Software Design Documentation (SDD)

Project – SyncGallery +

Revision History

Date	Revision	Description	Author
29-Jan-12	1.0	Document Created	W. Ghorishi
27-Feb-12	1.1	Document Updated	D. Liu

Table of Contents

1 Software Architecture and High-level Design

1.1 System's Interaction with External Systems and it's Environment

In the case of interaction with Android mobile phone devices, our image processing system aims to be built on the latest Android v4.x Ice Cream Sandwich SDK. It is also supposed to be compatible with all previous Android versions, including v2.2.x Froyo, v2.3.x Gingerbread and v3.x Honeycomb. We've considered producing an HD version for Android tablets, but currently we are focused more on mobile phone devices.

As one of the highlighted functions in our product, the synchronization will be developed and realized by using the open source Dropbox Android SDK. As a result, the Dropbox users can immediately see the processed images anywhere after their Android devices synchronize with the backend Dropbox cloud server.

The system will allow the user to share images with their friends through any social media application, supported on an Android ICS platform. Given the ever growing list of social media platforms (Facebook, myspace, Twitter, etc.), it is critical for SyncGallery+ to be compatible and allow sharing of images with all of these platforms.

SyncGallery+ will also leverage one of the most popular applications available on iOS and Android platforms called "Bump" to enable transfer of images between two phones by "Bumping" each other. For the purposes of this system, "Bump App" will be used to share links to photos that users upload between supported platforms.

1.2 High-level System Architecture

At the high level, the system will follow a Service-Oriented Architecture (SOA). The major components of the system are all interoperable, and will follow the format below.

Photo Editor (Aviary):

This component is responsible for all of the photo editing in the system.

Cloud Storage (Dropbox):

The cloud will store the images taken from the device, and make these images available for download from other devices.

Social Media:

The user can share pictures on any third-party software installed on the phone.

Mobile Device (Android 4.0):

The main component what will link the above services.

1.3 System Decomposition

Due to the fact that the system is following a Service-Oriented Architecture, the components have already been described in the above section but will be given more detail below.

Photo Editor (Aviary):

Aviary is a free photo editing tool that can be easily integrated onto a variety of platforms, including Android. Aviary will be built directly into the system where it will blend seamlessly with the rest of the system. It features many photo editing options that allow a user to fully customize their photos. Features include: crop, rotate, resize, boarders, brightness, contrast, grayscale and sharpness.

Cloud Storage (Dropbox):

Dropbox is a free web-based file hosting service that enables users to store and share files with others. Signing up for Dropbox will give users 2GB of free online storage for their photos.

Social Media:

Users will be able to put their customized image onto any social network where their friends will be able to see the, download it from the cloud onto their own device, add some enhancements to the image, and upload it back onto the cloud image if they installed the corresponding software.

Mobile Device (Android 4.0):

Having the system run on Android 4.0 will allow this application to run anywhere (the cloud storage component will only work when there is an internet connection). Integration of "Bump" smartphone application will provide an alternative method to sharing images between friends.

2 System Design

2.1 System Design in terms of Dropbox

The SyncGallery+ will interact with the backend online storage through the Dropbox API. It will periodically send regular HTTP requests to retrieve images stored on the Dropbox folder, and it will also upload the images from the Android phone to Dropbox as indicated by the user. The gallery creates a Dropbox album on the Android device and shows all uploaded images there. The Dropbox album plays a role of public folder, sharing images among all users' devices. Users can also share select images or albums with other Dropbox friends. Once the user move the shared image out of the Dropbox album, the image will get deleted from the online storage, but remain exist in the Android phone local storage.

2.2 System Design in terms of Photo Editing

The SyncGallery+ is doing partnership with Aviary, so the backend engine of photo editor will be Aviary with little customization. In Gallery, after the user clicks "Edit" button, the SyncGallery+ will switch to the edit mode, which means, the Aviary will take over the rest of the work. Once the user click "Done", the SyncGallery+ will switch back to the view mode, Aviary engine will be turned off.

2.3 System Design in terms of "Bump App"

There are two parts to "Bump": The app running on a supported smartphone and the "bump-matching" algorithms running on a cloud. The app uses the phones' sensors to detect the "bump" and then transmit the information collected from the phone to the "bump-matching" algorithm to allow pairing and IP tunnel creation between the phones.

