenesten	Applikasjonen som dokumenteres
Kunde	Gruppe som eier produktet
Push-notifikasjoner	Varsler på desktop eller mobil
Cookies	Informasjonskapsler
TOS	Terms of Service, liste over bruksrettigheter og bruksområde for brukerdatabaser

## 7 Brukerklasser

En oversikt over roller og egenskaper for de forskjellige brukerklassene.

### 7.1 Bruker

Brukerkontoen er en standardkonto som hvem som helst kan opprette. Som en bruker, kan man registrere sine egne parkeringsplasser og leie andre brukeres parkeringsplasser i perioder. Alle brukere har sin egen profilside der de kan redigere generell informasjon og innstillinger. Brukere kan gi anmeldesler/tilbakemelding på hverandre sine innlegg, og vil bli varslet dersom en annen bruker skriver en anmeldelse på sin parkeringsplass. Ved registrering av parkeringsplass, bestemmer man selv pris og tiden den er tilgjengelig, og den vil automatisk bli vist for andre brukere via feeden eller profil. Man må først registrere bilen man parkerer med før man kan reservere en parkeringsplass.

## 7.2 Bedrift

Det er kun organisasjoner som har tilgang til å opprette en bedriftskonto, da de må ha et organisasjonsnummer. Bedriftskontoen er sammensatt av flere brukerkontoer. En brukerkonto som er medlem av en bedrift har spesialtilgang til parkeringsplasser markert for generelle eller spesifikke bedrifter. Bedriftskontoen kan legge ut parkeringsplasser, men kan ikke leie en parkeringsplass. Det er kun bedrifter som kan markere parkeringsplasser for andre bedrifter.

## 7.3 Administrator

En administrator ligner en vanligbrukerkonto, men har noen ekstra rettigheter for administrering. De har full makt til å slette brukere, bedrifter, innlegg, parkeringsplasser, og anmeldelser, samt suspendere og oppheve brukerkontoer. Administrator er essensiell for å vedlikeholde en god brukeropplevelse og rettferdiggjøre systemet for alle brukere.

# 8 Personaer

## 8.1 Jørgen Moe



**Alder:** 34

**Jobb/Rolle:** Økonom

**Beskrivelse:**

Jobber som økonom. Han skal grave i gården og lage ny parkeringsplass. Han vet ikke hvor lang tid det kommer til å ta, og han kommer ikke til å ha et sted å parkere i mellomtiden.

**Motivasjon:**

Han har ikke parkeringsplass hjemme, så han trenger en parkeringsplass i nærheten.

**Behov:**

Han må kunne forlenge antall dager han låner en parkeringsplass

**Tekniske ferdigheter:** Gode

## 8.2 Siggurd Olson



**Alder:** 42

**Jobb/Rolle:** Ledelse i bedrift

**Beskrivelse:**

Jobber i ledelsen i en bedrift. Det er ikke alltid han får en plass til å parkere i byen når han skal på jobb.

**Motivasjon:**

Parkeringsplassene i byen er fullstappa.

**Behov:**

Han må kunne parkere nærmere arbeidsplassen og sikre seg parkeringplass ved et møte.

**Tekniske ferdigheter:** Gode

## 8.3 Karina Gausrud



**Alder:** 22

**Jobb/Rolle:** Uføre

**Beskrivelse:**

Karina er en rullestolbruker som trenger handicap-parkering. Hun vil helst parkere så nærme målet sitt som mulig, men vet ikke om det finnes en ledig handicap-parkeringsplass. Hun vil heller ikke risikere at stedet ikke tilbyr parkeringsplass for handicappede.

**Motivasjon:**

Hun trenger en oversikt over ledige handicap-parkeringsplasser.

**Behov:**

Hun må være sikker på at hun får en handicap-parkeringsplass

**Tekniske ferdigheter:** Gode

## 8.4 Karin



**Alder:** 34

**Jobb/Rolle:** Selger

**Beskrivelse:**

Karin er dør-til-dør selger, hun kjører til nabolagene hun jobber i hver dag, men det er vanskelig å finne parkering, som gjør at hun taper tid og penger mens hun leter. Derfor åpner hun appen og booker parkeringsplasser for områdene hun skal jobbe den neste måneden. Nå føler Karin at hun har en mindre ting å tenke på.

**Motivasjon:**

Han har ikke parkeringsplass hjemme, så han trenger en parkeringsplass i nærheten

**Mål:**

Karin ønsker å eliminere en del av hverdagen som er unødvendig og som koster henne tid og penger. Karin forventer at hun skal finne parkering, og kunne booke det for måneden fremover, eller på dagen om nødvendig.

**Reisen:**

For å oppnå dette er Karin nødt til å finne utleiere som ønsker kort-tid/dagsutleie, så hun er nødt til å søke området hun skal til, og filtrere til dette alternativet. Deretter kan hun velge sted å parkere basert på beliggenhet og pris i forhold til området hun utvalgte seg.

**Følelser:**

Karin føler seg frustrert når hun kjører rundt nabolaget hun skal banke dører i, men aldri klarer å finne et sted det er lov å parkere. Karin er på sitt lykkeligste når hun åpner appen for å sjekke hvor hun har bestilt parkering, og vet at hun ikke trenger å tenke på å lete igje

**Tekniske ferdigheter:** Gode



## 8.5 Jørn Berg



**Alder:** 56

**Jobb/Rolle:** Bonde

**Beskrivelse:**

Jørn bor i nærheten av et vakkert tjern som er populært blant unge å bade i, men siden det ikke er noen parkeringsplass der må folk parkere langs veien. Jørn eier en fin tomt i nærheten der folk kan parkere tryggere.

