# Student Use Case Description

**Login Use Case**

**Participating Actors:** Student

**Entry Conditions:**

* The user must have a valid School User account registered in the system.

**Exit Conditions:**

* The user session for the Student user will be created in the system.

**Flow of Events:**

1. The user launches the application/screen.
2. The system displays the login screen with the following inputs:
   1. Username
   2. Password
3. User inputs their Username and Password.
4. The system verifies the Username and Password. If the account is verified, the system loads the home screen for the User account.

**Alternative Flows:**

AF-S4: If the Username/Password is invalid.

1. The system displays a message saying “Invalid Username/Password! Please try again!”.
2. The system returns to step 2 of the Original Flow.

**Exceptions:**

EX1: If the Username/Password is invalid for 3 times.

1. The system displays with a message saying “Invalid Username/Password for 3 times! Account has been locked, please contact your system administrator or your lecturer!”.
2. The system returns to step 2 of the Original Flow.

EX2: If the system/internet network is not available/down.

1. A message should be shown “System network down! Please try again later or contact your system administrator or your lecturer!”.

**Logout Use Case**

**Participating Actors:** Student

**Entry Conditions:**

1. The user must have a valid Student account registered in the system.
2. The user must be already logged into the system.

**Exit Conditions:**

* The current user session for the Student user will be ended in the system.

**Flow of Events:**

1. The user launches the application/screen.
2. The user presses the ‘Logout’ button.
3. A popup box will appear with the message “Are you sure you want to log out?” with the options “Yes” and “No”.
4. If the user selects “Yes”, the system will end the Student user’s session in the system.

**Alternative Flows:**

AF-S4: If the user selects “No” to cancel the logout process.

1. The popup box will be closed.
2. The system returns to step 1 of the Original Flow.

**Exceptions:**

EX1: If the system/internet network is not available/down.

1. A message should be shown “System network down! Please try again later or contact your system administrator or your lecturer!”.

**Link to Facebook Accounts Use Case**

**Participating Actors:** Student

**Entry Conditions:**

1. The user must have a valid School User account registered in the system.
2. The user must be already logged into the system.
3. The user must have a valid Facebook account.

**Exit Conditions:**

* Facebook account session will be created, all the posts feed from the user timeline will be visible and able to post Facebook post for the user.

**Flow of Events:**

1. The user launches the application/screen.
2. The user selects “Linked Accounts” button.
3. The system will retrieve the list of accounts that can linked.
4. The user selects “Facebook”.
5. The system displays the login screen with the following inputs:
   1. Username
   2. Password
6. User inputs their Username and Password.
7. The system verifies the Username and Password. If the account is verified, A Advanced Permissions popup box will appear with the message “Facebook would like to manage Pages?” with the options “OK” and “Not Now”.
8. If the user selects “OK”, the system result will be shown by having a tick beside Facebook to show that Facebook has successfully linked.

**Alternative Flows:**

AF-S7: If the Username/Password is invalid.

1. The system displays a message saying “Invalid Username/Password! Please try again!”.
2. The system returns to step 2 of the Original Flow.

AF-S8: If user select the options “Not Now”.

1. The system returns to step 2 of the Original Flow.

**Exceptions:**

EX1: If the system/internet network is not available/down.

1. A message should be shown “System network down! Please try again later or contact your system administrator!”.

**Link to Twitter Accounts Use Case**

**Participating Actors:** Student

**Entry Conditions:**

1. The user must have a valid School User account registered in the system.
2. The user must be already logged into the system.
3. The user must have a valid Twitter account.

**Exit Conditions:**

* Twitter account session will be created, and able to read Tweets from the user’s timeline and post Tweets for the user.

**Flow of Events:**

1. The user launches the application/screen.
2. The user selects “Linked Accounts” button.
3. The system will retrieve the list of accounts that can linked.
4. The user selects “Twitter”.
5. The system displays the login screen with the following inputs:
   1. Username
   2. Password
6. User inputs their Username and Password.
7. The system verifies the Username and Password. If the account is verified, A Advanced Permissions popup box will appear with the message “Authorize School User to use your Twitter account?” with the options “Authorize” and “Cancel”.
8. If the user selects “Authorize”, the system result will be shown by having a tick beside Twitter to show that Twitter has successfully linked.

