

Website: Forum/bulletin board with user moderated forums

Student Information:

s3439870 – Harrison Slater

Table of Contents

- 1 [Executive summary](#)
- 2 [Scope of project](#)
- 3 [Gantt chart](#)
- 4 [Software Requirements Specification](#)
 - 4.1 [Requirements](#)
 - 4.1.1 [Functional Requirements](#)
 - 4.1.2 [Non-Functional Requirements](#)
 - 4.1.3 [Data Requirements](#)
 - 4.2 [Use Cases](#)
- 5 [Design](#)
 - 5.1 [ER diagram](#)
 - 5.2 [Sitemap](#)
 - 5.3 [Sketches](#)

Executive Summary

This document is the specification for a forum application. The application is a forum system that allows users to create their own forums where discussion with other users can take place. Forums would likely be centered around a specific theme such as 'video games'. Users can make threads in these forums about a subtopic or discussion relevant in the forum.

Each forum may have moderators which control what goes on in the forums by removing unallowed threads or comments and helping out in discussions.

Administrators are sitewide moderators.

Depending on whether the user is a moderator, admin, or a normal user the interface will show different links to their respective permitted actions.

Mission statement for the project: "To create a general web forum that is minimalist in design."

Scope

Main objectives:

To create a moderated forum application that has threads, comments, and users.

// Didn't finish the scope.

Gantt Chart

Made with (<https://www.gantt-chart.com/>) And (<https://trello.com>)



Software Requirements Specification

Requirements

Colour Legend

- Is required
- Originally required and planned for but didn't have time to implement
- Is a nice feature to have
- Is a complex feature only to be implemented when all else is done

Functional Requirements

- 1 Forums
 - 1.1 Create forum
 - 1.2 View forums
 - 1.3 Admin can close a forum
 - 1.4 Add moderator for a forum
 - 1.5 Remove moderator from a forum
 - 1.6 Sort forum threads by
 - 1.6.1 newest
 - 1.6.2 title
 - 1.7 Search forums for something in the title/thread description/thread comments
 - 1.8 Moderators can manage permissions on a forum for users
 - 1.9 Moderators can set up an automoderator
 - 1.10 Moderators can 'sticky' a thread in a forum (permanently there)
- 2 Threads
 - 2.1 Create thread
 - 2.2 View thread comments by page number
 - 2.3 View thread by infinite scroll
 - 2.4 Edit original thread post/title
 - 2.5 Moderator can delete a thread
- 3 Comments
 - 3.1 Create comment
 - 3.2 View comments
 - 3.3 Edit comment
 - 3.4 A moderator can delete a comment
 - 3.5 Quote a comment in a reply
 - 3.6 Alert a user when they've been quoted
- 4 Validation
 - 4.1 Must show error message for failure to validate user input
- 5 Roles
 - 5.1 Role with privileges should be shown on the user profile
 - 5.2 Admin
 - 5.2.1 Admin can only be added directly to the database with an SQL insert for security
 - 5.3 Moderators
 - 5.3.1 A user is made a moderator for a forum when they create a new forum
- 6 Authorization – All authorization is required
 - 6.1 'view all' and 'view single' pages are public unless otherwise specified
 - 6.2 Users – 'view all users' page is not public
 - 6.2.1 Users can:
 - 6.2.1.1 View any user's profile
 - 6.2.1.2 Edit their own profile

