Non-functional & Functional Security Test Cases

Sprint 1:

Use case: *As a user, I want to register, so that I get a dynamic experience*.

Test Case #	F-S1000
Test Item	Ensure user can successfully register an account.
Pre-Conditions	Must first be signed out on the website.
Post-Conditions	The account fully registered into the system.
Input Specifications	● Type in Username ● Type in Email ● Type in password ● Click register button
Expected Output Specifications	● Confirmation screen that the account is registered ● A new account is recorded into database
Pass/Fail Criteria	● The user can type in and enter into the input fields ● An account is recorded into the database using those inputs
Assumptions and Constraints	● User types in an existing email
Dependencies	N/A

Test Case #	NF-S1000
Test Item	Securely register for an account
Pre-Conditions	● user is trying to register for an account ● User must be first signed out
Post-Conditions	● the creation of an account is not exposed
Input Specifications	● Type in Username ● Type in Email ● Type in password ● Click register button
Expected Output Specifications	● error screen if account creation process was insecure ● Account is made with no signs of vulnerabilities in security

Pass/Fail Criteria	● fails if user information is retrievable through sources outside of this system
Assumptions and Constraints	● Assume user is creating an account and not using OAuth
Dependencies	● N/A

Use case: *As a user, I want to login, so that I can return to my account.*

Test Case #	F-S1001
Test Item	Logging into the website.
Pre-Conditions	● Not currently signed in on the website
Post-Conditions	● User signs out of website when done
Input Specifications	● Type in Email ● Type in Password ● Click login button ● Click sign out button
Expected Output Specifications	● The website appearance and information tailored uniquely to the user
Pass/Fail Criteria	● Fails if the website in the signed out state remains the same when signed in ● Information unique to the user appears ● Fails if signing out button does not change the screen to its default signed out appearance
Assumptions and Constraints	● User has a registered account
Dependencies	F-S1000

Test Case #	NF-S1001
Test Item	Securely login to website
Pre-Conditions	● User must be first signed out
Post-Conditions	● User is logged in without user credentials not exposed to outside sources once logged in

Input Specifications	● Type in Email ● Type in Password ● Click login button
Expected Output Specifications	● an error screen if after user enters credentials the process to authenticate is insecure ● User logs in with no vulnerabilities in security during the process
Pass/Fail Criteria	● fails if authentication process exposes user credentials to sources outside of the system
Assumptions and Constraints	● User has a registered account
Dependencies	F-S1000

Use case: As a user, I want to login with Google gmail account, so that I don't have to create and remember another password.

Test Case #	F-S1000
Test Item	Logging in with Gmail account.
Pre-Conditions	● Not signed into the website
Post-Conditions	● User successfully signs out from website
Input Specifications	● Signing in to google email ● Button to sign in to gmail ● Clicking sign out button
Expected Output Specifications	● The website appearance and information tailored uniquely to the user
Pass/Fail Criteria	● Fails if website in its signed out state remains the same even when logged in to gmail ● Information unique to the user appears ● Sign out button does not change the site to its default signed out state
Assumptions and Constraints	● User has a gmail account
Dependencies	● N/A

Test Case #	NF-S1000
Test Item	Securely use gmail account for login
Pre-Conditions	● Not signed into the website
Post-Conditions	● gmail information is not insecurely exposed after gmail account is used to login
Input Specifications	● login with gmail button
Expected Output Specifications	● error screen that triggers in the middle of authentication process if it is insecure ● User is signed in without gmail information being exposed
Pass/Fail Criteria	● fails if gmail credentials are not exposed
Assumptions and Constraints	● User has a gmail account

Dependencies	● N/A

Use cases: *As a user, I want be able to reset my password, so that I don't get locked out of my account if I forget my credentials.*

