­­Test Plan

Last Revision Date: 11/14/11

1. Purpose
2. Unit Testing
   1. User
   2. Network
   3. Subscription
   4. Activity
   5. Event
   6. Message
   7. Comment
   8. Review
   9. ConnAction
   10. Picture
3. Integration Testing
4. System Testing
5. Acceptance Testing
6. Revision History

1. Purpose

This test plan is to test a social networking platform for active people.

1. Unit Testing
   1. User Module: This module maintains all data manipulation of variables related to the current user.
      1. Equivalence classes:
         1. “first\_name”
            1. A string with length greater than or equal to 1 and less than 256 – Acceptable
            2. A string with length equal to 0 or greater than 256 – Error
         2. “last\_name”
            1. A string with length greater than or equal to 1 and less than 256 – Acceptable
            2. A string with length equal to 0 or greater than 256 – Error
         3. “street”
            1. A string with length greater than or equal to 1 and less than 256 – Acceptable
            2. A string with length equal to 0 or greater than 256 – Error
         4. “city”
            1. A string with length greater than or equal to 1 and less than 256 – Acceptable
            2. A string with length equal to 0 or greater than 256 – Error
      2. Checklist

[ ]: Login with correct username and password

[ ]: Login with wrong username  
[ ]: Login with wrong password  
[ ]: Login with wrong username, and password  
[ ]: Leave feedback for user  
[ ]: Change photo  
[ ]: Log out  
[ ]: View news feed  
[ ]: Register user  
[ ]: View user profile

[ ]: Edit personal information

* 1. Network Module: This module maintains all the networks in the database
     1. Equivalence classes
        1. “network\_id”:
           1. The database ID number corresponding to the appropriate network to be accessed – Acceptable
           2. The database ID number NOT corresponding to the appropriate network to be accessed - Error
        2. “area”:
           1. A string corresponding to a locale or neighborhood NOT already in the database – Acceptable
           2. A string corresponding to a locale or neighborhood already existing in the database - Error
     2. Checklist

[ ]: Create a new network

[ ]: Add a network to a user’s profile

[ ]: Ensure newly-added network displays correctly on user’s profile

[ ]: Ensure newly-added network displays correctly in user’s news feed

[ ]: Remove user from a network

[ ]: Ensure unsubscribed network no longer is displayed on user’s profile

[ ]: Ensure user no longer sees the network in his/her news feed

* 1. Subscription Module: This module maintains all of the subscriptions a user has
     1. Equivalence classes:
        1. “subscription\_id”
           1. A number generated by the database higher by one value than the last subscription\_id that was generated – Acceptable
           2. A number generated by the database less than or equal to the value of the last subscription\_id generated – Error
        2. “network\_id”
           1. The database ID number corresponding to the appropriate network to be accessed – Acceptable
           2. The database ID number NOT corresponding to the appropriate network to be accessed - Error
        3. “user\_id”
           1. The database ID number corresponding to the appropriate user to be accessed – Acceptable
           2. The database ID number NOT corresponding to the appropriate user to be accessed - Error
     2. Checklist

[ ]: Subscribe to a network

[ ]: Subscribe to an activity

[ ]: Unsubscribe from a network

[ ]: Unsubscribe from an activity

* 1. Activity Module: This module maintains all of the activities in the activities database
     1. Equivalence classes:
        1. “activity\_id”
           1. Any activity that the user who is created the ConnAction is in – Acceptable
           2. Any activity that the user who is created the ConnAction is not in – Error
     2. Checklist

[ ]: click Add new Activity button

[ ]: Try to add a new activity < 20 characters that does not exist in the activities table.

