**SIMPLE FACEBOOK USE CASES**

**Uses case name:** Register

**Actor:** New user

**Goal:** Let new user to create own account to login Simple Facebook

**Main success narrative:**

1. Ask E-mail, password, name, age, birthday, phone, gender.
2. Send E-mail to hash function and get position of hash table.
3. If there are same position, this data will keep in tree.

**Alternative narrative 1:**

1. If user input the same E-mail that have in data structure.
2. Print error message and ask again.

**Alternative narrative 2:**

1. Password must contain at least one character and one number, some allowed punctuation.
2. But if not, print error and ask again.

**Uses case name:** Sign in

**Actor:** User that have account of simple facebook

**Goal:** Let user to input E-mail and password to access to Simple Facebook

**Main success narrative:**

1. Ask E-mail and password. (\*Note: Our ID is E-mail\*)
2. send E-mail to hash function and search in hash table.
3. If found user data, let user login.

**Alternative narrative 1:**

1. If doesn’t found data of user by E-mail.
2. Print not found message and ask again or suggest for register.

**Alternative narrative 2:**

1. If found data of user by E-mail. But, the password doesn’t correctly.
2. Print an error message and ask again.

**Uses case name:** Suggest friend

**Actor:** User that sign in Simple Facebook already

**Goal:** Suggest new friend to user

**Main success narrative:**

1. Check another user that may relative to user that login.
2. Print suggest friend to user.

**Alternative narrative 1:**

1. If there are friends of friend.
2. Suggest friend of friend to user.

**Alternative narrative 2:**

1. If user don’t have any friend or any friend of friend.
2. Random suggest friend to user.

**Alternative narrative 3:**

1. If random but can’t found any user.
2. Print that don’t have any friends to suggest.

**Uses case name:** Update Status

**Actor:** User that sign in Simple Facebook already

**Goal:** Let user to update status on Simple Facebook

**Main success narrative:**

1. Get text status that user want to update.
2. Keep text that user input into data structure.
3. When user’s friend signs in to our Simple Facebook, Our program should manage to display that status to user’s friend.

**Uses case name:** Comment

**Actor:** User that sign in Simple Facebook already.

**Goal:** Let user to comment status that current display.

**Main success narrative:**

1. Get text comment that user want to comment.
2. Keep text into data structure of status that now display on the screen.
3. Can display both status and comment correctly.

**Uses case name:** Find friends

**Actor:** User that sign in Simple Facebook already.

**Goal:** Let user to find another user by E-mail.

**Main success narrative:**

1. Get E-mail of another user that want to find.
2. Send to hash function and get address of data back.
3. Display that user’s profile.
4. Then ask user to add friend or delete friend.

**Alternative narrative 1:**

1. If not found user.
2. Print error message.

**Alternative narrative 2:**

1. If both user are friend.
2. Allowed user to use command only delete friends or home.

**Alternative narrative 3:**

1. If both user are not friend.
2. Allowed user to use command only add friends or home.

**Uses case name:** Add friends

**Actor:** User that sign in Simple Facebook already and find another user.

**Goal:** Let user to add another user to be friend by send pending request.

**Main success narrative:**

1. Get command add friend from user.
2. Check that both user are not friends.
3. Keep pending request to user that we want to be friend.

**Alternative narrative 1:**

1. If both users are friend already.
2. Print error message and suggest command for delete friend.

**Uses case name:** Pending Request Friend

**Actor:** User that sign in Simple Facebook already.

**Goal:** Let user to check pending request and add to friend.

**Main success narrative:**

1. See which user are send pending request.
2. Ask user to accept or deny.
3. If user accept, manage to data to keep both user to be friend.

**Alternative narrative 1:**

1. If user deny to be friend.
2. Reject that data and let user to see another friend.

**Alternative narrative 2:**

1. User can’t ignore current pending request to see next pending request.
2. User must to decide current pending request to be friend or not.
3. Or can ignore it by go back to home.

**Uses case name:** Delete friends

**Actor:** User that sign in Simple Facebook already and find another user.

**Goal:** Let user to add another user to be friend.

**Main success narrative:**

1. Get command delete friend from user.
2. Check that both user are friends.
3. Manage data to keep both data of user to be unfriend.

**Alternative narrative 1:**

1. If both users are not friend.
2. Print error message and suggest command for add friend.