

NKUNet Design Document

Last Modified: March 14th, 2017

Tables of Contents:

| | |
|-------------------------|---|
| Introduction | 1 |
| Class Diagram | 2 |
| Design Sequence Diagram | 3 |
| Rails Commands | 4 |

1. Introduction

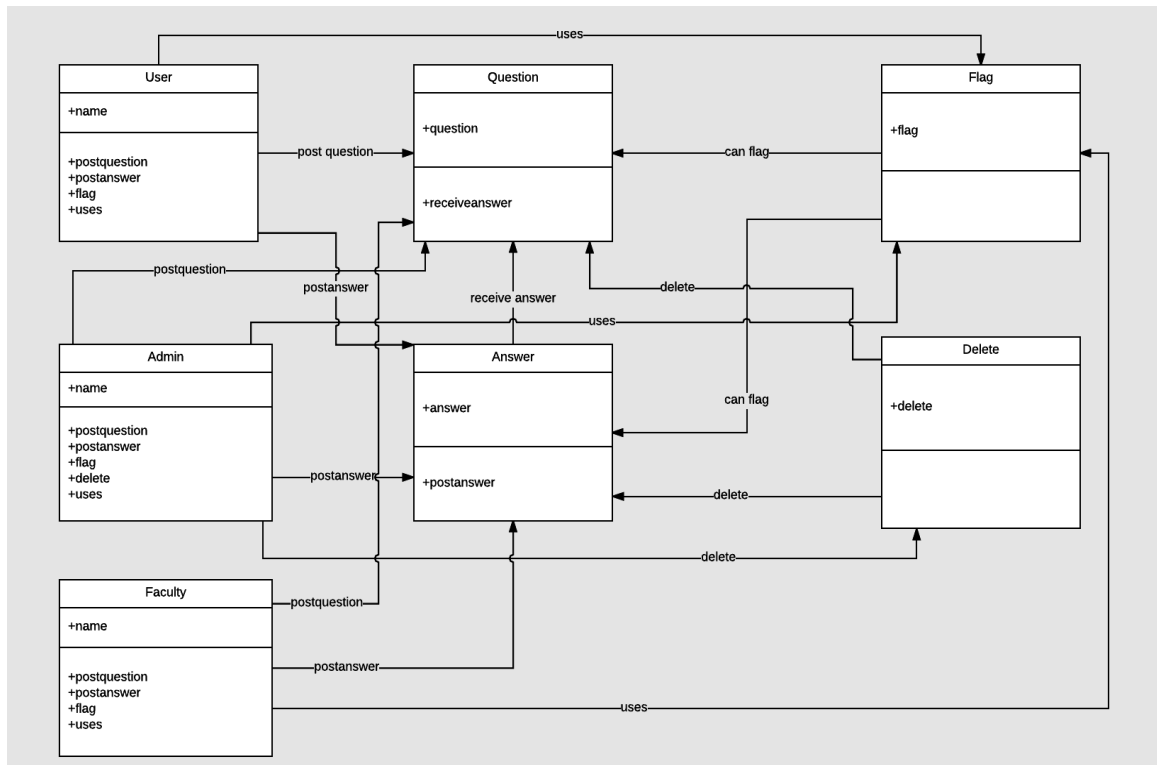
1.1 Background

NKUNet is a forum system designed for people within the NKU community. The ideal outcome of this is to provide a forum for faculty, students and registrars which are able to ask questions and get answers from others within the community. The idea behind this is to grant a social area for both faculty and students to share their questions and thoughts and to be able to communicate with people on a wider network.