[ ]: Try to add a new activity that already exists in the activities table < 20 characters. (should result in an error message)

[ ]: Try to add an Activity that is greater than 20 characters. (Should result in an error message)

* 1. Event Module: This module maintains the events database
     1. Equivalence classes:
        1. “event\_id”
           1. A number generated by the database higher by one value than the last event\_id that was generated – Acceptable
           2. A number generated by the database less than or equal to the value of the last event\_id generated – Error
        2. “start\_date”
           1. A time that is equal to the current time or in the future – Acceptable
           2. A time that is in the past – Error
        3. “end\_date”
           1. A time that is equal to the current time or in the future – Acceptable
           2. A time that is in the past – Error
        4. “location”
           1. Any network that the user is created the ConnAction is in – Acceptable
           2. Any network that the user who is created the ConnAction is not in – Error
     2. Checklist

[ ]: Create event with valid start or end date

[ ]: Create event with invalid start or end date

[ ]: Create event in a valid location

[ ]: Create event in an invalid location

* 1. Message Module: This module maintains all data manipulation of variables related to messages.
     1. Equivalence classes:
        1. “content”
           1. A string with length greater than or equal to 1 and less than 1024 – Acceptable
           2. A string with length equal to 0 or greater than 1024 – Error
        2. “recipient”
           1. A string with length greater than or equal to 1 and less than 256 – Acceptable
           2. A string with length equal to 0 or greater than 256 – Error
     2. Checklist

[ ]: Send user message

[ ]: View message  
[ ]: Delete message  
[ ]: Cancel send message  
[ ]: Archive current message

* 1. Comment Module: This module contains all of the comments in the database
     1. Equivalence classes:
        1. “from\_user”
           1. The database ID number of the user who sent the comment – Acceptable
           2. The database ID number of not the user who sent the comment – Error
        2. “to\_user”
           1. The database ID number of the user who receives the comment – Acceptable
           2. The database ID number of the user who did not receive the comment – Error
        3. “comment”
           1. A string value of length 1 to 256 characters – Acceptable
           2. A string value of length 0 or greater than 256 characters - Error
     2. Checklist

[ ]: Click “Leave a Comment” button

[ ]: Write a message less than 256 characters

[ ]: Write a message greater than 256 characters

[ ]: Attempt to send each comment

[ ]: Ensure comment displays on correct profile and displays correct sender

* 1. Review Module: This module maintains a database of all of the reviews that users leave other users
     1. Equivalence classes:
        1. “user\_id”
           1. Any user\_id that exists in the connaction\_attending table for the ConnAction identified by the connaction\_id in this module and is not the same as from\_user – Acceptable
           2. Any number that does not correspond to a user\_id in the Users table – Error
        2. “from\_user”
           1. The ID number of the user that is submitting the review. - Acceptable
           2. Any number that does not correspond to a user\_id in the users table – Error
        3. “is\_anonymous”
           1. An integer value in the set {0, 1} -Acceptable
           2. Any value other than 0 or 1 – Error
        4. “connaction\_id”
           1. Any integer that corresponds to a Connaction\_ID in the Connaction table -Acceptable.
           2. Any Integer that does not correspond to a Connaction\_ID in the Connation Table or Any integer that does not exist in the connaction\_attending table for the connactionID identified in this module – Error
        5. “is\_positive”
           1. An integer value in the set {0, 1} -Acceptable
           2. Any value other than 0 or 1 – Error
        6. “review\_date”
           1. Any Date value that is > the start time of the ConnAction identified by the connaction\_id in this module. - Acceptable
           2. Any Date value that is < the start time of the ConnAction identified by the connaction\_id in this Module. -Error
        7. “review”
           1. Any string that is less than 4000 characters – Acceptable
           2. Any String > 4000 characters - Error
     2. Checklist

[ ]: Click Review ConnAction Button for a valid user after the start time of the connaction\_id identified in this module.

[ ]: Click the Review Button before the start time of the ConnAction identified by the connaction\_id in this module. (should result in error)

[ ]: Click the review button and try to leave a review for the user that is logged in (yourself). (should not be an option)

[ ]: Try to leave a review for someone that did not attend the ConnAction. (should not be an option)

[ ]: Try to review a user on a ConnAction that you did not attend.

