

Collaboration Project
Conceptual Datamodel
Gruppe 34

Bjørn Åge Tungesvik

Tina Syversen
Eivind Kvissel

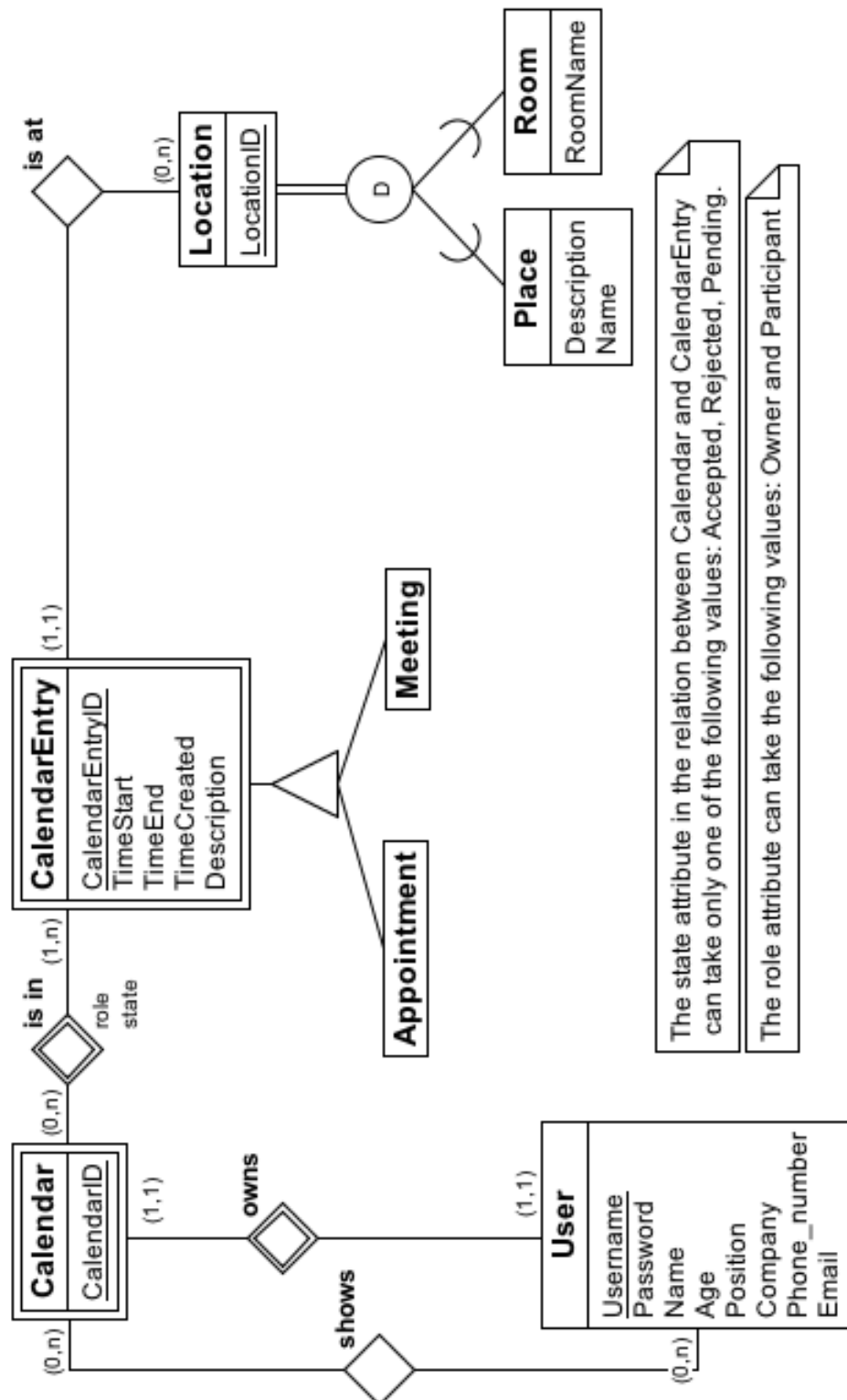
André Philipp
Håvard Høiby

Odd Magnus Trondrud

March 14, 2012

1 EER-Diagram

Please rotate your head 90° counter-clockwise. Thank you.



