Social Networking App Design Manual

By

Robert Timmons

Name: Robert Timmons
Student ID: C00120820
Project Supervisor: Dr Chris Meudec
Date: 16/01/12

Table of Contents

Introduction

3

Purpose

This document describe the design of the social network application for an android tablet device.

Overview

This document will be broken into four main sections:

- High level design of the project, which contains the Data Structures, Architecture and the UI prototypes.
- Low level design of the project, describes domain diagram, system sequence diagrams and interaction diagram
- How to go about connecting to the social networks
- How to go about uploading Data to the social networks.

High Level Detail Design

Data Structures

User Information Structure
The application will store the details of the current user's information for allowing the application access to the user's social network pages.

Details that will be required:

User Information Component	Data Type	Sample Input	Comment
Username	String	Jason	stores the users specified username for logging into the application
Password	String	Pass1234	stores the users specified password for logging into the application
Facebook Username/email	String	example@hotmail.com	stores the users email address or user name for their Facebook account
Facebook Password	String	Kkevsfp	stores the users password for their Facebook account
Google+ Username/email	String	example@hotmail.com	stores the users email address or user name for their Google+ account
Google+ Password	String	Firefly00	stores the users password for their Google+ account
Twitter Username/email	String	example@hotmail.com	stores the users email address or user name for their Google+ account
Twitter Password	String	Firefly00	stores the users password for their Twitter account

Architecture
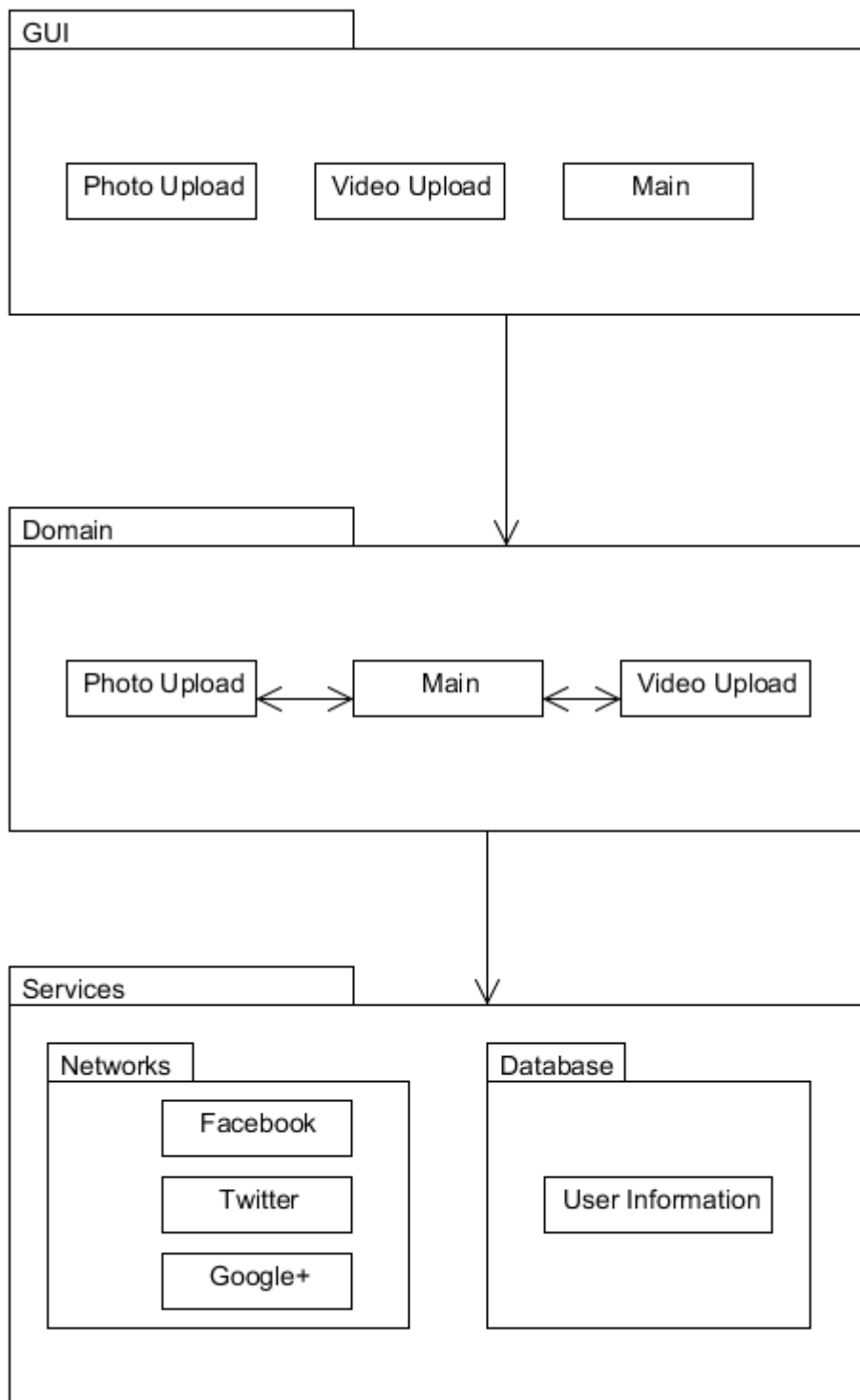
Interface Prototypes

Login Screen

This screen presents the user with a login screen upon loading the app.