6.2.2 Moderators can:

6.2.2.1 Do what a user can

6.2.3 Administrators can:

6.2.3.1 Do what a user can

6.2.3.2 View all users in the system

6.2.3.3 Create any user

6.2.3.4 Delete any user

6.2.3.5 Edit any user

6.3 Forums

6.3.1 Users can:

6.3.1.1 Add a forum

6.3.1.2 View all forums

6.3.1.3 View a forum

6.3.2 Moderators can:

6.3.2.1 Do what a user can

6.3.2.2 Edit a forum if it is theirs

6.3.3 Administrators can:

6.3.3.1 Do what a user can

6.3.3.2 Edit a forum

6.3.3.3 Delete a forum

6.4 Threads – 'view all threads' page is not public

6.4.1 Users can:

6.4.1.1 Add a thread

6.4.1.2 View a thread

6.4.2 Moderators can:

6.4.2.1 Do what a user can

6.4.2.2 Edit a thread in their forum

6.4.2.3 Close a thread in their forum

6.4.3 Administrators can:

6.4.3.1 Do what a user can

6.4.3.2 View all threads

6.4.3.3 Edit any thread

6.4.3.4 Delete any thread

6.5 Comments – 'view all comments' page is not public

6.5.1 Users can:

6.5.1.1 Add a comment

6.5.1.2 View a comment

6.5.1.3 Edit their own comment

6.5.1.4 Delete their own comment

6.5.2 Moderators can:

6.5.2.1 Do what a user can

6.5.2.2 Edit any comment in their forum

6.5.2.3 Delete any comment in their forum

6.5.3 Administrators can:

6.5.3.1 Do what a user can

6.5.3.2 Edit any comment

6.5.3.3 Delete any comment

6.6 Moderators section – 'view all' and 'view single' pages are not public

6.6.1 Users can:

6.6.1.1 Do nothing

6.6.2 Moderators can:

6.6.2.1 Do nothing

- 6.6.3 Administrators can:
 - 6.6.3.1 View all moderators
 - 6.6.3.2 View a single moderator
 - 6.6.3.3 Add a new moderator
 - 6.6.3.4 Delete a moderator

7 Time

- | | |
|-----|--|
| 7.1 | Stored in UTC |
| 7.2 | Should automatically convert to the user's local time in browser |

Non-functional

- 1 Server
 - 1.1 AWS instance
- 2 Software
 - 2.1 LAMP stack
 - 2.1.1 PHP 5.6.x
 - 2.2 Cakephp 3 framework
 - 2.3 Dependency management using composer

Data Requirements

- 1 Forum
 - 1.1 Title
 - 1.2 Moderators
 - 1.3 Time created
- 2 Thread
 - 2.1 Title
 - 2.2 Comments (including original post)
 - 2.3 Time posted
- 3 Comment
 - 3.1 User that submitted
 - 3.2 Comment body
 - 3.3 Time posted
 - 3.4 If the post was edited
 - 3.5 Time edited
- 4 User
 - 4.1 Username
 - 4.2 Email
 - 4.3 Password
 - 4.4 Date created
- 5 Moderator
 - 5.1 Forums
 - 5.2 User
- 6 Administrator

Use Cases

Initial scenarios for brainstorming use cases

The user browses and chooses a forum. The user browses and chooses a thread. The user posts a comment. System validates comment and returns a success message.

The user enters their login info and submits. The system authenticates the user and displays the main page.

The user changes their password. The system validates it and displays success.

A moderator sees a comment that is not ok according to the forum rules. The moderator deletes the comment and confirms the delete. The system replaces it with a 'moderator deleted' message.

A moderator sees a thread where the original creator has had his issue resolved but people are still posting for no reason. The moderator closes the thread to stop further posting.

Use Cases:

Posting a comment

Main Success Scenario:

1. User browses main page and chooses a forum
2. User browses the forum and chooses a thread
3. User posts a comment
4. System validates comment
5. System confirms comment posted

Extensions:

- 1a: User is not authorized for this forum
- .1: System denies user
- 3a: Comment is not valid
- .1: System attempts to validate
 - .2: System displays error message for user

Logging in

Main Success Scenario:

1. User enters login info
2. System authorises login
3. System displays main page

Extensions:

- 1a: Login info is not valid
- .1: System denies login
 - .2: System displays error message
- 3a: User is an Admin
- .1: System displays Admin page

Creating a thread

Changing password

Main Success Scenario:

1. User goes to their profile
2. User updates their password
3. System validates password
4. System confirms password updated

Extensions:

- 3a. New password is not valid
 - .1 System denies password update
 - .2 System displays error

Creating a forum

Precondition: User is logged in

Main Success Scenario:

1. User fills in forum details
2. System validates forum details
3. System creates forum
4. System sets User as a moderator

Extensions:

- 2a: User details are not valid
 - .1 System displays error

Moderating a forum – Deleting a comment

Main Success Scenario:

1. Moderator browses a thread
2. Moderator identifies a comment to be deleted
3. Moderator deletes
4. Moderator confirms the delete
5. System replaces comment with a 'moderator deleted' msg

A moderator sees a thread where the original creator has had his issue resolved but people are still posting for no reason. The moderator closes the thread to stop further posting.

