

Team One

John O'Kelly, Owen Morrissey, Steve Lesicka, Chris Kobierowski
For CCSU Journalism Dept.

Iteration Report 1

Current Functionality

- User accounts supported
- Permissions implemented through JDBC Realms
 - Two user classes so far: Admin, User
 - Restricted directories control access properly
- Baseline CSS/HTML display of static content
- Content and links can be written to the database through a web form
 - Content can be written in rich HTML

Implemented User Stories

1. Post Content Link

User Role: Content Publisher

Goal: To post a link to an article hosted on a 3rd party website.

Reason: To provide users with content relevant to the elections and local news.

Pre-Conditions: Content publisher is logged in, an article has been found, and a link is available to the content.

Post-Conditions: The link is posted to the database.

2. Post Content

User Role: Administrator, Content Publisher

Goal: To post content for all users to view.

Reason: To provide users with well-presented media for consumption.

Pre-Conditions: Content publisher or administrator is logged in, and has content to post.

Post-Conditions: The content is posted to the database.

8. Create Account

User Role: Unregistered User

Goal: Allows user to create their own personal account.

Reason: To allow users to join and contribute to the community.

Pre-Conditions: User does not have an account.

Post-Conditions: User now has access to privilege given to their user class.

Partially Implemented (Ongoing)

7. View Content

User Role: Administrator, Content Publisher, Registered User, Unregistered User

Goal: Allow all users to view content posted.

Reason: To see content that other users have posted.

Pre-Conditions: Content has been posted.

Post-Conditions: Users can respond to post, Administrator can edit and delete posts.

Team One

John O'Kelly, Owen Morrissey, Steve Lesicka, Chris Kobierowski
For CCSU Journalism Dept.

Synopsis

Our user stories changed slightly for this iteration. It turns out that posting content links and content to the database was an easily combined process, so both are accomplished from one page. This means user story 2 was implemented when it was not planned to, which is good. However, user story 7 proves to be an ongoing process with the site. It is implemented very basically, without any solid dynamic rendering based on the logged in user (e.g. adding a logout button when the user is logged in.) This will have to be an ongoing task tacked on to the next sprint, probably with the same point value.

Lessons Learned

We learned it is very important to sort out all of the JDBC and database configurations together and early in the process. It becomes very difficult to set up all of the different security realms and other glassfish configurations when everyone's database is named something different. So, it is crucial to work together to implement and configure server-side attributes early in the process to ensure everyone's configurations are standardized.

User Stories

3. Update Live Coverage

User Role: Administrator

Goal: To post real-time updates of the electoral process.

Reason: So users can enjoy and stay informed with up-to-the-minute updates.

Pre-Conditions: Account with administrator permissions is logged in.

Post-Conditions: The update is pushed out live to all viewers on page.

4. Comment on Content

User Role: All above Registered User (Admin, Content Publisher)

Goal: For a user to comment on posted content.

Reason: To allow discussion on site content.

Pre-Conditions: There exists some content to reply to.

Post-Conditions: The reply is visible on the site when viewing aforementioned content.

5. Post in Forum

User Role: All above Registered User (Admin, Content Publisher)

Goal: For a user to post a reply to a forum topic.

Reason: To allow users of the site to participate in discourse.

Pre-Conditions: There exists a forum topic to post a reply to.

Post-Conditions: The reply is posted and is available for viewing in the forum.

6. Create Forum Topic

User Role: Administrator

Goal: Create a new topic.

Reason: To provide a forum for users to post in.

Pre-Conditions: There is not an identical forum topic.

Post-Conditions: Users can now post content inside of the topic.

Team One

John O'Kelly, Owen Morrissey, Steve Lesicka, Chris Kobierowski
For CCSU Journalism Dept.

7. View Content

User Role: Administrator, Content Publisher, Registered User, Unregistered User

Goal: Allow all users to view content posted.

Reason: To see content that other users have posted.

Pre-Conditions: Content has been posted.

Post-Conditions: Users can respond to post, Administrator can edit and delete posts.

9. Update Map Data

User Role: Administrator

Goal: Designate points on map that link with stories.

Reason: To provide users an easy way to view the location to which a story is connected.

Pre-Conditions: A story has been posted.

Post-Conditions: A point on the map is linked with the story.

10. Delete Post/Comment

User Role: Administrator

Goal: To delete a post or comment.

Reason: Unsavory content finds its way on the internet in all shapes and forms, so an administrator can delete posts or comments.

Pre-Conditions: There exists a comment or post to be deleted.

Post-Conditions: The post or comment is removed from the page.

11. Edit Post/Comment

User Role: Administrator

Goal: To edit a post or comment.

Reason: An administrator may want to edit a post or comment on the site.

Pre-Conditions: A post or comment exists to be edited.

Post-Conditions: The post or comment is updated with the new content.

12. View Analytics

User Role: Administrator

Goal: View analytics about the website such as demographics and page views

Reason: To give administrators meaningful statistics about the website.

Pre-Conditions: Analytics have been collected and administrator is logged in.

Post-Conditions: The administrator is able to view analytics.

Team One

John O’Kelly, Owen Morrissey, Steve Lesicka, Chris Kobierowski
For CCSU Journalism Dept.

Fibonacci Distribution						
1	2	3	5	8	13	21
Delete Post/Comment	Post Content Link	View Analytics	Post Content	Comment on Content	Update Live Coverage	Create Forum Topic
Edit Post/Comment			Post in Forum	Update Map Data	Create Account	View Content
Total						
106						
Target Sprint						
36						

Selected User Stories – Second Iteration

Story	Value
View Content	21+
Delete Post/Comment	1
Edit Post/Comment	1
Comment On Content	8
Total	31

Information

As you can see, we have chosen to add viewing content back to the sprint with a value of 21+. It is by far and away the most complicated part of the system, as we need to implement JSF or some other framework in order to display the stories from the database in a logical, attractive, and procedural manner, as well as display certain parts of the site dependent on user class. We predict this will be the most time consuming portion of the application.

At the end of this iteration, the content system should be fleshed out so that dynamic content can be posted and viewed on the site as well as commented on by users. We missed the target of 36, but I believe we will have more than enough on our plate to bring us to the next iteration with a solid partially implemented system.