Project Report

By Anders Baeroe, Isak Hauge, Mathias Ahrn, Thora Mothes.

**Project report created in association with our final**

**project for CST336 – Internet Programming.**

# Entity Relationship Diagram

A close up of a map

Description automatically generated

Apart from all the tables, we have 13 stored procedures, 1 stored function, and 1 event.

The most important features the database provides are as following:

* Automatically populating a “wall of shame” table called overdues.
* Automatically selecting the next “on duty”/whose turn it is.
* Resetting and keeping track of how many times each chore should be done, and has currently been done.
* General CRUD on all tables.
* Views for reports etc.

# User stories:

As a *Guest* I want to *search for collectives and their members.*

As a *Guest* I want to be able to *create an account.*

As a *User* I want to *join a collective.*

As a *User* I want to *leave a collective.*

As a *User* I want to *create a collective.*

As a *User* I want to *create a chore.*

As a *User* I want to *edit a chore*.

As an *Admin* I want to *delete a collective*.

As an *Admin* I want to *edit a collective*.

# Mockups:

Figure 1 - Mockup 1

A screenshot of a cell phone

Description automatically generated

Figure 2 - Mockup 2

A screenshot of a cell phone

Description automatically generated

Figure 3 - Mockup 3

A screenshot of a cell phone

Description automatically generated

Figure 4 - Mockup 4

A screenshot of a cell phone

Description automatically generated

Figure 5 - Mockup 5

A screenshot of a cell phone

Description automatically generated

# Screenshots:

Landing Page:A screenshot of a cell phone

Description automatically generated

This is the first thing you see. If you have an account you can chose to log in, or if you don’t have an account you can create one.

Create Account Page:

A screenshot of a cell phone

Description automatically generated

Join/Create Collective Page:

A screenshot of a cell phone

Description automatically generated

When you have logged in for the first time, you’ll have the option of either joining a pre-existing collective or creating a new one. If you want to join a pre-existing one you’ll have to be provided with a unique key by one of the members.

Collective Page:A screenshot of a cell phone

Description automatically generated

This is the general “dash” of a collective. Here you’ll get an overview of all the chores, if there are any. If not, you can add a chore easily.

Chores operate on a weekly basis, and they will reset each Monday at midnight.

Chores not completed will be added into the “Overdue”/wall of shame table.

Chores are displayed as 0/7, which means the chore is planned to be “done” 7 times, and has currently been done 0 times.

For each click on “Done” the chore will increment until completion.

Overdue Page:

This is where you can keep an eye out for people who keeps missing their chores. Each Overdue is marked with the name, chore, and time of the person who didn’t do the chore.

This page acts as a “scoreboard” of missed chores. We didn’t have time to implement this.

|  |  |  |
| --- | --- | --- |
| Description | Where | Points |
| Documentation must include 1) ERD, 2) user stories, 3) mockups and related 4) screenshots (If any of the elements is missing, you won't get any of the 20 points) | This document. | 30 |
| Project must use at least three database tables and one of the database tables must have at least 20 records. (If any of the tables or data are missing, you won't get any of the 10 points) | 20 records in the user table. | 20 |

|  |  |  |
| --- | --- | --- |
| Description | Where | Points |
| There is a "user" section in which anyone can search and filter data by at least three fields and sort the results without having to login (there must be at least one JOIN relationship between your two of your database tables to provide the searchable data) | Press the search button in the top left corner.  File:  Routes/search.js | 15 |
| Users can login and logout from the system | Press the login button. | 10 |
| Users can create new records in the database | Create user and/or collective. | 10 |
| Users can edit records in the database | Edit user in your profile page. (Can also edit collective if you add “/edit” in the url after “collective/”. | 10 |
| Users can delete records in the database | Delete your own user from the profile page | 10 |
| Users can generate at least three reports, which use aggregate functions and at least one OUTER JOIN (e.g. average price of all products in the table) | We didn’t implement this. | 15 |
| Project uses at least one 3rd party API | The profile picture API, called “TinyFac.es”. | 10 |
| Project uses only AJAX for at least one page | Public/js/page/collective/home.js  We use only Ajax for updating the page dynamically. | 10 |
| Project has a nice and consistent design | Well, yes. | 10 |
|  | TOTAL: | 135 |