

## PS3 Problem Analysis

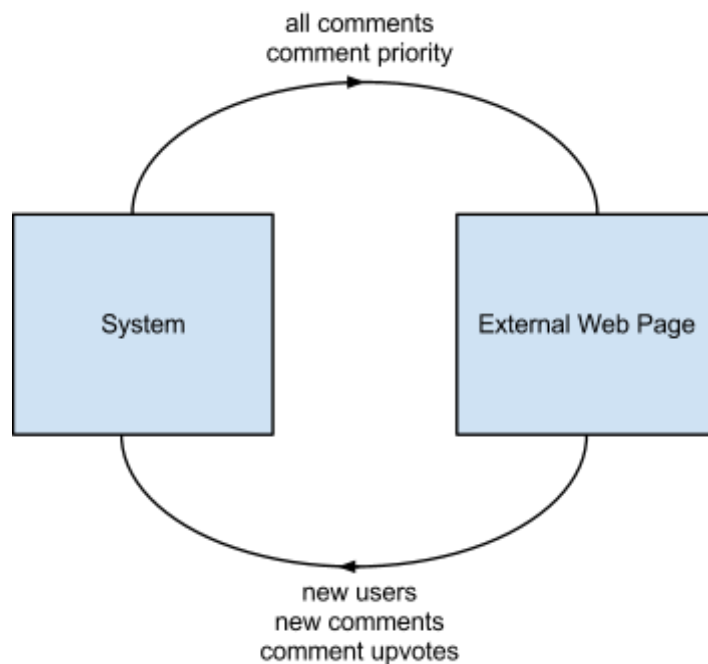
### Phase 2 and Phase 3

#### 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 components. 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.

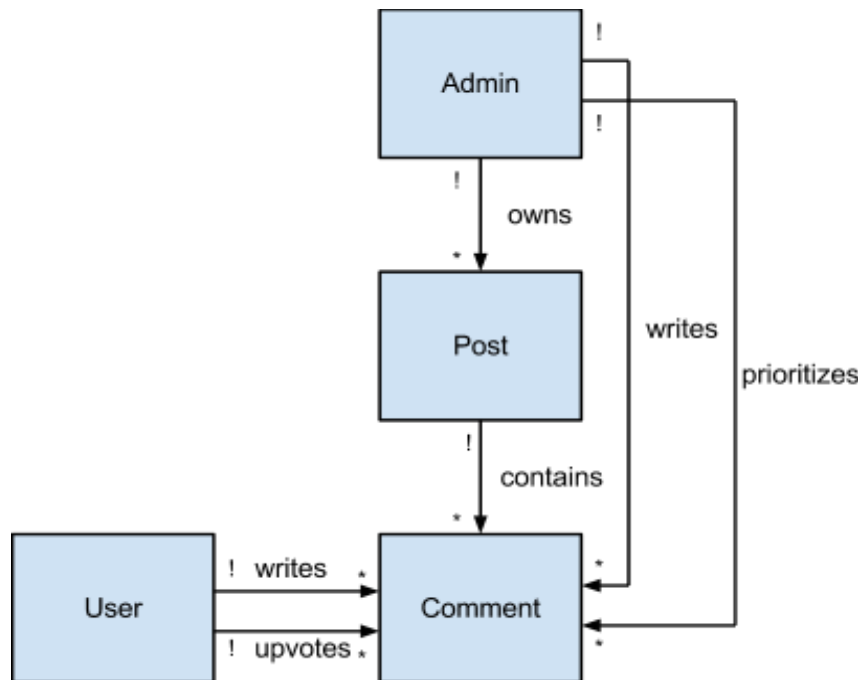
##### *Context diagram*



Phase 3 will implement comment priority.

#### Domain

##### *Object model*



Again, only Phase 3 will have the prioritize relationship.

#### *Event model*

Admin registration

User registration

Admin login

User 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

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 (only in phase 3)

Events ::= Admin | User

Admin ::= AdminReg AdminLogin\* CreatePost\* (CreateComment | PrioritizeComment)\*

User ::= UserReg\* UserLogin\* (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 (only in phase 3)

### *Security concerns*

The system must have a solid authentication 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*

#### User registration

pre: not logged in as a user or admin, on embedded site

post: new user created and logged in

frame: all fields filled out and valid

#### Admin registration

pre: not logged in as a user or admin

post: new admin created and logged in

frame: all fields filled out and valid

#### User login

pre: not logged in as user or admin, registered as user, on embedded site

post: logged in as user

frame: all fields filled out and valid

#### Admin login

pre: not logged in as user or 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

#### Create a comment

pre: logged in as a user (on embedded site) or admin (on system site)

post: comment created

#### Upvote a comment

pre: logged in as a user, comment exists

post: comment upvoted by 1 increment

#### Prioritize a comment (only in phase 3)

pre: logged in as an admin, comment exists

post: comment added to priority list

### *User interface*

login

welcome

Homepage

login

username:  
password:

Login

login

welcome mari!

view posts

create post

Admin home

login

link to post

comment2\*

comment0

Sample post: numbers represent upvotes and stars represent admin priority (the stars only in phase 3)

post

comment 2 \*

comment 0

write new comment here login

Embedded on web page: numbers represent upvotes and stars represent admin priority (the stars only in phase 3)

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