The user enters their account name and password into the textboxes supplied. The user selects the Login button, which checks the details form the username and password textboxes against the details from the user's details.

The user can create a new account by selecting the create account link.

The Keep me logged in check box allows the user to not have to log into the app every time the app is run.

New Feed Screen

This screen displays the news feeds from the users Twitter, Facebook and Google+ account on the one screen. The news feeds will be displayed in order of time received. The news feeds can be scrolled through.

The user can update their status by typing it into the status textbox and clicking the update button.

The user can move to the other features by selecting one from the top of the screen.
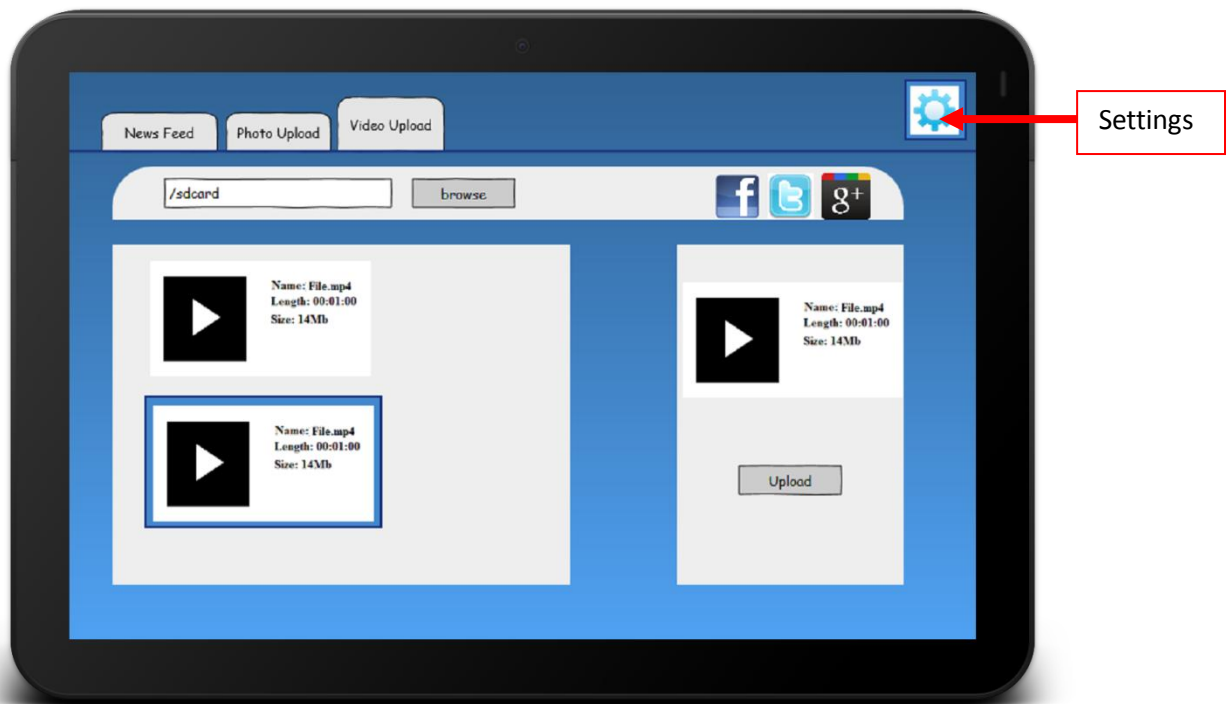
Photo Upload

This screen allows the user to upload photos to their social network accounts. The user can select the location of the images by either typing in the file path or bring up a clicking the browse button to bring up a GUI of the files on the storage device.

To select an image to be upload the user clicks the image they wish to be uploaded, a ticked check box at the bottom right corner show that the image has been selected.

To unselect an image to be upload the user clicks the image they wish not to be uploaded, check box becomes unchecked.

When the user has finished selecting the images they wish to upload, they then click the upload button, which uploads the images to the user's accounts.

Video Upload

This screen allows the user to upload photos to their social network accounts. The user can select the location of the images by either typing in the file path or bring up a clicking the browse button to bring up a GUI of the files on the storage device.

To select an image to be upload the user clicks the image they wish to be uploaded, a ticked check box at the bottom right corner show that the image has been selected.

To unselect an image to be upload the user clicks the image they wish not to be uploaded, check box becomes unchecked.

When the user has finished selecting the images they wish to upload, they then click the upload button, which uploads the images to the user's accounts.

Settings Button

The settings button brings up a list of the app settings. From there the user can edit their app username and password and their social network account username and password. The user can also select the refresh rate of the news feed.

