UPair Use Cases

1. Create account
2. Log in
3. Log out
4. Select group to be added to
5. Create group
6. Accept group invite
7. Change minimum / maximum number of classes shared
8. Change specifics for class match (time, professor, subject)
9. Limit who can join a group by time / professor / subject

**Use Cases**

1. Create account
   1. System asks for username
   2. System verifies username is free
      1. ***VARIANT***: System returns failure if username is taken
      2. Continue with **Ia**
   3. System asks for password
   4. System verifies password is acceptable
      1. ***VARIANT***: System returns failure if password doesn’t meet requirements
      2. Continue with **Ic**
   5. System adds user to database
2. Log in
   1. System asks for username
   2. User inputs username
   3. System asks for password
   4. User inputs password
   5. System verifies username / password combo
      1. ***VARIANT 1***: System returns failure if username password combo does not match
      2. Continue with **IIa**
      3. ***VARIANT 2:*** Multiple consecutive sign in failures
         1. System returns lock out message
         2. System requests for password reset
         3. Continue with **Ic**
   6. System changes user status to logged in
3. Log out
   1. User selects log out option
   2. System requests for verification of log out
      1. ***VARIANT***: User cancels log out procedure
      2. User status maintained as logged in
   3. User status changed to logged out
4. Select group to be added to
   1. System loads group list database
   2. System prints group list database
   3. User selects group from list
      1. ***VARIANT 1***: User modifies list of groups by number of classes shared
      2. System returns updated list
      3. Continue with **IVc**
      4. ***VARIANT 2***: User modifies list of groups by class time
      5. System returns updated list
      6. Continue with **IVc**
      7. ***VARIANT 3***: User modifies list of groups by professor name
      8. System returns updated list
      9. Continue with **IVc**
      10. ***VARIANT 4***: User modifies list of groups by class subject
      11. System returns update list
      12. Continue with **IVc**
   4. System sends a join request to group leader
   5. System receives acceptance from group leader
      1. ***VARIANT***: System receives rejection notice from group leader
      2. System prints rejection notice to user
      3. Continue with **IVb**
   6. System adds group to user group list
5. User selects the option to create a new group
   1. System loads the group list database
   2. System requests new group name
   3. User inputs group name
   4. System adds new group to the group list database
   5. System makes user group leader
      1. ***VARIANT 1***: group leader creates join restriction based on class meet up time
      2. System updates group restriction status
      3. ***VARIANT 2***: group leader creates join restriction based on professor
      4. System updates group restriction status
      5. ***VARIANT 3***: group leader creates join restriction based on class subject
      6. System updates group restriction status