

# **ProjIO**

Av: Kristoffer Lind

# Innehållsförteckning

ProjIO.....	1
1. Problembeskrivning.....	3
2. Konceptuell modell med tabellprecisering.....	4
3. Fysisk Modell med tabellprecisering.....	5
4. Exempeldata.....	6
5. Mockup Formulär – Funktionalitet – Validering.....	7
5.1 Skapa Projekt.....	7
Validering.....	7
Påverkade fält.....	7
5.2 Lägg till uppgift.....	8
Validering.....	8
Påverkade fält.....	8
5.3 Tilldela uppgift.....	9
Validering.....	9
Påverkade fält.....	9
5.4 Ändra status på uppgift/Sätt uppgift till Avslutad.....	10
Validering.....	10
6. Rapporter/Utskrifter.....	11
6.1 Visa Personer.....	11
6.2 Visa Projekt.....	11
6.3 Visa uppgifter.....	11
6.4 Mina Uppgifter.....	12
6.5 Avslutade Uppgifter.....	12
7. MS SQL Diagram från databasen.....	13
8. Lagrade Procedurer.....	14
9. Referentiell Integritet.....	15
10. Volymberäkning.....	16
11. Historik.....	17
12. Prestanda.....	18
13. Miljökrav (IT-Miljö).....	18
14. Installation.....	18
15. Manualer.....	18
16. Rättigheter.....	18
17. Sammanfattning.....	19

# 1. Problembeskrivning

För egen räkning tänker jag bygga uppgiftshanteringen i ett projekthanteringssystem. Hela projekthanteringssystemet skulle vara för ambitiöst att bygga under en 3 veckors period, men jag skulle ganska lätt kunna plocka in fler delar om det här går för fort att implementera. Systemet ska kunna användas av både små och stora organisationer/företag eller användargrupper, varje användare ska kunna ha flera projekt.

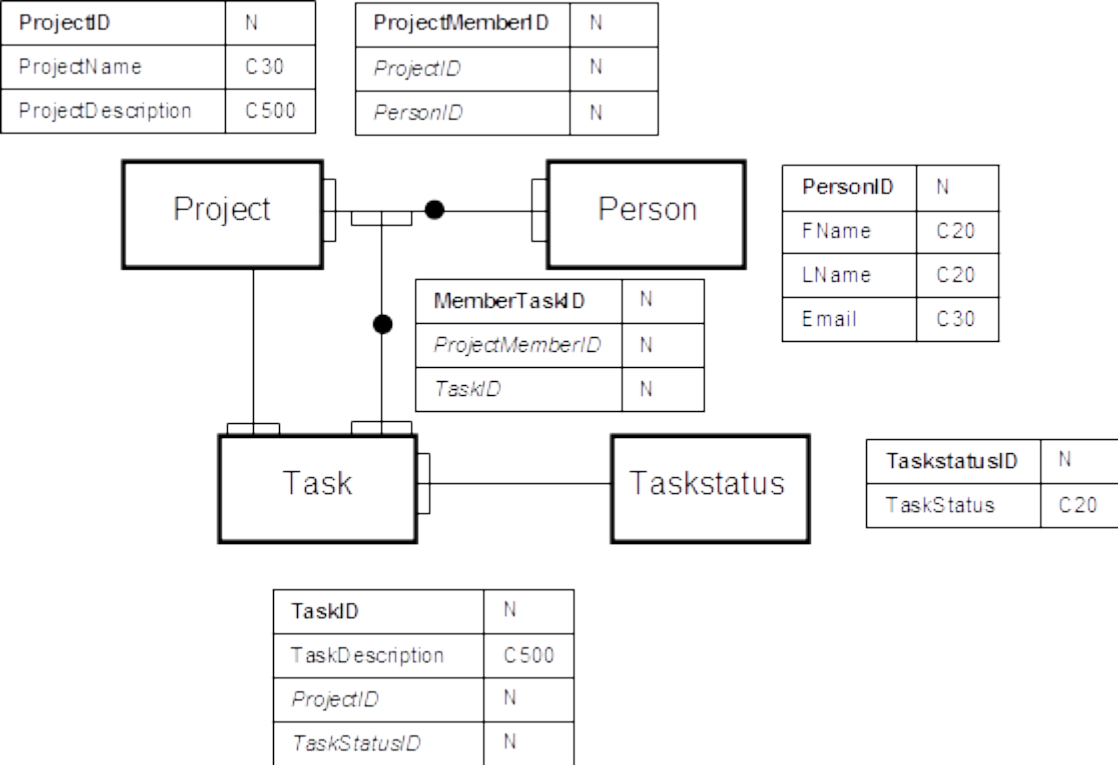
- Systemet ska vara ett hjälpmedel för att hålla ordning på vilka uppgifter som behöver utföras i ett projekt, samt vilka medlemmar som tilldelats dessa uppgifter.
- Systemet ska kunna hantera flera projekt, samt hålla ordning på vilka personer som tillhör vardera projekt.

