OverfloZZ app specifications
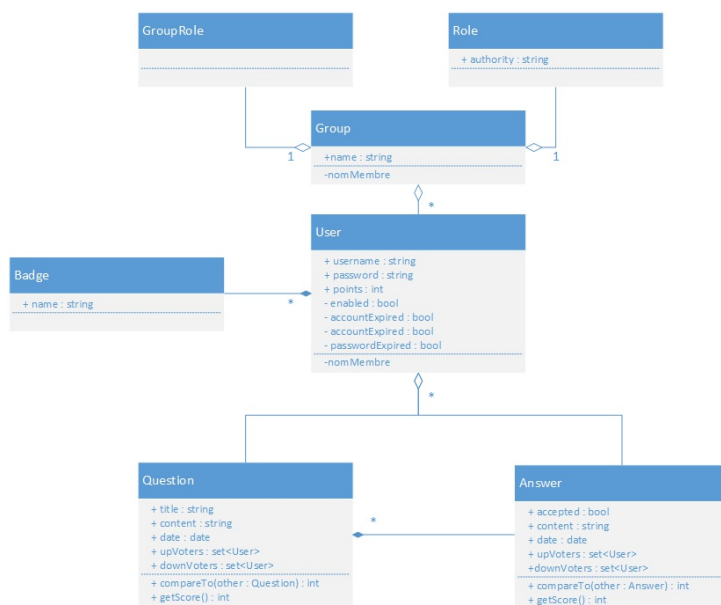
Benjamin Barbesange and Pierre-Loup Pissavy

Overall

The goal is to create a Grails app to ask and answer questions. A gamification system will be in place to reward some user actions.
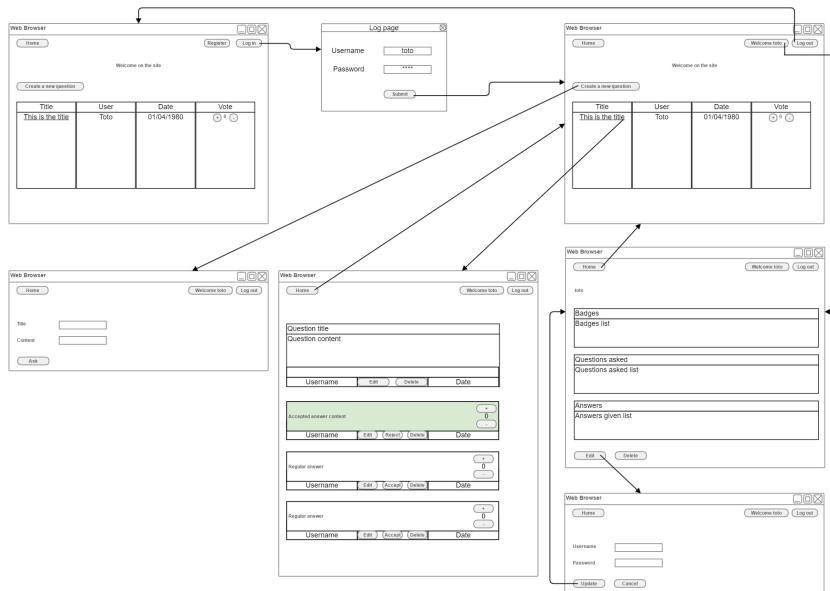
App conception

The global app conception is defined in the UML file diagram.

Preview

App layout and routing

You can see the main app layout and routing in the following file.

Preview

App specifications

The app allows anyone to consult questions and answers associated without having an account on the platform.

However, having an account will enable you to perform some actions, for example :

- **Create** a **question**,
- **Answer** a **question**,
- **Vote** for the answer your found the most interesting.

Questions

The **questions** are the main entities in the app. The questions will contain a **title** to quickly identify the topic and a **content** to develop what the user is asking for. When the question is posted, the user will still be able to **edit** the content but not to delete the question.

The question **author** and **creation date** will be visible for everyone so they can judge if the content is recent enough to solve the problem they are facing.

On the questions **index**, a **vote** section is available for connected users. This allows to sort the questions on the page so the most interesting will be displayed first for the users.

Answers

When a question is posted, logged users are allowed to answer the question. The answer will provide other information such as **author name** and **creation date**.

If the author of the question finds your answer helpfull, he can **mark** it as the best one and it will be displayed on top of other questions. If other users find another answer more important, they can simply **up vote** it (or **down vote** it if they find it not usefull) so it will appear on top of other answers given.

Answers are listed in that order :

- The **accepted** answer is always on top,

- The answers are then ordered by **score** (determined by users votes),
- Finally they are ordered by **date** (most **recent first**).

Any answer provided can be **edited** by the author only to correct mistakes or provide additionnal information.

Users

Users are the main actors in the app. We will find two types of users, described below.

Regular users

This part deals with the regular users that are able to **ask** questions and **answer**.

A user has a lot of information on himself and can consult it on his personnal profile page which will regroup :

- A summary of the **badges** he earned,
- A list of **questions** he asked,
- A list of **answsers** he gave. This page will also enable the user to change his **password** and **username**.

The users will gain some badges by executing tasks on the app. For example a user can earn badges by asking questions or answering questions. When a user question or answer is up voted by the community, this increases its reputation and his points. The more points the user has, the closer he is from winning a giant prize (including a Ferrari !).

Administrators

Along side with the regular users we have the administrators. These users can perform **at least** the same actions of the regular users. But they can also execute moderation tasks such as :

- **Edit** a question or answer content,
- **Delete** a question or answer,
- **Mark** an answer as accepted for a question.

They are also in charge of users accounts and can set a password expiration date, reset a password or delete the account.