Imageshare.io

COP 4934 CS Senior Design 1

Group 15
Scott Anderson
Khanh Bui
Deni Karuli
04/28/2016
Table of Contents

PROJECT MOTIVATION ........................................................................................................... 4
  Scott Anderson’s Personal Motivation ................................................................................. 4
  Khanh Bui’s Personal Motivation ....................................................................................... 5
  Deni Karuli’s Personal Motivation .................................................................................... 5
BROADER IMPACTS ............................................................................................................... 5

TECHNICAL OBJECTIVES, GOALS, SPECIFICATIONS, AND REQUIREMENTS ..................... 6
BACKEND .................................................................................................................................. 6
  Goals and Objectives ........................................................................................................... 6
  Specifications ...................................................................................................................... 7
MOBILE .................................................................................................................................... 7
WEB ......................................................................................................................................... 8
  Register An Account ........................................................................................................... 9
  Login To Account .............................................................................................................. 9
  Manage Account ............................................................................................................... 9
  Create, Edit, and View a Story .......................................................................................... 10
  Create, Edit, and View a Moment ..................................................................................... 10
  View a Public Moment or Story .................................................................................... 10

RESEARCH AND INVESTIGATIONS ...................................................................................... 10
BACKEND ................................................................................................................................ 10
  Google Drive ..................................................................................................................... 11
  Dropbox ............................................................................................................................ 12
  Laravel PHP framework .................................................................................................. 12
  Dropbox APIs for Python .................................................................................................. 15
  Django REST framework ................................................................................................ 26
Writing tests ....................................................................................................................... 28
  Django’s unit tests use a Python standard library module: unittest. This module defines tests using a class-based approach .............................................................................. 28
  Here is an example which subclasses from django.test.TestCase, which is a subclass of unittest.TestCase that runs each test inside a transaction to provide isolation .................................................................................................................. 28
  Figure: writing test case in Django. Reprinted with permission from djangoproject.com .................................................................................................................................................. 28
  Figure: running test case in Django. Reprinted with permission from djangoproject.com ............................................................................................................................................. 28
  The Test Client .................................................................................................................. 29
  Provided test case classes ............................................................................................... 29
DATABASE ............................................................................................................................... 33
  Using OAuth 2.0 Access Google APIs .............................................................................. 34
  Using OAuth 2.0 work with Dropbox APIs ....................................................................... 35
MOBILE ..................................................................................................................................... 37
WEB ......................................................................................................................................... 41
  Devise - User Authentication ............................................................................................ 42
  Fast Image - Image Identification ..................................................................................... 42
  Paperclip - File Uploads .................................................................................................... 42
  Spud Photos - Gallery Views ............................................................................................ 42
  Crack - JSON Parsing ...................................................................................................... 43
Executive summary

The goal of Imageshare is to create a easy to use image sharing platform for users to collaborate, share, and view their stories online and mobile in a meaningful way. Imageshare’s platform allows users to create a story that is comprised of moments that are described by a collection of map locations, descriptions, images, and other tag able information to convey their message. All media used in the platform is stored to a user’s linked Dropbox or Google Drive account leaving data storage to the user for them to decide to allocate the space they require and to be able to easy access the photos they upload to later be retrieved when a story is viewed or edited.

Project Motivation

Every picture tells a story… this is ours. The motivation to make Imageshare was the need to fill a void in online photo sharing, the ability to share photos in a meaningful manner with friends and family. All members of our team come from different backgrounds in software development from backend databases, mobile development, and web development bringing our own unique perspective on how we will accomplish our goals and it is this diversity factor that drives us to work together and innovate our take on a modern storytelling platform.

Scott Anderson’s Personal Motivation

I love to tell stories! Whenever I travel alone, with friends or family I love to share my experience with others… expressed through images taken and my personal account of what I experienced. The act of storytelling to me is a way to take someone with me on an adventure and experience what I had… the challenge I always face is how can I share my story when I’m not around to tell it? The solution is simple use a photo sharing platform like Google Photos right? I can simply share photos and even add a description on what is happening… but to me my experiences with these platforms seem hollow and meaningless and better yet what happens if my sister also wants to add who part in this story? How will I share my story once it is written?

Imageshare was my solution to the age old challenge of sharing a story in the modern century, to create a simple and easy to use platform to collaborate and share my experience with others and to archive the wonderful moments that go along with them. Through my collaboration on this project with a wonderful team with complementary skill sets and similar experience and challenges faced I know we can solve this challenge in front of us. Now it is time for us to tell our story.
Khanh Bui’s Personal Motivation

I have done a lot of road trips, taken quite a lot of pictures and always wanted to share them with my family. However, I found it difficult to share those pictures with them since they are living far away and have little knowledge about computer and social networks. I would like to have a photo sharing platform that is simple yet powerful to make the process as easy as possible. My hope is that everyone can use it to share photos and pictures that capture moments of their life. In addition, I have gained experiences through my work in developing web services and backend for data-driven web application using PHP over the years. This, this project seems well fitted to my knowledge and interest.

Deni Karuli’s Personal Motivation

People have long said, a picture speaks a thousand words. Such statement holds true if only people could access those pictures easily and affordably! In a world driven by technology my team members and I found it aggravating that a simple photo management platform does not exist. This issue was never more prevalent than when a small group of friends and I took a road trip last year. We drove to up to Pittsburg, PA, making many stops and taking many photos along the way. When our trip was over we all had photos of one another that we wanted to share and since some of us were not computer savvy or had social media accounts, this became a large inconvenience. Ultimately, we lost many of our photos and memories because the lack of a simple photo sharing platform. Since that exciting trip to Pittsburgh, I’ve been searching for way to solve this very common photo sharing problem. Sr. Design, presented me with the perfect team and resources to solve this issue, it also helped that my teammates were passionate and motivated about the subject as well. With our powerful yet simple design, I can upload and deliver photos to even the most computer illiterate users at the tap of a screen or the click of a mouse. My grandfather in Albania can receive and view photos of me, all the meanwhile sending his personal photos to his friends and family. This project motivates me because it helps make an enormous world smaller.

Broader Impacts

“To every story there is a moment, and it is within these moments the the world around us comes alive!” -Scott Anderson.

It is defined within this quote that defines our passion for Imageshare, to create a platform for users to share their story the way they want to on a platform
made just for them. The impact of Imageshare is to provide a platform for users to share their online photos in a meaningful way again, not to just create a dump for photos to exist in an endless stream. Imageshare exists to provide context to the moments in which photos are taken to help the user better express their viewpoint in that moment.

Impacts of Imageshare are limited on to which a user restricts his or her audience to. A family might tell their story of a road trip to friends and family, to a church wishing to engage with their local communities and share the services and stories of their church. A global movement wishing to document the progression of their cause and the impacts of a society at large. Imageshare can serve as a tool to help individuals or organizations share their story the only limitations are the bounds of our user’s imagination.

Technical Objectives, Goals, Specifications, and Requirements

Backend

Goals and Objectives

The goal of the project was to provide the user the tools to easily manage and share their photos stored on cloud storage services: google drive and dropbox. User can upload, delete retrieve their photos. There are several objectives that the backend should meet: integration, security, reliability and usability.

**Integration** - the backend shall be able to integrate with Google Drive and Dropbox. Furthermore, it provides the services and API to web and mobile applications to consume. To achieve this goal, all the data requests and responses are through JSON (JavaScript Object Notation)

**Security** - the backend shall be able to ensure that all stored user information is encrypted and safely stored in the database. Oauth2 protocol will be employed to access to Google API and Dropbox API for authentication and authorization in getting user photos securely and conveniently.

**Reliability** - to ensure the availability of the system, user’s photos are also stored and cached on the web server. This would help in reducing the access time to user’s data and avoid flooding the network bandwidth with huge number of requests to cloud storage. In worst case scenario, when the database is down, user still has full access to his/her data on the cloud storage and the system is still fully operational.
Usability - The backend will be designed with ease of usability in mind. All the services and functions from the backend will be ensured to have simple interface but are fully functional for the web and mobile application having little effort in making requests and receiving responding data.

Specifications
The system shall conform to the following specifications:
- The system shall be able to store user information on the database
- The system shall be able to use OAuth 2.0 to authorize requests
- The system shall be able to get authorization token to send request to Google Drive API and Dropbox
- The system shall be able to get access token from Google Drive and Dropbox to access user data
- The system shall be able to use refresh tokens to acquire new access tokens
- The system shall be able to get the permission from user to access their data
- The system shall be able to read user file content
- The system shall be able to download user photos
- The system shall be able to upload photos to Google Drive Dropbox
- The system shall be able to delete photos.
- The system shall be able to store comments on photos or album of photos
- The system shall be able to create a link for viewing shared photos.