1.2 Design Goals

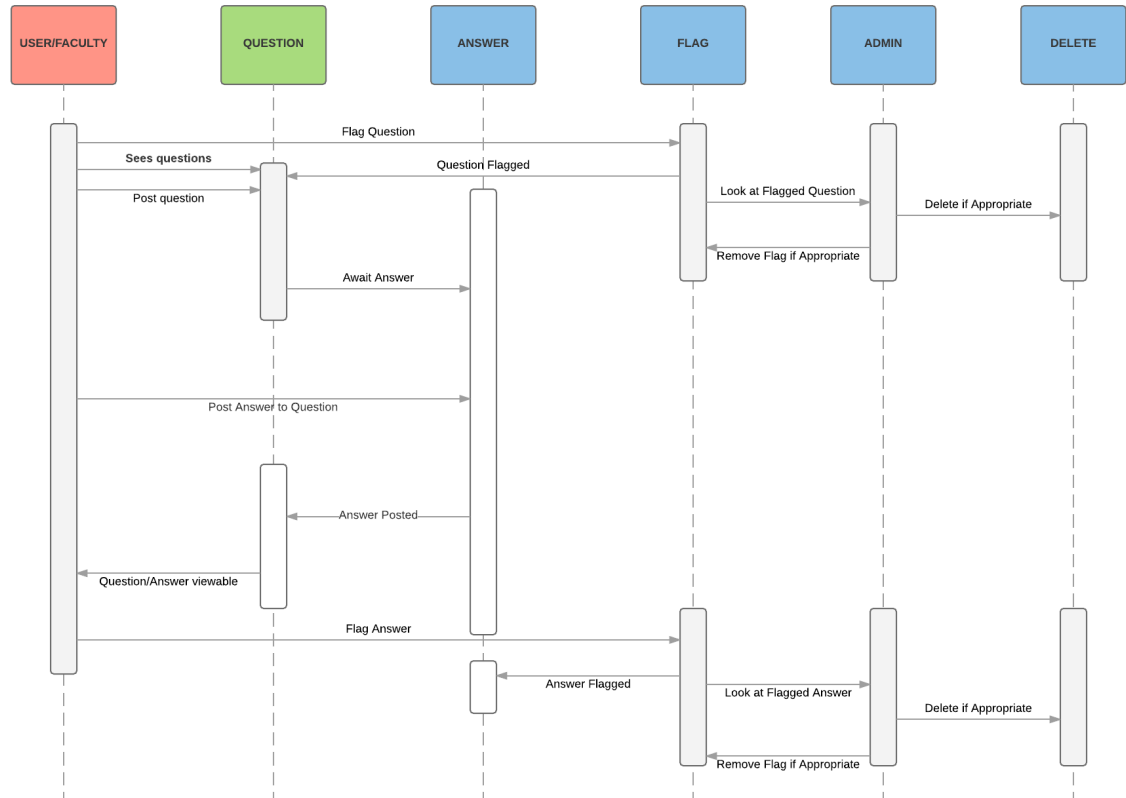
The basic design goal is the creation of a forum for those within the NKU community. The users should have the ability to post questions, receive answers, upvote/downvote responses, have their schedules attached to their user account and add friends, for personal messaging. The idea has potential to expand further beyond this, given development time and resources available should initial implementation be done ahead of schedule.

2. Class Diagram



The basic premise behind the class choices here is to get a functional, early model of the system up and running. The user, when posting, will have their names displayed and the ability to post questions or answers on NKUNet. They will also have the ability to use the flag command for posts that need possible looking into for various reasons. Faculty will follow the same permissions as the user currently. The Admin will have the ability to see everything, including posts that are flagged. Admin will also be the class that is granted the delete functionality, for removing either an answer or an entire question. These will likely be expanded upon as time progresses forward.

3. Design Sequence Diagram



The Sequence diagram is a current basic flow chart for the current design. The idea starts with a user/faculty having the ability to see questions and answers. If either a question or answer is inappropriate, misplaced or not needed then it can be flagged to be looked at by the admin. The admin has the right to observe the flagged question or answer and if desired delete the answer or the entire question and subsequent answers attached to it. User/Faculty also has the ability to post a question, post an answer, and observe current questions and answers and whom posted them.

4. Rails Commands

4.1 – rails generate scaffold User name: string email: string

This command has a name field and an email field.

4.2 – rails generate scaffold Question content: text user_id: integer

This command we have a content field and an id to differentiate questions

4.3 - rails generate model Question content: text user: references –force

This command we have a content field and a user reference foreign key so that we can easily see who asked. The –force means we overwrite the db table.

4.4 - rails generate scaffold Answer content: text user_id: integer

This command we have a content field and an id to differentiate answers.

4.5 – rails generate model Answer content: text user: references question: references –force

This command we have a content field and a user reference foreign key so that we can easily see who asked. The question reference also has a key to the question foreign key. The –force means we overwrite the db table.