Test Case #	F-S1000
Test Item	Resetting password
Pre-Conditions	● Singed out of website ● Has an account registered to system
Post-Conditions	● Password gets updated in the user's record inside database
Input Specifications	● Type in email ● Clicking the resend button ● Type in new password ● Clicking reset password button
Expected Output Specifications	● Confirmation that password changed ● Password in database is changed
Pass/Fail Criteria	● Pass if password gets updated in database ● User can type into input fields ● Resend buttons emails the link ● The link sent in works ● Clicking reset password button works
Assumptions and Constraints	● User has a registered account
Dependencies	● N/A

Test Case #	NF-S1000
Test Item	Securely change the password
Pre-Conditions	● Singed out of website
Post-Conditions	● the new password made is not exposed outside of the system
Input Specifications	● Type in email

	● Clicking the resend button ● Type in new password ● Clicking reset password button
Expected Output Specifications	● error screen when process of updating password is insecure ● Password is updated without it being leaked
Pass/Fail Criteria	● fails if new password made is exposed to sources outside of the system ● Fails if the emailed link is exposed
Assumptions and Constraints	● User has a registered account
Dependencies	● N/A

Use cases: As a user, I want to be able to create, update, and delete bucket list items so that I can reach my goals.

Test Case #	F-S1000
Test Item	Create, update, delete bucket list item
Pre-Conditions	● Signed into the website
Post-Conditions	● User has an updated bucket list
Input Specifications	● Type in a bucket list item ● Clicking delete button ● Clicking edit button ● Clicking bucket list button in navigation bar

Expected Output Specifications	● Visible changes to bucket list page
	  ○ Added items to list
	  ○ Items removed
	  ○ Items changed

Pass/Fail Criteria	● Items added to list ● Items added into database ● Items removed to list ● Items removed from database ● Items changed on the list ● The change of an item is updated on database ● User can type into input field

	● Delete button works ● Edit button works ● bucket list button in navigation bar works
Assumptions and Constraints	● Assume user interacting with CRUD functions in current session
Dependencies	● N/A

Test Case #	NF-S1000
Test Item	Submitting changes to bucket list is not slow
Pre-Conditions	● Signed into the website
Post-Conditions	● updated bucket
Input Specifications	● Type in a bucket list item ● Clicking delete button ● Clicking edit button ● Clicking bucket list button in navigation bar
Expected Output Specifications	● bucket list is updated without slowing down
Pass/Fail Criteria	● pass if one change to bucket list is less than 2 seconds
Assumptions and Constraints	● assume data is updated in database
Dependencies	● N/A

Sprint 2:

Use cases: *As a user, I want be able to create a personal profile, so that people can know more about me.*

Test Case #	F-S2000
Test Item	Edit personal profile page.
Pre-Conditions	User has authenticated account.
Post-Conditions	Test user has been removed.
Input Specifications	- upload profile image - edit bio
Expected Output Specifications	- allow upload of an image successfully - enable successful update of user bio - rendering of personal profile page in edit mode
Pass/Fail Criteria	- user can get response of profile page - upload a photo - edit a bio - successfully submit and see success response
Assumptions and Constraints	- assume image size can be resized to amount - assume user uploads an appropriate profile image
Dependencies	F-S1000

Test Case #	NF-S2000
Test Item	Secure a user's profile editing page.
Pre-Conditions	Existing user is unauthorized.
Post-Conditions	Test user has been removed.
Input Specifications	unauthorized user attempts request to edit page
Expected Output Specifications	- user is redirected - user gets response message stating unauthorized

Pass/Fail Criteria	● unauthorized user is redirected when trying to get page
Assumptions and Constraints	● assume that user knows url path to another user's edit profile page
Dependencies	F-S1000

Use cases: As a user, I want be able to have control over my account, so that I can control who can see my profile.