	14-02-12	14-02-18	14-02-26	14-03-05	14-03-12	14-03-19
Problembeskrivning	x					
AKS steg 1	x	x				
AKS slutlig			x	x		
Databas implementerad				x		
Gränssnitt				x	x	
Testning					x	x
Redovisning						x

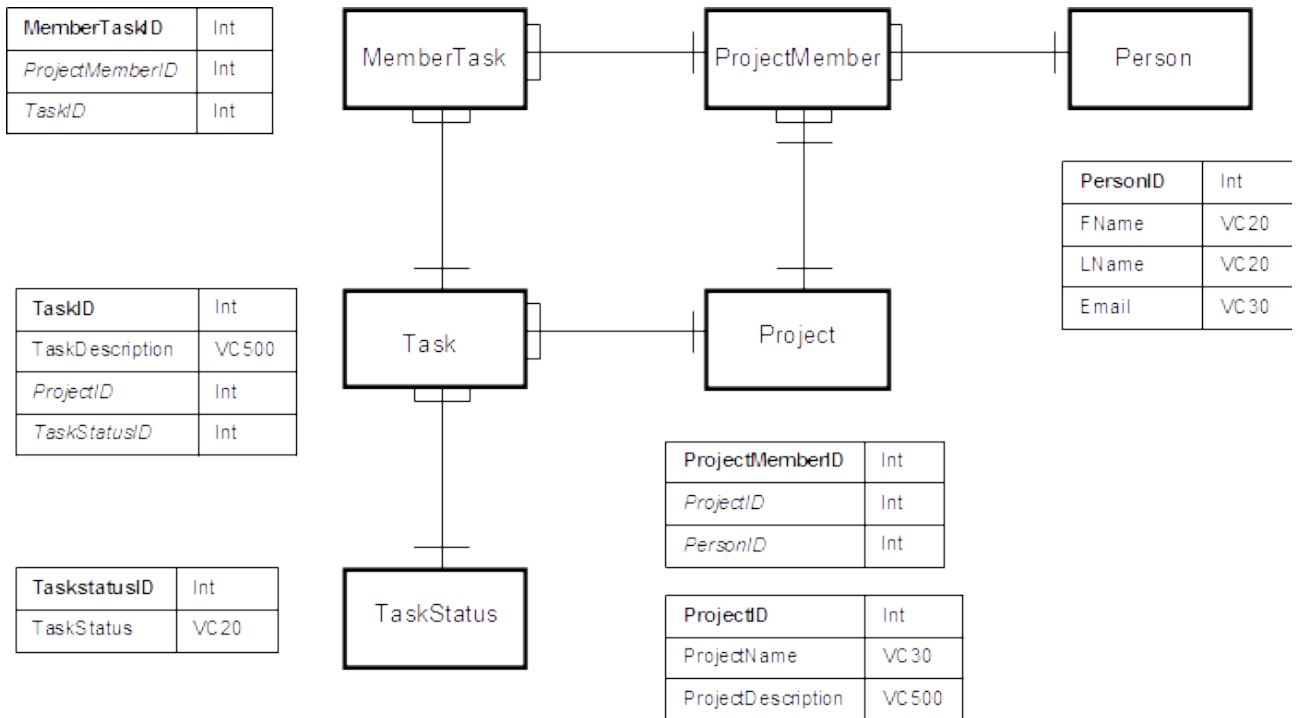
En mer utförlig beskrivning av systemet i sin helhet finns på:

<https://github.com/kl222jy/Uppgift-234> (denna dokumentation är inte färdigställd, men förhoppningen är att hinna färdigställa även denna innan kursens slut)

