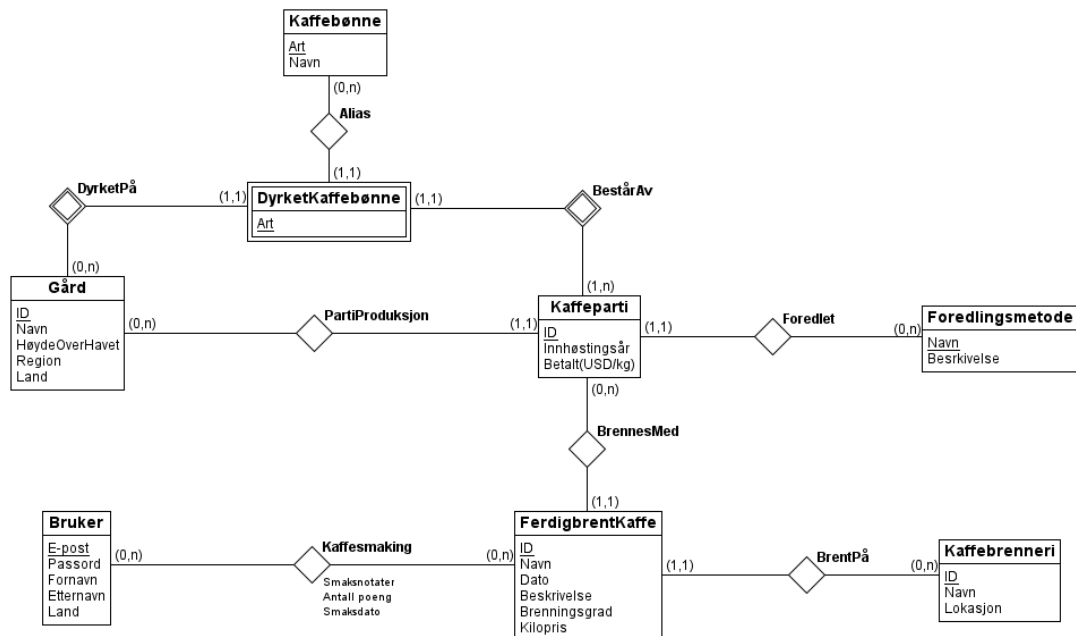


TDT4145 Prosjekt 2022 Delinnlevering 1

1. ER-diagram for modellen



2. Relasjonstabeller for modellen og drøfting normalform

- **Bruker** (Epost, Passord, Fornavn, Etternavn, Land)
 - E-post er primærnøkkelen til tabellen
 - Passord, Fornavn, og Etternavn er påkrevd

Bruker-tabellen er på normalform 4NF, ettersom tabellen er i BCNF, og vi ikke har noen flerverdi funksjonelle avhengigheter. Tabellen er i BCNF fordi alle felter kun er funksjonelt avhengig av Epost, som er kandidatnøkkelen for tabellen. Det finnes ingen ytterlige funksjonelle avhengigheter som ikke er trivielle.

Vi bruker Epost som naturlig nøkkel, ettersom det å tillate kun en bruker per epost er en god begrensning for de fleste applikasjoner, i tillegg til at dette er en veldig vanlig måte å identifisere brukere på.

- **Kaffesmaking** (BrukerEpost, KaffeID, Smaksnotat, Poeng, Dato)
 - BrukerEpost + KaffeID er primærnøkkelen til tabellen
 - BrukerEpost er fremmednøkkel til **Bruker**
 - KaffeID er fremmednøkkel til **FerdigbrentKaffe**
 - Poeng og Dato er påkrevd

Kaffesmaking-tabellen er på 4NF, ettersom vi ikke kan ha flerverdi-avhengigheter, og at alle felter blir bestemt av BrukerEpost + KaffeID, samtidig som vi ikke har noen transitive avhengigheter. Tabellen oppfyller BCNF-kravet, ettersom verken Smaksnotat, Poeng, eller Dato, bestemmer nøkkelattributtene. Som resultat av at BrukerEpost og KaffeID er primærnøkkelen, vil det ikke være mulig at en bruker legger ut flere anmeldelser på samme kaffe.

