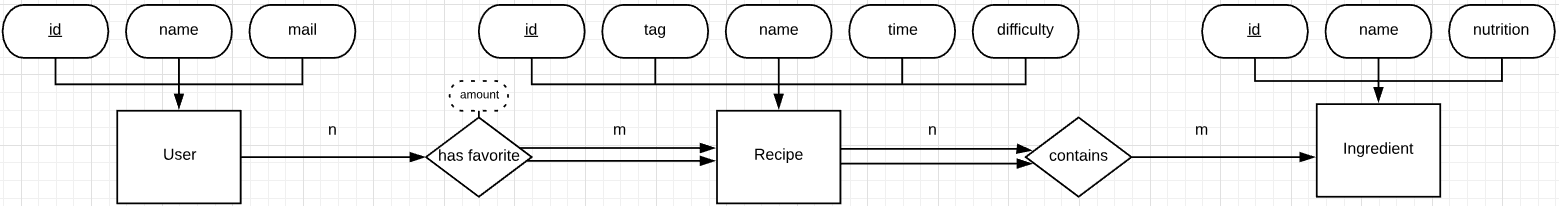
# **Arbetsrapport**

För att lösa denna uppgift skapades först ett ER-diagram som fungerar som ett konceptuellt schema för databasen. Två timmars arbete spenderades för att fundera ut hur schemat skulle se ut och för att skapa det. En dåligt uppbyggd databas har stor chans att ge varaktiga svårigheter, därför bestämde jag mig tidigt för att lägga en stor vikt i planering.

Diagrammet består av tre olika entitetstyper och två sambandstyper. Mellan sambandstyperna och entitetstyperna finns många-till-många-samband. Detta för att göra det möjligt för användare att ha flera olika favoritrecept och att recepten kan vara favoriter av flera olika användare, samt att recepten ska kunna innehålla flera olika ingredienser och ingredienserna kan användas i flera olika recept. De olika entitetstyperna har en mängd olika attribut, varav ID-nummer fungerar som unika värden.

ED-diagram:



Efter att diagrammet konceptualiserats skapades tabellerna i SQL. Först skapades entitetstyperna och tillhörande attribut, och sedan två kopplingstabeller för att para ihop användare och recept samt recepten och dess ingredienser. Arbetet gick snabbt eftersom tanken bakom tabellerna redan var genomtänkt. Det tog ungefär en timme totalt att skapa tabellerna och att mata in data som krävdes för att kunna utföra de olika queries som efterfrågades.

**Queries**

Nedan följer förklaring på de olika queries:   
1. För att lista recept som innehåller en specifik ingrediens (i detta fall tomat) användes select name från recept-tabellen, sedan användes join för kopplingstabellen ”recipeIngredient” där recept-id är samma som kopplingstabellens-recept-id och där kopplingstabellens ingrediens-id är tomatens id.

2. För att lista de olika recepten och hur många ingredienser dem innehåller användes select name och count(\*) från recept-tabellen. Sedan användes återigen join för recipeIngredient där recept-id är samma som kopplingstabellens recept-id. Detta grupperades med avseende på recept-id och sorterades så att receptet med flest antal ingredienser hamnade längst upp.

3. En lista på recept som har vinter som tagg togs fram med hjälp av select name från recept-tabellen där attributen ”tag” är lika med ”winter”.

4. Sedan listades de tre användare som har flest antal favoritmarkerade recept genom select name och count(\*) från användar-tabellen. Sedan användes join för favorit-kopplingstabellen där användar-id är samma som kopplingstabellens användar-id. Detta grupperades med avseende på användar-id, sorterades baserat på antal favoritmarkeringar och begränsades så bara tre användare listades. Antalet favoritmarkeringar finns med i diagrammet som ett härlett attribut. Alltså ett attribut som inte är med i någon tabell, utan kan räknas ut med hjälp av tabellerna.

Det tog en timme för att formulera ovanstående queries i databashanteraren.

**Reflektion**

Jag fann det enbart positivt att lägga mycket vikt vid planeringen av databasen. I detta projekt fokuserade jag extra på planering, just för att jag lagt märke till en dålig planerings påverkan i tidigare arbeten. I de fall där det behövs göra en större förändring p.g.a. oduglig planering kan man behöva ändra på väldigt mycket i databasen för att korrigera det felaktiga.

Jag använde mig av det digitala verktyget *Lucidchart* för att skapa mitt ER-diagram i detta projekt. Verktyget har all funktionalitet jag behöver för att kunna skapa och konceptualisera ett schema för en databas. För att skapa själva databasen använde jag mig av *Khan Academy*, som jag också anser har den funktionalitet jag finner nödvändig för mitt arbete.

Jag tyckte att projektet var intressant och hade en intressant uppgift att lösa. Projekt som detta, där jag arbetar för en fiktiv kund är något jag upplever är både roligt och användbart för att kunna tänka sig in i en utvecklar-situation. Det är ett arbetssätt jag tror är bra för framtida arbete med databaser.

Länk till databasen: <https://www.khanacademy.org/computer-programming/recipes/4675267192061952>

Länk till Diagram: https://www.lucidchart.com/invitations/accept/95e2dbe0-c9eb-45d5-9755-07f1b7e27a0d