## 2. Konseptuell modell med tabellprecisering



### 3. Fysisk Modell med tabellprecisering



## 4. Exempeldata

MemberTask

MemberTaskID	ProjectMemberID	TaskID
1	1	2
2	2	3
3	3	3

ProjectMember

ProjectMemberID	ProjectID	PersonID
1	2	1
2	2	2
3	3	3

TaskID	TaskDescription	ProjectID	TaskStatusID
1	Design rev 1	2	3
2	Design rev 2	2	2
3	Implementera databas för recept	3	2

TaskStatus

TaskStatusID	TaskStatus
1	Ej påbörjad
2	Pågående
3	Avslutad

Project

ProjectID	ProjectName	ProjectDescription
1	ProjIO	Projekthanteringssystem
2	Sportbutiken.se	E-butik för sportartiklar
3	Matlagning.nu	Mattips på nätet

PersonID	Fname	Lname	Email
1	Anders	Andersson	<a href="mailto:a.andersson@test.com">a.andersson@test.com</a>
2	Erik	Eriksson	<a href="mailto:e.eriksson@test.com">e.eriksson@test.com</a>
3	Per	Persson	<a href="mailto:p.persson@test.com">p.persson@test.com</a>

## 5. Mockup Formulär – Funktionalitet – Validering

### 5.1 Skapa Projekt

[TaskManager](#) [My Projects](#) [Tasks](#) [Current Tasks](#) [Finished Tasks](#) [Create Task](#) [About](#)

### My Projects

Click on a project to activate it, by default your newest project is active.

You are now logged in. X

Name	Description	Edit
<input type="text"/>	<input type="text"/>	<a href="#">Spara</a> <a href="#">Avbryt</a>
<a href="#">Proj10</a>	Ett grundläggande projekthanteringssystem.	<a href="#">Edit</a> <a href="#">Delete</a>
<a href="#">testproject</a>	testdescription	<a href="#">Edit</a> <a href="#">Delete</a>

Skapande av nytt projekt.

Vid skapande av nytt projekt ska projektnamn och projektbeskrivning väljas. När projektet skapas ska personen som skapat det sättas in i projectmember med koppling till projektet och sättas till projektledare.

### Validering

Fält	Validering
ProjectID	Räknare, PK, Unik
ProjectName	Får inte vara null, max 20 tecken
ProjectDescription	Default=""-", max 500 tecken
ProjectLeader	Måste finnas som ProjectMemberID

### Påverkade fält

Fält	Validering
ProjectMemberID	Räknare, PK, Unikt
ProjectID	FK
PersonID	FK

## 5.2 Lägg till uppgift

[TaskManager](#) [My Projects](#) [Tasks](#) [Current Tasks](#) [Finished Tasks](#) [Create Task](#) [About](#)

### Create Tasks

**Task Description**

Create

Vyn för att lägga till en uppgift hanterar enbart tabellen Task.

Användaren väljer beskrivning av uppgiften och om den tillhör någon av projekten denna är medlem i, eller om det ska vara en allmän uppgift.

### Validering

Fält	Validering
TaskID	Räknare, PK, Unik
TaskDescription	Får inte vara null, max 500 tecken
ProjectID	FK, projektmedlemmen måste tillhöra projektet
TaskStatusID	FK

### Påverkade fält

Inga fält påverkas, ProjectMember måste dock anropas.



## 5.3 Tilldela uppgift

TaskManager   My Projects   Tasks   Current Tasks   Finished Tasks   Create Task   About

### Tasks

List of all tasks that need to be done.

- [test](#)

Vyn presenterar användaren med en lista över aktuellt projekts uppgifter där användaren kan välja att börja arbeta med en uppgift. Då användaren väljer att arbeta med en uppgift skapas en post i member task och status på uppgiften sätts till pågående.

### Validering

Fält	Validering
MemberTaskID	Räknare, PK, Unik
ProjectMemberID	FK
TaskID	FK

### Påverkade fält

Fält	Validering
TaskStatusID	FK

## 5.4 Ändra status på uppgift/Sätt uppgift till Avslutad

[TaskManager](#) [My Projects](#) [Tasks](#) [Current Tasks](#) [Finished Tasks](#) [Create Task](#) [About](#)

### My Tasks

List of all tasks i'm currently working on.

