**Hvordan modellen oppfyller kravspesifikasjonen**

1. *Logge på. Ansatte får tilgang til kalendersystemet ved å logge seg på kalenderklienten med brukernavn og passord.*

* Vi har opprettet en entitetsklasse BRUKER der hver forekomst vil ha et unikt brukernavn av datatype VARCHAR(45) og samtidig et personlig passord av datatype VARCHAR(100).

1. *Legge inn avtale. Ansatte skal kunne legge inn avtaler i kalendrene sine. En avtale legges inn på avtaledato med et start- og sluttidspunkt, samt en kort beskrivelse av avtalen ("Bil på verksted") og eventuelt sted for avtalen ("Strandveien Auto"). Det skal være lov å legge inn flere avtaler for overlappende tidspunkt.*

* Vi har laget AVTALE som en entitetsklasse der vi har starttidspunkt, sluttidspunkt, beskrivelse og sted som attributter i tillegg til avtaleID og avtalenavn. Ved at vi ikke bruker unike start- og sluttidspunkter har vi også tilrettelagt for at avtaler kan overlappe hverandre. Vi har trukket en relasjon (AVTALEBRUKER) mellom BRUKER og AVTALE slik at en BRUKER kan opprette en avtale. Brukeren som oppretter avtalen skal være administrator for denne.

1. *Håndtere møtedeltakere. Den som har lagt til en avtale skal også kunne legge til (eller fjerne) (potensielle) deltakere, ved å angi enkeltpersoner og/eller grupper. Grupper administreres utenfor systemet og kan ses på som en samling personer uten noen spesiell struktur (roller/hierarkier). Inviterte deltakerne skal kunne bekrefte evt. avkrefte at de deltar. Dersom en deltaker avkrefter deltakelse, så kan han/hun også velge om den skal skjules i kalenderen sin.*

* Vi håndterer grupper som en entitetsklasse, GRUPPE, der vi viser at en bruker tilhører en gruppe, ved å knytte BRUKER og GRUPPE gjennom relasjonen GRUPPEBRUKER. Den forekomsten av BRUKER som har administratorrettigheter til en avtale (den som har opprettet avtalen) kan legge til andre forekomster av BRUKER/GRUPPE til avtalen eller eventuelt velge å fjerne forekomster av BRUKER/GRUPPE fra avtalen dersom de har tilgang til denne fra før av. For hver BRUKER som har tilgang til en AVTALE skal det gjennom relasjonen AVTALEBRUKER komme frem om brukeren deltar eller ikke og dette har vi implementert ved å ha «deltar» som et eget attributt i AVTALEBRUKER som en BOOLEAN der default-verdi er NULL, mens dersom brukeren huker av for «delta» vil denne bli TRUE og FALSE dersom den velger «ikke delta». Vi skal også tilrettelegge for at brukeren kan velge om den ønsker å vise/skjule avtaler i kalenderen, men ønsker ikke å benytte et eget attributt til å håndtere dette.

1. *Slette avtale. Ansatte skal kunne slette avtaler de har opprettet, og disse forsvinner da fra kalenderen til alle deltakere.*

* En BRUKER med administrasjonsrettighet til en forekomst av AVTALE skal kunne fjerne forekomsten og da vil «ON DELETE CASCADE» på AVTALEBRUKER sørge for at alle forekomster av BRUKER ikke lenger vil være knyttet til avtalen.

1. *Endre avtale. Ansatte skal kunne endre på avtaler de har opprettet. Alle feltene kan endres. Alle deltakere blir varslet om endringen (på en eller annen måte), så de kan forholde seg til det og evt. endre på om de deltar eller ikke.*

* Dersom en forekomst av BRUKER har administrasjonsrettigheter til en AVTALE har den tilgang til å endre på verdiene for attributtene i den bestemte forekomsten av AVTALE. AvtaleID kan ikke endres. Forekomster av BRUKER som har tilgang til AVTALE vil få varsel om endringene både på mail og inne i systemet.

1. *Reservere møterom. I stedet for å skrive inn sted for en avtale eller et møte, skal kunne velge møterom blant de som er ledige i det angitte tidsrommet. Ved av tidsrommet, skal kalendersystemet automatisk reservere møterommet for nye tidsrommet, men ikke holde på det om andre har reservert det da. Det være mulig å angi antall deltakere som rommet skal kunne ta, uavhengig av antallet deltakere som er invitert og evt. har bekreftet at de kommer.*

* Vi har her brukt en egen entitetsklasse MØTEROM med attributtene romID, navn, sted, tidsrom, og størrelse. En forekomst av AVTALE vil da automatisk bli knyttet til et rom som er ledig i tidsrommet avtalen foregår i og som samtidig har plass til minimum så mange personer som er invitert til avtalen.