**Motivasjon:**

Han vil at tilby parkeringsplasser slik at ungdommene mer sannsynligvis kommer tilbake til tjernet og bader.

**Behov:**

En omtale system som kan bidra å spre det gode rykte videre. Han vil leie ut tomt for en inntekt ved siden av gårdsdriften.

**Tekniske ferdigheter:** Dårlig

## 9 Funksjonelle krav

### 9.1 Kontoopprettelse

**Bruker skal kunne opprette en brukerkonto**

**ID: FK1**

For å opprette en brukerkonto, må bruker oppgi brukernavn, passord, e-post og telefonnummer.

**ID: FK1.1** Brukernavn og passord må være unike.

**Bedrifter skal kunne opprette en bedriftskonto**

**ID: FK2**

For å opprette en bedriftskonto, må bedriften må oppgi brukernavn, passord, e-post, bedriftsnavn og organisasjonsnummer.

**ID: FK2.1** Brukernavn, passord, bedriftsnavn og organisasjonsnummer må være unike.

**Kunden skal kunne opprette brukerkonto med administratorrettigheter**

**ID: FK3**

Bare kunde av applikasjonen kan opprette denne brukerkontoen. For å opprette denne kontoen må

kunden oppgi et brukernavn, passord og telefonnummer.

**ID: FK3.1** Brukernavn, passord og telefonnummer må være unike.

### **Systemet skal verifisere registreringen**

**ID: FK4**

Systemet må verifisere registreringen via verifiseringskode som sjekkes. Denne må bli sendt til brukeren slik at de kan skrive inn koden til programmet. Hvis koden stemmer med den som er lagret i databasen, er registreringen verifisert. Hvis ikke skal de få en feilmelding på samme siden. Etter verifisering lagres brukerdataen i databasen.

### **Derigere opprettet bruker**

**ID: FK5**

Etter opprettelse av konto skal bruker automatisk bli ført til startsidene.

## **9.2 Innlogging**

### **Bruker skal kunne logge inn som en brukerkonto eller bedriftskonto**

**ID: FK6**

Bruker skal kunne velge å logge inn som en brukerkonto eller bedriftskonto, og så fylle ut tekst-felter med de samme opplysningene som da de opprettet kontoen sin.

### **Systemet skal verifisere innloggingen**

**ID: FK7**

Dersom innloggingsopplysningene er korrekte i forhold til opplysningene i brukeropprettingen, skal brukeren bli logget inn. Dersom de er ukorrekte, skal brukeren få en feilmelding på hva som gikk galt.

### **Derigere innlogget bruker**

**ID: FK8**

Etter vellykket innlogging skal bruker bli ført til startsidene.

### **Bruker skal kunne tilbakesette passord**

Brukere har valg om å tilbakesette passord hvis de har glemt dette. I så fall skal bruker bli sendt verifiseringskode som skrives inn i et nytt felt for å verifisere at det er samme bruker. Hvis koden stemmer kan bruker skrive inn nytt passord som skal brukes fremover.

## **9.3 Verifiseringskode**

### **Systemet skal verifisere bruker gjennom en kode**

Denne koden blir brukt til blant annet Kontoopprettelse, passordbytte og to-faktor autentisering. Hensikten er å gjenkjenne brukeren som gjør sikkerhetskritiske handlinger på sin profil. Når bruker gjør en av disse handlingene blir koden midlertidig lagret i databasen, og videre sendt til bruker. Bruker må skrive koden inn i et tekst-felt, der den blir sjekket opp med den opprinnlige koden lagret i databasen. Hvis den stemmer er brukeren verifisert.

**ID: FK8.1** Koden skal være fire-sifret

**ID: FK8.2** Koden skal bli sendt via sms

## 9.4 Presentasjon av parkeringsplasser (Feed)

### Feeden skal oppliste relevante parkeringsplasser

#### ID: FK9

Alle brukertyper skal ha tilgang til en opplistingen som består av innlegg fra andre brukere. Det er her brukere ettersøker og reserverer parkeringsplasser. Standaropplistingen rangeres etter nærmest avstand fra posisjonen til brukeren.

**ID: FK9.1** Posisjonen hentes via cookies.

### Bruker skal kunne sette et filter på feeden

#### ID: FK10

Opplistingen rangeres via et brukervalgt filter. Filteret bestemmer om opplistingen rangeres basert på nærmest avstand, lavest til høyest pris, høyest til lavest pris, parkeringsplasstype og om den leies ut av en brukerkonto eller bedriftskonto.

### Bruker skal kunne søke etter en lokasjon

#### ID: FK11

Feeden skal rangere parkeringsplassene basert på avstand i forhold til denne lokasjonen.

**ID: FK11.1** Søkefeltet skal akseptere adresse.

**ID: FK11.2** Søkefeltet skal akseptere koordinater.

## 9.5 Leie parkeringsplass

### Bruker skal kunne leie parkeringsplass som en brukerkonto

#### ID: FK12

Som en brukerkonto skal bruker kunne leie en eller flere parkeringsplasser. Dette gjøres ved at systemet oppretter en reservasjon for der parkeringsplassen knyttes opp til brukeren som leier den.

### Varsel på utløpstid

#### ID: FK13

Når en reservert parkeringsplass holder på å gå tom for tid, så skal brukeren som har leid parkeringsplassen få en varsel om at den holder på å gå ut minst 30 minutter før.

**ID: FK13.1** Varselet skal gå gjennom SMS og push-notifikasjoner

## 9.6 Tilbakemelding på innlegg (Anmeldelser)

### Bruker skal kunne gi tilbakemelding på et innlegg

#### ID: FK14

Dersom brukeren har brukt en parkeringsplass, skal de kunne gi tilbakemelding på innlegget til den parkeringsplassen. Innlegget må minst bestå av en poengsum. Om ønskelig kan bruker også legge til en kommentar. Tilbakemeldingen skal lagres i databasen og tilknyttes innlegget.