- [test3](#)

För ändra status gäller enbart att TaskStatusID påverkas, validering=FK. Uppgiftstilldelningen i membertask behålls, för att kunna hålla reda på vilka användare som arbetade med en uppgift när den markerades som klar. Dessa är troligen mest lämpade att fortsätta arbete med uppgiften om vidare arbete krävs på uppgiften.

### Validering

Fält	Validering
TaskStatusID	FK

## 6. Rapporter/Utskrifter

Här hittar du all utdata som används i gränssnittet.

### 6.1 Visa Personer

Syfte: Lista med person används som en förenklad inloggning, användaren klickar på en person och blir då inloggad som denna användare.

[TaskManager](#) [My Projects](#) [Tasks](#) [Current Tasks](#) [Finished Tasks](#) [Create Task](#) [About](#)

### TaskManager

Choose a user to be logged in as (default setup is Kristoffer Lind and ProjO).

Name	Email
<a href="#">Anders Andersson</a>	<a href="#">a.andersson@test.com</a>
<a href="#">Erik Eriksson</a>	<a href="#">e.eriksson@test.com</a>
<a href="#">Per Persson</a>	<a href="#">p.persson@test.com</a>
<a href="#">Kristoffer Lind</a>	<a href="#">kristoffer@krad.se</a>

### 6.2 Visa Projekt

Syfte: Lista med projekt visas, val av aktivt projekt görs genom att klicka på ett projekt. Det går här också att skapa nya, ändra och ta bort projekt.

[TaskManager](#) [My Projects](#) [Tasks](#) [Current Tasks](#) [Finished Tasks](#) [Create Task](#) [About](#)

### My Projects

Click on a project to activate it, by default your newest project is active.

You are now logged in. X

Name	Description	Edit
<input type="text"/>	<input type="text"/>	<a href="#">Spara</a> <a href="#">Avbryt</a>
<a href="#">ProjO</a>	Ett grundläggande projekthanteringssystem.	<a href="#">Edit</a> <a href="#">Delete</a>
<a href="#">testproject</a>	testdescription	<a href="#">Edit</a> <a href="#">Delete</a>

### 6.3 Visa uppgifter

Syfte: Lista med uppgifter som varken är klara eller under arbete i det aktiva projektet. Det går här också att ta bort, ändra och välja att börja arbeta med en uppgift.

[TaskManager](#) [My Projects](#) [Tasks](#) [Current Tasks](#) [Finished Tasks](#) [Create Task](#) [About](#)

### Tasks

List of all tasks that need to be done.

- [test](#)

## 6.4 Mina Uppgifter

Syfte: Lista med uppgifter som den inloggade användaren arbetar med i det aktiva projektet.

[TaskManager](#) [My Projects](#) [Tasks](#) [Current Tasks](#) [Finished Tasks](#) [Create Task](#) [About](#)

### My Tasks

List of all tasks i'm currently working on.

- [test3](#)

## 6.5 Avslutade Uppgifter

Syfte: Visar lista med avslutade uppgifter i det aktiva projektet. Det går på denna sida att sätta tillbaka en uppgift i arbete, de som arbetade med uppgiften då den markerades som avslutad får du upp din under sina aktiva uppgifter.