Test Case #	F-S2001
Test Item	Use user settings to dictate rendering.
Pre-Conditions	Authenticated user account
Post-Conditions	Remove test user account
Input Specifications	● user clicks on toggle to make account private
Expected Output Specifications	● receives a successful flash message in response
Pass/Fail Criteria	● toggle privacy settings in UI ● reflect changes in database
Assumptions and Constraints	● assume user wants to set profile to private
Dependencies	F-S1000

Test Case #	NF-S2001
Test Item	Ensure fast toggle update for changing profile privacy settings
Pre-Conditions	Test User account created
Post-Conditions	Remove User account
Input Specifications	● user clicks on toggle to make account private
Expected Output Specifications	● successful toggle of privacy settings
Pass/Fail Criteria	● Update for privacy settings displayed <= 500ms

Assumptions and Constraints	● assume user wants to change privacy settings
Dependencies	F-S1000

Use cases: *As a user, I want be able to have a personal and dynamic page, to see my past history and recent activity of my groups.*

Test Case #	F-S2002
Test Item	Display recent activity in activity feed.
Pre-Conditions	● two users with authenticated accounts ● activity in accounts regarding new bucket list task
Post-Conditions	● remove bucket list subscriptions ● remove test user accounts
Input Specifications	● user clicks on Activity Feed page
Expected Output Specifications	● display previous post based on most recent
Pass/Fail Criteria	● previous post are populated and rendered
Assumptions and Constraints	● authorized user ● previous post from friend group or recent activity
Dependencies	F-S1000

Test Case #	NF-S2002
Test Item	Ensure fast rendering of post in Activity Feed
Pre-Conditions	● user account ● create test post
Post-Conditions	● remove user account ● remove test post
Input Specifications	● user clicks to view Activity Feed
Expected Output Specifications	● user sees post in chronological order
Pass/Fail Criteria	● display post <= 2 seconds

Assumptions and Constraints	● assuming that post are already created
Dependencies	F-S1000

Use cases: *As a user, I want to connect with people with the same bucket list item, so that I can connect and accomplish more.*

Test Case #	F-S2003
Test Item	Enable the adding/removal of friends
Pre-Conditions	● Two authenticated user accounts.
Post-Conditions	● Remove authenticated accounts.
Input Specifications	● user clicks to Friends page ● user sees public accounts ● user clicks to add friends
Expected Output Specifications	● expect rendering of public accounts ● expect flash success message
Pass/Fail Criteria	● flash success message shown ● successfully add friend to db
Assumptions and Constraints	● assume public accounts existing ● assume friend acceptance
Dependencies	F-S1000

Test Case #	NF-S2003
Test Item	Ensure fast rendering of Public Accounts
Pre-Conditions	● test user ● user has profile image,bio, and a bucket list
Post-Conditions	● remove profile image, bio, and bucket list ● remove test user
Input Specifications	● user views a public account
Expected Output Specifications	● fast rendering to see the profile image

Pass/Fail Criteria	● display list of public accounts <= 2 seconds
Assumptions and Constraints	● assume user has a public account that attempting to view
Dependencies	F-S1000

Use cases: *As a user, I want to share new post with images and text, so that I can contribute to the group.*

Test Case #	F-S2004
Test Item	Create and display a post
Pre-Conditions	User test account created.
Post-Conditions	Remove User test account.
Input Specifications	● user goes to Activity Feed page ● user uploads a photo ● user enters some text ● user clicks Post
Expected Output Specifications	● new post shows up in Activity Feed ● displays image, text, and time created
Pass/Fail Criteria	● new Post includes image, text, and date created ● changes are reflected in database
Assumptions and Constraints	● Assume that user has an image to upload ● Assume image and text are appropriate
Dependencies	F-S1000

Test Case #	NF-S2004
Test Item	Ensure fast rendering of adding new post to Activity Feed
Pre-Conditions	● test user
Post-Conditions	● remove test user
Input Specifications	● user submits a new post with an image and text
Expected Output Specifications	● page successfully renders the post ● post info is reflected in the database

Pass/Fail Criteria	● display new post <= 1 second
Assumptions and Constraints	● assume that post is appropriate ● assume user has image to upload
Dependencies	F-S1000

Use cases: As a user, I want to like posts, so that I can show my support for other group members.