**ID: FK14.1** Hver bruker kan bare skrive en anmeldelse per innlegg.

**ID: FK14.2** Kommentaren skal være på maks 130 tegn.

**ID: FK14.3** Poengsummen skal være fra 1 til 5, der 5 er den høyeste poengsummen.

## **Tilbakemeldinger skal vises på hvert innlegg**

### **ID: FK15**

Alle anmeldelser som tilhører et innlegg skal vises på listeform på innlegget, der den nyeste tilbakemelding ligger øverst.

**ID: FK15.1** Hver tilbakemelding skal vise poengsummen, kommentaren og brukernavn til de som lagde den.

## **Innlegg skal vise en endelig poengsum**

Poengsummen består av gjennomsnittet av poengsummer fra alle tilbakemeldinger knyttet til innlegget.

## **Utleier skal kunne få tilbakemeldingene via e-post**

Ved valg har utleier mulighet til å få tilsendt alle tilbakemeldinger på e-post.

## **9.7 Opprette parkeringsplasser**

### **Brukerkonto og bedriftskonto skal kunne registrere en eller flere parkeringsplasser**

#### **ID: FK16**

For å registrere en parkeringsplass må følgende oppgis:

- Bredde, lengde, høyde i centimeter
- Antall parkeringsplasser
- Type parkeringsplass (om det er handikapparkering, elektriskparkering, osv.)
- Etasje
- Pris (dette skal kunne endres manuelt eller automatisk i etterkant)
- Postadresse
- Gateadresse
- Gatenummer

Etter opprettelsen lagres parkeringsplassen i databasen.

**ID: FK16.1** Bedriftskonto skal kunne markere en parkeringsplass for en spesifikk bedrift eller for bedrifter generelt

**ID: FK16.2** Parkeringsplassen skal knyttes til en unik ID

**ID: FK16.3** Parkeringsplassen skal knyttes til brukeren som opprettet den.

## **9.8 Leiebetingelser**

### **Pris per time skal bestemmes av bruker**

#### **ID: FK17**

Dette er prisen som parkeringsplassen leies ut for (per timer) og bestemmes av bruker som registrerte parkeringsplassen. Den bestemmes på hver individuelle parkeringsplass.

**ID: FK17.1** Prisen skal kunne oppdateres etter opprettelsen.

**Tiden skal bestemmes av bruker**

**ID: FK18**

Dette er fra-tid og til-tid som parkeringsplassen kan leies av andre brukere.

**Brukere skal kunne se ut ifra innlegget om parkeringsplassen er ledig eller ikke**

Ledig-statusen regnes ut dersom den tilgjengelige tiden samsvarer med nåværende klokkeslett.

**ID: FK18.1** Hvis en annen bruker låner parkeringsplassen, så skal tiden den er opptatt bli trukket fra det totale tilgjengelige tidsintervallet.

## **9.9 Betaling**

**Bruker skal betale utleieren for å opprette en reservasjon**

**ID: FK19**

Bruker betaler for antall timer de ønsker å leie parkeringsplassen.

**Tredjepart**

**ID: FK20**

Når en bruker sender inn en forespørsel om å kunne leie en parkeringsplass, så må tredjepart verifisere kjøpet før en bruker kan leie.

## **9.10 Administrering**

**Administrator skal kunne bannlyse bruker**

**ID: FK21**

Administrator skal ha full rett til å bannlyse en vilkårlig bruker.

**ID: FK21.1** Den bannlyste brukeren skal få beskjed om bannlysingen via e-post og direkte i systemet.

**Administrator skal kunne slette bruker**

**ID: FK22**

Administrator skal ha full rett til å slette individuelle brukere slik at all tilhørende data blir også slettet fra databasen.

**ID: FK22.1** Den slettede brukeren skal få beskjed om slettingen via e-post.

**Administrator skal kunne slette tilbakemeldinger**

**ID: FK23**

Administrator skal ha full rett til å slette individuelle tilbakemeldinger på andre innlegg slik at de fjernes både fra grensesnittet og databasen.

**Administrator skal kunne slette innlegg**

**ID: FK24**

Administrator skal ha full rett til å slette individuelle innlegg.

**ID: FK24.1** Bruker som lagde innlegget skal få beskjed om slettingen via e-post og direkte i systemet.

## **Administrator skal kunne slette parkeringsplasser**

### **ID: FK25**

Administrator skal ha full rett til å slette individuelle parkeringsplasser.

**ID: FK25.1** Bruker som opprettet parkeringsplassen skal få beskjed om slettingen via e-post og direkte i systemet.

## **Administrator skal kunne slette reservasjoner**

### **ID: FK26**

Administrator skal ha full rett til å slette individuelle reservasjoner.

**ID: FK26.1** Bruker som opprettet reservasjoner skal få beskjed om slettingen via e-post og direkte i systemet.

## **Bedrift skal ha administrerende rettigheter på deres egne innlegg**

### **ID: FK27**

Dette innebærer at bedriften kan blokkere brukere av deres egne parkeringsplasser fra å bruke de igjen og slette tilbakemeldinger på deres egne innlegg.

## **Bruker skal kunne rapportere andre brukere sine innlegg**

### **ID: FK28**

Brukere har frihet til å rapportere andre brukere sine innlegg. Rapporteringen skal bestå av en begrunnelse.

**ID: FK28.1** Brukeren som har lagd innlegget skal få varsel via e-post om rapporteringen.

## **Bruker skal kunne rapportere andre brukere**

### **ID: FK29**

Brukere har frihet til å rapportere andre brukere. Rapporteringen skal bestå av en begrunnelse.