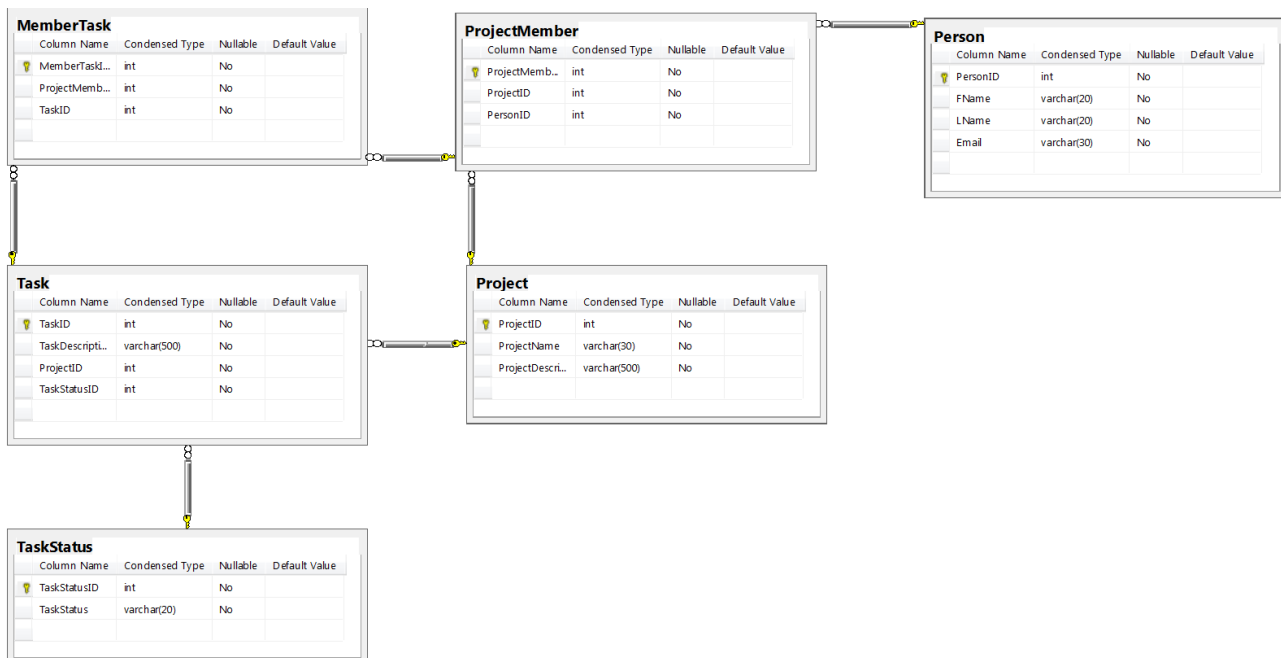
[TaskManager](#) [My Projects](#) [Tasks](#) [Current Tasks](#) [Finished Tasks](#) [Create Task](#) [About](#)

### Done Tasks

List of all tasks that are finished.

- Implement basic functionality.
- test
- test2

## 7. MS SQL Diagram från databasen



## 8. Lagrade Procedurer

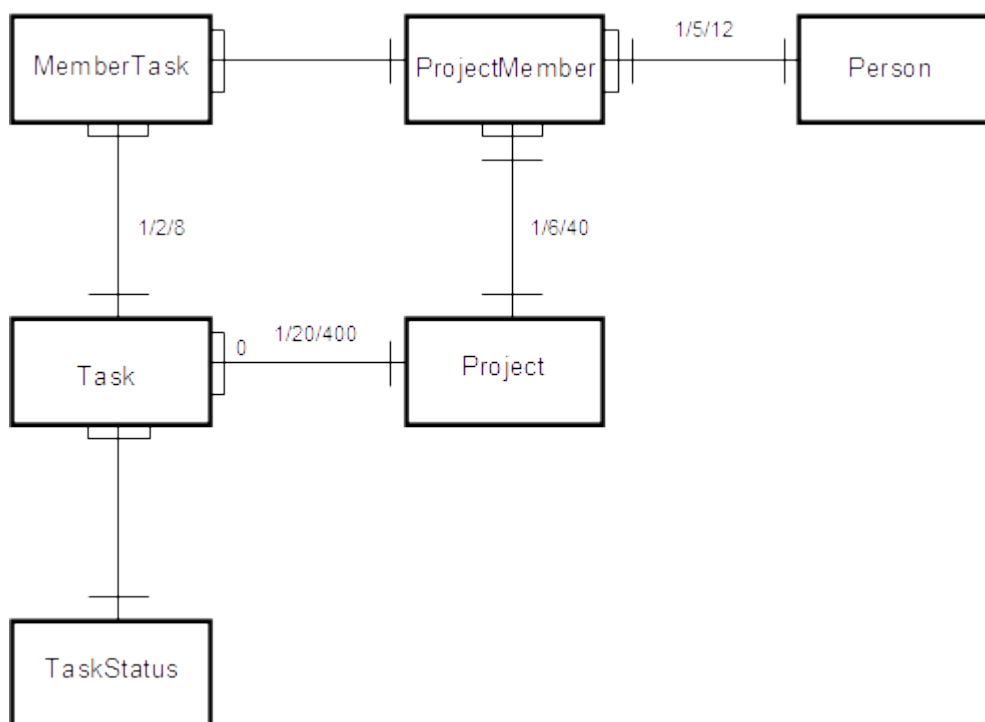
I nuläget finns det inga lagrade procedurer i databasen.