Test Case #	F-S2005
Test Item	Enable Like component for post
Pre-Conditions	● two authenticated accounts ● Post and Activity Feed components are functional
Post-Conditions	● Remove test post ● remove test accounts
Input Specifications	● Click 'like' icon to perform event
Expected Output Specifications	● expect component to render successfully ● 'like' to be reflected in database
Pass/Fail Criteria	● component displays state of having been 'liked' ● post in database updates and shows recent like number
Assumptions and Constraints	● Post has been created ● Post rendering successfully in Activity Feed
Dependencies	F-S1000

Test Case #	NF-S2005
Test Item	Ensure fast rendering of 'Like' component upon toggle
Pre-Conditions	● test user ● test post
Post-Conditions	● remove post ● remove user
Input Specifications	● user clicks on like component

Expected Output Specifications	● post successfully renders the 'like' state ● change is reflected in the database
Pass/Fail Criteria	● display toggle 'like' update <= 1 second ● Fails if > 1 second or change is not rendered
Assumptions and Constraints	● assume post is appropriate ● assume users are friends
Dependencies	F-S1000

Use cases: *As a user, I want to interact with other member's post with comments, so that I can respond and share with my groups.*

Test Case #	F-S2006
Test Item	Enable commenting on a post.
Pre-Conditions	● two authenticated user accounts ● existing post
Post-Conditions	● removal of post ● removal of test user accounts
Input Specifications	● click on comment component to render textarea and submit button ● user enters text and presses submit
Expected Output Specifications	● comment is rendered under the post after submission
Pass/Fail Criteria	● comment successfully shown in Activity Feed ● comment is reflected in database
Assumptions and Constraints	● Assume Activity Feed rendering post ● Assume post exist
Dependencies	F-S1000

Test Case #	NF-S2006
Test Item	Ensure fast rendering of comment
Pre-Conditions	● two test users ● post submitted

Post-Conditions	● remove test users ● remove post ● remove comment
Input Specifications	● user clicks to make a comment ● user enters comment and submits
Expected Output Specifications	● successfully renders comment
Pass/Fail Criteria	● user can successfully submit a comment ● see the change reflected ● renders in <= 2 seconds
Assumptions and Constraints	● assume comment is appropriate
Dependencies	F-S1000

Sprint 3:

Use cases: *As a user, I want to be able to add/remove friends, so that I can further connect with other users.*

Test Case #	F-S3000
Test Item	User can a remove someone added as a friend.
Pre-Conditions	● User is logged in and authenticated. ● User has a friend
Post-Conditions	● Both test accounts are removed
Input Specifications	● Search for friend user on user search. ● Select test friend to be removed. ● Click on 'Remove as Friend' button
Expected Output Specifications	● Current logged-in test user is no longer friends with the target test friend user.
Pass/Fail Criteria	● Fails if the user cannot remove the test friend as a friend.
Assumptions and Constraints	● Logged-in test user is already a friend with the target test friend.
Dependencies	F-S1001

Test Case #	NF-S3000
Test Item	User can add as many friends without performance problems client and server side.
Pre-Conditions	● User is logged in. ● There are countless test users as friends. ● Server is in stable condition
Post-Conditions	● Original test user is deleted. ● Test friend accounts are deleted.
Input Specifications	● Select user from user search. ● Click 'Add Friend' ● Repeat with multiple users ● Repeat x1000, x10000, x100000
Expected Output Specifications	● User should have exponential increases in number of friends without page loading times taking a hit.

	● Server should not have performance issues while performing the test.
Pass/Fail Criteria	● Fail if any page implementing the 'Friends' class has a longer load time than other pages. ● Fail if AWS Server monitoring displays any performance issues.
Assumptions and Constraints	● Server is originally in a relatively stable condition. ● AWS Monitoring services are working.
Dependencies	F-S1001

Use cases: *As a user, I want to be able to direct message my friends, so that I can get to know them better and make plans to accomplish bucket list items.*

