



Data Modell för Yrkesco

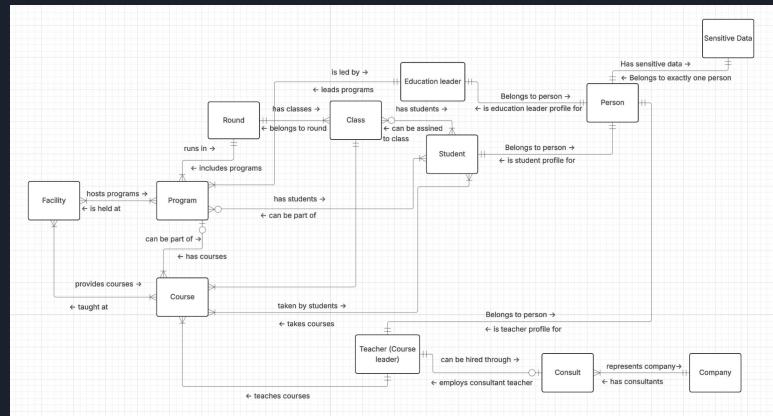
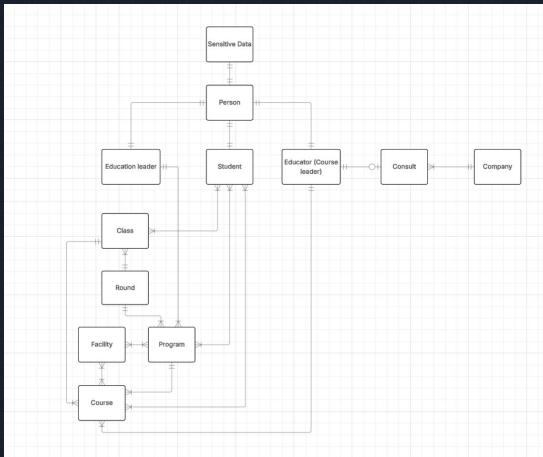
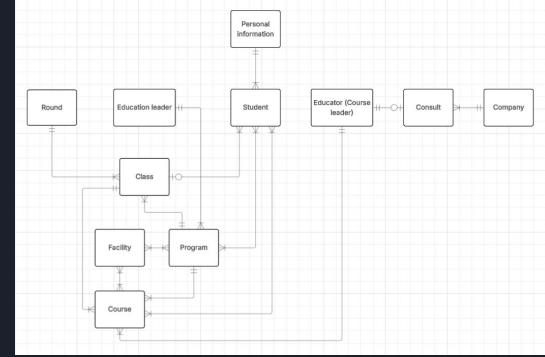
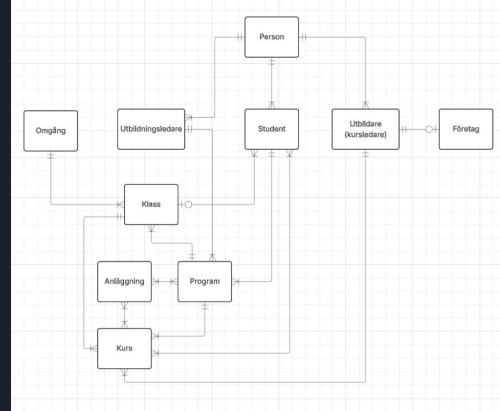
Sammanfattning av byggnation av datamodell



