# 1. General structure

## 1.1. Users

Every type of user should be registered on the website. They are:

* Students
* Parents
* Teachers
* Directors

The necessary fields are *during registration* available in order to select which type the user is.

## 1.2. Ruimte

Ruimte is created for each student. The aim of this is the ability to create a special space for each student to follow up their relationship with UCCIB and their school.

The users for that Ruimte should be (a) teachers who are giving lessons for the student, (b) his/her parents, (c) the student himself and (d) the director who created that Ruimte (who is by default the admin of the Ruimte). (a) should also be assigned as admin.

# 2. Modules

In each Ruimte, there are different modules which can be activated. This can be anytime updated if desired. The following modules describe which/how the modules are used

## 2.1. Stream

This is the default module when ruimte is visited, which is comparable with the wall of facebook which means that the latest activities and updates are available to see as an overview.

People can obtain the activities, their dates, their status and interact with it etc. dependent on the activity. People can like, comment, change, stick and edit if allowed.

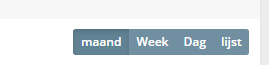
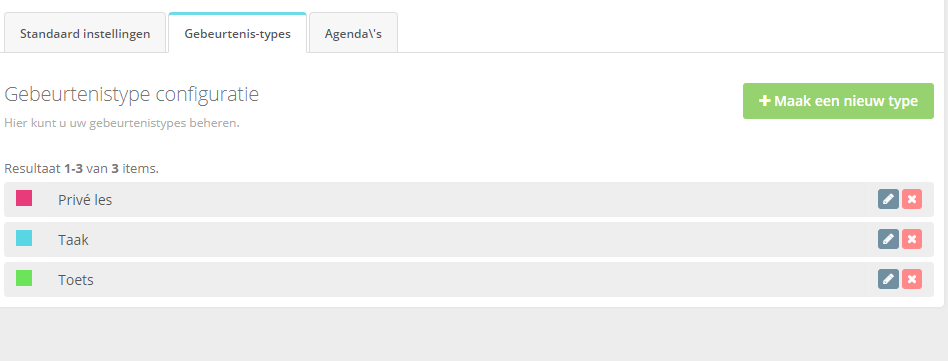
## 2.2. Agenda (Calendars for different tasks)

The agenda of the student is noted in this module.

In agenda:

* Presence/absence of the student is given
* Evaluations (tasks/exams etc.) are given
* The daily notes of the student is given

General nice-to-know functions are:

* On the righthand side, user can switch between the view modes (month-week-day-list)
  + 
* Reminder can be set for agenda items
* Event type (gebeurtenis type) can be explicitly given
  + 

## 2.3. Linklist

This module is to note the list of external links which can be handy for the users.

Although it can be used in user specific modules, it can be more useful when it is used in common modules.

## 2.4. Stembus

This is the module in which different kinds of enquetes can be defined, which can be cleverly used for various cases.

Although it can be used in user specific modules, it can be more useful when it is used in common modules.

## 2.5. Taken (Homework and different tasks)

The module in which the teacher can assign tasks to the student. The functions are:

* Teacher can create list of task
* Teacher can add task to the list of task
* Student and teacher can discuss about the tasks using comments
* The status of the tasks can be obtained (assigned/busy/finished)
* Tasks can be planned (which can also be obtained from agenda)

## 2.6. Scoreboard (report cards)

## 2.7. Attendance (Calendars for different class)

# 3. Development

## 3.1 Scoreboard plugin development

The aim of the project is creating a general plugin for HumHub which follows the structure described in their official website ( <https://docs.humhub.org/docs/develop/modules/> ).