**ID: FK29.1** Brukeren som har lagd innlegget skal få varsel via e-post om rapporteringen.

## **9.11 Persistent data**

### **Systemet skal lagre informasjon om parkeringsplasser**

#### **ID: FK30**

Systemet skal lagre all informasjon relatert til enhver parkeringsplass i en tabell i en database, slik at endringene gjenstår.

## **10 Ikke funksjonelle krav**

### **10.1 Tilgjengelighet**

#### **ID: IFK1**

All brukere skal ha tilgang til applikasjonen til enhver tid.

#### **Bruker kvalifikasjon**

#### **ID: IFK2**

Brukere må minimum være minimum 16 år, og ha gyldig førerbevis.

## **10.2 Sikkerhetskopi(Backup)**

### **ID: IFK3**

Sikkerhetskopiering av database skal skje hver 24 time kl 04.00

## **10.3 Dependency på tredjepart**

### **ID: IFK4**

Betaling verifiseres og håndteres av NETS

## **10.4 Legal and licensing issues**

### **ID: IFK5**

Brukeren skal få en Terms of service hvor de får informasjon om hvordan informasjonen deres brukes, og hva de har rettigheter på, og hvilke rettigheter vi har ovenfor deres bruk av tjenesten.

## **10.5 Nettverkstopologi**

## **10.6 Preformance / response time**

## **10.7 Personvern**

## **10.8 Kvalitet**

## **10.9 Sikkerhet**

## **10.10 Supportability**

## **10.11 Testability**

## **10.12 Interface**

# **11 Prototype**

## **11.1 Sammendrag**

Vi har laget en forenklet versjon som viser hvordan strukturen på programmet kan se ut. Vårt fokus ved prototypen var å vise en oversikt over parkeringsplasser der man kan trykke inn på en, få mer spesifikk info om selve parkeringsplassen, og å legge den til som en reservasjon.

Det andre fokuset var å ha en admin bruker som kan kontrollere bruken av siden ved suspensjon og sletting av brukere, parkeringsplasser og reservasjoner

I denne forenklet versjonen, så gjør vi all lagring mot .json filer.

Vi har lagdelt back-end koden til controller, model, datahandler og repository. Dette tillater at vi har splittet funksjonaliteten til forskjellig seksjoner av programmet og kan gjøre endringer uten å påvirke den generelle funksjonaliteten i programmet. Dette gjør for eksempel at et switch til en database-modell ville kun trenge endring i datahandler seksjonen av programmet.

Vi har valgt å bruke Javalin med Vue som rammeverk for front-end. Dette har gjort at det var relativt enkelt å få en pen og oversiktlig MVP. Dette er hovedavhengigheten i programmet vårt.

## **11.2 Innstallasjon**

For å bygge og kjøre programmet åpner man hele mappen som et intellij prosjekt, da vil også alle avhengigheter bli installert, eller det vil komme en installasjonsprompt med beskrivelse i terminalen i

intelliJ(dette skjer for npm installasjon). Dersom dette skjer fullfører du installasjonen som beskrives i terminalen. Deretter må man trykke på den grønne hammeren i høyere hjørnet eller ctrl+f9 for å bygge programmet. Om det fortsatt ikke kjører, last inn maven avhengigheter på nytt ved å trykke på søkeknappen i høyre hjørne av intelliJ, skriv maven, og trykk reload maven dependencies.

### 11.3 Bruksmanual

For å starte programmet velger man main metoden i drop-down menyen og trykker grønn pil eller shift+f10. Da vil javalin kjøre i konsollen i intelliJ, og det vil åpne seg port på <http://localhost:7000/>. Når man trykker på den kommer man en hovedside med tre knapper. En for bruker, en for corporation og en for admin.

Når man har kommet inn til localhost:7000 blir man presentert med tre knapper. En for bruker, en for corporation og en for admin. Bruker knappen leder til parkeringsplass oversikten der alle tilgjengelige plasser vises. Derfra kan bruker trykke seg inn på en parkeringsplass, trykke seg inn til bestillings-siden, og fylle ut ordren, og trykke submit knappen for å bestille, da skal den dukke opp i “My parking spots” på bunnen av siden der det står reservasjoner.

Inne på denne siden kan brukeren opprette parkeringsplasser de ønsker å leie ut ved å trykke på “Create new parking spot”-knappen der de taes til en skjema for utfylling. Dersom skjema ikke er tilstrekkelig utfylt vil en feilmelding vises med all info som er nødvendig i skjemaet. Når skjema er riktig utfylt kan man trykke submit knappen, og den nye parkeringsplassen vil umiddelbart være postet til både hovedoversikten og til brukerens “My parking spots”-side. Foreløpig fungerer corporation-brukeren på lik linje som bruker-typen.

Da man trykker admin-typen kommer man til en oversikt over brukere, parkeringsplasser, poster og reservasjoner. Admin kan suspendere eller slette kontoer, og den kan slette parkeringsplasser, poster og reservasjoner.

### 11.4 Implementerte krav og tester

For å kjøre backend testene må man åpne test-mappen i intelliJ og høyre-klikke på den grønne java-mappen, og velge Run ‘All Tests’ With Coverage. Da kjører man testene og kan se andelen av programmet der det er test-dekning.

### 11.5 Avhengigheter

Avhengigheter blir automatisk lastet inn av maven ved åpning av programmet. Dersom dette ikke skjer kan man gjøre det manuelt i intelliJ.

### 11.6 Begrensninger

De begrensningene vi har møtt på i utviklingen har for det meste vært praktiske. Vi har valgt å holde corporation og bruker som like bruker-typer i prototypen ettersom at de i hovedsak deler mange egenskaper.