Low Level Detail Design

Use Cases
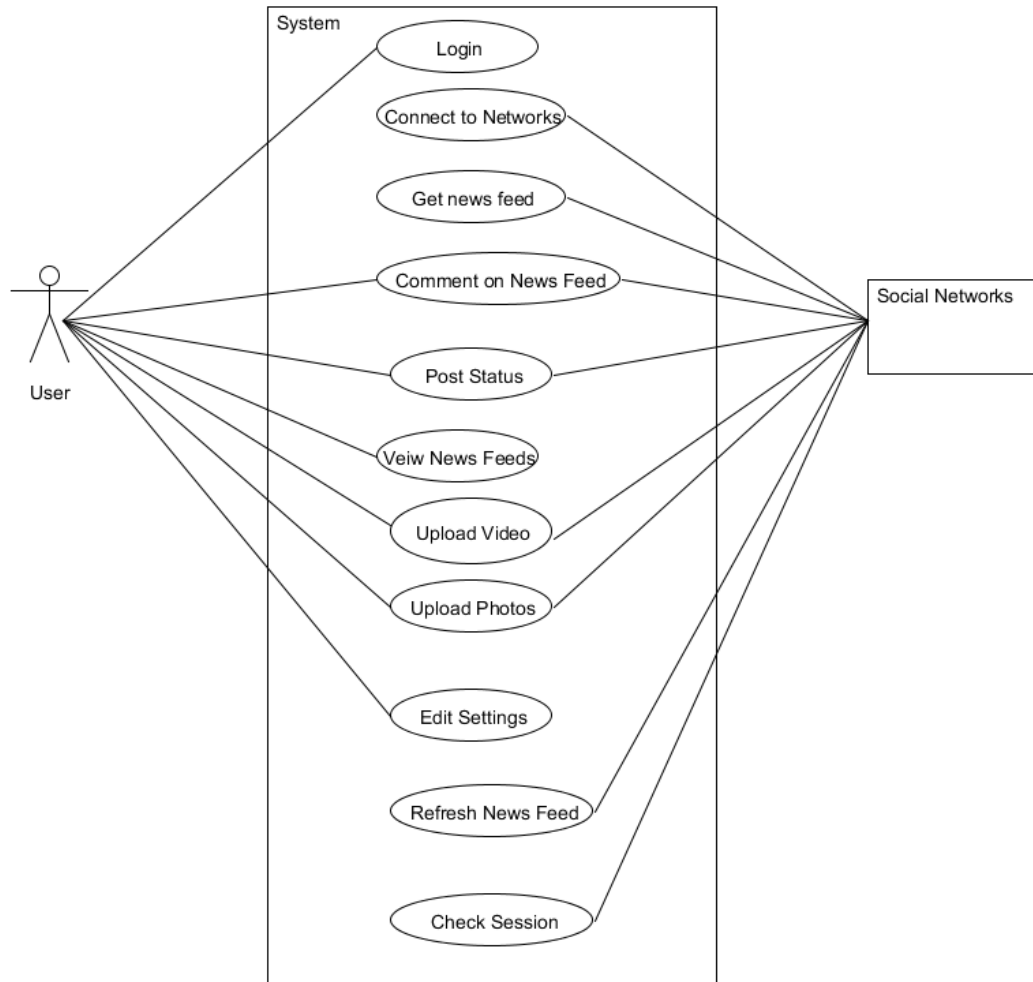
Detailed Use Cases

Name: Comment on News feed

Actor: User, Social Networks

Description:

This use case begins when the user selects an entry in the list of news feed they wish to comment on. The user then enters their message in the textbox supplied. The message is then uploaded to the social network.

Main Success Scenario:

1. The user selects the post they wish to comment on.
2. The user enters their comment in the textbox supplied
3. The user posts their comment

3a. The app did not make a connection to the social network

1. The user is prompted with an error message.

Name: Post Status

Actor: User, Social Networks

Description:

This use case begins when the user enters a message in the status update textbox. When the user selects to post the message, the message is then uploaded to the users account.

Main Success Scenario:

1. The user enters in their status in the textbox
2. The user selects to post status

2a. The app did not make a connection to the social network

2. The user is prompted with an error message.

Name: View News Feeds

Actor: User

Description:

This use case begins after the app has retrieved and displayed to the device. The user can scroll through the list of news feeds by using the touch function of the device.

Main Success Scenario:

1. The user uses the touch function to scroll through the data displayed on screen

Name: Edit Settings

Actor: User

Description: This use case begins when the user selects the settings option. The user can set and change their account details, and set the way the application updates the news feeds.

Main Success Scenario:

1. The user selects the fields they wish to edit
2. The user enters the new data.
3. The user saves their new settings

Name: Connect to Networks
Actor: User Actor: User, Social Networks
Description: This use case begins when the user signs into the application. A connection is made to the users Google+, Facebook and Twitter accounts if they have their details set up on the app.

Main Success Scenario:

1.   The application sends a connection request to the social networks, providing the users account detail and the applications unique ID

2.   The Social networks responds, acknowledging the applications request

Alternatives:

2a. The Social networks do not acknowledge one or more of the details supplied by the application.