Mobile

The mobile portion of this project will be constructed for the iOS only, therefore NO application will be supported for android. Furthermore, the mobile application will be designed and build for the iPhone 5 or newer and will not support previous devices. To communicate with the backend the application will be using JSON. In order to code and design the mobile application we will be using the Swift programing language along with SwiftyJSON framework to easily parse JSON calls and MWPhotoBrowserSwift framework to beautifully display photo arrays. Some form of internet access is required for the mobile application to work. In order to use the mobile application a user must have or create an account with google drive or dropbox, as this will be where the images are stored. A user will have the ability to create an account or log in if one already exists. The application will be deployed for iOS version 8.0.

Each account will have the specified storage platform linked to it. Once a registered user has logged in, they will enter the main menu. The main menu will provide access to the four separate portals: Add Story, view/edit story, Account and Logout. Each portal will open up to their own separate sections with unique abilities. Additionally, the main menu will provide information on the amount of storage available on cloud of the user currently logged in. Add story will allow
the user to access and view all photos/videos and albums currently stored on their phone, a user will also have the ability to perform simple photo editing such as crop, or changing background filter. A user can select photos/videos on his or her phone and chose to add them to an existing story/moment or create a new one, the photos will be stored on the platform linked to the user's account. View/edit story section will allow the user to access all stories/moments/photos on their storage platform. This section will also allow the user to select the photos or moment of their choosing and construct a shareable downloadable link. The user can delete and add comments/remove comments to any story/moment/photo stored on the cloud. Account Information section will give the user access to their account credentials. Here a user can view their name, the amount of storage space left and their username. Additionally, the user has the ability to update and change their password. Logout section is self-explanatory. This button simply allows the user to logout of the account and exit the main menu.

The photos will be displayed via the collection view controller on Swift. The collection view will be populated with thumbnails of the photos. Selecting on a photo/video will allow viewing of the item in full quality, if a photo is selected that is in your local directory, you will have the ability to perform simple editing before pushing the change to the cloud. Likewise, if you select a photo that is stored on the cloud you will have the ability to remove it or add/delete comments to it.

Since Imageshare is primarily a image sharing platform, the decision to implement a video feature will be decided based on user feedback. The main goal of our platform is to provide a simple way for users to share their photos and stories, if the impact of videos on the user's bandwidth takes away from user experience it will be removed feature.

The frameworks used for this project will be SwiftyJSON, an open framework that allows for simple handling of JSON data in Swift, and MWPhotoBrowserSwift, a Swift photos framework that allows for simple viewing of photo collections. Swift has the ability to handle JSON on its own however the default handling is very verbose and clunky. This framework simplifies the syntax and reduces the likelihood of syntax errors. I installed the framework using cocoa pods. Cocoa pods is a dependency manager for Swift and Objective C, allowing the additions and updates of frameworks quickly via the terminal.

Web

Imageshare’s web application is the main place users are expected to spend their time the most on the platform, for this reason all major web browsers are being supported including latest releases of Apple Safari, Mozilla Firefox, Google Chrome, and Internet Explorer 10 or newer. The web application is going to be built on a ruby on rails backend to communicate with the backend services layer and an AngularJS front end for display of all UI elements. In order for a user
to be able to use the web application outside of simple shared viewing a user will be required to register an account with the Imageshare platform and link an existing cloud service provider account. Related services provided by the web application is the ability to create, edit, and view stories and sub moments independently or collaboratively.

The following are detailed activities provided by the web application.

Register An Account

When a user first visits the Imageshare website http://www.Imageshare.io they will be required to register for an account in order to begin creating stories. In order to register a user must click on the Register button on the website's main menu navigation system, after doing so they should be presented a three step register page. The register process will contain three steps where a user must enter their basic information to create an account, link and existing cloud service, and then be able to to use the application (pictures W.1.1, W.1.2, W.1.3).

The objective of this activity is to allow the user to successfully create an account for the Imageshare website, once completed they will have fully access to all of Imageshare services both on the web application or through the compatible iOS companion application.

Login To Account

Once a user has successfully created a Imageshare account with an associated cloud storage they will be able to login to the application by the Login button in the website's main navigation menu. Once on the login page a user will be able to enter their email and password and click a Login button from which they will be granted access to the web application see picture W.2.0. The Objective of this activity is to allow users gain easy secure access to their Imageshare content.

Manage Account

The objective of this activity is to provide easy access for a user to manage their account information and view their current data usage on their cloud storage. Once a user is successfully logged into the web application they will be able to click on Account on the site’s main menu and will then be redirected to their accounts management page see picture W.3.0. The account management page must present the user’s current email and the ability to update it, the ability to change password, and to see a usage meter of their linked cloud storage and the
ability to change their existing cloud setting including but not limited to provider and allocated space.

Create, Edit, and View a Story

The objective of this activity is to provide an easy way for a user to be able to create a story, edit and update a story, and view already made stories. In context a story is similar to an album. Requirements for a story is to include a title, description, location, a defined storyteller, collaborators, tags, and a collection of moments within a story. Once a story has been fully developed it must also be shareable publicly via a url link or privately by password protection.

Create, Edit, and View a Moment

The objective of this activity to an easy way for a user to be able to create a moment, edit and update a moment, and view already made moments within a story. Requirements for a moment is to include a title, description, location, tags, date range, and a collection of photos that contain optional captions. Once a moment has been completed it must be able to be shared publicly via a url link or privately by password protection.

View a Public Moment or Story

The objective of this activity is to provide an easy way for people to view a public or private story without requiring a Imageshare account. When a url is provided a public link they must be able to view a targeted moment or all moments in a shared story, if the url is private they must be prompted for a password to verify access rights to the story. For an example of what a story view will look like see picture W.4.0.

Research and investigations

Backend

Google Drive and Dropbox both give a group of APIs along with client libraries and documentation to help developing apps that integrate with them.
Google Drive

Google Drive and Dropbox both give a group of APIs along with client libraries and documentation to help developing apps that integrate with them.

Google Drive

Google Drive REST API supports quite a few client libraries: .Net, Go, iOS, Java, JavaScript, Node.js, PHP, Python and Ruby.

**Google APIs Client Libraries**

The Google APIs Client Libraries are great examples of how to use the Google APIs Discovery Service to build a good quality client library for Google APIs. All libraries are open-sourced and ready for you to use as a sample.

In the tables below, the first column shows each library's stage of development (note that some are in early stages), and links to documentation for the library. The second column links to available samples for each library.

<table>
<thead>
<tr>
<th>Documentation</th>
<th>Samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google API Client Library for Java</td>
<td>Java samples</td>
</tr>
<tr>
<td>Google API Client Library for JavaScript (beta)</td>
<td>JavaScript samples</td>
</tr>
<tr>
<td>Google API Client Library for .NET</td>
<td>NET samples</td>
</tr>
<tr>
<td>Google API Client Library for Objective-C</td>
<td>Objective-C samples</td>
</tr>
<tr>
<td>Google API Client Library for PHP (beta)</td>
<td>PHP samples</td>
</tr>
<tr>
<td>Google API Client Library for Python</td>
<td>Python samples</td>
</tr>
</tbody>
</table>

These early-stage libraries are also available:

<table>
<thead>
<tr>
<th>Documentation</th>
<th>Samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google APIs Client Libraries for Dart (beta)</td>
<td>Dart samples</td>
</tr>
<tr>
<td>Google API Client Library for Go (alpha)</td>
<td>Go samples</td>
</tr>
<tr>
<td>Google API Client Library for Node.js (alpha)</td>
<td>Node.js samples</td>
</tr>
<tr>
<td>Google API Client Library for Ruby (alpha)</td>
<td>Ruby samples</td>
</tr>
</tbody>
</table>

Figure B.1: Google APIs Client Libraries. Reprinted with the permission from developers.google.com with Creative Commons Attribution 3.0 License.
Dropbox

The Dropbox API allows developers to work with files in Dropbox, including advanced functionality like full-text search, thumbnails, and sharing. Dropbox supports quite a few platforms:

![Dropbox API platforms](figure)

Figure B.2: Platform supported by Dropbox APIs. Reprinted with the permission from [www.dropbox.com/developers](http://www.dropbox.com/developers)

Laravel PHP framework

Initially, with the first glance, it was so great that both Google Drive and Dropbox has API for PHP. Thus the team decided to use Laravel, a PHP framework, design and built our system.

Laravel is a web application framework with expressive and nice syntax. It’s motto is: “development must be an enjoyable, creative experience to be truly fulfilling”. It make the development easier by taking care of common tasks used in the majorities of web projects: authentication, routing, sessions, queueing and caching. Moreover, with tightly integrated unit testing support, it facilitates us in making a robust application. Although it is a new framework to us, but the learning curve would be short since we all have a background in PHP. We thought it was a perfect framework to build up our system.
Unfortunately, thing did not happen as we expected. When we looked more closely to the PHP API client on Dropbox and Google Drive we found out that:

- Dropbox does not yet have an official PHP SDK.

