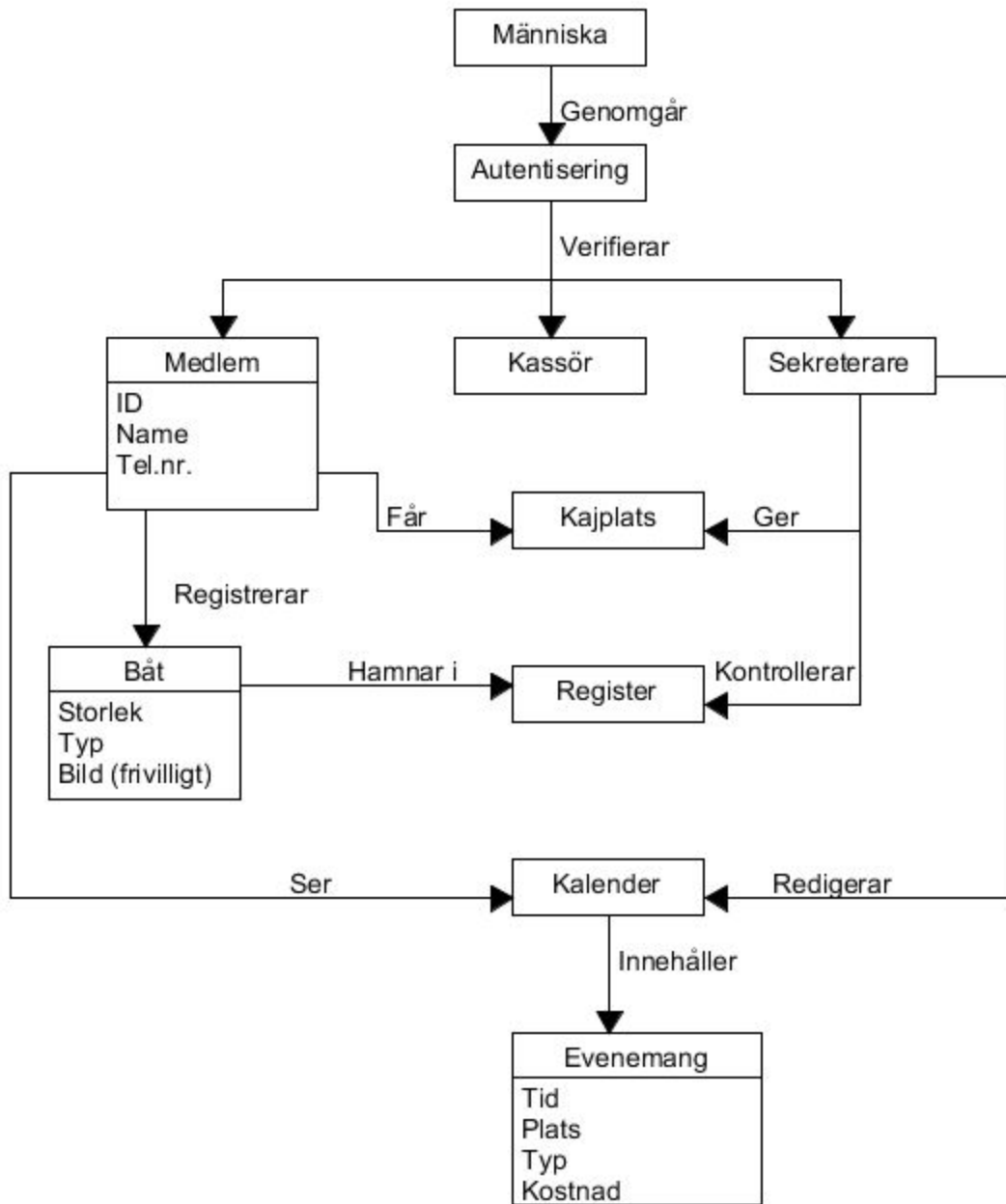


# Patrik Erlandssons grupp



# Peer review

## Som utvecklare, hjälper modellen dig och varför/varför inte?

Modellen är rörig och har med även det dynamiska flödet, till exempel att en Människa genomgår en Autentisering eller att en Sekreterare redigerar Kalender. Det finns även med vaga associationer som att Sekreteraren kontrollerar Register. Vad kontrolleras egentligen?

## Tror du att en expert på domänen, till exempel sekreteraren, skulle förstå modellen och varför/varför inte?

Då modellen är rörig kan det vara svårt att förstå den, även om sekreteraren säkert förstår modellen eftersom sekreteraren är väl insatt i domänen. Dock kan det vara svårare för utvecklarna att förstå vad som är viktigt och hur den statiska vyn ser ut.

## Vad är modellens starka sidor och varför?

Bra namnval på klasserna; de flesta klasserna utgår från substantiv i problembeskrivningen [1, s.139]. Alla väsentliga klasser finns med.

## Vad är modellens svaga sidor och varför?

Den är rörig, vilket anmärktes under första rubriken. UML-notation är felaktig, det ska inte vara några pilar i ett domän diagram utan vyn är statiskt [1, s.253]. Det finns ett överflöd av attributer vilket för med sig att man inte ser vilka attributer som är extra viktiga i modellen [1, s.158]. Det tillför till exempel inte någon förståelse att evenemang har attributen tid, plats, typ och kostnad. Överflöd av associationer, till exempel mellan medlem och kalender. Modellen behöver inte visa allt som kan hända, att en medlem kan se kalendern, utan bör fokusera på hur klasserna hänger samman på ett statiskt plan [1, s.151]. En sådan association är bara relevant om det är av vikt att över tid se vilken medlem som ser eller har sett kalendern [1, s.149-151]. Flera av associationerna förvirrar också mer i det här skedet än vad de tillför, exempelvis Medlem-Registrerar-Båt [1, s.151]. Den processen uppstår säkert någonstans men i en statisk vy skulle det snarare vara Medlem-Äger-Båt eller liknande [1, s.152-153, 157]. Överflöd av klasser, till exempel klasserna Människa, Autentisering, Kalender och Kassör [1, s.144]. Vidare är både Register och Kalender rapportklasser och bör användas med försiktighet i den statiska vyn. [1, s.145]. Exempelvis har all information i Kalender sitt ursprung i Evenemang.

## Anser du att modellen uppfyller kraven för betyg 2?

Nej. För betyg 2 krävs det att se över vilka klasser och relationer som är relevanta i sammanhanget. Sträva efter en statisk karta över domänen [1, s.145] med de krav som specificeras istället för allt som kan hända i domänen. UML-notationen bör också kontrolleras [1, s.253].

## Referenser

1. Larman C., Applying UML and Patterns 3rd Ed, 2005, ISBN: 0131489062