1. Error is displayed to the user stating the error occurred.
2. The application allows the user to fix the problem.
3. Go back to main step 1

Name: Get News Feeds

Actor: Social Networks

Description: This use case begins after a connection has been made to the each of the Social networks. The application retrieves the latest news feeds from each of the social networks.

Main Success Scenario:

1.   The social networks send the users news feed to the application

2.   The application displays all of the news feeds to screen in date order.

Alternatives:

1a The app did not make a connection to the social network

    3.   This use case does not continue.

2a There are no news feeds

    1.   The app continues as normal

Name: Login to App

Actor: User

Description:

Main Success Scenario:

1.   The user enters their password

2.   The user is logged into the app

Alternatives:

1a. The user does not have an account.

1.   The user can create an account
2.   User creates an account
3.   Go back to main step 2

2a. The user's password was incorrect.

1.   An error message is displayed and the user is prompted to enter password again
2.   Go back to main step 1

Name: Refresh News Feeds

Actor: Social Networks

Description:

Main Success Scenario:

1.   The app checks the refresh time
2.   The app then gets new news feeds from the social networks and adds them to the top of the news feeds list

1a The app did not make a connection to the social network

1.   This use case does not continue.

Name: Upload Video

Actor: User, Social Networks

Description:

Main Success Scenario:

1. User selects the video
2. User uploads video
3. Video sent to users social network accounts

1a The app did not make a connection to the social network

1. No videos to select.
2. User changes file path
3. Go back to main step 1

2a The app could not make a connection to social network

1. The user is presented with an error message and prompted to try again.
2. Go back to main step 2

Name: Upload Photo

Actor: User, Social Networks

Description:

Main Success Scenario:

1. User selects the photos
2. User uploads photos
3. Photos sent to users social network accounts

1a The app did not make a connection to the social network

1. No Photos to select.
2. User changes file path
3. Go back to main step 1

2a The app could not make a connection to social network

1. The user is presented with an error message and prompted to try again.
2. Go back to main step 2

Name: Check Session
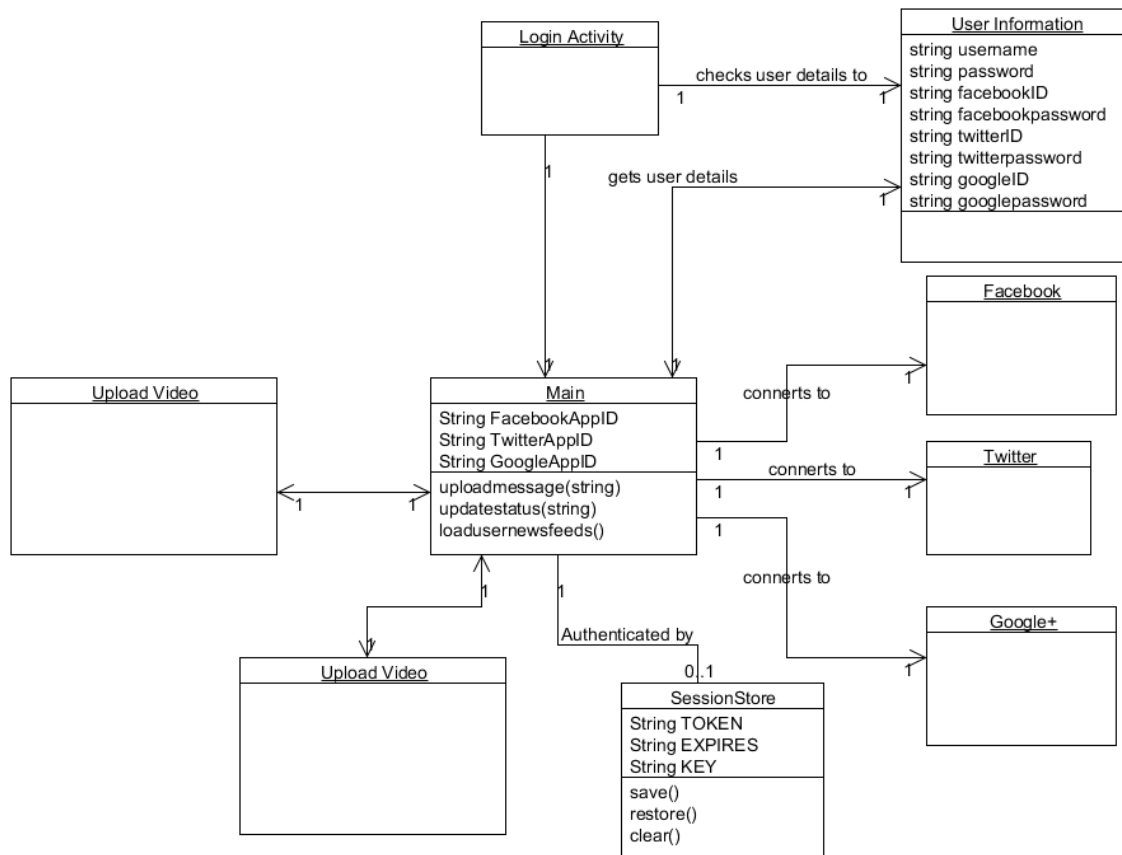
Actor: Social Networks

