

## 1. **Functionality:**

### 1.1. **USER PROFILE - FUNCTIONAL FEATURE CHOSEN BY TEAM** - *Implemented in prototype*

1.1.1. Users can view their own profile and other users' profiles

*Validation: Functional feature chosen by the team*

*Verify by: Viewing user profile*

1.1.1.1. User's own profiles will display notifications for the users (ie. notifying them of threads they have subscribed to)

*Validation: Functional feature chosen by the team*

*Verify by: Displaying notifications for the current user*

1.1.1.2. User's own profiles will display and link groups they are a member of

*Validation: Functional feature chosen by the team*

*Verify by: Displaying and linking groups the current user is a member of*

1.1.1.3. User's own profiles will display and link topics they are subscribed to.

*Validation: Functional feature chosen by the team*

*Verify by: Subscribed topics are displayed and linked on the current user's profile*

1.1.1.4. User's own profiles and other user's profiles will display comments and topics the user has created.

*Validation: Functional feature chosen by the team*

*Verify by: Display and linking topics and comments the user has created*

1.1.1.5. Users can search usernames to find a user's profile

*Validation: Functional feature chosen by the team*

*Verify by: Searching for a username.*

### 1.2. **USER AUTHENTICATION** - *Implemented in prototype*

1.2.1. Users are able to register for an account, providing a username, password, and email

*Validation: Clear implication from project requirements*

*Verify by: Registering using user's credentials*

1.2.2. Users are able to authenticate by providing a username and password that is validated by the application

*Validation: Clear implication from project requirements*

*Verify by: Logging in using credentials that were provided in registration*

1.2.2.1. Users must authenticate before viewing application web pages other than the sign in and registration page (e.g. unauthenticated users cannot view topics)

*Validation: Clear implication from project requirements*

*Verify by: Logging in using credentials that were provided in registration*

- 1.2.2.2. Users are able to logout, ending their authenticated session  
*Validation: Clear implication from project requirements*  
*Verify by: Ability to logout*

### **1.3. TOPICS AND COMMENTS - Implemented in prototype**

- 1.3.1. Authenticated users can create topic threads  
*Validation: Clear implication from project requirements*  
*Verify by: Creating or editing a topic or commenting on an existing topic.*
  - 1.3.1.1. Users are able to give the topic a title and a body.  
*Validation: Clear implication from project requirements*  
*Verify by: Creating a new topic with supplied title and body*
  - 1.3.1.2. Users are able to comment on an existing topic  
*Validation: Clear implication from project requirements*  
*Verify by: Commenting on an existing topic*
    - 1.3.1.2.1. Users are able to edit their own comments
    - 1.3.1.2.2. *Validation: Clear implication from project requirements*  
*Verify by: Ability to edit a comment that a user is the author of*
  - 1.3.1.3. Users are able to edit the title and body of topic threads they have created  
*Validation: Clear implication from project requirements*  
*Verify by: Ability to edit a topic that a user is the author of*

### **1.4. USER SUBSCRIPTION AND NOTIFICATIONS - Implemented in prototype**

- 1.4.1. Users can subscribe to a topic  
*Validation: Clear implication from project requirements*  
*Verify by: Subscribing to a topic*
  - 1.4.1.1. Users are notified of new posts on topics they are subscribed to  
*Validation: Clear implication from project requirements*  
*Verify by: Receiving notification after new comment on subscribed threads*
  - 1.4.1.2. Users can unsubscribe to a topic  
*Validation: Clear implication from project requirements*  
*Verify by: Unsubscribing to a topic*
  - 1.4.1.3. Users are automatically subscribed to topics they create  
*Validation: Clear implication from project requirements*  
*Verify by: Topic is added to author user's subscriptions after creation*

**1.5. GROUPS - Implemented in prototype**

1.5.1. Users will have the ability to create groups, supplying a group name

*Validation: Clear implication from project requirements*

*Verify by: Creating a group with required details.*

1.5.1.1. A user that is a member of a group is able to add other users to the group

*Validation: Clear implication from project requirements*

*Verify by: Add a member to a group*

1.5.1.2. Users can search other users by username for purposes of adding them to a group

*Validation: Clear implication from project requirements*

*Verify by: Searching for a user using their username.*

1.5.1.3. Users can create discussion threads in groups they are a member of that are only accessible by members of that group

*Validation: Clear implication from project requirements*

*Verify by: Creating a topic in a group*

1.5.1.4. Users can leave groups they are a member of

*Validation: Clear implication from project requirements*

*Verify by: Leaving a group the user is a part of.*

**2. Usability - Implemented in prototype**

2.1. The application will utilize Bootstrap for its design

*Validation: Need for a user friendly design*

*Verify by: Bootstrap library used for html design*

**3. Reliability - Implemented in prototype**

3.1. The application will utilize a database for its persistence model

*Validation: Need to persist and maintain data*

*Verify by: Data is stored in a database*

**4. Performance - Implemented in prototype**

**5. Supportability - Implemented in prototype**

5.1. The application will utilize a database interface (databasemanager.py)

*Validation: Clear implication from design requirements*

*Verify by: Utilization of interface for database functions*

5.2. The application will utilize SQLAlchemy for its back end database

*Validation: Need to persist and maintain data*

*Verify by: SQL Alchemy used for database*