**Alternative Flows:**

AF-S7: If the Username/Password is invalid.

1. The system displays a message saying “Invalid Username/Password! Please try again!”.
2. The system returns to step 2 of the Original Flow.

AF-S8: If user select the options “Cancel”.

1. The system returns to step 2 of the Original Flow.

**Exceptions:**

EX1: If the system/internet network is not available/down.

1. A message should be shown “System network down! Please try again later or contact your system administrator!”.

**View Journal Use Case**

**Participating Actors:** Student

**Entry Conditions:**

1. The user must have a valid Student account registered in the system.
2. The user must be already logged into the system.

**Exit Conditions:**

1. System will show the list of journal entries that the student submitted.

**Flow of Events:**

1. The user launches the application/screen.
2. The user selects “View journal” button.
3. If the system user’s entries exist, System will show the list of journal with sorted by date that student submitted.

**Alternative Flows:**

AF-S3: If the user does not have any entries.

1. The system displays a message saying “No entries yet. Please create your own journal first and try again!”.

The system returns to step 1 of the Original Flow.

**Exceptions:**

EX1: If the system/internet network is not available/down.

1. A message should be shown “System network down! Please try again later or contact your system administrator!”.

**Comment Journal Use Case**

**Participating Actors:** Student

**Entry Conditions:**

1. The user must have a valid Student account registered in the system.
2. The user must be already logged into the system.
3. Journal must have an entry or entries.

**Exit Conditions:**

* System must notify the both parties who has comment/reply the journal.

**Flow of Events:**

1. The user launches the application/screen.
2. The user selects “View journal” button.
3. If the system user’s entries exist, System will show the list of journal with sorted by date that student submitted.
4. User selects his/her post.
5. System will show the full details of that post that user select with a comment button.
6. User selects comment button.
7. User select submit button after finish key in his/her comment.
8. A successfully message will be shown “Comment successfully submitted.”

**Alternative Flows:**

AF-S3: If the user does not have any entries.

1. The system displays a message saying “No entries yet. Please create your own journal first and try again!”.

The system returns to step 1 of the Original Flow.

AF-S8: If the comment field is an empty field.

1. The system displays a popup dialog message saying “input field cannot be empty. Please key in your comment and try submit again”.

The system returns to step 6 of the Original Flow.

**Exceptions:**

EX1: If the system/internet network is not available/down.

1. A message should be shown “System network down! Please try again later or contact your system administrator!”.

**Create Journal Use Case**

**Participating Actors:** Student

**Entry Conditions:**

1. The user must have a valid Student account registered in the system.
2. The user must be already logged into the system.

**Exit Conditions:**

* New journal entries will be registered in to the system database.

**Flow of Events:**

1. The user launches the application/screen.
2. The user selects “Create journal” button.
3. The user selects “submit” button after finish keying their own entry.
4. A successfully message should be shown “Entry has successfully submitted”!

**Alternative Flows:**

AF-S4: If the entry is an empty field.

* The system displays a message saying “input field cannot be empty. Please key in your journal and try submit again”.

The system returns to step 3 of the Original Flow.

**Exceptions:**

EX1: If the system/internet network is not available/down.

* A message should be shown “System network down! Please try again later or contact your system administrator!”.

|  |  |
| --- | --- |
| Use case no. |  |
| Name |  |
| Description |  |
| Version | 1.0.0 |
| Last edited | 05-September-2016 |
| Participating actor(s) |  |
| Entry condition(s) |  |
| Exit condition(s) |  |
| Main flow |  |
| Alternate flow |  |
| Exception(s) | EX1: The server is not available to process the user request   1. The system displays the message “Server is busy, please try again later!” 2. The system exits logout use case |