Description:

When the app is connecting to the social networking sites, it first checks whether it already has an existing connection to them

Main Success Scenario:
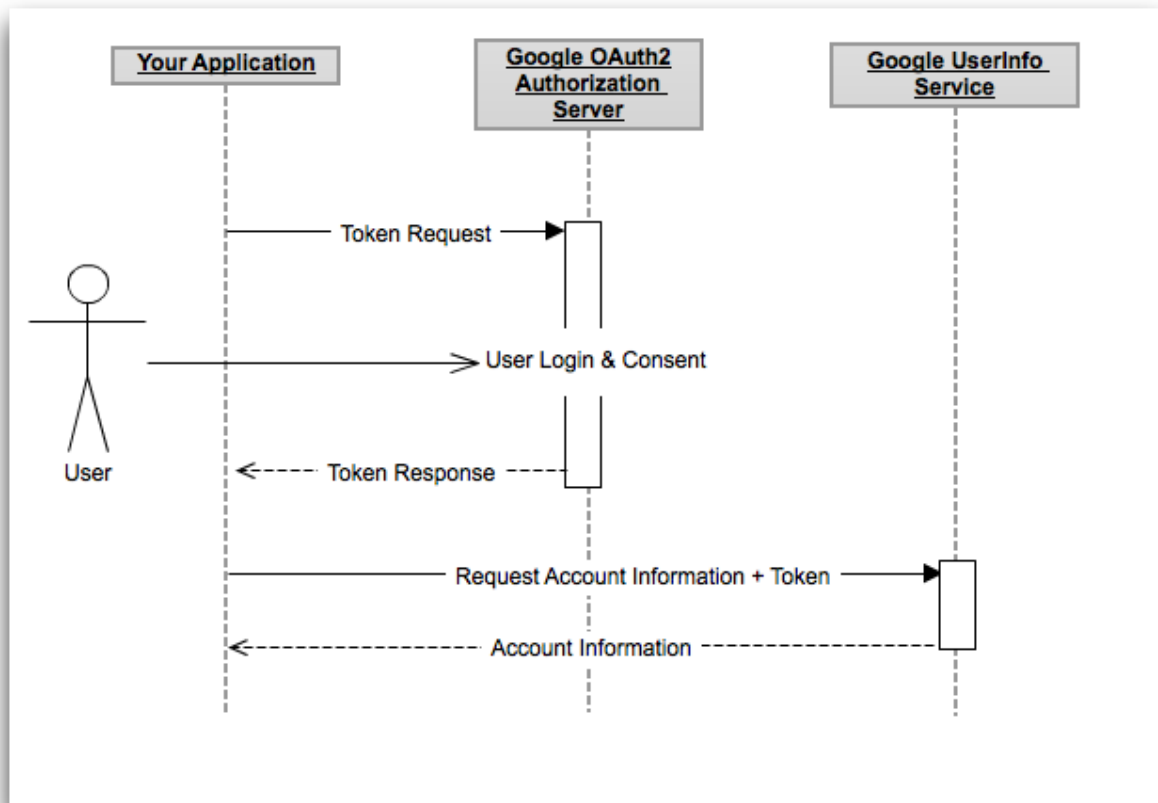
1. The app checks to see if a session is active between the app and social networks
2. The app gets news feeds

1a. There is no existing session

1. The app connects to the social networks
2. The app stores the session
3. Go back to main step 2

Domain Diagram

System Sequence Diagram
Connect to Google+

[google1]