1. *Visning. Kalenderklienten skal vise en ukeoversikt der alle avtaler og møter i den personlige kalender vises. Det skal være et tydelig skille mellom avtaler som en har opprettet selv og avtaler som en er invitert til. For avtaler en er invitert til så skal det være tydelig angitt status for egen deltakelse (om en har bekreftet/avkreftet deltakelsen). Det skal være tydelig angitt om en avtale er endret, så brukeren kan forholde seg til endringen. (Merk at det ikke er definert hva som gjør at systemet tror brukeren har forholdt seg til endringen.)*

* For de ulike forekomstene av AVTALE har vi en fremmednøkkel til en forekomst av BRUKER, som er administrator for avtalen og dette kan vi bruke til å tydeliggjøre gjennom brukergrensesnittet at brukeren er admin for en avtale ved at avtalen har et litt annerledes format enn de avtalene som brukeren bare er invitert til. Vi vil på tilsvarende måte benytte oss av deltar-attributtet som ligger i relasjonen AVTALEBRUKER. At en avtale har blitt endret kan vi vise ved at den bestemte avtalen vises med fet skrift i ukesoversikten.

1. *Status for deltakelse. For møter med inviterte deltakere så skal det være tydelig angitt om 1) alle har svart eller ikke og 2) om det er noen som har avslått invitasjonen.*

* Dette kan vi på tilsvarende måte gjøre gjennom brukergrensesnittet ved å sjekke deltar-attributtet til de ulike brukerne som er invitert. Dersom dette attributtet for noen brukere fortsatt har NULL som verdi vil det vises at ikke alle brukerne har svart, og dersom én eller flere inviterte brukere har avslått invitasjonen ved at deres deltar-attributt står som FALSE vil dette også komme frem.

1. *Melde avbud for møte. En ansatt kan melde avbud på en møteinnkalling ved å slette avtalen i sin personlige kalender. Når en ansatt melder avbud, sendes melding til alle de andre møtedeltakerne. Møteleder kan da velge om møtet skal avlyses eller om han/hun skal endre tidspunkt på møtet.*

* Når en forekomst av BRUKER melder avbud ved å slette avtalen vil, brukerens sin relasjon til avtalen, AVTALEBRUKER bli slettet og systemet vil da varsle de andre møtedeltakerne og møteleder om dette, slik at møteleder kan foreta endringer på møte dersom det er ønskelig.

1. *Reservere møterom. I stedet for å skrive inn sted for en avtale eller et møte, skal brukeren kunne reservere møterom. Databasen skal lage en liste med tilgjengelige møterom (tilgjengelig betyr ikke reserverte) i tidsperioden for avtalen/møtet. Brukeren kan da velge møterom fra denne listen. Om en avtale med reservert møterom slettes, skal reservasjonen slettes på databasen. Det samme gjelder for møter som avlyses.*

* For å oppfylle dette vil vi benytte oss av entitetsklassen MØTEROM, og dersom møteleder ikke ønsker å benytte møterommet som avtalen automatisk fikk tildelt, kan den velge blant de forekomstene av MØTEROM, som er ledige i det gitte tidsrommet og som har plass til alle deltakerne av møtet, som vil finnes i en liste. Dersom en forekomst av AVTALE blir slettet eller om et møte avlyses skal relasjonen mellom MØTEROM og AVTALE også bli slettet og møterommet blir nå ledig.

1. *Visning. Kalenderklienten skal vise en ukekalender der alle avtaler og møter i den ansattes personlige kalender vises. Det skal være enkelt å bla mellom ukene.*

* Dette skal vi få implementert på brukergrensesnittet ved at de forekomstene av AVTALE som er knyttet til den bestemte forekomsten av BRUKER skal vises i ukeoversikter der den aktuelle uken med tilhørende avtaler vises som default ukesoversikt.

1. *Spore møteinnkallinger. Kalenderklienten skal indikere i ukekalenderen om a) en møteinnkalling venter på svar fra en eller flere deltakere, b) en eller flere møtedeltakere har avslått møteinnkalling, eller c) om alle innkalte har godtatt møteinnkallingen.*

* *På tilsvarende måte som for krav 8 lar vi det at møteinnkallingen venter på svar komme frem gjennom brukergrensesnittet ved å sjekke om det finnes noen NULL-verdier på deltar-attributtet for brukere som er invitert. Vi vet også at en eller flere møtedeltakere har avslått møteinnkallingen dersom deltar-attributtet på minst én av brukerne er FALSE, og at alle innkalte har godtatt møteinnkallingen dersom samtlige møtedeltakere har TRUE som verdi for dette attributtet. Hvilken av disse tre scenariene som er gjeldende vil komme frem ved at vi for eksempel bruker forskjellige fargekoder på avtalen i ukesoversikten i brukergrensesnittet.*

1. *Vis flere kalendre. Det skal være mulig å se kalendere (avtaler og møter) for flere ansatte i samme ukesoversikt.*

* Dette kan vi implementere ved å gjøre det mulig å spørre om avtalene som tar for seg mer enn en forekomst av BRUKER og deretter vise avtalene som gjelder for de ulike ansatte i stedet for at vi bare viser avtalene som angår én ansatt.

1. *Alarm. Det skal være mulig for hver ansatt å konfigurere enhver avtale slik at avtalen genererer en alarm en gitt tid før møtet.*

* For hver avtale som en bruker har tilgang til skal den ha mulighet til å kunne huke av for «alarm» og velge hvor lenge før møtet den ønsker å bli varslet. En alarm vil da bli sendt til brukerens e-post i ønsket tid før møtet.