The pairing algorithm leverages unique information from each handset including accelerometer data, location, IP address and other information to generate a unique bump vector allowing two bumped phones to be paired. This is an adaptive process which means that the tolerances used to establish a match can vary based on the state of the system. In a dense usage area, the system adapts itself to form tighter-bound bump-vectors, thereby ensuring a correct match. To ensure security, when the system is unable to perform a 100% match, the system will trigger the user to "bump" again.

2.4 System Design in terms of Social Networking

Like other photo gallery products, the SyncGallery+ allows the user to share the image via Social Networking platforms. Instead of choosing a specific social network tools, we will use the leave the sharing options to the user. The user is able to use any social network software installed on the phone to share their images.

2.5 System Design in terms of Error Handling

In SyncGallery+, 2 levels of errors will be mainly taken care of. Local validation will be implemented and run before the data send to the remote server in order to minimize the possibility of response failure. Remote validation is all done on the server side and we can't interfere with its processing and return. Local validation will run again after the response data is received from the server. It ensures that there is no error occurred during the data manipulation on the server and the response data makes sense to the users' requests. Users' operation will also be monitored all the time by the Local validation system.

3 Information Representation

3.1 Dropbox

Dropbox will be used as the cloud site to provide secure storage. Dropbox is a web-based file hosting service. The user will acquire their own accounts and passwords to log onto the site, which will grant them access to the files stored on their personal account. The users are able to upload onto, download and delete files from their account's storage.

The Dropbox API can provide the best performance compared to other tools.
Dropbox uses OAuth v1 for authentication. It allows the user to store files on one site with another without typing the account and password, which improves the work efficiency and user experience. Secure Sockets Layer (SSL) and AES-256 bit encryption is used for encryption by Dropbox, which can provide secure login process.

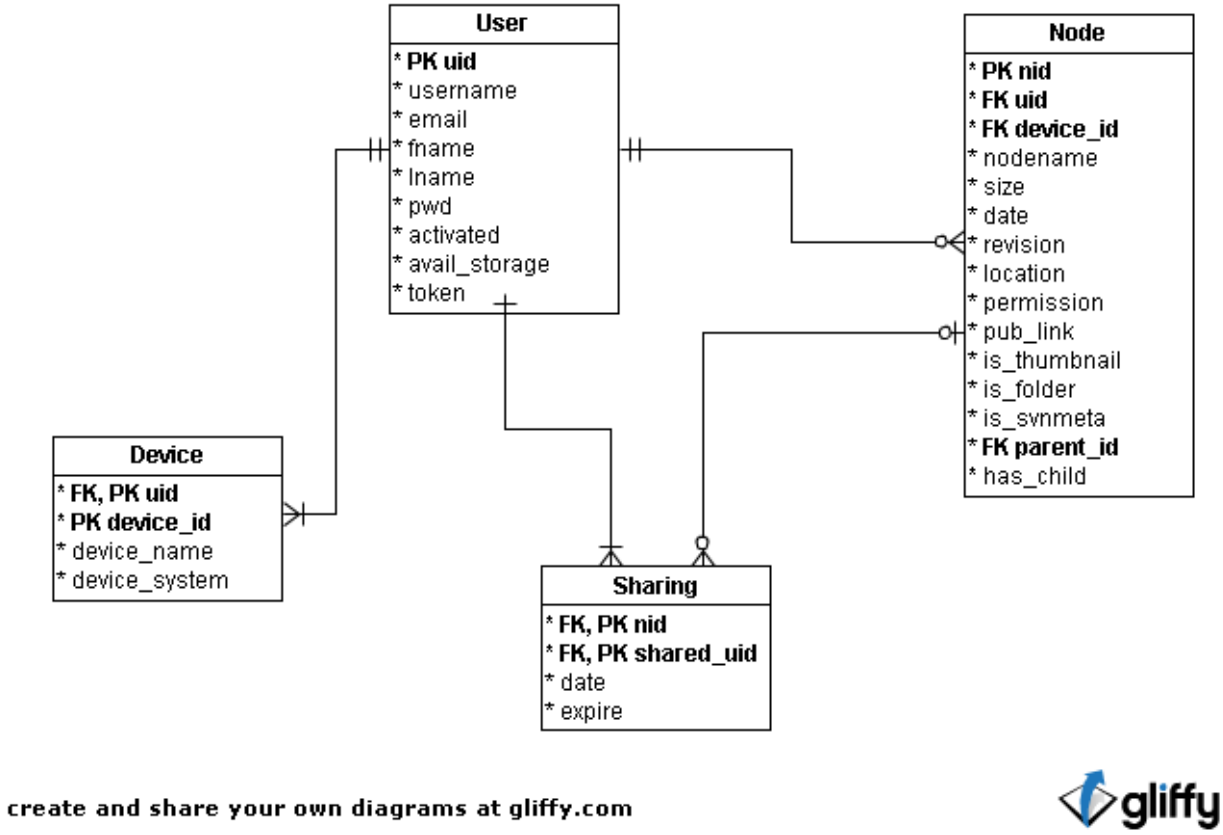
Dropbox is also very user friendly. Any modification, such as delete and rename, on the files can be done easily as in the Windows. Similar to file storage of operating systems, Dropbox also supports usage of directories, which can give a better file management experience for the users. Moreover, users can turn on auto-synchronization to upload files instantly when they are created or modified.