Connect to Facebook
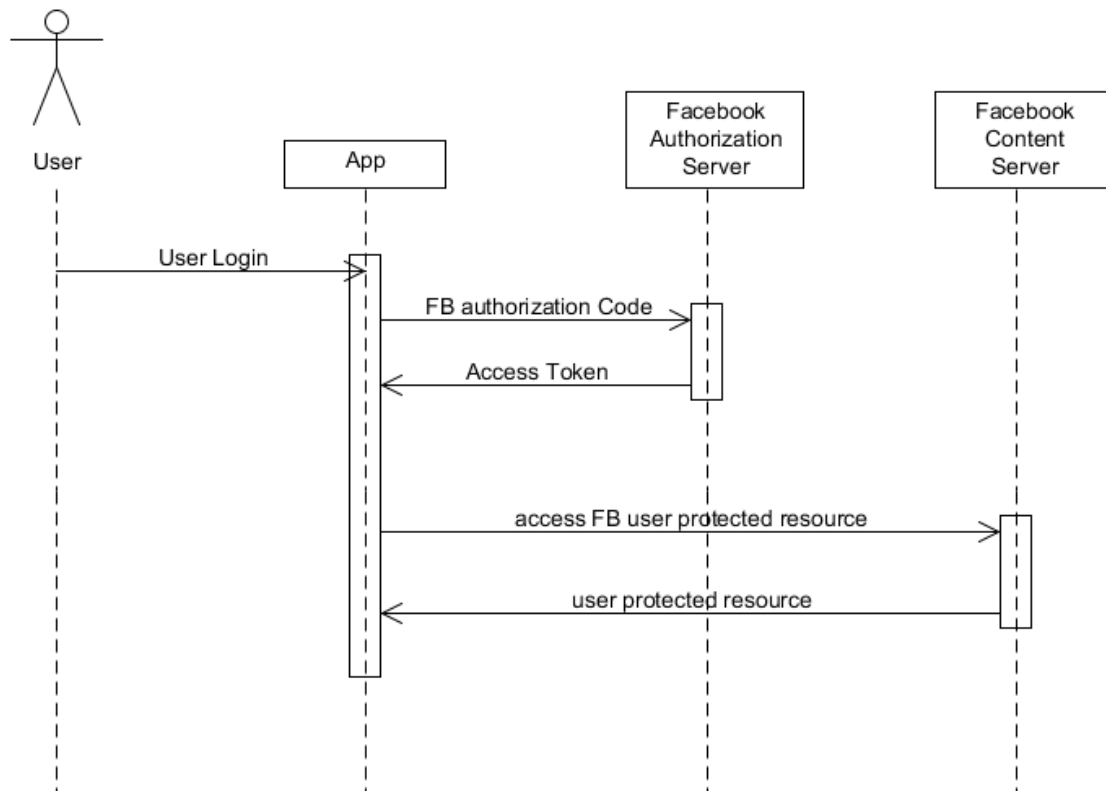
Connect to Twitter

Interaction Diagram

User	App	User Information	Facebook	Twitter	Google+

Login(Username, Password)

CheckDetails(Username, Password)

success(bool)

If success = true

facebookConnect(AppID)

facebookConnection(Token)

twitterConnect(AppID)

twitterConnection(Token)

googleConnect(AppID)

googleConnection(Token)

Session

saveSession(token)

returnSuccess()
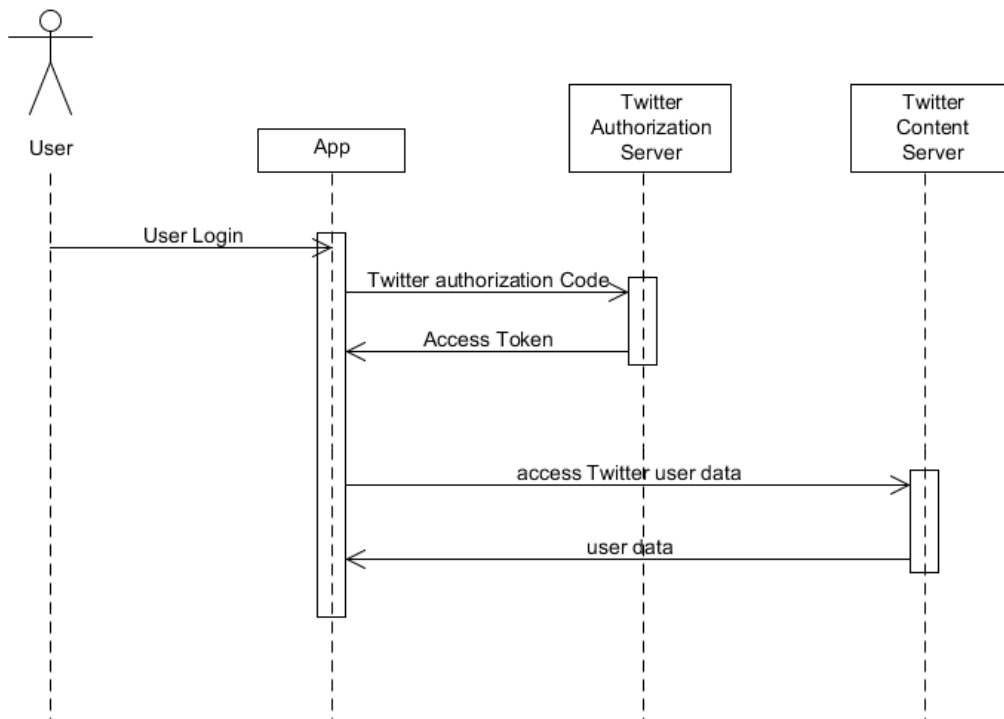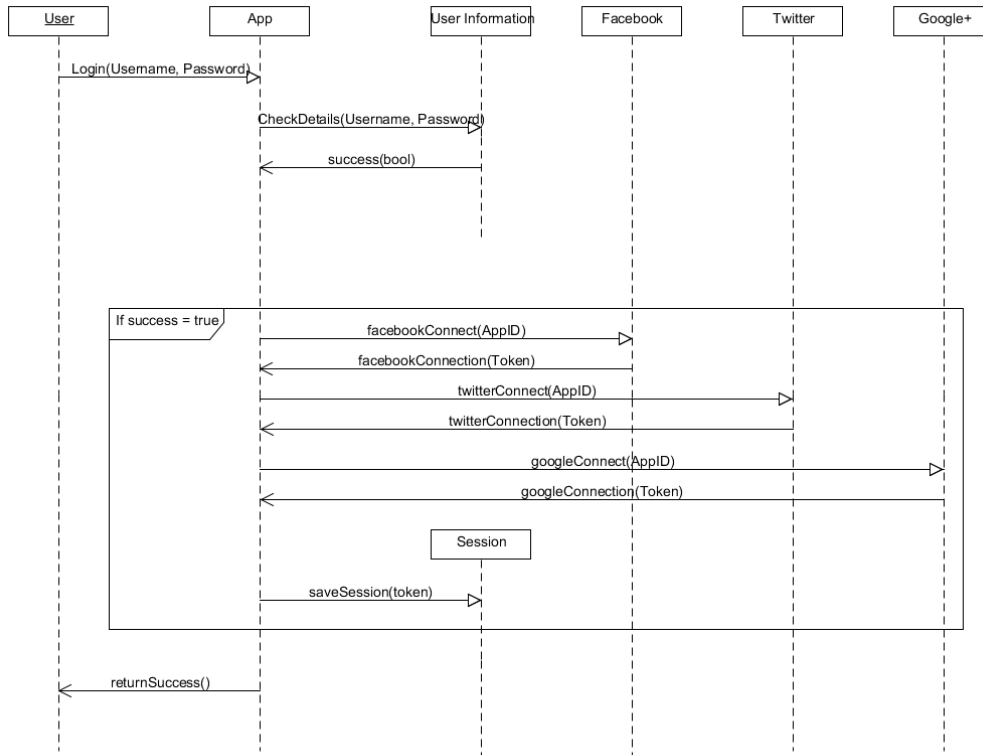
Connecting to social networks

