PS3 Problem Analysis

**Overview**
Purpose and goals
For both phase 2 and phase 3, the goals of the system are the same. I will be designing and implementing a service that allows for comments on admin specified web page sections.
For example, if an admin wanted to allow comments for a photo on their website, they could embed the given code into that page. Users will be able to see new comments in real time, post comments, and upvote comments. In phase 3, admins will be able to give certain comments priority over others. While existing solutions exist for this type of service, my system will be very simple and easy to use, with the focus being on upvoting and correctly prioritizing the ordering of comments.
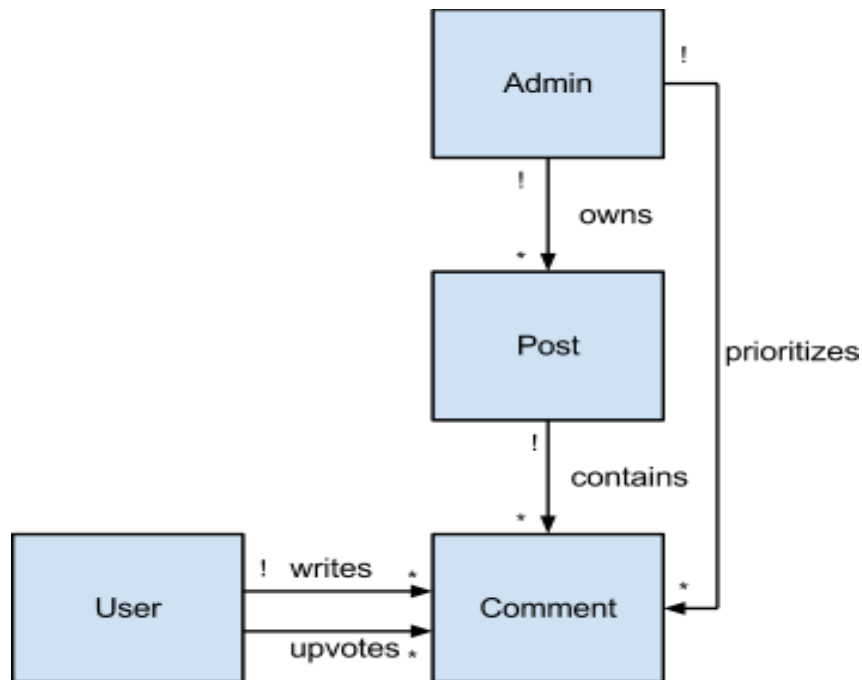
Context diagram
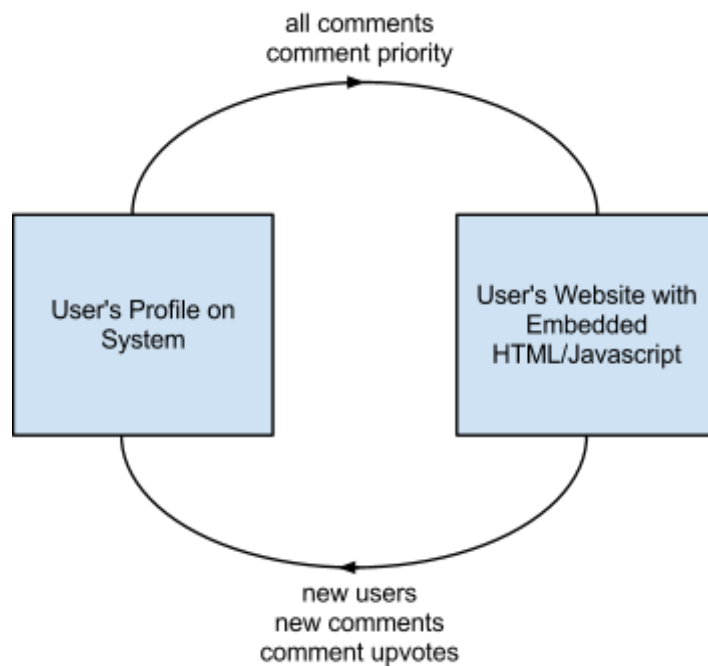
Phase 3 will implement comment priority.

Domain
Object model

Admin

!

owns

*

Post

!

prioritizes

contains

*

! writes *

User

Comment

*

upvotes *

Again, only Phase 3 will have the prioritize relationship.

Event model
Admin registration
Admin login
Create a post: on the system web page, create a log for the web page you would like to embed the comment system within
View a post summary on website: only accessible to admins
View a post on external site
Create a comment
Upvote a comment: only users can upvote comments on the external embedment
Prioritize a comment: only admins can prioritize a comment on the system web page

Events ::= Admin | User
Admin ::= AdminReg AdminLogin* (CreatePost | ViewPost Internal)* (CreateComment | PrioritizeComment)*
User ::= ViewPostExternal (CreateComment | UpvoteComment)*

Behavior

Feature descriptions
User registration
Admin registration
User login
Admin login

Create a new post for comments: an admin can register a post on their website and implement the comment service on that post
Create a comment: a user or admin can create a comment on an existing post
Upvote a comment: a user can upvote an existing comment
Prioritize a comment: an admin can prioritize an existing comment; prioritized comments will appear higher than unprioritized comments, but still organized by vote counts

Security concerns
The system must have a solid authentications system for logging in both admins and users. Passwords will be used for both logins, and sessions used to authenticate both as well. Another security concern is user spam. To address this, admins have the capability to prioritize comments, which implies both increasing the visibility or decreasing it in the event of comment spam.

Operations
Admin registration
        pre: not logged in as admin
        post: new admin created and logged in
        frame: all fields filled out and valid
Admin login
        pre: not logged in as admin, registered as admin
        post: logged in as admin
        frame: all fields filled out and valid
Create a post
        pre: logged in as an admin
        post: post created
        frame: post not deleted
View a post on internal site
        pre: post exists, logged in as admin
        post: post exits, logged in as admin
        frame: post not deleted
View a post on external site
        pre: post exists
        post: post exists
        frame: post not deleted
Create a comment
        pre: post exists
        post: comment created
        frame: post not deleted
Upvote a comment
        pre: comment exists
        post: comment upvoted by 1 increment
        frame: post not deleted
Prioritize a comment

pre: logged in as an admin, comment exists
post: comment added to priority list
frame: post not deleted, comment not deleted

User interface

login
welcome

Homepage

login
username: password:

Login

welcome mari!

view posts

create post

Admin home

link to post

comment   2   *

comment   0

Sample post: numbers represent upvotes and stars represent admin priority

post

comment | 2 | *

comment | 0

write new comment here | login

Embedded on web page: numbers represent upvotes and stars represent admin priority

Homepage leads to login leads to admin home which leads to sample post
Service embedded on web page allows for user login, commenting, and upvoting