Vi møtte motstand ved lastning av fler forskjellige typer data til samme vue side, som har limitert dens totale omfang for visning, og reservasjon. Dette spesifikt da single-parking-spot seksjonen av programmet.

Vi har forsøkt å begrense prototypen til skallet som er nødvendig før implementering av database, betalingssystem og brukerregistreringer.

Vi har ikke lagt inn noen bil-registrering eller bruker-registrering da dette er undøvelig for funksjonaliteten i prototypen.



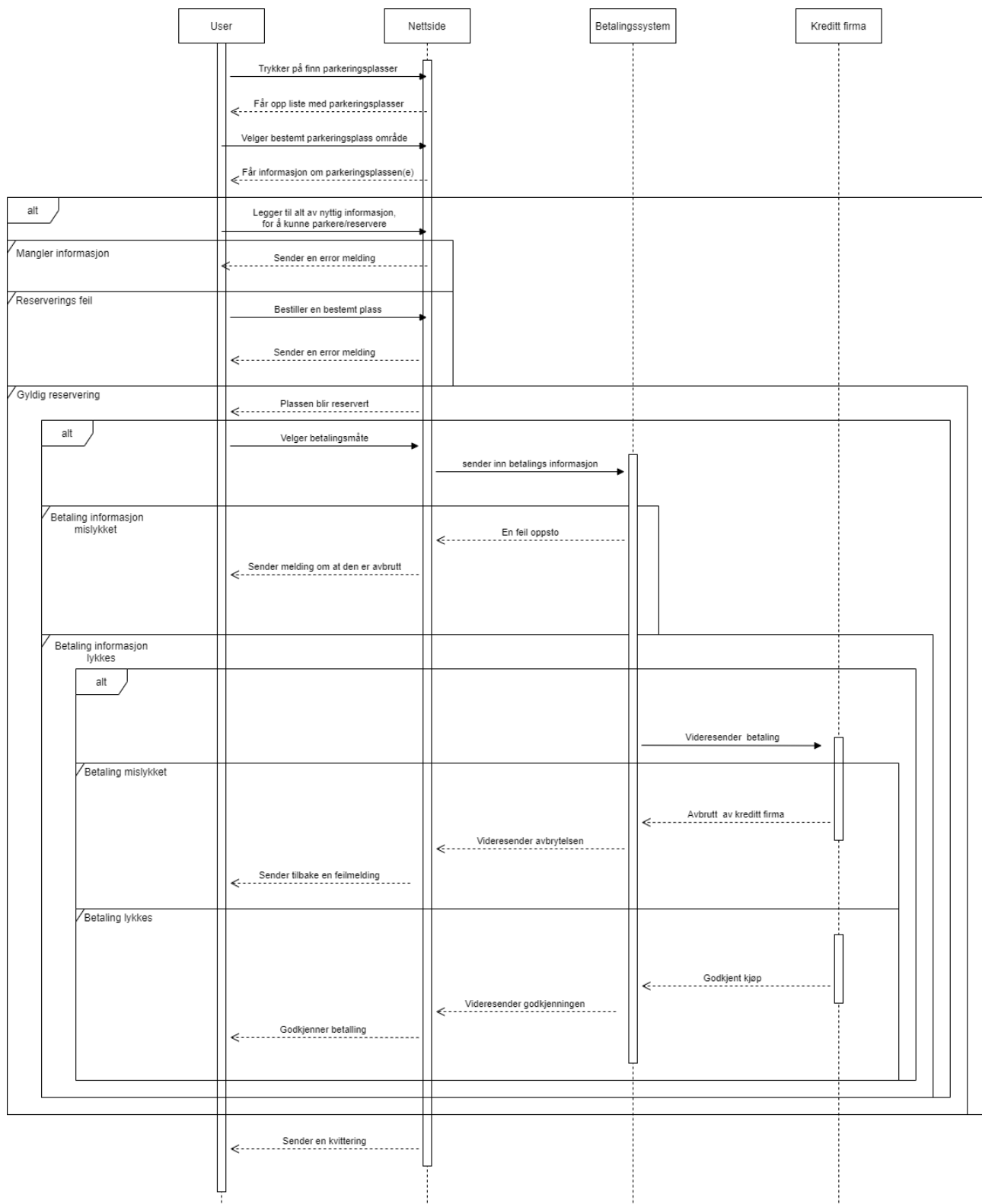
Vi ønsket å legge inn corporation-bruker som kun kunne gjøre utleiing av p-plass, men også som kunne legge til medlemmer som kan leie av de plassene spesifikt. Dette ble en for stor utfordring, og vi valgte å holde den i sin basis-form.

Det var ment at vi skulle ha flere brukere til prototypen, men valgte å fokusere på å få én bruker til å fungere på alle stegene i utlegging og leie-prosess.