- **FerdigbrentKaffe** (ID, KaffebrenneriID, KaffepartiID, Navn, Dato, Beskrivelse, Brenningsgrad, Kilopris)
 - ID er primærnøkkelen til tabellen
 - KaffebrenneriID er fremmednøkkel til **Kaffebrenneri**
 - KaffepartiID er fremmednøkkel til **Kaffeparti**
 - Navn, Dato, KaffebrenneriID og KaffepartiID er påkrevd

Tabellen er på normalform 4NF, ettersom vi kun har en kandidatnøkkel, og vi ikke har noen flerverdi funksjonelle avhengigheter.

Her bruker vi en generert nøkkel (ID) for å identifisere entiteten. Vi har valgt dette for å unngå å lage restriksjoner for klassen som kan medføres dersom for eksempel klassen hadde vært svak, og vi hadde brukt Navn + KaffebrenneriID som primærnøkkel. I dette tilfellet hadde det ikke vært mulig å ha to entiteter med samme navn, oppført på forskjellige datoer.

Å bruke en generert nøkkel medfører noen ekstra operasjoner når vi skal slå opp i tabellen, men som vi ser, medfører dette mindre begrensninger.

- **Kaffeparti** (ID, *Foredlingsmetode*, *GårdID*, *Innhøstningsår*, *Betalt(USD/kg)*)
 - ID er primærnøkkelen til tabellen
 - Foredlingsmetode er fremmednøkkel til **Foredlingsmetode**
 - GårdID er fremmednøkkel til **Gård**
 - Foredlingsmetode og GårdID er påkrevd

Tabellen er på 4NF. Tabellen har ingen flerverdi funksjonelle avhengigheter, og alle attributter er kun bestemt av primærnøkkelen (ID).

- **Kaffebrenneri**(ID, *Navn*, *Lokasjon*)
 - ID er primærnøkkel til tabellen
 - Navn og Lokasjon er påkrevd

Tabellen er på 4NF, ettersom Navn og Lokasjon er uavhengige, alle attributter er bestemt kun av ID, og at det ikke forekommer flere rader med samme ID.

- **Foredlingsmetode** (Navn, *Beskrivelse*)

- Navn er primærnøkkelen til tabellen

Tabellen er på 4NF. Vi bruker Navn som naturlig nøkkel, ettersom vi ikke kan ha forskjellige foredlingsmetoder med like navn.

- **DyrketKaffebønne** (Art, *GårdID*, *KaffepartiID*)
 - Art og GårdID og KaffepartiID er primærnøkkelen til tabellen
 - Art er fremmednøkkel til **Kaffebønne**
 - GårdID er fremmednøkkel til **Gård**
 - KaffepartiID er fremmednøkkel til **Kaffeparti**

Tabellen er på 4NF, ettersom alle kolonnene i tabellen er kandidatnøkkelen. Vi har dermed kun trivielle funksjonelle avhengigheter, og ingen flerverdi funksjonelle avhengigheter.

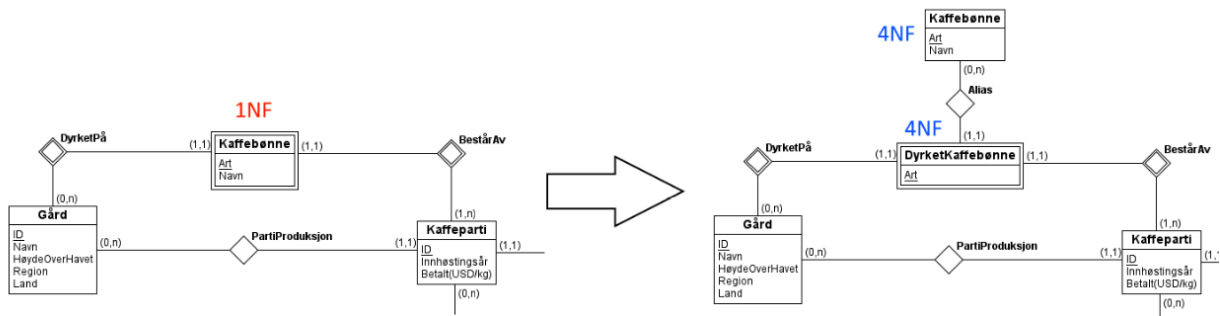
- **Kaffebønne**(Art, *Navn*)
 - Art er primærnøkkelen til tabellen

- Navn er påkrevd

Tabellen er på 4NF, ettersom tabellen er på normalform BCNF, og vi kun har to kolonner.