**Dropbox for PHP Developers**

- API client library for PHP only in Beta only
After taking more time to investigating, we finally come to an agreement to use Python which is fully supported by both of them to build up our system.
Dropbox APIs for Python

- dropbox.oauth – OAuth

  class dropbox.oauth.DropboxOAuth2Flow(consumer key, consumer secret, redirect Uri, session, csrf_token_session_key, locale=None)

  This is the OAuth 2 authorization helper for web application. OAuth 2 has a two-step authorization process. The first step is having the user authorize your app. The second involves getting an OAuth2 access token from Dropbox.

```python
from dropbox import DropboxOAuth2Flow

def get_dropbox_auth_flow(web_app_session):
    redirect_url = "https://my-web-server.org/dropbox-auth-finish"
    return DropboxOAuth2Flow(
        APP_KEY, APP_SECRET, redirect_uri, web_app_session,
        "dropbox-auth-csrf-token")

@url handler for /dropbox-auth-start
def dropbox_auth_start(web_app_session, request):
    authorize_url = get_dropbox_auth_flow(web_app_session).start()
    redirect_to(authorize_url)

@url handler for /dropbox-auth-finish
def dropbox_auth_finish(web_app_session, request):
    try:
        access_token, user_id, url_state = \
            get_dropbox_auth_flow(web_app_session).finish(\
                request.query_params)
    except BadRequestException, e:
        http_status(400)
    except BadStateException, e:
        # Start the auth flow again.
        redirect_to("/dropbox-auth-start")
    except CsrfException, e:
        http_status(403)
    except NotApprovedException, e:
        flash('Not approved? Why not?')
        return redirect_to("/home")
    except ProviderException, e:
        logger.log("Auth error: %s" % (e,))
        http_status(403)
```

Figure B.7: Example of OAuth 2 authorization helper for web application. Reprinted with the permission from dropbox.com/developers.

- dropbox.dropbox – Dropbox
class dropbox.dropbox.Dropbox(oauth2_access_token, max_retries_on_error=4, max_retries_on_rate_limit=None, user_agent=None, session=None, headers=None)

- files_copy(from_path, to_path)
  Copy a file or folder to a different location in the user’s Dropbox. If the source path is a folder all its contents will be copied.

  Parameters:
  - from_path (str) – Path in the user’s Dropbox to be copied or moved.
  - to_path (str) – Path in the user’s Dropbox that is the destination.

  Return type: dropbox.files.Metadata

  Raises: dropbox.exceptions.ApiError

  If this raises, ApiError.reason is of type:
  dropbox.files.RelocationError

  Figure B.8: copy file in Dropbox. Reprinted with the permission from dropbox.com/developers.

- files_create_folder(path)
  Create a folder at a given path.

  Parameters: path (str) – Path in the user’s Dropbox to create.

  Return type: dropbox.files.FolderMetadata

  Raises: dropbox.exceptions.ApiError

  If this raises, ApiError.reason is of type:
  dropbox.files.CreateFolderError

  Figure B.9: create folder in Dropbox. Reprinted with the permission from dropbox.com/developers.

- files_delete(path)
files_delete(path)

Delete the file or folder at a given path. If the path is a folder, all its contents will be deleted too. A successful response indicates that the file or folder was deleted. The returned metadata will be the corresponding FileMetadata or FolderMetadata for the item at time of deletion, and not a DeletedMetadata object.

Parameters:
- path (str) – Path in the user’s Dropbox to delete.

Return type:
- dropbox.files.Metadata

Raises:
- dropbox.exceptions.ApiError

If this raises, ApiError.reason is of type:
- dropbox.files.DownloadError

Figure b.10: delete file in Dropbox. Reprinted with the permission from dropbox.com/developers.

files_download(path, rev=None)

Download a file from a user’s Dropbox.

files_download_to_file(download_path, path, rev=None)

Download a file from a user’s Dropbox.

Parameters:
- download_path (str) – Path on local machine to save file.
- path (str) – The path of the file to download.
- rev (Nullable) – Deprecated. Please specify revision in path instead

Return type:
- (dropbox.files.FileMetadata, requests.models.Response)

Raises:
- dropbox.exceptions.ApiError

If this raises, ApiError.reason is of type:
- dropbox.files.DownloadError

Figure B.11: download file from Dropbox. Reprinted with the permission from dropbox.com/developers.

files_get_metadata(path, include_media_info=False, include_deleted=False, include_has_explicit_shared_members=False)
Returns the metadata for a file or folder. Note: Metadata for the root folder is unsupported.

```python
files_get_metadata(path, include_media_info=False, include_deleted=False, include_has_explicit_shared_members=False)
```

Returns the metadata for a file or folder. Note: Metadata for the root folder is unsupported.

**Parameters:**
- `path (str)` – The path of a file or folder on Dropbox.
- `include_media_info (bool)` – If true, `FileMetadata.media_info` is set for photo and video.
- `include_deleted (bool)` – If true, `DeletedMetadata` will be returned for deleted file or folder, otherwise `NotFoundError.not_found` will be returned.
- `include_has_explicit_shared_members (bool)` – If true, the results will include a flag for each file indicating whether or not that file has any explicit members.

**Return type:** `dropbox.files.Metadata`

**Raises:** `dropbox.exceptions.ApiError`

If this raises, `ApiError.reason` is of type:

`dropbox.files.GetMetadataError`

Figure B.12: get file metadata from Dropbox. Reprinted with the permission from dropbox.com/developers.

* files_get_thumbnail(path, format=ThumbnailFormat('jpeg', None), size=ThumbnailSize('w64h64', None))
  Get a thumbnail for an image. This method currently supports files with the following file extensions: jpg, jpeg, png, tiff, tif, gif and bmp. Photos that are larger than 20MB in size won’t be converted to a thumbnail.
Figure B.13: get image thumbnail from Dropbox. Reprinted with the permission from dropbox.com/developers.

- `files_get_thumbnail(path, format=ThumbnailFormat('jpeg', None), size=ThumbnailSize('w64h64', None))`

  Get a thumbnail for an image. This method currently supports files with the following file extensions: jpg, jpeg, png, tiff, tif, gif and bmp. Photos that are larger than 20MB in size won’t be converted to a thumbnail.

  **Parameters:**
  - `path (str)` - The path to the image file you want to thumbnail.
  - `format (dropbox.files.ThumbnailFormat)` - The format for the thumbnail image, `jpeg` (default) or `png`. For images that are photos, `jpeg` should be preferred, while `png` is better for screenshots and digital arts.
  - `size (dropbox.files.ThumbnailSize)` - The size for the thumbnail image.

  **Return type:** `(dropbox.files.FileMetadata, requests.models.Response)`

  **Raises:** `dropbox.exceptions.ApiError`

  **If this raises, ApiError.reason is of type:** `dropbox.files.ThumbnailError`

Figure B.14: get image thumbnail to file from Dropbox. Reprinted with the permission from dropbox.com/developers.

- `files_get_thumbnail_to_file(download_path, path, format=ThumbnailFormat('jpeg', None), size=ThumbnailSize('w64h64', None))`

  Get a thumbnail for an image. This method currently supports files with the following file extensions: jpg, jpeg, png, tiff, tif, gif and bmp. Photos that are larger than 20MB in size won’t be converted to a thumbnail.

  **Parameters:**
  - `download_path (str)` - Path on local machine to save file.
  - `path (str)` - The path to the image file you want to thumbnail.
  - `format (dropbox.files.ThumbnailFormat)` - The format for the thumbnail image, `jpeg` (default) or `png`. For images that are photos, `jpeg` should be preferred, while `png` is better for screenshots and digital arts.
  - `size (dropbox.files.ThumbnailSize)` - The size for the thumbnail image.

  **Return type:** `(dropbox.files.FileMetadata, requests.models.Response)`

  **Raises:** `dropbox.exceptions.ApiError`

  **If this raises, ApiError.reason is of type:** `dropbox.files.ThumbnailError`

- `files_upload(f, path, mode=WriteMode('add', None), autorename=False, client_modified=None, mute=False)"
Create a new file with the contents provided in the request. Do not use this to upload a file larger than 150 MB. Instead, create an upload session with `upload_session_start()`.

| `files_upload(f, path, mode=WriteMode(add', None), autorename=False, client_modified=None, mute=False)` |

Create a new file with the contents provided in the request. Do not use this to upload a file larger than 150 MB. Instead, create an upload session with `upload_session_start()`.