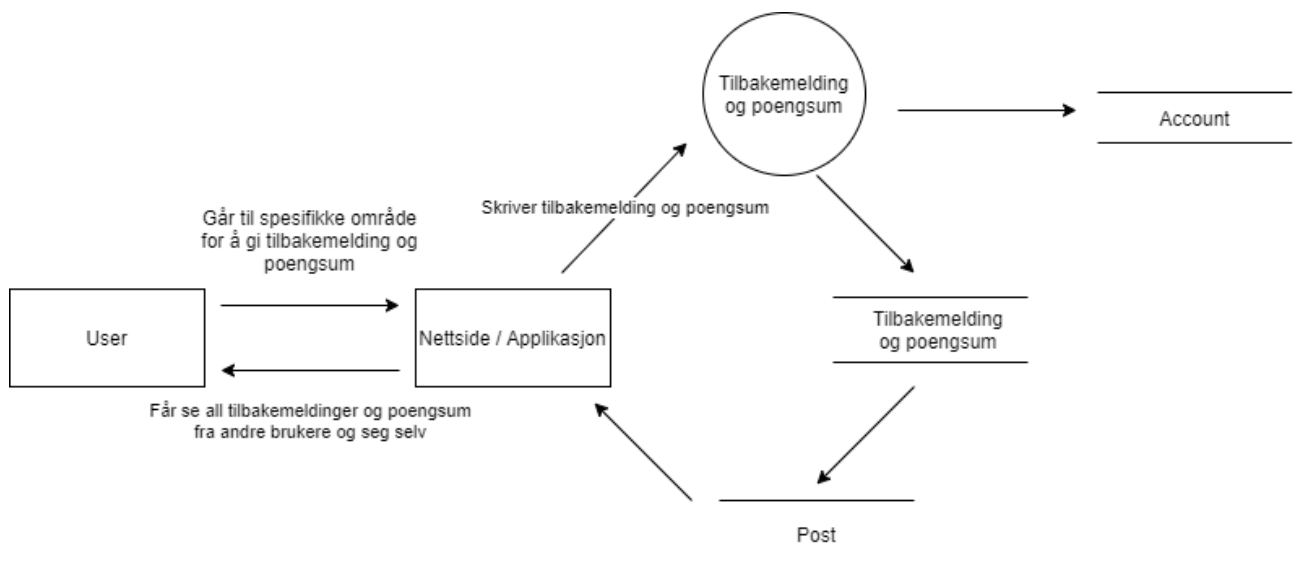
MVP-en har ikke fått inn noe rangerings-system eller filtrering, da vi ønsker å filtrere på lokasjon, men ikke har kunnskapen til å gjøre denne typen filtrering.

## 12 Diagrammer

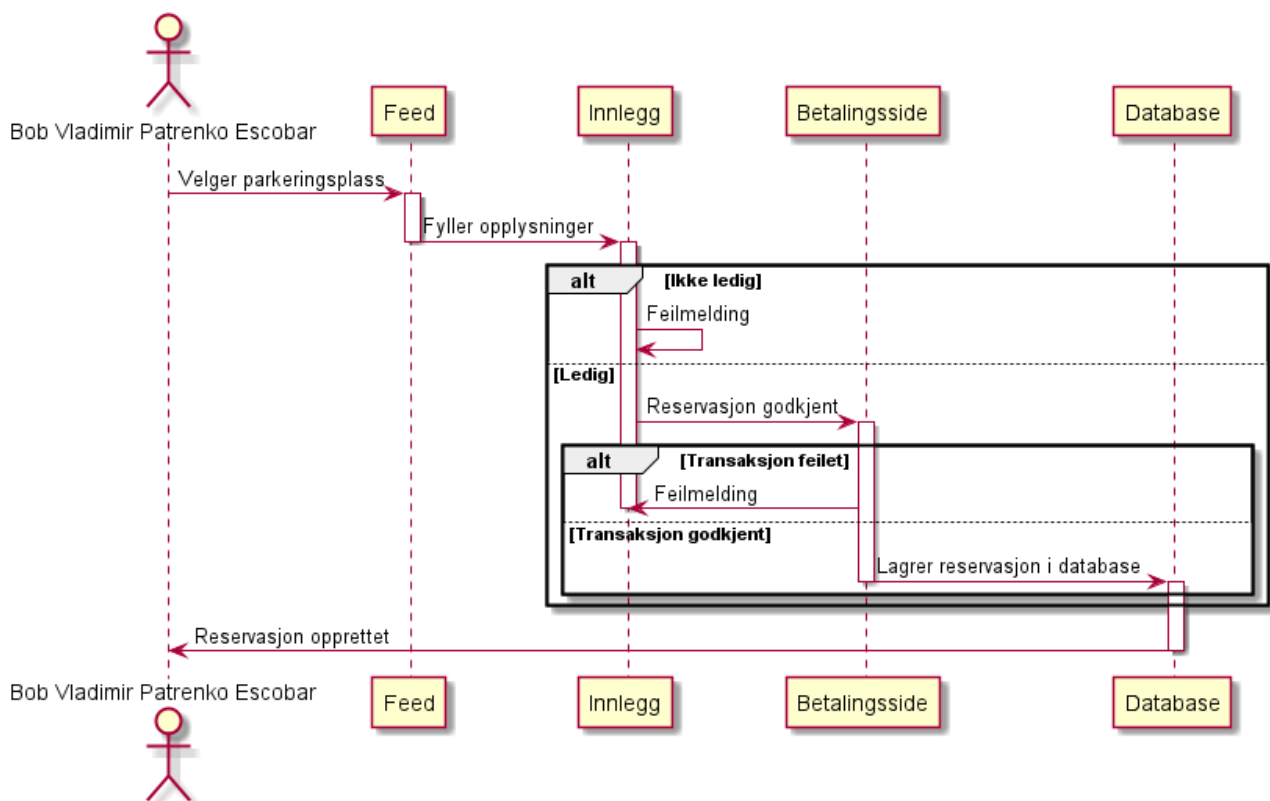
### 12.1 Sekvesndiagram - Leie en parkeringplass.



