

## 0.1 Proces/task kommunikation og synkronisering

### Kommunikation

TrackABusprovideren er en klasse, som bliver kaldt fra Main tråden når MySQL databasen skal tilgås. Dette sker ved at TrackABusProvideren laver en ny tråd, der kalder ned i data tilgangs laget til SoapProvideren der tilgår databasen, og returnerer til TrackABusProvider med det relevante data. For at tråden i TrackABusProvideren kan sende data tilbage til main tråden, bliver der brugt en message handler, der lytter på, om der kommer en besked fra TrackABusProvideren.

Både SetFavoriteBusRoute og RemoveFavorite bliver begge startet af main tråden, men da de blot skal indsætte og slette fra SQLite databasen, har de ikke behov for at kommunikerer med andre tråde.

Administrations hjemmesidens server tråd og client tråd snakker sammen ved brug af AJAX kald, *Se afsnit 8.2.3 Komponent 3: Administrations hjemmeside* for en beskrivelse af, hvordan dette virker og er implementeret. Dette kald starter en asynkron task der bliver kørt på serveren. Når serveren er færdig med at udføre tasken vil der blive sendt et response tilbage til klienten, der lavede AJAX kaldet.

Simulatorens bus tråde kommunikerer med UI tråden ved hjælp af to custom events. Det ene sørger for at ændre viewet når det er nødvendigt, og det andet sørger for at sætte en log-besked. Dette sker igennem BeginInvoke funktionen, som sørger for, at selv hvis en view opdatering sker direkte fra en tråd, håndteres det i UI-tråden.

### Synkronisering

Trådene SetFavoriteBusRoute og RemoveFavorite er de eneste tråde på mobil applikationen, der har behov for at blive synkroniseret. Grunden til at TrackABusProvideren ikke har brug for Synkronisering er, at den ikke kan skrive til MySQL databasen, men kun hente. SetFavoriteBusRoute skal skrive til SQLite databasen, hvilket skal synkroniseres så der ikke bliver skrevet flere gange til SQLite databasen på samme tid. Dette er blevet opnået ved at knappen, der bliver brugt til at favorisere en bus, bliver deaktiveret indtil

tråden er færdig med at skrive til databasen, dette gør, at det ikke er muligt at starte to tråde der skriver til SQLite databasen på samme tid.

Administrations hjemmesidens client og server skal synkroniseres, så der ikke bliver skrevet til databasen flere gange på samme tid. Dette sker på samme måde som i mobil applikationen ved, at det ikke er muligt at skrive til databasen, hvis en anden task allerede er igang med at skrive.

I simulatoren bliver tråd synkroniseringen håndteret, ved hjælp af binære semaforer. Dette er kun relevant, når MySQL databasen skal tilgås. Skrivnings- og læsnings funktionerne er statiske, og det samme er deres relevante semaforer, således at når tråd starter skrivningen, tager denne semaforen, og frigiver den når den er færdig. I tilfælde af, at simuleringen stoppes, og trådene lukkes, før semaforen er frigivet, frigives semaforerne ved hver afslutning af simulering. Dette gøres for at undgå deadlocks.