**Parameters:**
- `f` - A string or file-like obj of data.
- `path` (str) – Path in the user’s Dropbox to save the file.
- `mode` (drobox.files.WriteMode) – Selects what to do if the file already exists.
- `autorename` (bool) – If there’s a conflict, as determined by `mode`, have the Dropbox server try to autorename the file to avoid conflict.
- `client_modified` (Nullable) – The value to store as the `client_modified` timestamp. Dropbox automatically records the time at which the file was written to the Dropbox servers. It can also record an additional timestamp, provided by Dropbox desktop clients, mobile clients, and API apps of when the file was actually created or modified.
- `mute` (bool) – Normally, users are made aware of any file modifications in their Dropbox account via notifications in the client software. If `True`, this tells the clients that this modification shouldn’t result in a user notification.

**Return type:** `drobox.files.FileMetadata`

**Raises:** `drobox.exceptions.ApiError`

**If this raises, ApiError.reason is of type:**

| `drobox.files.UploadError` |

Figure B.15: upload file to Dropbox. Reprinted with the permission from dropbox.com/developers.
Google Drive APIs functions

Google Drive API
- Authorizing requests with OAuth 2.0.
  All the requests to the Drive API must be authorized by an authenticated user.
  Here is the OAuth 2.0 scope information for the Drive API:
This method supports a /upload URI and accepts uploaded media with the following characteristics:

- Maximum file size: 5120 GB
- Accepted Media MIME type

This method provides media upload functionality through two separate URIs.

- Upload URI, for media upload requests:
- Metadata URI, for metadata-only requests:
  POST https://www.googleapis.com/drive/v3/files
**Parameter for Google Drive http request.** Reprinted with the permission from developers.google.com

### Parameter name | Value | Description
--- | --- | ---
**Required query parameters**
uploadType | string | The type of upload request to the /upload URI. Acceptable values are:
- media - Simple upload. Upload the media only, without any metadata.
- multipart - Multipart upload. Upload both the media and its metadata, in a single request.
- resumable - Resumable upload. Upload the file in a resumable fashion, using a series of at least two requests where the first request includes the metadata.

### Optional query parameters
ignoreDefaultVisibility | boolean | Whether to ignore the domain’s default visibility settings for the created file. Domain administrators can choose to make all uploaded files visible to the domain by default; this parameter bypasses that behavior for the request. Permissions are still inherited from parent folders. (Default: false)
keepRevisionForever | boolean | Whether to set the ‘keepForever’ field in the new head revision. This is only applicable to files with binary content in Drive. (Default: false)
ocrlanguage | string | A language hint for OCR processing during image import (ISO 639-1 code).
useContentAsIndexableText | boolean | Whether to use the uploaded content as indexable text. (Default: false)

**Figure B.18: Parameter for Google Drive http request.** Reprinted with the permission from developers.google.com

- **Delete**

**HTTP request**

```
DELETE https://www.googleapis.com/drive/v3/files/fileId
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path parameters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>fileId</td>
<td>string</td>
<td>The ID of the file.</td>
</tr>
</tbody>
</table>

**Figure B.19: Delete file http request.** Reprinted with the permission from developers.google.com

- **Get**

Get a file’s metadata or content by Id
HTTP request

GET https://www.googleapis.com/drive/v3/files/fileId

Parameters

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path parameters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>fileId</td>
<td>string</td>
<td>The ID of the file.</td>
</tr>
<tr>
<td>Optional query parameters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>acknowledgeAbuse</td>
<td>boolean</td>
<td>Whether the user is acknowledging the risk of downloading known malware or other abusive files. This is only applicable when alt=media. (Default: false)</td>
</tr>
</tbody>
</table>

Figure B.20: Get file http request. Reprinted with the permission from developers.google.com

- List
  Lists or searches files.

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>corpus</td>
<td>string</td>
<td>The source of files to list.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Acceptable values are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- &quot;domain&quot;: Files shared to the user's domain.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- &quot;user&quot;: Files owned by or shared to the user. (default)</td>
</tr>
<tr>
<td>orderBy</td>
<td>string</td>
<td>A comma-separated list of sort keys. Valid keys are 'createdTime', 'folder', 'modifiedByMeTime', 'modifiedTime', 'name', 'quotaBytesUsed', 'reocency', 'sharedWithMeTime', 'starred', and 'viewedByMeTime'. Each key sorts ascending by default, but may be reversed with the 'desc' modifier. Example usage: orderBy=folder,modifiedTime desc,order=desc. Please note that there is a current limitation for users with approximately one million files in which the requested sort order is ignored.</td>
</tr>
<tr>
<td>pageSize</td>
<td>integer</td>
<td>The maximum number of files to return per page. Acceptable values are 1 to 1000, inclusive. (Default: 100)</td>
</tr>
<tr>
<td>pageToken</td>
<td>string</td>
<td>The token for continuing a previous list request on the next page. This should be set to the value of 'nextPageToken' from the previous response.</td>
</tr>
<tr>
<td>q</td>
<td>string</td>
<td>A query for filtering the file results. See the &quot;Search for Files&quot; guide for supported syntax.</td>
</tr>
<tr>
<td>spaces</td>
<td>string</td>
<td>A comma-separated list of spaces to query within the corpus. Supported values are 'drive', 'appDataFolder' and 'photos'.</td>
</tr>
</tbody>
</table>

Figure B.21: list file http request. Reprinted with the permission from developers.google.com

- Update
Updates a file’s metadata and/or content patch semantics. This method supports an /upload URI and accepts uploaded media with the following characteristics:

- Maximum file size: 5120 GB
- Accepted Media MIME types

Upload URI, for media upload requests:


Metadata URI, for metadata-only requests:

- PATCH https://www.googleapis.com/drive/v3/files/fileId

### Parameters

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path parameters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>fileId</td>
<td>string</td>
<td>The ID of the file.</td>
</tr>
<tr>
<td>Required query parameters</td>
<td>string</td>
<td>The type of upload request to the /upload URI. Acceptable values are:</td>
</tr>
<tr>
<td>uploadType</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- media - Simple upload. Upload the media only, without any metadata.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- multipart - Multipart upload. Upload both the media and its metadata, in a</td>
</tr>
<tr>
<td></td>
<td></td>
<td>single request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- resumable - Resumable upload. Upload the file in a resumable fashion, using a</td>
</tr>
<tr>
<td></td>
<td></td>
<td>series of at least two requests where the first request includes the metadata.</td>
</tr>
<tr>
<td>Optional query parameters</td>
<td>string</td>
<td>A comma-separated list of parent IDs to add.</td>
</tr>
<tr>
<td>addParents</td>
<td></td>
<td></td>
</tr>
<tr>
<td>keepRevisionForever</td>
<td>boolean</td>
<td>Whether to set the 'keepForever' field in the new head revision. This is only applicable to files with binary content in Drive. (Default: false)</td>
</tr>
<tr>
<td>ocrLanguage</td>
<td>string</td>
<td>A language hint for OCR processing during image import (ISO 6939-1 code).</td>
</tr>
<tr>
<td>removeParents</td>
<td>string</td>
<td>A comma-separated list of parent IDs to remove.</td>
</tr>
<tr>
<td>useContentAsIndexableText</td>
<td>boolean</td>
<td>Whether to use the uploaded content as indexable text. (Default: false)</td>
</tr>
</tbody>
</table>

Figure B.22: update file http request parameter. Reprinted with the permission from developers.google.com

Django Python framework

Django was chosen to be the web framework to build up our system:
- Django is a free and open-source web framework written in Python follows the model-view-controller (MVC) architecture pattern.
- It was designed to help developers take applications from concept to completion as quickly as possible.
- It takes security seriously and helps developers avoid many common security mistakes.
- It is exceedingly scalable.
- Django is not a new framework, it has been around for more than a decade.
- Django has been crowd-tested. It powers many of the Web’s most-used sites, like Pinterest and Instagram. Facebook also uses Django.
- Django has excellent documentation, and supportive community. This would help a lot in reducing the time to learn about the framework and having more time to develop the system.

Moreover, there are packages available on PyPI - the Python Package Index for Django to support Google Drive and Dropbox. This would be very convenient for integrating Django and those cloud storage services together.

Django architecture

![Django architecture](image)

Figure B.23: Django Framework. Reprinted with the permission from djangoproject.com

**Django REST framework**

More than a decade after its introduction, REST has become one of the most important technologies for Web applications. Its importance is likely to continue...
growing quickly as all technologies move towards an API orientation. Every major development language now includes frameworks for building RESTful Web services.

While REST stands for Representational State Transfer, which is an architectural style for networked hypermedia applications, it is primarily used to build Web services that are lightweight, maintainable, and scalable. A service based on REST is called a RESTful service. REST is not dependent on any protocol, but almost every RESTful service uses HTTP as its underlying protocol.

In a RESTful API, endpoints (URLs) define the structure of the API and how end users access data from our application using the HTTP methods: GET, POST, PUT, DELETE.

Django REST framework is a powerful and flexible toolkit for building Web APIs.

Some reasons you might want to use REST framework:

- The Web browsable API is a huge usability win for your developers.
- Authentication policies including packages for OAuth1a and OAuth2.
- Serialization that supports both ORM and non-ORM data sources.
- Customizable all the way down - just use regular function-based views if you don't need the more powerful features.
- Extensive documentation, and great community support.
- Used and trusted by large companies such as Mozilla and Eventbrite.
Writing tests
Django’s unit tests use a Python standard library module: unittest. This module defines tests using a class-based approach. Here is an example which subclasses from django.test.TestCase, which is a subclass of unittest.TestCase that runs each test inside a transaction to provide isolation:

```python
from django.test import TestCase
from myapp.models import Animal

class AnimalTestCase(TestCase):
    def setUp(self):
        Animal.objects.create(name="lion", sound="roar")
        Animal.objects.create(name="cat", sound="meow")

    def test_animals_can_speak(self):
        """Animals that can speak are correctly identified""
        lion = Animal.objects.get(name="lion")
        cat = Animal.objects.get(name="cat")
        self.assertEqual(lion.speak(), 'The lion says "roar"')
        self.assertEqual(cat.speak(), 'The cat says "meow"')
```

Figure: writing test case in Django. Reprinted with permission from djangoproject.com

When you run your tests, the default behavior of the test utility is to find all the test cases (that is, subclasses of unittest.TestCase) in any file whose name begins with test, automatically build a test suite out of those test cases, and run that suite.

```bash
# Run all the tests in the animals.tests module
$ ./manage.py test animals.tests

# Run all the tests found within the 'animals' package
$ ./manage.py test animals

# Run just one test case
$ ./manage.py test animals.tests.AnimalTestCase

# Run just one test method
$ ./manage.py test animals.tests.AnimalTestCase.test_animals_can_speak
```

Figure: running test case in Django. Reprinted with permission from djangoproject.com
The Test Client

The test client is a Python class that acts as a dummy Web browser, allowing you to test your views and interact with your Django-powered application programmatically.

Some of the things you can do with the test client are:

- Simulate GET and POST requests on a URL and observe the response – everything from low-level HTTP (result headers and status codes) to page content.
- See the chain of redirects (if any) and check the URL and status code at each step.
- Test that a given request is rendered by a given Django template, with a template context that contains certain values.

To use the test client, instantiate `django.test.Client` and retrieve Web pages:

```python
>>> from django.test import Client
>>> c = Client()
>>> response = c.post('/login/', {'username': 'john', 'password': 'smith'})
>>> response.status_code
200
>>> response = c.get('/customer/details/')
>>> response.content
b'<!DOCTYPE html...'
```

Figure: using test client Django. Reprinted with permission from djangoproject.com

Provided test case classes

Normal Python unit test classes extend a base class of `unittest.TestCase`. Django provides a few extensions of this base class: Converting a normal `unittest.TestCase` to any of the subclasses is easy: change the base class of your test from `unittest.TestCase` to the subclass. All of the standard Python unit test functionality will be available, and it will be augmented with some useful additions as described in each section below.
Figure: Hierarchy of Django unit testing classes. Reprinted with permission from djangoproject.com

TransactionTestCase
class TransactionTestCase
TransactionTestCase inherits from SimpleTestCase to add some database-specific features:
- Resetting the database to a known state at the beginning of each test to ease testing and using the ORM.
- Database fixtures.
- Test skipping based on database backend features.
- The remaining specialized assert* methods.

TestCase
class TestCase
This is the most common class to use for writing tests in Django. It inherits from TransactionTestCase (and by extension SimpleTestCase). If your Django application doesn't use a database, use SimpleTestCase.

The class:
Wraps the tests within two nested atomic() blocks: one for the whole class and one for each test. Therefore, if you want to test some specific database transaction behavior, use TransactionTestCase.
LiveServerTestCase

class LiveServerTestCase

LiveServerTestCase does basically the same as TransactionTestCase with one extra feature: it launches a live Django server in the background on setup, and shuts it down on teardown. This allows the use of automated test clients other than the Django dummy client such as, for example, the Selenium client, to execute a series of functional tests inside a browser and simulate a real user's actions.

By default the live server listens on localhost and picks the first available port in the 8081-8179 range. Its full URL can be accessed with self.live_server_url during the tests.

```python
from django.test import TestCase

class MyTests(TestCase):
    @classmethod
    def setUpTestData(cls):
        # Set up data for the whole Test Case
        cls.foo = Foo.objects.create(bar="Test")
        ...

    def test1(self):
        # Some test using self.foo
        ...

    def test2(self):
        # Some other test using self.foo
        ...
```

Finally, you may run the test as follows:

```
$ ./manage.py test myapp.tests.MySeleniumTests.test_login
```
class RequestFactory
The RequestFactory shares the same API as the test client. However, instead of behaving like a browser, the RequestFactory provides a way to generate a request instance that can be used as the first argument to any view. This means you can test a view function the same way as you would test any other function – as a black box, with exactly known inputs, testing for specific outputs.

The API for the RequestFactory is a slightly restricted subset of the test client API:
- It only has access to the HTTP methods `get()`, `post()`, `put()`, `delete()`, `head()`, `options()`, `options()`, and `trace()`.
- These methods accept all the same arguments except for `follows`. Since this is just a factory for producing requests, it’s up to you to handle the response.
- It does not support middleware. Session and authentication attributes must be supplied by the test itself if required for the view to function properly.

```python
from django.contrib.auth.models import AnonymousUser, User
from django.test import TestCase, RequestFactory

from .views import MyView, my_view

class SimpleTest(TestCase):
    def setUp(self):
        # Every test needs access to the request factory.
        self.factory = RequestFactory()
        self.user = User.objects.create_user(  
            username='jacob', email='jacob@example.com', password='top_secret'
        )

    def test_details(self):
        # Create an instance of a GET request.
        request = self.factory.get('/customer/details')

        # Recall that middleware are not supported. You can simulate a logged-in user by setting request.user manually.
        request.user = self.user

        # Or you can simulate an anonymous user by setting request.user to an AnonymousUser instance.
        request.user = AnonymousUser()