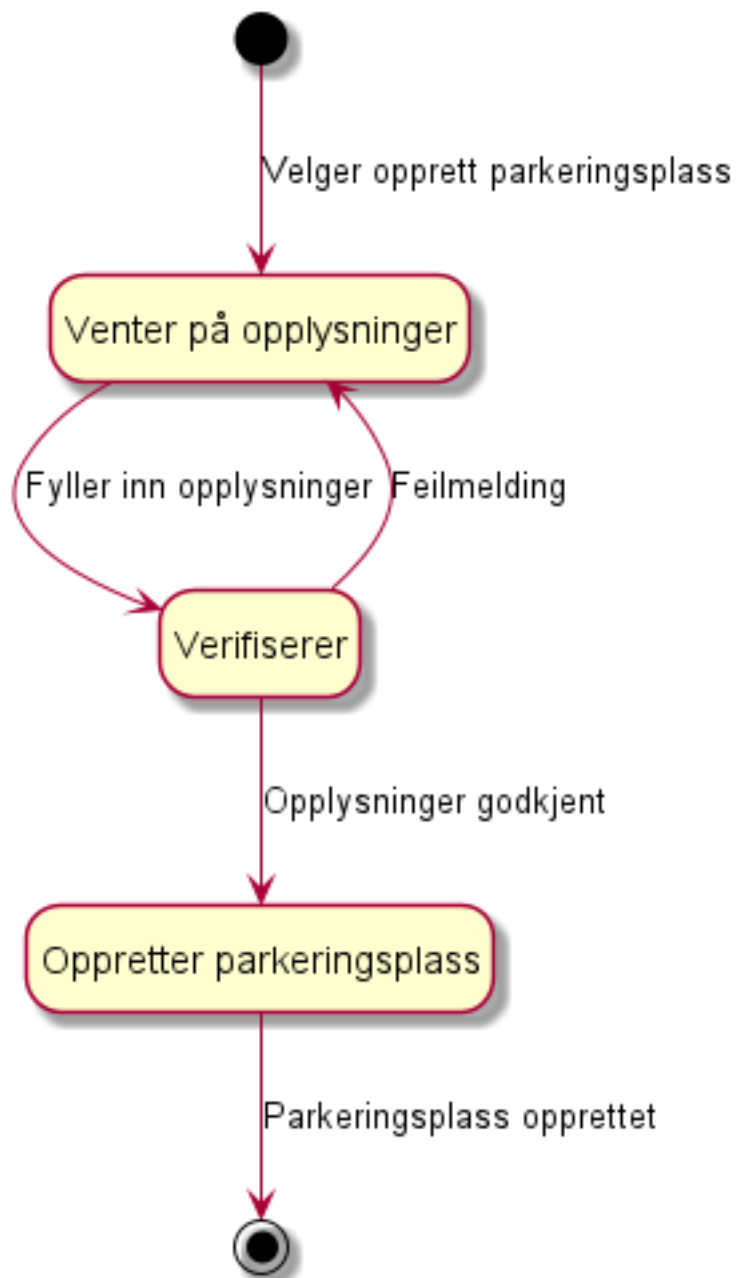
## 12.2 Dataflytdiagram - Tilbakemelding



## 12.3 Sekvensdiagram - Opprette reservasjon



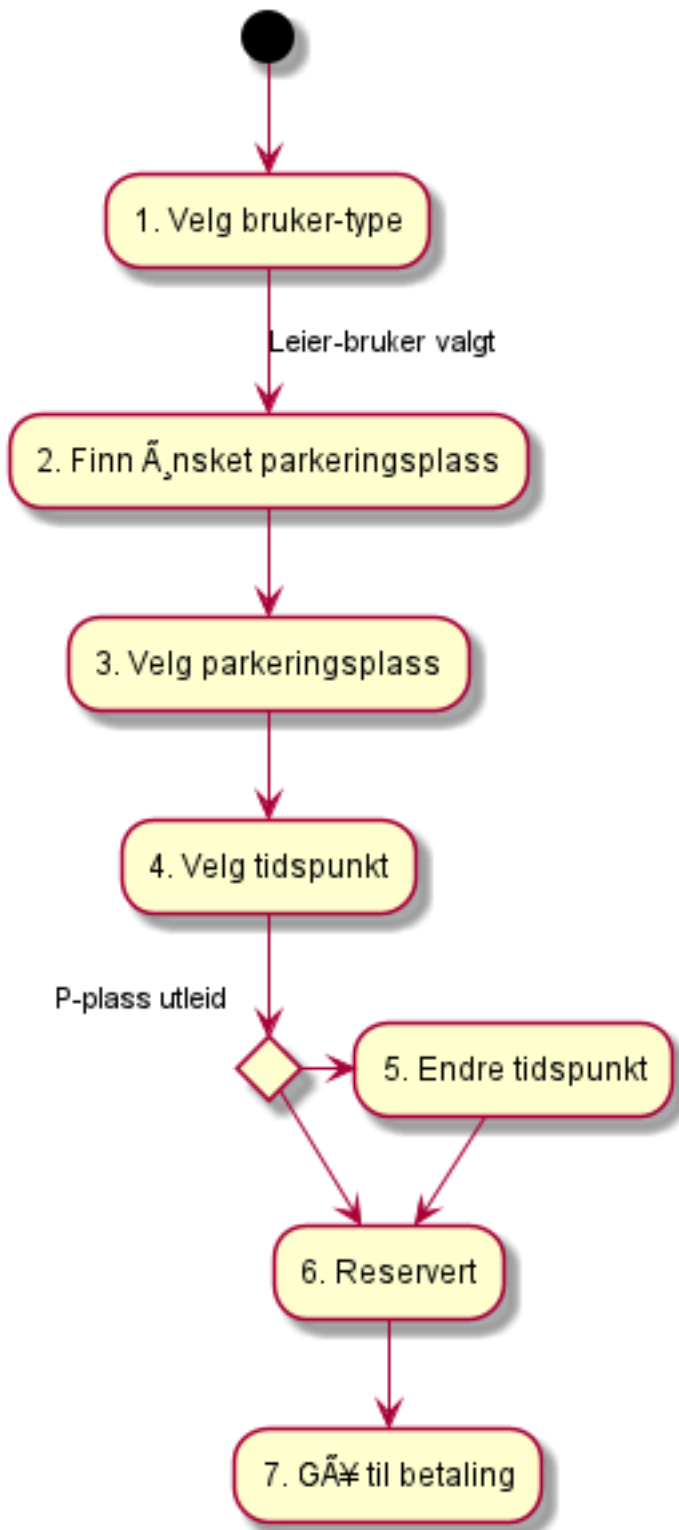
## 12.4 Tilstandsdiagram - Registrere parkeringsplass