[ ]: Try to leave a review that has a message longer than 4000 characters. (should result in an error)

* 1. ConnAction Module: This module maintains the different ConnActions users will have with other users.
     1. Equivalence classes:
        1. “connaction\_id”
           1. A number generated by the database higher by one value than the last connaction\_id that was generated – Acceptable
           2. A number generated by the database less than or equal to the value of the last connaction\_id generated – Error
        2. “location”
           1. Any network that the user is created the ConnAction is in – Acceptable
           2. Any network that the user who is created the ConnAction is not in – Error
        3. “activity\_id”
           1. Any activity that the user who is created the ConnAction is in – Acceptable
           2. Any activity that the user who is created the ConnAction is not in – Error
        4. “date”
           1. A time that is equal to the current time or in the future – Acceptable
           2. A time that is in the past – Error
     2. Checklist

[ ]: Click to create a ConnAction

[ ]: Attempt to select an invalid network

[ ]: Select a valid network

[ ]: Attempt to select an invalid activity

[ ]: Select a valid activity

[ ]: Attempt to select a time in the past

[ ]: Select a time that is equal to current or sometime in the future

[ ]: Click post this ConnAction

[ ]: Ensure the ConnAction is in the correct stream (location and/or activity stream)

* 1. Picture Module: This module maintains the user’s profile picture
     1. “profile\_pic”:
        1. A string value corresponding to the URL which points to where the image is hosted on the web – Acceptable
        2. A string value NOT corresponding to the URL which points to where the image is hosted on the web - Error
     2. Checklist

[ ]: Click Upload Profile Picture button

[ ]: In popup window, browse files on user’s computer and select the image to upload

[ ]: Select ‘Ok’ to upload the photo from user’s machine to website server and verify that the image now displays as the user’s profile picture

[ ]: Delete profile picture

[ ]: Click on a profile picture in a news feed to ensure that it links to the posting user’s profile

* 1. Testers: Ray and Rob

1. Integration Testing
   1. Purpose: To ensure all of the modules work and communicates correctly with the appropriate modules.
   2. Integration Testing Checklist

[ ]: Register and create a new profile

[ ]: Fill in my profile and add a picture

[ ]: Subscribe to various networks and activities

[ ]: Post a ConnAction

[ ]: Ask to Join a ConnAction

[ ]: Send a message

[ ]: Receive a message

[ ]: Leave a review

[ ]: Comment on a ConnAction

[ ]: Log out

* 1. Testers: Vincent

1. System Testing
   1. Purpose: To ensure that the website is able to use all modules and meets the requirements
   2. System Testing Checklist

[ ]: Queries take a reasonable amount of time

[ ]: All pages work (UI does not break, appropriate sections are visible)

[ ]: Security settings work

[ ]: Databases do not run out of space

[ ]: Can handle a 100+ concurrent users

* 1. Testers: Everyone

1. Acceptance Testing
   1. Purpose: To ensure the website meets the functionality requirements
   2. Acceptance Testing Checklist

[ ]: Accounts can be created

[ ]: Profiles can be edited

[ ]: Networks can be created

[ ]: Activities can be created

[ ]: Networks can be joined

[ ]: Activities can be joined

[ ]: ConnActions can be posted to the appropriate activity

[ ]: ConnActions are visible to those who are subscribed to that particular activity

[ ]: Messages can be sent

[ ]: Users can be reviewed after a ConnAction

* 1. Testers: Everyone

1. Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Person | **Date Modified** | **Section** | **Description** |
| **Ray** | 11/11/11 | 2.1, 2.6 | First Draft |
| **Rob** | 11/12/11 | 2.7, 2.9 | First Draft |
| **Kim** | 11/13/11 | 2.2, 2.10 | First Draft |
| **Vince** | 11/14/11 | 1, 2.3, 2.5, 3, 4, 5 | First Draft |
| **Dave** | 11/14/11 | 2.4, 2.8, | First Draft |