        # Test my_view() as if it were deployed at /customer/details
        response = self_view(request)
        # Use this syntax for class-based views.
        response = MyView.as_view()(request)
        self.assertEqual(response.status_code, 200)
```

Figure: Using LiveServerTestCase in Django. Reprinted with permission from djangoproject.com

A summary of the MVC request process can be described as the following:

- A user requests to view a page by entering a URL.
- The application matches the URL to a predefined route.
- The controller action associated with the route is called.
- The controller action uses the models to retrieve all of the necessary data from a database, places the data in an array, and loads a view, passing along the data structure.
- The view accesses the structure of data and uses it to render the requested page, which is then presented to the user in their browser.

Database

MySQL was chosen to be the database for our system. MySQL was designed and optimized for Web applications. The performance, scalability, reliability and ease of use of the world’s most popular open source database make MySQL the number 1 choice for web application. It is also one of the component of the leading open source web platform, LAMP stack (standing for Linux, Apache, MySQL, PHP/Perl/Python), adopted by a large majority of popular Web sites across the world including Facebook, Google, Twitter, Youtube, Craigslist and Zappos.

![Lamp stack web platform. Reprinted with the permission from mysql.com](image)

OAuth 2.0 Authorization Framework
The OAuth 2.0 authorization framework enables a third-party application to obtain limited access to an HTTP service such as Facebook, Github, Google. It works by delegating user authentication to the service that hosts the user account and authorizing third-party applications to access user account. OAuth 2.0 provides authentication flows for web and desktop applications, mobile devices.

OAuth defines four roles:
- Resource Owner: the user who authorize an application to access their account. The application’s access to the user’s account is limited to the “scope” of the authorization granted (e.g. Read or write access).
- Resource/Authorization Server: the resource server hosts the protected user accounts, and the authorization server verifies the identity of the user then issues access tokens to the application. From application developer’s view, a service’s API fulfills both the resource and authorization server roles.
- Client: the application that wants to access the user’s account. Before it may do so, it must be authorized by the user, and the authorization must be validated by the API.

Figure B.25: Abstract Protocol Flow For OAuth 2. Reprinted with the permission from digitalocean.com under Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

Using OAuth 2.0 Access Google APIs
Google APIs use the OAuth 2.0 protocol for authentication and authorization. Google supports common OAuth 2.0 scenarios such as those for web server, installed, and client-side applications.
OAuth 2.0 client credentials can be obtained from the Google Developers Console. Then the client application requests an access token from the Google Authorization Server, extracts a token from the response, and sends the token to the Google API that we want to access. Access tokens have limited lifetimes. If the application needs access to a Google API beyond the lifetime of a single access token, it can obtain a refresh token. A refresh token allows our application to obtain new access tokens.

Figure B.26: Using OAuth 2.0 for Web Server Applications. Reprinted with the permission from developers.google.com

The authorization sequence begins when the application redirects a browser to a Google URL. The URL includes query parameters that indicate the type of access being requested. Google handles the user authentication, session selection, and user consent. The result is an authorization code, which the application can exchange for an access token and a refresh token. The application should store the refresh token for future use and use the access token to access a Google API. Once the access token expires, the application used the refresh token to obtain a new one.

Using OAuth 2.0 work with Dropbox APIs

When working with the Dropbox APIs, an application will access the Dropbox service on behalf our users. We will need to have each user of our app authenticate with Dropbox to verify their identity and give our app permission to access their data on Dropbox. To fulfil this purpose, Dropbox uses OAuth 2.0. Once completed by a user, the OAuth process returns an access token to our app. This access token is a string generated by
Dropbox that we’ll need to send with each subsequent API request to uniquely identify both our app and the end user.

Figure B.27: Dropbox OAuth2 on the web. Reprinted with the permission from dropbox.com/developers.

The first step in the OAuth process is to redirect the user to a Dropbox webpage. Typically the user takes some action on our site, such as clicking a "Connect to Dropbox" button, and our app will need to redirect the user to a particular Dropbox authorization URL. The Dropbox authorization URL is specific to our app and is composed of our app key, redirect URI, response type, and state. Each of the Dropbox SDKs contains a method that will help us generate this URL when we need it. At this point in the flow, the user may need to login to Dropbox or create a Dropbox account. Once the user is logged into Dropbox, they will be presented with a screen to authorize our app to access their Dropbox data.

After the user approves your app, they'll be redirected from Dropbox back to our app using the redirect URI provided in the Dropbox authorization URL. For security, this redirect URI must match one of the redirect URIs we have specified for our app. If we forget to specify a redirect URI we will get a friendly error message from Dropbox to go and set one up.

The redirect back to our app will include an authorization code from Dropbox. Our app will then use it to make a request to Dropbox to exchange the authorization code for a reusable access token. This request to exchange the authorization code for an access token takes places behind-the-scenes with a call to the token API endpoint. This access token is
the key to making successful requests to the Dropbox API. We will store it in our database and use it each time your app needs data from Dropbox for that user.

Mobile

When deciding on the programming language to use for the mobile app, we were limited to Apple programming languages, Swift and Objective C. Objective C was used by apple to design all software used on their devices before the release of Swift. We ultimately chose to design the product with Swift due to many important factors. Setting aside our prior experience designing with Swift, we did further research into the topic and found that there were many advantages with using swift.

Although we’ve had experience with swift programming, we’ve never created an application of this magnitude and so the learning curve was an important factor. Swift provides an easy to read and interpret type of syntax compared to objective C which contained massive amounts of legacy programming that made interpretation of its syntax incredibly difficult. Swift provides a programming environment that is easier to maintain and edit via the elimination of their two file system with objective C. Swift allows for an easier way to manage memory, this is because Swift automatically keeps track of all memory references and prevents the programmer from creating memory leaks. Objective C has no way of tracking memory references and allows for an environment where memory leaks can occur. Swift is a faster programming language by design than Objective C, this is because Swift was designed to perform application logic at a faster speed and more efficiently. There was a study performed to prove this case in 2014. Three separate tests were performed on Swift and C++ to determine the fastest and most efficient language for each category, Mandelbrot algorithm, a computation bound algorithm, GEMM, a memory tests that accesses large arrays in small blocks, and FFT, a memory test that performs irregular access to large arrays.
In this diagram, each Gflop represents the computation of 1 billion floating point operations per second. Simply, this chart means that the greater the Gflop the better the performance. Since C++ is the industry standard high level language in compute bound tasks and memory bound tasks, this table represents very impressive data suggesting that Swift is performing equally in compute-bound tasks and showing vast improvement in memory bound tasks. Considering is a very young language, the data suggests that one day swift could set the bar for such tests. Lastly, Swift has one massive advantage to Objective C in that it allows for dynamic libraries. Dynamic libraries are chunks of code shared by multiple applications. Through dynamic libraries apple can make important updates to swift that will automatically sync to all applications using them. Simply put, Swift is the future of apple and Objective C is the past.

When designing the application, choosing the cutoff line for compatibility was an important decision. The more more inclusive the application was, the more limiting it was in terms of design and capability. Ultimately we chose the iPhone 5 as the cutoff line based on the user base for apple.
Based on the survey performed by Localytics, the pie chart represents that customers using an iPhone 5 or newer represent 84.2% of the iPhone owning population. Furthermore, the processor and GPU are dated on the iPhone 4 so much so that an application this demanding would proved to be clunky and slow, ruining customer satisfaction.

The applications store contains individual applications that access dropbox or google drive directly however they only give you the simple ability to store files and leave the tedious task of organizing and accessing them to you. Meanwhile applications such as Instagram or Facebook give you the ability to organize and store your photos with the burden of owning them and losing certain rights. This application will provide full ownership to the user and provide a simple interface with which to organize, label, and privately share. The satisfaction of such services is measured by the American customer satisfaction index. By looking at current photo sharing / social media services it can be seen that Facebook, the leading photo sharing platform is currently underperforming as of 2014 statistics.

Figure M3 - Consumer Satisfaction Index-Public Survey
This chart shows that Facebook only has a CSI of 67%, 4 points below the average for all social media. This table suggests that many people are starting looking elsewhere for their photo/video sharing needs. Also worth mentioning, Facebook and many other popular photo/video sharing sites will downgrade quality for storage. The popularity of a simple photo/video sharing application is proven by the fast rise of snapchat, a new social media application that sends and shares temporary snips of photos and videos. A study conducted in 2014 shows that snapchat has risen to the 3rd most popular social media application after only 3 years (released in 2011).

**Figure M4 - Leading Social Media Apps in Youth-Public Survey**

![Leading Social Media Apps Among 18-34 Year Olds by Smartphone Penetration](image)

Furthermore the same study showed a steady rise in popularity in the millenials user group.

**Figure M5 - Snapchat Popularity Change by Age-Public Survey**
The addition of video implementation proves to be a difficult one because of the burden on bandwidth that modern high quality videos impose. The feature to add videos will be the last feature to be implemented because its slow upload speed could have a negative effect on user experience. Since the video storage can prove to be timely and bandwidth consuming, which can cost users real money since most cellular companies allocate a given amount of bandwidth to users we will decided on this addition based on real time practice. Since our application is by design mean to be simplistic for even a computer novice can easily use, we want to take the worry of eating a user's entire monthly bandwidth in one video upload.

Web

When deciding on the programming framework Ruby on Rails prior knowledge and work experience unlike mobile was a primary factor in its selection over other frameworks including cake php, node.js, php laravel, and python Django. The web development part of the group is being tasked in learning the ruby on rails framework at the workplace and as such chose the rails framework to expedite the learning and up start process. Moving past the primary reasoning for the selection of ruby on rails over other web frameworks there are many technical reasons in its choice.

The two primary design philosophies of Ruby on Rails is convention over configuration and don’t repeat yourself coupled, these two paired with a rails strict enforcement of the MVC design architecture makes it an ideal choice for our web application framework. Like other frameworks rails provides a majority of all required site functionalities through libraries known as gems, unlike libraries in PHP however rail gems are designed with the convention convention over configuration this allows the gems to function more rapidly apply industry standard configurations over the need to write a lot of configuration code before you can begin using a library in php. Ruby’s second convention of don’t repeat yourself also helps to reduce in the amount of
boilerplate code throughout the application allowing for a smaller more readable codebase that can more easily be modified without having to refactor large portions of the application.

The following gems were chosen to help assist in the development of the web application.

Devise - User Authentication

The Devise gem helps with the development of the user authentication and session management system by providing pre-generated user forms, automatic password encryption, session management, and user record management features. Alongside these core services Devise also handles user routing and pages access to restricted content. Devise will be the primary gem used for the security of the website to ensure user’s data is secure and that only content they have access to will be presented.

Fast Image - Image Identification

The Fast Image gem will be used to help identify incoming image types as well as their file size. Fast image utilizes a file’s UMI to identify what the file type really is, this will help with security in the platform as incoming file types regardless of extension will be properly identified before being uploaded to a user’s cloud storage.

Paperclip - File Uploads

The Paperclip gem will be utilized to handle simultaneous file uploads. This gem is capable of handling simultaneous and enqueued file uploads, once a file has been uploaded it will then be sent to the Fast Image gem to properly filter file attributes before proceeding to image storage.

Spud Photos - Gallery Views

The Spud Photos gem will be used to handle gallery views in the user’s front end view. Spud photo is capable of generating gallery views which will then be extended to handle additional gallery attributes defined in the technical requirements. Galleries generated by spud photo are simple html outputs which allows for easy design handling with a css overlay reducing the need to make gallery components from scratch.
Crack - JSON Parsing

The Crack gem will be used to handle all JSON files, this will primarily be used with all backend services related to the storage backend layer. Crack is capable of receiving all JSON requests and parsing them to make the information more easily available to the ruby environment.

All ruby gems defined together help piece together the web application. With their use development time on non-platform related service is reduced to allow for more time to be spent on Imageshare exclusive services.

The biggest area for concern with Ruby on Rails is its design pattern of adopting the latest web technologies and make them available to general use, the issue with this is Rails failure to support legacy Rails features forcing gems to be constantly maintained or be replaced. All of these selected gems have shown continued support development for over two years and are actively maintained to help reduce the fear of deprecation. Through continued support and a treasure chest of available gems to choose from Ruby on Rails is the ideal platform for the development of our web application.

Detailed design content

Backend

The Imageshare platform transforms a user's basic storage platform such as google drive and dropbox into a powerful storytelling tool. This is done by taking the photos given by a user and organizing and presenting them in a unique way we call a story. A user will use our service to deposit a select group of images into his storage unit and our service will ask the user select an existing story or create one. The story is composed of a storyteller (this is the user who is currently storing the story), collaborator (other users whom the storyteller gives permission to add to and change to a story if any), title, description, date range, tags, location(s), and moment(s). Once the Story has been selected or created the user is required to select a moment or create a moment in that story. Moments are composed of description, photo(s), title, date, location, tag(s). Once that moment has been selected or created the photos that were given by the user are placed here. A user can further elaborate on the moment by adding a title, comment and description to each photo. This way a basic storage platform is transformed into a powerful yet simple tool to tell stories. Simply, stories hold moments and moments hold photos. All the data submitted by the user remains their property for which they are in full control of regardless of the state of Imageshare. Image share is a service not a storage platform. Figure C1 below demonstrates the Imageshare blueprint.
Figure C1 - Imageshare Blueprint

Database design
Figure B.28: ER diagram

User entity store all the information of the user: id, name, password, email. Each user can create album(s) or upload photo(s). Album includes: album id, album description, location, created date, and share key. Each album has one or more photo. Photo entity consists of: photo id, share key, location, created date and photo description. User can also own one or more Cloud storage. Cloud storage has: storage Id, storage token, storage key and storage account.

Each user can create one or more share link. Each share link has id and description.
User case diagram:

Figure B.29: Use Case diagram
Accessing Google Drive using PyDrive

Google documentation for Python APIs is not very clear and fully updated. Thus, this makes it quite complicated to access to Google Drive. For that reason, We use PyDrive, a wrapper library of google-api-python-client, which simplifies many common Google Drive API tasks.

- Feature of PyDrive
  - Simplifies OAuth2.0 into just few lines with flexible settings.
Wraps Google Drive API into classes of each resource to make your program more object-oriented.

Helps common operations else than API calls, such as content fetching and pagination control.

- OAuth made easy
  With the client_secrets.json downloaded from Google API Console, OAuth2.0 is done in two lines. We can customize behavior of OAuth2 in one settings file settings.yaml.