2 Compliance to the Requirement Specifications

The requirements' description is not presented in this document. The numbers correspond to the ones found in section 2.2 of the compendium. The rest of this document is in Norwegian. Sorry about that.

2.1 Logg på

Databasen inneholder informasjon som kan verifisere brukere (Username Password) under User tabellen, så personer som logger på får sin egen User-rad assosiert med den sesjonen.

2.2 Legge inn avtale

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").

CalendarEntry tabellen inneholder felt for start- og sluttidspunkt, felt for beskrivelse (Henholdsvis TimeStart, TimeEnd og Descripton feltene), og en relasjon til en tabell kalt Location for å holde orden på sted. (Sted ligger i Place tabellen som er subklasse til Location)Hver CalendarEntry rad er knyttet til den spesifikke brukeren som oppretter raden.

2.3 Slette avtale

Hver avtale er representert som en CalendarEntry i databasen. En bruker skal ha full kontroll over avtaler som han har laget, inkludert å slette dem.

2.4 Endre avtale

En bruker skal kunne gå inn i og endre (gjennom GUI'et) hvert felt bortsett ifra "timeCreated" i CalendarEntry og Location.

2.5 Kalle inn til møte

En User skal kunne oprette en rad i Meeting tabellen på samme måte som en avtale, og bli satt som "leader" for "Role" feltet i "Is in" - relasjonen mellom Calendar og CalendarEntry, og bruke "Owns" relasjonen mellom User og Calendar. Han skal også kunne invitere andre til møtet, de skal bli satt som "Participant" i "Role" feltet i "Is in" relasjonen. Kun den som har "Leader" i Role i "is in" relasjonen skal kunne endre feltene innenfor møtet. CalendarEntry - objektet skal også ha en liste over alle som er assosiert med møtet.

2.6 Motta møteinnkalling

"State" feltet innenfor "Is in" relasjonen mellom Calendar og CalendarEntry tabellen holder orden på statusen for de som er assosiert med møte. Dette felte skal bare kunne endres av deltakere i et møte, men skal kunne sees av alle som er innkalt

2.7 Endre møteinnkalling

Når lederen for en CalendarEntry endrer datoen for møte, skal det sendes ut en beskjed, og en melding om å oppdatere "state" til alle assosiert med møtet. "State" vil også bli resatt til "Pending" for alle deltakere når dette skjer. Om en ansatt setter statusen sin til "reject", vil møteleder motta en beskjed om at han må endre tidspunkt igjen, eller avlyse møtet. Disse notifikasjonene lagres ikke i databasen men kommer av hvilken state "Is in" relasjonen står i.

2.8 Avlyse møte

Når en leder avlyser et møte, skal det sendes ut en melding til alle deltakere, og raden slettes fra databasen.

2.9 Melde avbud for møte

Når en User melder avbud til en avtale som han står som participant i, vil “State”-en hans skiftast til rejected, og CalendarEntry-en vil forsvinne fra “Calendar”-raden hans. Det vil også sendes en melding til alle andre som har assosiasjon med det objektet. Den som står som Leader for det møtet vil i tillegg få beskjed om enten å endre TimeStart/TimeEnd for CalendarEntryen, eller å slette det (Avlyse).

2.10 Reservere møterom

Subklassen Room under Location holder orden på møterom. Relasjonen mellom Location og CalendarEntry gir tilgang på informasjon om når rommet er ledig.

2.11 Visning

Calendar klassen samler alle CalendarEntry - radene for en user.

2.12 Spore møteinnkallinger

Dette blir tatt hånd om av State - feltet under “is in” relasjonen mellom Calendar og CalendarEntry.

2.13 Vis flere kalendere

“Shows” relasjonen mellom user og Calendar gir mulighet for å requeste andres kalendere for å bli vist sammen med ditt eget view (representert med “owns” relasjonen).