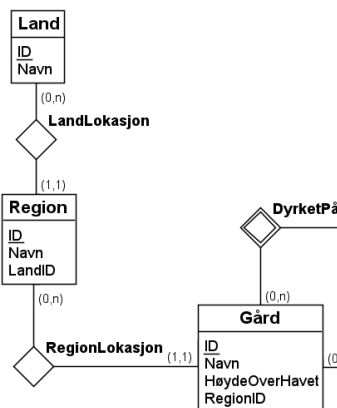
Grunnen til at DyrketKaffebønne og Kaffebønne er to separate tabeller, er at

DyrketKaffebønne ville vært på normalform 1NF dersom Navn-attributtet til Kaffebønne hadde vært med.



- **Gård** (ID, Navn, HøydeOverHavet, Region, Land)
 - ID er primærnøkkelen til tabellen
 - HøydeOverHavet, Region og Land er påkrevd

Tabellen er på 2NF ettersom tabellen er på 1NF, og at det ikke finnes delvis avhengigheter, ettersom ID er nøkkelen. Grunnen til at tabellen ikke er i 3NF, skyldes at det er en transitiv avhengighet mellom Region og Land. Navnet på en Region er avhengig av landet, og dersom for eksempel navnet på en region skulle endre navn (for eksempel Akershus -> Viken), måtte vi endre det for alle radene. Vi kunne ha valgt å splitte dette opp i tre tabeller, som vist under:



Land(ID, Navn),

Region(ID, Navn, LandID),

Gård(ID, Navn,

Her er LandID er fremmednøkkel til Land, og RegionID er

fremmednøkkel til Region. På denne måten kunne vi fått tabellen på normalform 4NF.

Likevel tenker vi dette er en litt ekstrem implementasjon for denne applikasjonen. Vi kan anta at det hender svært sjelden at en region endrer navn, i tillegg til at det ikke er mulig at en gård flytter seg til et annet land. Å ha tabellen på 2NF-formen gjør databasen enklere å arbeide med, samtidig som det krever færre spørreoperasjoner ved bruk. Dermed mener vi at de praktiske fordelene med formen den allerede er på, veier opp for ulempene og restriksjonene som kan forekomme i svært sjeldne tilfeller.

Nevneverdige restriksjoner:

- Gårder kan ikke bytte Region eller Land etter registrering.
- En DyrketKaffebønne må være med i et Kaffeparti, og være registrert dyrket på en Gård for å være med i databasen.
- En bruker kan ikke legge inn flere anmeldelser av den samme kaffen.
- For at en bruker skal kunne finne en instans av FerdigbrentKaffe basert kun på Navnet til kaffen og kaffebrenneriet det er brent på, må vi bruke dette som nøkkel fremfor ID. Derfor antar vi at brukeren velger den nyeste ferdigbrente kaffen (med tanke på Dato attributtet) i systemet, hvis ikke annet er oppgitt fra bruker-input.

3. Redegjørelse for brukerhistorier

Brukerhistorie 1)

- Bruker skriver inn **Navn** på kaffen: "Vinterkaffe 2022", "Jacobsen & Svart" som **Navn** på kaffebrenneriet, og "Trondheim" som **Lokasjon**. DBMS finner den siste FerdigbrentKaffe-entiteten i databasen som har navnet "Vinterkaffe 2022", og har **KaffebrenneriID** til en Kaffebrenneri-entitet som tilsvarer opplysningene for Kaffebrennerinavn og lokasjon (ved bruk av Kaffebrenneri-tabellen). **ID** til denne FerdigbrentKaffe-entiteten blir lagret som KaffeID i Kaffesmaking-tabellen.
- Bruker skriver inn 10 som antall **poeng**, og skriver et **notat**. "10" blir lagret som **Poeng**, og "Wow - en odysse for smaksløkene: sitrusskall, melkesjokolade, aprikos!" blir lagret som **Smaksnotat** i Kaffesmaking-tabellen. (Epost for den innloggede brukeren, samt ID for FerdigbrentKaffe blir automatisk lagt inn i **Kaffesmaking**-tabellen).