```
from pydrive.auth import GoogleAuth
from pydrive.drive import GoogleDrive

gauth = GoogleAuth()
gauth.LocalWebserverAuth()

drive = GoogleDrive(gauth)

- File management made easy
  Upload/update the file with one method. PyDrive will do it in the most efficient way.

```
file1 = drive.CreateFile({'title': 'Hello.txt'})
file1.SetContentString('Hello')
file1.Upload() # Files.insert()

file1['title'] = 'HelloWorld.txt' # Change title of the file
file1.Upload() # Files.patch()

content = file1.GetContentString() # 'Hello'
file1.SetContentString(content+' World!') # 'Hello World!'
file1.Upload() # Files.update()

file2 = drive.CreateFile()
file2.SetContentFile('hello.png')
file2.Upload()
print 'Created file %s with mimeType %s' % (file2['title'], file2['mimeType'])
# Created file hello.png with mimeType image/png

file3 = drive.CreateFile({'id': file2['id']})
print 'Downloading file %s from Google Drive' % file3['title'] # hello.png
file3.GetContentFile('world.png') # Save Drive file as a local file
```
- File listing pagination made easy

PyDrive handles file listing pagination for you.

```python
# Auto-iterate through all files that matches this query
file_list = drive.ListFile({'q': "'root' in parents"}).GetList()
for file1 in file_list:
    print 'title: %s, id: %s' % (file1['title'], file1['id'])

# Paginate file lists by specifying number of max results
for file_list in drive.ListFile({'maxResults': 10}):
    print 'Received %s files from Files.list()' % len(file_list)
    for file1 in file_list:
        print 'title: %s, id: %s' % (file1['title'], file1['id'])
```

Accessing to Dropbox using Python Core APIs

The Core API is based on HTTP and OAuth and provides low-level calls to access and manipulate a user's Dropbox account

```python
# Include the Dropbox SDK
import dropbox

# Get your app key and secret from the Dropbox developer website
app_key = 'INSERT_APP_KEY'
app_secret = 'INSERT_APP_SECRET'

flow = dropbox.client.DropboxOAuth2FlowNoRedirect(app_key, app_secret)

# Have the user sign in and authorize this token
authorize_url = flow.start()
print '1. Go to: ' + authorize_url
print '2. Click "Allow" (you might have to log in first)'
print '3. Copy the authorization code.'
code = raw_input("Enter the authorization code here: ").strip()

# This will fail if the user enters an invalid authorization code
access_token, user_id = flow.finish(code)

client = dropbox.client.DropboxClient(access_token)
print 'linked account: ', client.account_info()

f = open('working-draft.txt', 'rb')
response = client.put_file('/magnum-opus.txt', f)
print 'uploaded: ', response

folder_metadata = client.metadata('/')
print 'metadata: ', folder_metadata
```
Mobile

The mobile application is constructed with Xcode version 7.3 using the Swift programming language version 2.2. The mobile application will be require iOS 8 or newer powered by an iphone 5 or newer. The mobile application will be communicating with the backend API through JSON calls. In order to make the JSON calls we will be Utilizing the Swift framework SwiftyJSON that allows for simple and error proof parsing of JSON calls. The mobile application will give the user the ability to create an account or log in and access the storage platform of their choosing, currently only dropbox and google drive supported, and create/edit personal or collaborative stories of them and their friends with the ability to easily share those photos via downloadable link. Each story will have a owner (storyteller) , and a collection of moments. The moments of a story are composed by a collection of photos. A user can create a story and a moment from their selection of photos or chose to add those photos to an existing moment. This story -> moments -> photos architecture is exemplified in the image below.

Figure M6 - Story Composition
When the user opens the mobile application screen they will first encounter the login screen, here they are required to place their username and password to login or register if they are unregistered. Once the server has authorized the login credentials, the user will gain access to the main menu of the application screen. The main menu will contain the a progress bar representing the total storage room and storage room consumed on their storage platform along with four connecting portals: Add story, View/Edit story, Account, and logout. Add story will grant the user access to the photos in their local phone library. The photos will be displayed in collection view format exemplified by an image below. The user will have the option to select all photos they would like to add to a story/moment. Once the photos collection has been selected the user will have the option to go through each photo and perform simple editing then once they select to upload, they will chose an existing story/moment or to create one. User will be returned to the add story screen where they can chose to go back to main menu or add more photos. View/Edit story will grant the user access to all the stories in their storage device, here the user will select the story and moment they would like to view. The photos will be displayed in collection view format similar to Add story. User will have the freedom of going through the photos and adding comments or the user can opt to select a group of photos and create a downloadable link with which they can share with others or delete them. A user can return to the main menu from here at anytime. The accounts portal will give the user access to their user information with the option of changing their password. Lastley the logout button will terminate the user's access to the server and return them to the login screen. Below are a series of diagrams and photos to help further explain the design.

**Figure M7 - Mobile Application Case Diagram**
This case diagram represents the options a user has when using the mobile application, note that a logged in user is a type of user whom has already registered.

Figure M8 - Mobile Application Sequence Diagram
The Sequence diagram above represents a simplified version of the mobile application interpreting user input and communicating that with the server. Also note that this sequence diagram does not demonstrate the registration process which has its own unique sequence.