For apps to connect to the each of the social networks api, unique Ids are required. These ID's are sent to the social networks, who verify whether the app is valid or not.

Google+ connection

"Google+ uses OAuth 2.0 endpoint to gain access to a Google API. It's a Python web application running on AppEngine.

1. *When the application loads, it shows the user a "Login" link.*

2. *When you click that link, you are asked to login to Google and asked to release basic account information to the application (user consent).*

3. *If you grant consent, the application receives an access token.*

4. *Once it has the access token, the application presents the access token to the Google API that provides basic account information* (`https://www.googleapis.com/oauth2/v1/userinfo`)

5. *The application renders the basic account information in a simple table."*

[google1]

Using Google's API allows for improved security, and support for making authenticated calls, which will allow for easy access to Goolge+ features. The API supplies all the libraries which are currently available to interact with on Google+.

[google02]

Facebook connection

Facebook uses its own unique ID which it supplies to app developers. For extra security Facebook requires the user to supply the Android application signature, which is got form using the Keytool supplied with the android SDK and using the command line

```
keytool -exportcert -alias androiddebugkey -keystore
~/.android/debug.keystore | openssl sha1 -binary | openssl base64
```

Commands for connection set up :

```
URL command:
https://www.facebook.com/dialog/oauth?client_id=YOUR_APP_ID&redirect_uri=YOUR_URL
```

[Facebook01]

The Facebook API supplies all the necessary libraries for connecting to and interacting with data on Facebook. Using the supplied API allows for more secure calls and easier access.

Twitter Connection

Twitter supplies OAuth Token on request from the application developer.

URL code for gaining access to Twitter:

Description	Command URL
Request token URL	https://api.twitter.com/oauth/request_token
Authorize URL	https://api.twitter.com/oauth/authorize
Access token URL	https://api.twitter.com/oauth/access_token

[Twitter01]

For this app to connect to Twitter we will need to use Twitters REST API. This API supplies all the necessary libraries we need for the functions supplied in this app.

[Twitter02]

Photo Upload

For Uploading photos to the social networking sites, the app must first have made a connection to the sites, otherwise the sites will not validate the apps upload call.

Facebook

To upload photos to Facebook:

Command	Description	Arguments
`//graph.facebook.com/ALBUM_ID/photos`	Upload a photo to an album	`name,message`

To publish a 'photo' object you need

- a valid `access token`
- `publish_stream` permission

With that granted, you can upload a photo by issuing an HTTP POST request with the photo content and an optional description to one these to Graph API connections:

- https://graph.facebook.com/USER_ID/photos - The photo will be published to an album created for your app. We automatically create an album for your app if it does not already exist. All photos uploaded this way will then be added to this same album.
- https://graph.facebook.com/ALBUM_ID/photos - The photo will be published to a specific, existing photo album, represented by the ALBUM_ID. Regular albums have a size limit of 200 photos. Default application albums have a size limit of 1000 photos.