Test Case #	F-S3001
Test Item	Messages should appear in chronological order.
Pre-Conditions	● Both test users are friends.
Post-Conditions	● Both test users are deleted. ● Chat history between both users is deleted.
Input Specifications	● User 1: Visits test friend page ● User 1: Selects 'Message' ● User 1: Sends random messages at random times ● User 2: Selects message ● User 2: Sends random messages at random times
Expected Output Specifications	● Messages are displayed in the message page in ascending chronological order. ● Messages have markers for the 'author'/'sender'
Pass/Fail Criteria	● Fail if any of the test messages are not in ascending chronological order by YYYY-MM-DD-HH-SS
Assumptions and Constraints	● Order of any two messages sent at the same time is not important. ● One user cannot send two messages at the same time.
Dependencies	F-S1001

Test Case #	NF-S3001
Test Item	A conversation should have indefinite amount of messages without performance issues.
Pre-Conditions	● Users are logged in ● Users have an extent chat history (x10000+ messages)
Post-Conditions	● Chat history between both users is deleted ● Both users are deleted
Input Specifications	● User 1: Visits the chat page ● User 1: Refreshes the chat page multiple times

Expected Output Specifications	● The average time for refresh should be within 2 seconds of loading time for any other page.
Pass/Fail Criteria	● Fails if the average refresh time for chat page is greater than 2 seconds of site global average loading time.
Assumptions and Constraints	● AWS deployment services are in stable condition. ● There are no other performance hindrances
Dependencies	F-S1001

Use cases: As a user, I want to be able to report offensive content, so that I can feel safe and respected.

Test Case #	F-S3002
Test Item	A user cannot report the same post multiple times.
Pre-Conditions	● User is logged in ● Post to be reported exists ● User has already reported the post.
Post-Conditions	● User is deleted. ● Post is deleted. ● Report is deleted.
Input Specifications	● User selects post from feed. ● Clicks post options ● Clicks on 'Report this post'
Expected Output Specifications	● User should be prompted whether or not they have reported the post prior to current attempt.
Pass/Fail Criteria	● Fails if user is able to make multiple reports of the same post.
Assumptions and Constraints	● N/A
Dependencies	F-S1001

Test Case #	NF-S3002
Test Item	Usability, a user should be able to navigate the report function to make a proper report as they seem fit.
Pre-Conditions	● User is logged in ● Post has not been reported by the user prior to test.
Post-Conditions	● User is deleted. ● Test post is deleted.
Input Specifications	● User should select options for post to report ● User should select 'Report this post" ● User should click tier-1 category for reporting ● User should click tier-2 category for reporting ● User should click tier-3 category for reporting, if any. ● User should be able to enter extra details for report.

	● User should be able to click 'Submit Report' button
Expected Output Specifications	● User should receive confirmation with read-back details of their report.
Pass/Fail Criteria	● Fail if user cannot reach the report reason on their first attempt without backtracking. Options unintuitive.
Assumptions and Constraints	● Assumption that user can understand pre-determined categories for reporting.
Dependencies	F-S1001

Use cases: As a user, I want to see the recent activity of my friends, so that I can make new bucket list items and stay connected.

Test Case #	F-S3003
Test Item	Refreshing should update feed if new items are available.
Pre-Conditions	● User is logged in ● User has friends with posts/activities
Post-Conditions	● User is deleted ● Friends' test posts/activities are deleted ● Friends' test accounts are deleted

Input Specifications	● User visits recent activity feed.
	● Branch 1: Browser refresh
	   ○ User should click built-in 'Refresh' button on their browser.
	● Branch 2: Site dynamic refresh
	   ○ User should scroll up when at top of activity feed.

Expected Output Specifications	● Activity feed should render new posts/activity items.
Pass/Fail Criteria	● Fails if refreshing does not render new post/activity items on user feed.
Assumptions and Constraints	● User's browser has a built-in refresh function.
Dependencies	F-S1001

Test Case #	NF-S3003
Test Item	User should be able to scroll their feed indefinitely without client-side performance hinderance.
Pre-Conditions	● User is logged in ● User is at top-most feed item. ● User's activity feed has large number of items to list.
Post-Conditions	● Feed is refreshed to reset loaded items.
Input Specifications	● User visits activity feed ● User continuously scrolls down until no more available items.
Expected Output	● User CPU performance should not increase the further down