Moderating a forum – Closing a thread

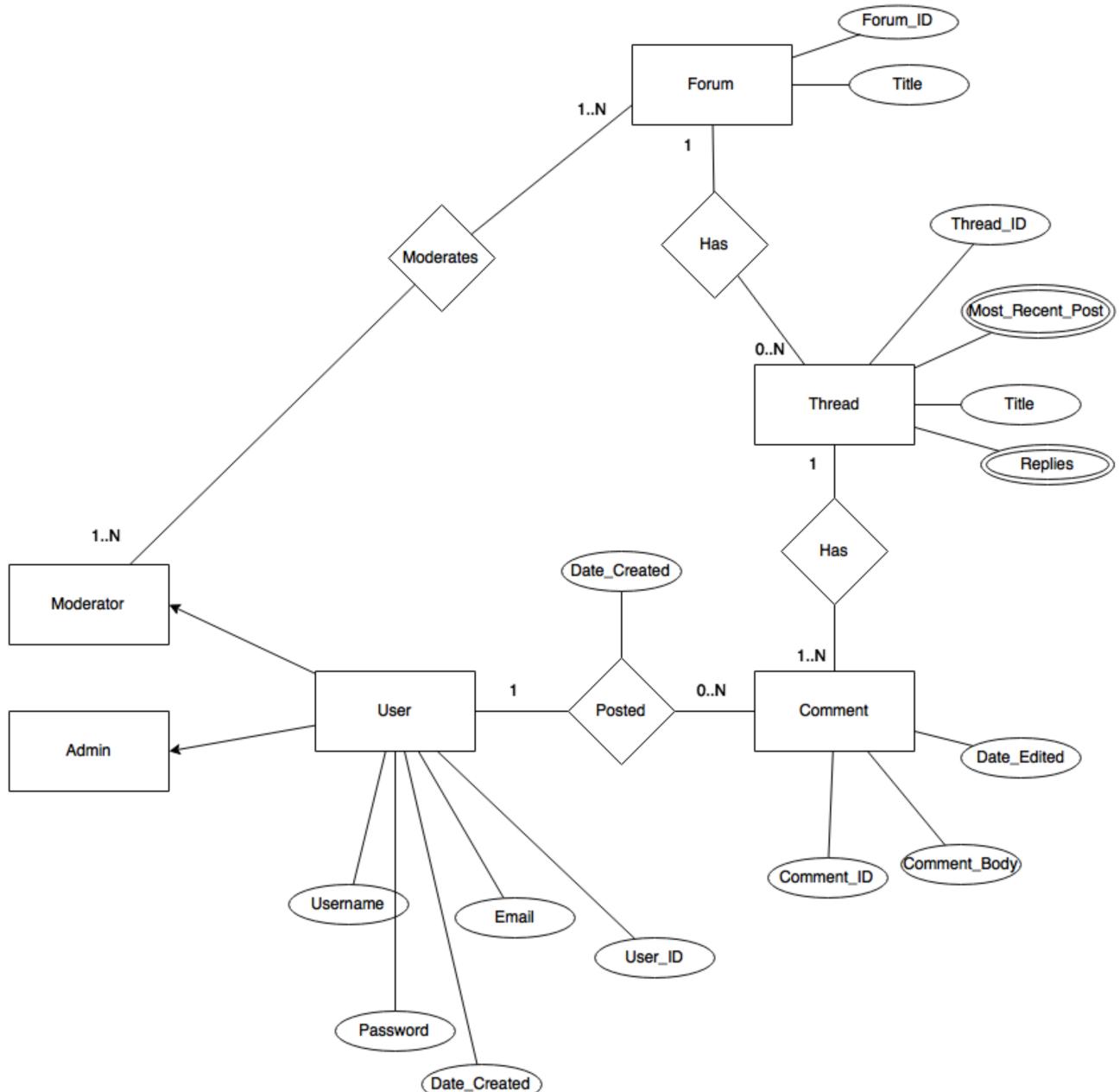
Main Success Scenario:

1. Moderator identifies thread that needs to be closed
2. Moderator closes thread

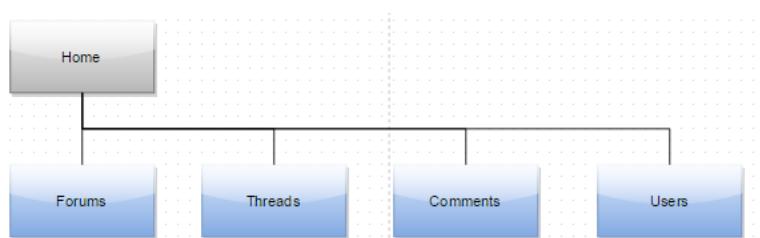
Design

ER diagram

ER diagram done with (<https://www.draw.io/>)



Sitemap



Basic sitemap done with (<https://www.draw.io/>)

Sketches

Basic sketches of the ui

