

2 Heads Are Better Than 1

Iteration 3

Test Plan

- **Iteration 1**

- **Tests**

- *Login* (Manual) – This use case is the most critical to the success of the iteration because without allowing the user to sign up or login to Event Social, the application is essentially useless since no one can access the site. This use case needs to pass 100% of the tests thrown at it. To test this use case, we will do the following:
 - Create an account on Event Social through the internal sign up form. It is important to enter incorrect formatting characters in the form to ensure that the system is validating user input.
 - Create an account on Event Social using the Sign In With Twitter button and providing the application access to Twitter credentials.
 - Log in to an account that was created using the internal form by entering the correct email address and password. It is also important to enter incorrect credentials to ensure that the system does not incorrectly log in somebody who should not have access to the site.
 - Leave all of the fields blank and try to submit to ensure that the page will display the errors to the user telling them that every field is required.
 - Log in to an account that was created using the Twitter sign up by clicking on the Sign In With Twitter button.
 - *View Tweets* (Manual) – This use case is important to the success of the iteration because it is the beginning of what will be the most used feature of Event Social. To test this use case, we will do the following:
 - Navigate to the Tweets page; all tweets for the first event should be displayed.
 - Make sure that all events that the user is subscribed appear in the sidebar on the left, and if you click on one of the events, the tweets are refreshed for the current event's hashtag.

- **Iteration 2**

- **Tests**

- *Edit User Profile* (Manual) – This use case is important because we want the user to be able to customize their profile that is displayed to other users. They should be able to edit their profile at any time, so it is important to make sure that this form works correctly almost every time. It isn't necessarily a core feature of importance to the application such as the use cases dealing with events, but we need to make sure it is not normal for users to run into errors when editing their profile. It should be something that runs perfectly smoothly 9.5 out of 10 times in the very least. To test this use case, we will do the following:
 - Enter normal data in every field to ensure that it the feature works as it should.
 - Enter incorrect data in every field and try to submit to ensure that the user cannot save this information to their profile.
 - Enter incorrect data in one field and correct data in all others and try to submit to ensure that each field is validating the information correctly. Do this multiple times, changing which field you enter the incorrect data in every time to make sure each field works the way it should.

- Leave all of the fields blank and try to submit to ensure that the page will display the errors to the user telling them that every field is required.
- Leave one field blank and fill in correct data for every other field and try to submit to ensure that each individual field will display to the user that it is required. Do this multiple times, changing which field we leave blank every time to make sure each field works the way it should.
- *Create New Event (Manual)* – This use case is important because it leads to the most important feature of Event Social: allowing users to see tweets and comments from an event that they are subscribed to. We want all users to be able to create an event at any time, and because this use case is so vital to the application, we need this to work 100% of the time without fail. To test this use case, we will do the following:
 - Enter normal data in every field to ensure that it the feature works as it should.
 - Enter incorrect data in every field and try to submit to ensure that the user cannot save this information to their profile.
 - Enter incorrect data in one field and correct data in all others and try to submit to ensure that each field is validating the information correctly. Do this multiple times, changing which field you enter the incorrect data in every time to make sure each field works the way it should.
 - Leave all of the fields blank and try to submit to ensure that the page will display the errors to the user telling them that every field is required.
 - Leave one field blank and fill in correct data for every other field and try to submit to ensure that each individual field will display to the user that it is required. Do this multiple times, changing which field we leave blank every time to make sure each field works the way it should.
 - Create events with and without images and make sure that it works whether or not you choose to upload an image for your event.
 - After we have thoroughly tested the form with the above steps, we will ensure that the event was created successfully by completing the following steps.
 - Navigate to the My Events page and ensure that the event that you just created now appears there. Expand the tab and click on the View More button and confirm that all of the data displayed matches what you had input when you created the event.
 - Navigate to the My Events page, expand the tab of the event you created, and click on the Edit Event page. Ensure that all of the data populated in the input fields is the same as the information you filled out when you created the event.
 - Log out of the profile that you used to create the event and log back in with a different profile. Navigate to the Search Events page and verify that the event you just created shows up in the table. Click on the View More button and confirm that this page is the same as the one that you viewed when you completed the test two bullets above.
- *Subscribe to Event (Manual)* – Similar to Create New Event, this is an extremely important use case for the success of Event Social. The whole point of the application is to build hype for an event, and without allowing users to subscribe to events, we would not be able to meet the basic functionality we envisioned from the start. Because of the importance of this use case, we also need this to work 100% of the time. To test this use case, we will do the following:

- Create an event and log in with a different account than the one that you used to create it. Navigate to the Search Events page and click the View More button of that event. Click the subscribe button at the bottom of the page to subscribe to the event, and ensure that this button now changes to an unsubscribe button.
- Navigate to the My Events page and make sure that this page is now displayed under your Subscribed Events section.
- Navigate to the Tweets page and locate the hashtag of the event you just subscribed to under the Subscribed Events section on the left sidebar. Click on the hashtag and ensure that the tweets that are displayed on the page all contain the hashtag specified for this event.
- Log out of this profile and log in to a third profile. Navigate to the Search Events page and find the event you created. Click the View More button and then on the next page, click the Subscribe button. This makes sure that multiple users are allowed to subscribe to an event. Then follow the above steps for extra confirmation that you are subscribed to this event.

• **Iteration 3**

○ **Tests**

- *View Comments* (Manual) – This use case is important to the success of the iteration because it is a feature of Event Social that is heavily integrated with events and that will be extremely beneficial to users who do not have Twitter accounts. To test this use case, we will do the following:
 - Navigate to the comments page. The page should display all comments regarding to the first event on the left sidebar, and selecting a different event from the sidebar should populate the page with all of the comments for that event.
 - Make sure that all events that the user is subscribed appear in the sidebar on the left, and if you click on one of the events, the comments are refreshed for the current event.
- *Add Comments* (Manual) – This use case is important because it allows users to provide their feedback and input on events, as well as communicate with others through replies. Since users without Twitter can't provide feedback through tweets, it is important to ensure that the comments feature works so that Event Social users without Twitter accounts can participate at a similar level as users with Twitter accounts. To test this use case, we will do the following:
 - Enter normal data in the comment textbox and submit it to ensure that the comment feature works as it should.
 - Enter normal data in the reply textbox and submit it to ensure that the reply feature works as it should.
 - Try to submit a comment without entering any characters in the comment textbox to ensure that the user cannot enter blank comments or replies.
 - Try to submit a reply without entering any characters in the reply textbox to ensure that the user cannot enter blank comments or replies.
 - All of the above tests should be completed on both a moderated event and an event that is not moderated. If the event is not moderated, make sure that the comment or reply is displayed on the page immediately after submitting it. If the event is moderated, make sure that an alert appears under the navigation bar that specifies that the comment has been submitted for approval.
- *Moderate Events* (Manual) – This use case is extremely important to the success of the iteration because it gives event managers the power to control their events at a high level. This use case will

be especially useful for business conferences so that they can make sure that all tweets and comments that are displayed are appropriated and related to the event. It is important to ensure that users can only manage the events that they created and that all of the features within this piece work correctly. To test this use case, we will do the following:

- Try to navigate to the moderate events page before creating any events on that account and ensure that the button is not in the top bar and you cannot access the page.
- Create an event and navigate to the Manage Events page before there are any tweets or comments. Ensure that the text on the page indicates that there are currently no comments or tweets for this event.
- Navigate to the Manage Events page for an event that has tweets. Click the trashcan icon next to a tweet and ensure that it is now deleted from that page and does not show up there or on the tweets page.
- Enter in a comment on the comments page of an event that is moderated and that you created. Go to the Manage Events page for that event and confirm that the comment that you just made appears on the page. Click the checkmark icon next to this comment and ensure that this comment is now gone from that page. Navigate back to the comments page for that event and confirm that the comment is now displayed on the page since you marked it as approved.
- Enter in a comment on the comments page of an event that is moderated and that you created. Go to the Manage Events page for that event and confirm that the comment that you just made appears on the page. Click the trashcan icon next to this comment and ensure that this comment is now gone from that page. Navigate back to the comments page for that event and confirm that the comment is not displayed anywhere on the page since you marked it as deleted.