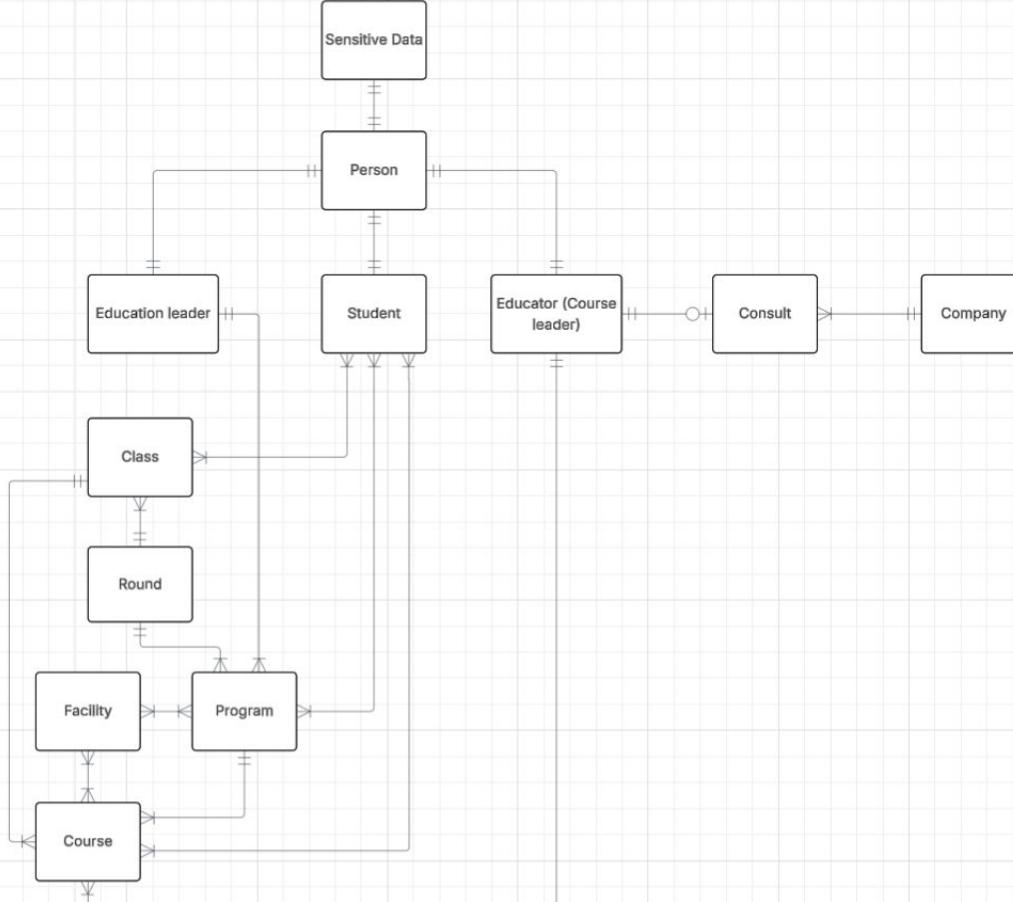
Kravspecifikationen

- Jag har tittat på kravspecifikationen samt QA med Elvin för att få en tydlig bild av hur datamodellen bör se ut
- Utifrån det har jag skapat ett dokument som jag har utgått ifrån när jag skapat den konceptuella modellen
- Jag har gjort en del små ändringar för att det ska bli en så bra modell som möjligt samtidigt som jag behåller integriteten av kravspecifikationen.
- Jag har också lagt till några saker som jag tycker bör finnas med för att datamodellen ska bli så komplett som möjligt

Konceptuell Modell utveckling







Relationship statements

Person

- En person kan ha en eller flera roller: student, utbildare och/eller utbildningsledare.
- Varje student, utbildare och utbildningsledare är exakt en person.

Student

- En student är inskriven i ett eller inget program.
Fristående kurser organiseras under standalone i kurser.
- En student kan vara en del av en klass.
- En student kan gå på en eller flera kurser.

Utbildningsledare

- En utbildningsledare ansvarar för exakt två program.
- Varje program ansvaras av en utbildningsledare.

Utbildare (kursledare)

- En utbildare kan undervisa i en eller flera kurser.
- En utbildare kan vara kopplad till ett företag.

Klass

- En klass består av flera studenter.
- En klass genomför flera kurser.
- En klass tillhör exakt ett program och en omgång.

Program

- Ett program består av flera klasser.
- Ett program består av flera kurser.

Anläggning

- En anläggning erbjuder ett eller flera program.
- En anläggning erbjuder en eller flera kurser.