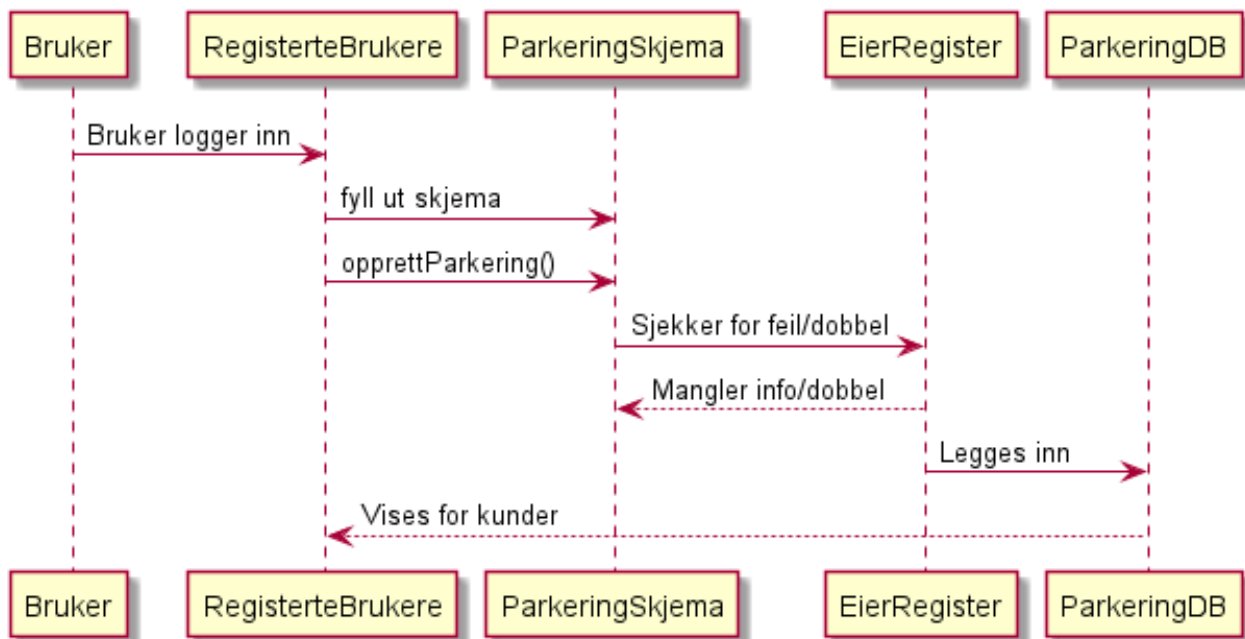
### Forklaring

Når bruker oppretter parkeringsplass venter systemet på en POST forespørsel med informasjon om gatenavn, adresse, bredde, lengde og høyde, samt informasjon angående parkeringsplasstype (handicap, el-bil, osv.). Dersom opplysningene ikke tilfredstiller forventningene i systemet, får bruker en feilmelding og blir nødt til å fylle ut på nytt. Dersom opplysningene er godkjent blir parkeringsplassen opprettet.

## 12.5 Aktivitetsdiagram - Leie en parkeringsplass



## 12.6 Sekvensdiagram - Leie ut en parkeringsplass



## 13 Estemering

### 13.1 Enheter

	Utviklingsstørrelse			
Verdi	X-Large	Large	Medium	Small
X-Large	0	4	6	7
Large	-4	0	2	3
Medium	-6	-2	0	1
Small	-7	-3	-1	0

### 13.2 Estimeringstabell - Funksjonelle krav

EsID	Vikitighet	Vanskelighetsgrad	Utviklingsstørrelse
FK1	NULL	NULL	NULL
FK2	NULL	NULL	NULL
FK3	NULL	NULL	NULL
FK4	NULL	NULL	NULL
FK5	NULL	NULL	NULL
FK6	NULL	NULL	NULL
FK7	NULL	NULL	NULL
FK8	NULL	NULL	NULL
FK9	NULL	NULL	NULL
FK10	NULL	NULL	NULL
FK11	NULL	NULL	NULL
FK12	NULL	NULL	NULL
FK13	NULL	NULL	NULL
FK14	NULL	NULL	NULL
FK15	NULL	NULL	NULL
FK16	NULL	NULL	NULL
FK17	NULL	NULL	NULL
FK18	NULL	NULL	NULL
FK19	NULL	NULL	NULL
FK20	NULL	NULL	NULL
FK21	NULL	NULL	NULL
FK22	NULL	NULL	NULL
FK23	NULL	NULL	NULL
FK24	NULL	NULL	NULL
FK25	NULL	NULL	NULL
FK26	NULL	NULL	NULL
FK27	NULL	NULL	NULL
FK28	NULL	NULL	NULL
FK29	NULL	NULL	NULL
FK30	NULL	NULL	NULL
FK31	NULL	NULL	NULL
FK32	NULL	NULL	NULL
FK33	NULL	NULL	NULL
FK34	NULL	NULL	NULL
FK35	NULL	NULL	NULL
FK36	NULL	NULL	NULL

### 13.3 Estimeringstabell - Ikke funksjonelle krav

EsID	Vikitighet	Vanskelighetsgrad	Utviklingsstørrelse
IFK1	NULL	NULL	NULL