Namn	Användningsområde
usp_deleteTask	Tar bort en uppgift
usp_doneTask	Markera en uppgift som klar
usp_getDoneTasks	Hämtar uppgifter som är klara
usp_getTaskById	Hämtar uppgift med ett specifikt id
usp_getTasks	Hämta lista med uppgifter
usp_getUserTasks	Hämta lista med uppgifter för en specifik användare
usp_joinTask	Börja arbeta på en uppgift
usp_newTask	Skapar en ny uppgift
usp_updateTask	Uppdaterar uppgiftsbeskrivning
usp_getAllUsers	Hämtar samtliga användare
usp_getTaskUsers	Hämta lista med användare som arbetar med en viss uppgift
usp_createProject	Skapar ett projekt och kopplar inloggad användare som medlem
usp_deleteProject	Tar bort projekt
usp_getProjects	Hämtar lista över projekt som användaren tillhör
usp_updateProject	Uppdatera projekt

## 9. Referentiell Integritet

Relation	Relationstyp	RI	Delete	Update
Person->ProjectMember	1:N	x	Cascade	No Action
ProjectMember->MemberTask	1:N	x	Cascade	No Action
Task->MemberTask	1:N	x	Cascade	No Action
Project->ProjectMember	1:N	x	Cascade	No Action
Project->Task	1:N	x	Cascade	No Action
TaskStatus->Task	1:N	x	No Action	No Action