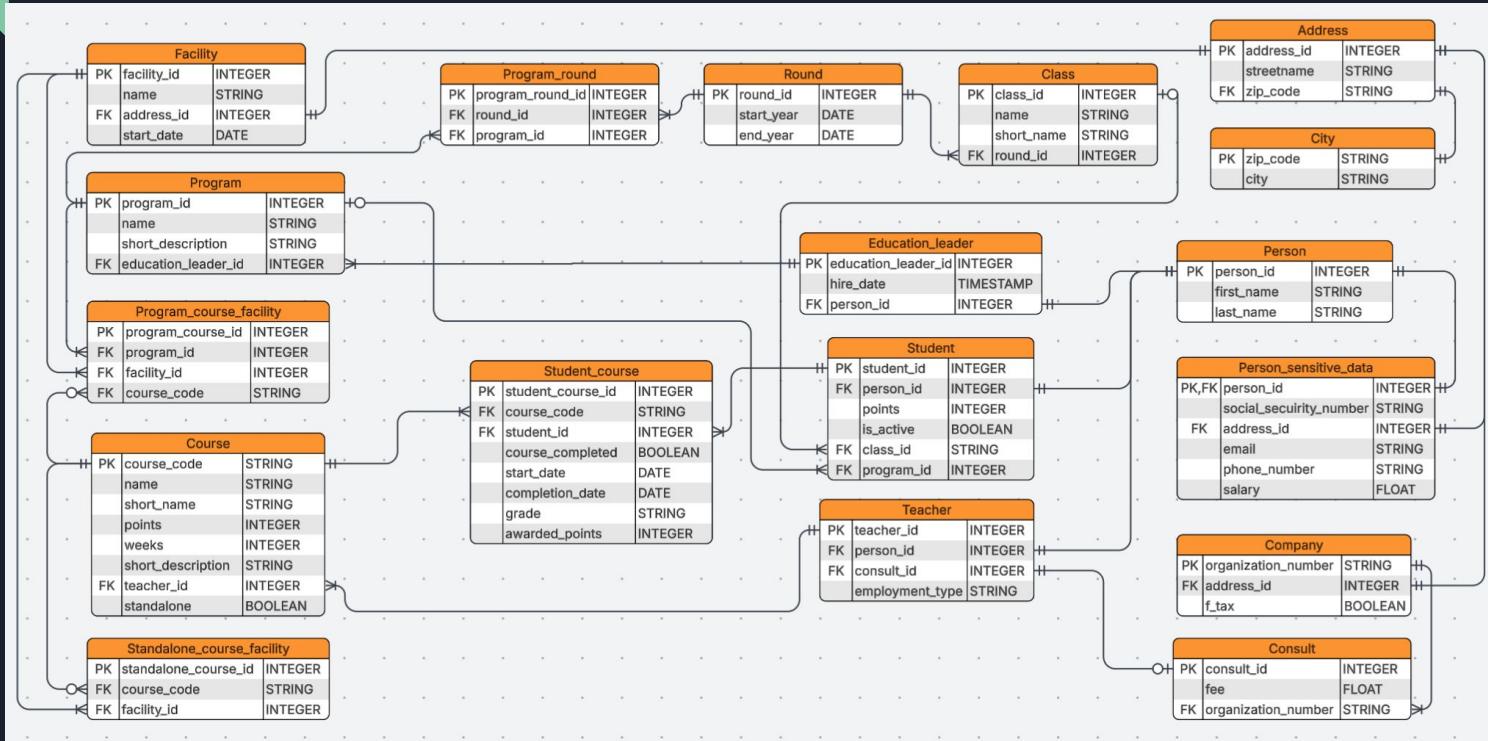
Kurs

- En kurs kan tillhöra ett program eller vara fristående.
- En kurs kan ha flera studenter.
- En kurs har exakt en utbildare.

Omgång

- En omgång kan ha en eller flera klasser.
- En omgång kan ha ett eller flera program.

Logiska Modellen





Fysiska modellen

- DB diagram
- Relativt smärtfritt med en bra logisk modell
- Modifieringar efter att den skapats
- Constraints
- Function och triggers



Fake data och queries för att testa

Test för att se så att databasen, funktioner och triggers fungerar som tänkt.

- Vi har generat fake data med LLM för att fylla alla tables med data.
- Vi har gjort ett antal queries med olika joins för att säkerställa att allt fungerar som tänkt.



Testa constraints

Varför har vi constraints?

- För att säkerställa att det inte går att databasen inte blir korrumperad med felaktig information så testar vi att mata in incorrect data i olika tables.
- Detta för att minimera att vi fyller databasen med ogiltig data.

Exempel

- När en student klarar en kurs så ska den inte kunna ge minus poäng.
- Ett personnummer ska vara korrekt (inga bosktäver, 12 siffror, inga bindestreck i databasen det görs via ett ui senare om så önskas).



Eventuella förbättringar

- Koppling mellan Class -> Round -> Program.
- Lägga till administrativa tjänster och personal.
- Standalone_course_facility överflödig?
- Lägga till flera funktioner om så önskas.