Specifications	the feed they are.
Pass/Fail Criteria	● Fails if user's device CPU usage increases the further down they are on their feed.
Assumptions and Constraints	● No other CPU-intensive processed are being used.
Dependencies	F-S1001

Sprint 4:

Use cases: *As a user, I want to be able to find new bucket list items, so that I can have fun things to look forward to*

Test Case #	F-S4001
Test Item	Item Recommendations show up on feed
Pre-Conditions	● User has authenticated account ● Activity in users bucket list
Post-Conditions	Remove test user account
Input Specifications	● Add items to Bucket list ● Search random items to add to bucket list
Expected Output Specifications	● Allow user to add items to bucket list ● Recommend items according to users search ● Allow user to add bucket list items to list
Pass/Fail Criteria	● Fail if user is not able to add items to bucket list
Assumptions and Constraints	● User has items in bucket list ● User has search history ● User has a authenticated account
Dependencies	● N/A

Test Case #	NF-S4001
Test Item	Efficiency of item recommendations showing up
Pre-Conditions	● User has authenticated account ● User has items in users bucket list ● Server is running
Post-Conditions	Remove test user account
Input Specifications	● Add items to Bucket list ● Search random items to add to bucket list ● Refresh feed

Expected Output Specifications	● Feed should be updated in less that 2 sec for performance
Pass/Fail Criteria	● Fail if feed takes too long to load
Assumptions and Constraints	● Server is up and running ● User has previous search history
Dependencies	● N/A

Use cases: As a user, I want to learn about cool events, so that I can find events I did not know about.

Test Case #	F-S4002
Test Item	Event Recommendations show up on feed
Pre-Conditions	● User has authenticated account ● Activity in users bucket list ● Events related to those items
Post-Conditions	Remove test user account
Input Specifications	● Add items to bucket list ● Search groups for those items ● Refresh pages for recommendations
Expected Output Specifications	● User should see event recommendations in feed ● User should be able to search recommendation genre
Pass/Fail Criteria	● User can add items to list ● User gets event recommendations ● User can select recommended events
Assumptions and Constraints	● User has items in bucket list ● User has search history ● User has a authenticated account
Dependencies	● N/A

Test Case #	NF-S4002
Test Item	Reliability of event recommendations

Pre-Conditions	● User has authenticated account ● Activity in users bucket list ● Server is running
Post-Conditions	Remove test user account
Input Specifications	● Add items to bucket list ● Search groups for those items ● Refresh pages for recommendations
Expected Output Specifications	● Events/feed should be updated in less that 2 sec for performance ● Server should not fail to recommend events if multiple items added
Pass/Fail Criteria	● Fail to recommend events ● Takes too long to update the feed ● Server fail if multiple events added
Assumptions and Constraints	● Server is running ● Events related to items are in database ● User has bucket list items
Dependencies	● N/A

Use case: As a user, I want to meet new people with similar interest, so that I can have a bigger social circle.

Test Case #	F-S4003
Test Item	Friend Recommendations show up in friends list
Pre-Conditions	● User has authenticated account ● User has friends ● More than 2 user accounts ● Users friends have friends
Post-Conditions	Remove test user account
Input Specifications	● User adds friends ● User opens friends tab ● User sees recommendations
Expected Output Specifications	● User sees friends as recommendations

Pass/Fail Criteria	User can add friends in recommendations to friends list
Assumptions and Constraints	● All Users have friends ● All Users have an authenticated account
Dependencies	● N/A

Test Case #	NF-S4003
Test Item	Efficiency of Friend recommendations showing up
Pre-Conditions	● User has authenticated account ● User has Friends with multiple friends(X1000) ● Server is running
Post-Conditions	Delete test user accounts
Input Specifications	● User adds friends ● User opens friends tab ● User sees recommendations
Expected Output Specifications	● User sees friends as recommendations in less that 2 seconds ● User should be able to add friends
Pass/Fail Criteria	● Server takes too long to load recommendations ● User unable to add friends
Assumptions and Constraints	● Server is running ● User has friends(x1000) ● Friends have mutual friends
Dependencies	● N/A