## 10. Volymberäkning

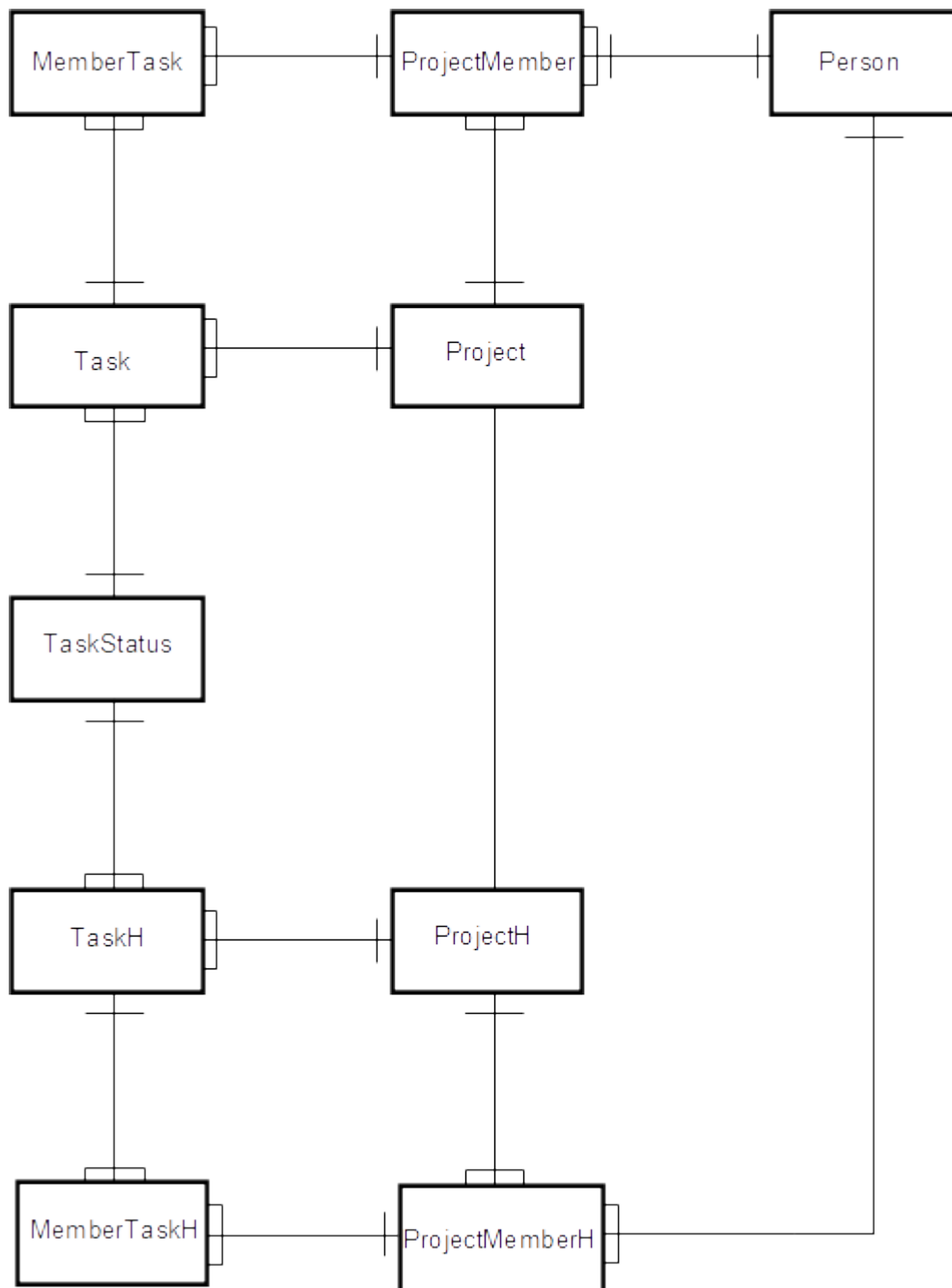


Utgångspunkt (antal projekt): 100  
 Ökning per år (antal projekt): 100

Storlek	Tabell	Storlek	År 1	År 2	År 3	År 4	År 5	År 1	År 5
4-1064	Project	300	100	200	300	400	500	30 000	150 000
4-144	Person	110	120	240	360	480	600	13 200	66 000
12	MemberTask	12	4 000	12 800	19 200	25 600	32 000	48 000	384 000
12	ProjectMember	12	600	1 200	1 800	2 400	3 000	7 200	36 000
12-1012	Task	212	2 000	4 000	6 000	8 000	10 000	424 000	2 120 000
4-44	TaskStatus	44	3	3	3	3	3	132	132
	Totalt (b):	690	6 823	18 443	27 663	36 883	46 103	522 532	2 756 132
	Totalt (kb):	0,67	6,66	18,01	27,01	36,02	45,02	510,29	2 691,54
	Totalt (mb):	0,00	0,01	0,02	0,03	0,04	0,04	0,50	2,63
	Totalt (gb):	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00



## 11. Historik



Färdiga projekt skulle kunna läggas i historiska tabeller. Tabellen för personuppgifter lämnas orörd.

## **12. Prestanda**

Databasens volym beräknas inte ge några prestandaproblem. Avslutade projekt förflyttas till egna tabeller, dessa skulle kunna förflyttas till en annan server med lägre prestanda.

## **13. Miljökrav (IT-Miljö)**

Databasen är utvecklad i MS SQL 2008, applikationen är utförd med ASP.NET WebForms.

## **14. Installation**

Applikation körs från webbplats. För egen installation krävs IIS och MS SQL 2008 Server.

## **15. Manualer**

Det finns i nuläget inga manualer till applikationen, framtida hjälpavsnitt kommer att dyka upp i gränssnittet.

## **16. Rättigheter**

Applikationen är fri att använda, men får inte säljas vidare i någon form.

## 17. Sammanfattning

Applikationen som helhet är begränsad till den gräns att den är närmast oanvändbar, systemet behöver utökas väsentligt för att faktiskt komma till användning. För den del av systemet det faktiskt fokuserats på (uppgiftshanteringen) tycker jag dock att det fungerar bra.

Jag skulle egentligen vilja att gränssnittet var betydligt mer interaktivt (exempelvis drag&drop för uppgiftshantering), men då skulle jag behöva lägga väldigt mycket energi på att få javascript och webforms att arbeta i symbios. Känslan jag har nu är att WebForms lämnar en hel del att önska för gränssnittsutveckling. ASP.NET tillsammans med MSSQL känns dock väldigt stabilt som backend.

Planen med att göra just detta som projekt var att få klarhet i fler saker inför att bygga det egentliga systemet i sin helhet. Detta har gått bra, jag har modellerat systemet i helhet och har väldigt mycket klart för mig inför projektet i nästa kurs.

För framtida utveckling kan jag ta med mig dessa tankegångar och till viss del datamodellen. Både MSSQL och MySQL är väldigt bra och jag kommer troligen använda mig av dem fler gånger, men under vidareutvecklingen tänkte jag prova Angular + Firebase under nästa kurs och sedan MEAN stack under en sommarkurs.