### 3.1.1 HumHub Integration

* Module should be enabled on space level
* Module will generate content (for specific user) (described in following sections)
* Module should give an overview about latest changes in stream
* No sidebar snippets needed
* No special permission is needed
* A brief notification will be fired after any change
* Guest users cannot have access to it

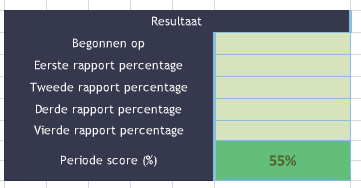
### 3.1.2. General Guidelines

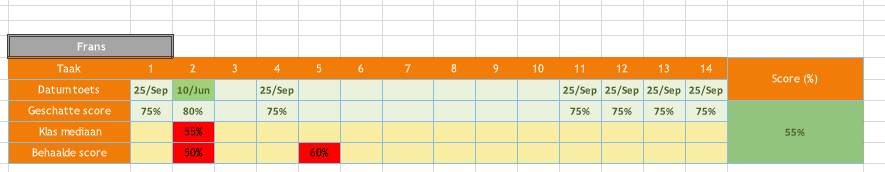
* Preferably, the following technologies should be known by developer:
  + PHP, MYSQL, HUMHUB, *design patterns*
* Bootstrap (and their generic classes) are preferred.
* Css codes should be created in separated file
* Meaningful classes should be created (in order to change it easily)

### 3.1.3 Plugin Scoreboard

The main aim of the plugin is a system where the admin of the space can give in scores for the member of the module (there can be multiple users but the user who gets the score is just one user).

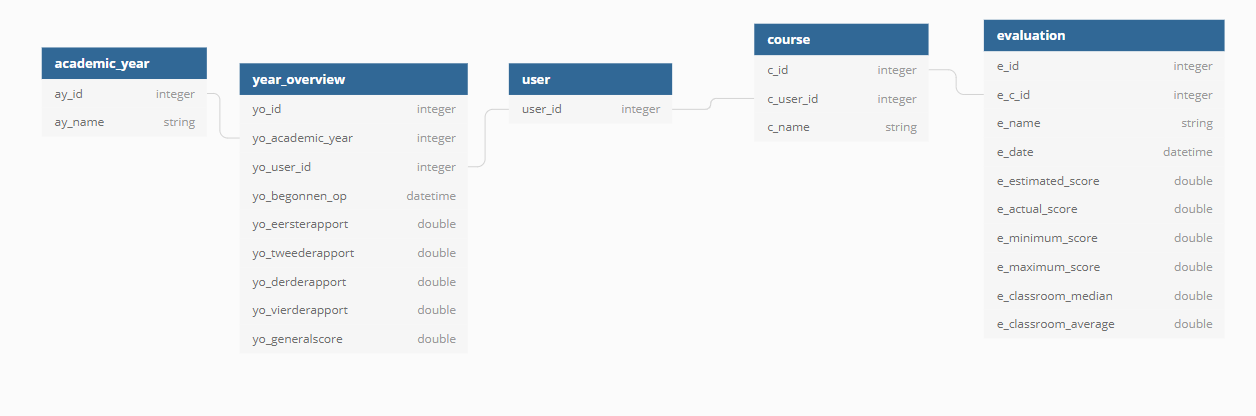
#### 3.1.3.1. Functional requirements

* An overview of the user should be shown (which can be fetched by HumHub). This is the member of that module whose type is “student”.
  + Name, last name, telephone, school, niveau, klas, docent
  + Super admin can edit the needed fields by at self
  + 
* The following details should be able to given/shown at the top:
  + 
  + This is a record for one specific academic year.
  + Academic years (starting from 2019-2020) can be created globally
  + A dropdown can be placed at the top of this grid to be able to see/edit the
* A new course should be able to added
  + Name is the only property
* For each course, an evaluation can be added
  + An evaluation has:
    - Name (default name is “Taak”)
    - Date
    - Estimated score
    - Actual score
    - Minimum required score
    - Classroom median
    - Classroom average
* Every courses and their evaluations should be shown in a grid as (more or less) follows:



* Add a color system in the scoreboard. Like points up to 60 RED, op to 80 ORANGE and above 80 color GREEN. This also for the total score.
* Adding/editing features should be privileged to the administrator and teachers. The students can add the Estimated score (Geschatte sore) for the test subjects before the date. The rest of the users can obtain all the details but cannot change any of it.
* Adding a page with an overall average scoreboard for all students for each report cards like school, study and homework.

#### 3.1.3.2. Database schema



<https://dbdiagram.io/d/5edba1969ea313663b3a23e6>

## 3.2 Smartschool plugin

...