[Facebook02]

Facebook Photo objects have the following fields

Name	Description	Permissions	Returns
id	The photo ID	Generic access_token or user_photos or friend_photos	string
from	The profile (user or page) that posted this photo	Generic access_token or user_photos or friend_photos	Object containing id and the name fields
tags	The tagged user and their positions in this photo	Generic access_token or user_photos or friend_photos	Array of objects, the x and x coordinates are percentages from the left and top edges of the photo, respectively
Name	The user provided caption given to this photo - do not include advertising in this field	Generic access_token or user_photos or friend_photos	string
Icon	The icon that Facebook displays when photos are published to the Feed	generic `access_token` or `user_photos` or `friend_photos`	`string` representing a valid URL
picture	The thumbnail-sized source of the photo	generic `access_token` or `user_photos` or `friend_photos`	`string` representing a valid URL
Source	The source image of the photo - currently this can have a maximum width or height of 720px	generic `access_token` or `user_photos` or `friend_photos`	`string` representing a valid URL
height	The height of the photo in pixels	generic `access_token` or `user_photos` or `friend_photos`	number
Width	The width of the photo in pixels	generic `access_token` or `user_photos` or `friend_photos`	number
images	The 4 different stored representations of the photo	Requires `access_token`	`array` of objects, containing `height`, `width`, and `source` fields
Link	A link to the photo on Facebook	generic `access_token` or `user_photos` or `friend_photos`	`string` representing a valid UR
`created_time`	The time the photo was initially published	generic `access_token` or `user_photos` or `friend_photos`	`string` containing ISO-8601 date-time
`updated_time`	The last time the photo or its caption was updated	generic `access_token` or `user_photos` or `friend_photos`	`string` containing ISO-8601 date-time
position	The position of this photo in the album	generic `access_token` or `user_photos` or `friend_photos`	number

[Facebook02]

Twitter
For uploading photos to twitter

Resource URL

https://upload.twitter.com/1/statuses/update_with_media.format

Parameters for uploading photos:

Status	*The text of your status update. URL encode as necessary. t.co link wrapping may affect character counts if the post contains URLs. You must additionally account for the characters_reserved_per_media per uploaded media, additionally accounting for space characters in between finalized URLs.*
media[]	*Up to max_media_per_upload files may be specified in the request, each named media[]. Supported image formats are PNG, JPG and GIF. Animated GIFs are not supported.*

[Twitter03]

When uploading photos, they must match the requirements that are stated in the documentary. Twitter can only handle PNG, JPG and GIF type file.

Video Upload

For Uploading videos to the social networking sites, the app must first have made a connection to the sites, otherwise the sites will not validate the apps upload call.

Facebook

In order to upload a video from your application, the user must grant your application the `publish_stream` extended permission.

For video upload using the video.upload call to http://api-video.facebook.com server.

The videos must be in the format specified in their documentation and the aspect ratio of the video must be between 9x16 and 16x9, and the video cannot exceed 1024MB or 180 minutes in length.

[Facebook03]

The Facebook API has detailed all the requirements needed for uploading videos from the API calls needed to the video requirements.

Fields

The Facebook `Video` object has the following fields.

Name	Description	Permissions	Returns
`id`	The video ID	`user_videos`	`string`
`from`	The profile (user or page) that created the video	`user_videos`	object containing `id` and `name` fields
`tags`	The users who are tagged in this video	`user_videos`	`array` of objects containing `id` and `name` fields
`name`	The video title or caption	`user_videos`	`string`
`description`	The description of the video	`user_videos`	`string`
`picture`	The URL for the thumbnail picture for the video	`user_videos`	`string`
`embed_html`	The html element that may be embedded in an Web page to play the video	`user_videos`	`string` containing a valid URL
`icon`	The icon that Facebook displays when video are published to the Feed	`user_videos`	`string` containing a valid URL
`source`	A URL to the raw, playable video file	`user_videos`	`string` containing a valid URL
`created_time`	The time the video was initially published	`user_videos`	`string` containing ISO-8601 date-time
`updated_time`	The last time the video or its caption were updated	`user_videos`	`string` containing ISO-8601 date-time
`comments`	All of the comments on this video	`user_videos`	`array` of objects containing `id`, `from`, `message`, `created_time`, and `likes` fields

[Facebook03]

Goolge+

Since Google have not released detail on most of the features, I cannot design for uploading videos to Google+. During the space of this project if the features needed to complete the google+ features, then I will design the video upload to use Youtube.

Twitter

Twitter does not support video upload, so there will be no need to design for this.

Glossary

API - application programming interface, essential tools needed to allow software to interact with other different software.

GUI – graphical user interface, a user interface that allows users to interact with electronic devices by the use of images.

Bibliography

[google1] Authentication and Authorization for Google APIs. *Google,* [Online] 2011.
https://code.google.com/apis/accounts/docs/OAuth2.html#Registering

[google02] Google+ Platform, Google. https://developers.google.com/+/downloads

[Facebook01] Authentication, Facebook. http://developers.facebook.com/docs/authentication/

[Facebook02] Photo. *Facebook*. http://developers.facebook.com/docs/reference/api/photo/

[Facebook03] video.upload, Facebook.
http://developers.facebook.com/docs/reference/rest/video.upload/

[Twitter01] REST API Resources, Twitter, https://dev.twitter.com/docs/api

[Twitter02] REST API Resources, Twitter, https://dev.twitter.com/docs/api

[Twitter03]   POST statuses/update_with_media, Twitter,
https://dev.twitter.com/docs/api/1/post/statuses/update_with_media