- FerdigbrentKaffe-entiteten har “lysbrent” som **Brenningsgrad**, og 20.01.2022 som **Dato**. 600kr er registrert som **Kilopris**. «En velsmakende og kompleks kaffe for mørketiden» er registrert som **Beskrivelse**.
- Ved å hente Kaffeparti-entiteten som er referert til i FerdigbrentKaffe, finner vi at **Innhøstningsåret** var 2021, at kaffepartiet har bærtørket som **Foredlingsmetode** og at gården fikk utbetalt 8 USD/KG kaffe. (ved **Betalt(USD/kg)**-attributtet).
- Vi finner ut at **Navn** på gården er “Nombre de Dios” og at **HøydeOverHavet** er 1500(moh). Disse opplysningene får vi ved å benytte oss av **GårdID** fremmednøkkelen i Kaffeparti.
- Vi finner ut at Kaffebønnene som ble brukt er av **Art** “Coffea Arabica”. Dette finner vi ut ved å slå opp i Kaffebønne-tabellen, og finne alle rader der **ID**en til Kaffeparti-entiteten samsvarer med **KaffepartiID**-fremmednøkkelen. Vi ser også at **Navn** på kaffebønnen er oppført som Bourbon.

Brukerhistorie 2)

- Brukeren bruker “toppliste” kommandoen i appen.
- En ny tabell blir laget. Vi teller alle forekomster av like **BrukerID**-er i Kaffesmaking tabellen. Siden brukere ikke kan legge in flere anmeldelser av samme kaffe, er alle kaffesmakingene på unike typer kaffe. **BrukerID** og Antall kaffe smakt blir lagt til i den nye tabellen for hver bruker.
- Den nye tabellen blir sortert synkende, basert på Antall Kaffe smakt.
- Den sorterte tabellen blir slått sammen med Bruker-tabellen med **BrukerID**, og **Fornavn** og **Etternavn** blir lagt til, og slått sammen til Fullt Navn. Denne tabellen blir skrevet ut til brukeren.

Brukerhistorie 3)

- Bruker velger å vise frem alle **FerdigbrentKaffe** og sorterer listen basert på gjennomsnittscore, som er summen av **Antall poeng** av alle som smakt den kaffen, delt på antallet som har smakt på kaffen. Listen blir sortert i synkende rekkefølge.
- Listen henter navnet på kaffebrenneriet fra **Kaffebrenneri**, ifølge **BrentPå** relasjonen.

- Listen henter navnet på kaffen fra **Navn** attributten til **FerdigbrentKaffe**, og bruker videre **Kilopris** fra samme tabell.
- Gjennomsnittscore henter den fra regnestykket som den bruker for å sortere listen.

Brukerhistorie 4)

- Bruker henter ut alle **FerdigbrentKaffe** og søker på “floral”. Listen blir oppdatert med entiteter hvor enten **Beskrivelse** inneholder ordet “floral”, eller hvor en bruker har skrevet **Smaksnotater** som inneholder ordet "floral".
- Listen returnerer **Navn** attributten i **FerdigbrentKaffe**, samt **Navn** på **Kaffebrenneri** til **FerdigbrentKaffe**.

Brukerhistorie 5)

- Bruker henter ut alle **FerdigbrentKaffe** hvor **Kaffeparti** kommer fra en **Gård** hvor **Land** er lik “Rwanda” eller “Colombia”.
- Videre skal **Kaffeparti** være foredlet med en **Foredlingsmetode** hvor **Navn** ikke er “Vaskede”
- Listen skal da gi tilbake **Navn** på **FerdigbrentKaffe** og **Navn** på **Kaffebrenneri** til **FerdigbrentKaffe**