Sharing of files and directories is simple when using Dropbox. Files and folders can be shared between Dropbox users. Furthermore, by linking accounts with Facebook or Twitter, users can share files with their friends on Facebook and Twitter. In addition, Dropbox is cross-platform supported and we can log in using different devices simultaneously, which will make sharing even more convenient.

The Dropbox API is open-sourced and provided online. Also, modification on it is going to be relatively easy, which will shorten our developing time.

In order to achieve better performance, the Dropbox API will be modified and some irrelevant functionalities, such as the event recorder, will be removed

3.1.1 Data Model (ER diagram)



3.1.2 Data Schema

User

Field	Type	Null	Default	Extra	Links to
uid	Bigint	No		auto_increment	
username	varchar(32)	No			
email	varchar(64)	No			
fname	varchar(16)	Yes	NULL		
lname	varchar(16)	Yes	NULL		
pwd	varchar(255)	No			
activated	int(11)	No	0		
avail_storage	Double	No	250.0		
token	varchar(255)	No			

Device

Field	Type	Null	Default	Extra	Links to
uid	Bigint	No			User.uid
device_id	int(11)	No			
device_name	varchar(255)	No			
device_system	int(11)	No			

Sharing

Field	Type	Null	Default	Extra	Links to
Nid	bigint	No			Node.nid

shared_uid	bigint	No			User.uid
Date	datetime	No			
Expire	datetime	Yes	NULL		

Node

Field	Type	Null	Default	Extra	Links to
nid	bigint	No		auto_increment	
uid	bigint	No			User.uid
device_id	int(11)	No			Device.device_id
nodename	varchar(255)	No			
size	double	No			
date	datetime	No			
revision	int(128)	No			
location	varchar(255)	No			
permission	int(11)	No			
pub_link	varchar(255)	Yes	NULL		
is_thumbnail	smallint(6)	No	0		
is_folder	smallint(6)	No	0		
is_svnmeta	smallint(6)	No	0		
parent_id	bigint	No	0		
has_child	smallint(6)	No	0		

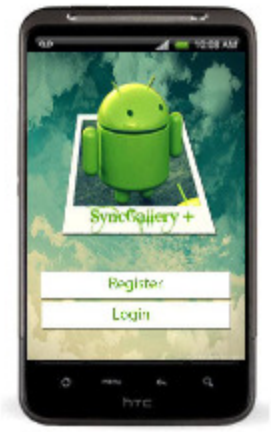
3.1.3 UML

Please refer to Dropbox_UML.jpg

4 High Level UI Design

The following diagram depicts a high-level UI design:

1st UI



SyncGallery IMAGES

2nd UI

SyncGallery IMAGES

3rd UI

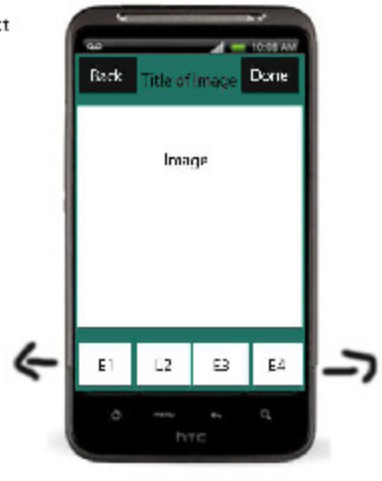
SyncGallery IMAGES

4th UI

SyncGallery IMAGES

5th UI

E=Effect



SyncGallery IMAGES