Although the proposed design of functionality of the Mobile application is finalized given the research and information we have acquired to this point, the GUI design has not. Since user input will influence GUI design process to a much larger degree it will be subject to changes, although current prototypes have been constructed which I demonstrate below.

**Figure M9 - Mobile Login Screen**
This is the prototype GUI login screen, which requests the user's login information. The login button will submit the user input and request access to the main screen. The forget password will redirect users to a different UI to answer verification questions in order to email their password. There will be a register option added to this screen, which will redirect users to a separate UI to fill out account information. This screen will in all likelihood remain the same outside of the addition of a register button.

**Figure M10 - Mobile Main Menu**
This is the GUI main menu prototype with which the user gains access after login credentials have been verified. This screen contains the four separate buttons described in detail above which grant the user access to the other application features. The main menu actively monitors users storage room in gigabytes.

*Figure M11- Collection View Screen*
This is a GUI prototype for the collection view, this is the method in which photos will be displayed both in add story and view/edit story. The blue check marks represent all photos selected and awaiting instruction.

**Figure M12- Mobile Photo Editing Screen**
This is currently the GUI prototype for the photo editing utility used by the add story feature. Here users will have simple photo editing options such as cropping a photo or adjusting lighting. Due to the importance we place on simplicity of use and fluidity of the mobile application this prototype is subject to many changes dependent on user experience and input.

Mobile Testing
As mentioned many times through the design detail we stress the importance of simplicity and user experience because this is a part of our mission not only for the mobile application but for Imageshare entirely. To ensure simplicity and positive user experience for the mobile application we have designed a thorough and descriptive testing phase.

First a qualified group of individuals whom already own an IPhone 5 or newer with iOS 8 or newer will be hand selected as our testing group. The mobile application will undergo rigorous testing far before it reaches the hands of the test group. A mobile application will be first tested via the xcode simulation device making sure all feature both display and function properly. Once simulation testing has been completed the mobile application will be uploaded to our IPhone 6 where it will undergo similar testing but in real world.

Second, after the mobile application has completed all testing phases it will be given to the select user group for further testing to check for stress testing, functionality testing and user feedback/opinion. Each user of the test group will be required to complete the mobile evaluation form requesting important information from the user about their experience for each version of the mobile application. All mobile evaluation forms will be collected and evaluated, the group will use the forms make a decision on necessary changes to the mobile application. If changes are required, the requested changes are made via Xcode and sent back through the testing loop described.

Finally, when the mobile application has completed all testing phases and the group has decided that the last round of mobile evaluation forms proved to not require any changes to the mobile application, the mobile application is finalized. At this time the mobile application will be submitted to apple to undergo their layer of testing before reaching the app store where anyone in the world with the necessary IPhone can download and use.

Below is an image of the mobile application evaluation form that will be completed by the testing group. The form covers such topics as bug/errors along with features the user like and disliked. The last question covers the video feature which our group feels will ultimately negatively impact user experience. A version of the mobile application will allow for videos to be a part of moments along with photos. Since our group values user input, we will reverse our choice to remove the video feature given that users find it to be a positive addition to the mobile application without adding complexity to much its use. Complexity of use needs to be low for the mobile application so that our user base can include the elderly and users who do not have the greatest smart phone / computer understanding.

**Figure M13- Mobile Application Evaluation Form**
This is the mobile evaluation form described in the testing portion of the mobile design. It contains five simple questions the users will fill out and return for each version of the mobile application they test.
Web

The design architecture chosen for the web application is that of Model View Controller (MVC), the default configuration of Ruby on Rails is to enforce MVC design patterns for easy of rapid prototype development in the web application. The web application is the staging platform for all services proposed for the Imageshare service and as such any accepted design or activity implemented in the web will be pushed to the mobile client.

User Web Workflow

The first time a visitor enter the web application they are presented with an introductory home page describing the services of the Imageshare platform and offers clear links to sign-up for the service. Once a user proceeds through the initial register process they will be able to access the core services of the web application as a logged in user with the first primary options being view current stories from the site's main menu bar.

Once a user is directed to the view a story screen they are presented a album view of all their currently existing stories from which they may view or edit. Within the story view when a story is hovered over two icons appear over the story view in the form of a COG and and a X icon from which they may modify the album, COG opening up a edit pane and the X prompting the user if they would like to delete the current story. If a story is selected by clicking on it they are then shown the the moments view screen containing all related moments to that story see picture W.4.0 all features in the story view are applied to the moments view.

In order to create a new story the user must be at the story album view screen from there a large button titled New Album will be presented, once clicked a dialog will appear with fields for a user to initialize a new album with basic default information. To create a new moment a user must select an existing story and within the moments album view in place where New Album was located a New Moment button will be presented, once clicked a dialog will appear for a user to initialize a new moment.

Web User Workflow Diagram
Image W.1.1
Caption Text
Image W.1.1
This is the current first step in the registration page. The user must enter a valid email address and password before they can proceed by pressing **Next**. Once they do ajax transitions the page elements to step two and saves the user’s information in a data register object. The primary purpose of this screen is to receive the user’s basic information for site login credentials.

Image W.1.2
This is the current second step in the registration process. The user is presented a list of all the supported hosting platforms (at this current time only google drive and dropbox have support). Once a user clicks on their host of choice a dialog containing a iframe will appear to allow a user to valid Imageshare’s access to their cloud storage. Once a user has
successfully linked an account a green checkmark will appear their associated cloud provider and then they may proceed by clicking **Next**.

**Image W.1.3**

This is the current screen for step three in the registration process. At this point a user’s credentials have been validated and an account created with an associated cloud storage account at this point the user may login to the web application by clicking **Login** in the menu bar or button on the page, or by logging in via the mobile application.

**Image W.2.0**
This is the current user login screen. The user is prompted to enter their email and password for the Imageshare account they created in the registration process and then click **Login**. Once a user has submitted their login request their credentials are validated and if valid will be forwarded into the application, else if the information is invalid the user will be prompted with the message “matching email and password are not found”.

---

**Image W.3.0**

This is the current account management page. The user has the ability to enter a new email address and click on update or change their current password for a new one. Located at the bottom of the account management page is a the Usage Meter showing the user how much available storage they have left with their current cloud configuration settings, once a user goes past their warning level an email will be sent out to them notifying them to increase their storage. If a user chooses to change image host or increase their allocated storage or warning notification level they may click on the **Update Cloud Settings** button which will present a dialog to update settings.
This is a mockup view of what viewing a story screen looks like. Once the use has selected a story to view the story’s moments are listed in a album view which each contain a snapshot of what is contained within the view the default containing a map of where the event is from, relevant photos, and moment title. Once a user selects a moment to view a dialog will appear with the moments view.

Explicit Design Summary with diagrams

Backend

Since both Google Drive and Dropbox both have fully functional APIs for Python, we decided to use Django web framework. Django is designed to encourage loose coupling and strict separation between pieces of an application. If we follow this philosophy, it’s easy to make changes to one particular piece of the application without affecting the other pieces. Those three pieces together – data access logic, business logic, and presentation logic – comprise a concept that’s sometimes called the Model-View-Controller (MVC) pattern of software architecture. In this pattern, “Model” refers to the data access layer, “View” refers to the part of the system that selects what to display and how to display it, and “Controller” refers to the part of the system that decides which view to use, depending on user input, accessing the model as needed.

Django follows this MVC pattern closely enough that it can be called an MVC framework. Here’s roughly how the M, V, and C break down in Django:
- M, the data-access portion, is handled by Django’s database layer, which is described in this chapter.
- V, the portion that selects which data to display and how to display it, is handled by views and templates.
- C, the portion that delegates to a view depending on user input, is handled by the framework itself by following your URLconf and calling the appropriate Python function for the given URL.

Because the “C” is handled by the framework itself and most of the excitement in Django happens in models, templates and views, Django has been referred to as an MTV framework. In the MTV development pattern,

- M stands for “Model,” the data access layer. This layer contains anything and everything about the data: how to access it, how to validate it, which behaviors it has, and the relationships between the data.
- T stands for “Template,” the presentation layer. This layer contains presentation-related decisions: how something should be displayed on a Web page or other type of document.
- V stands for “View,” the business logic layer. This layer contains the logic that access the model and defers to the appropriate template(s). You can think of it as the bridge between models and templates.

MySQL was chosen to be our database to store all user information and data which is retrieved from cloud storage by using Python APIs provided. All requests and responds
from the web server will be through REST api. This will make the communication among web and mobile application easy and effective.

Mobile

Mobile application design is composed of three main phases: Mobile application GUI, coding application functionality, and testing. Phases are designed and worked on in the order they were listed i.e. GUI is designed then its functionality is coded followed by testing of implementation. The mobile application is also designed in a loop format so that any changes required in the testing phase will send the design back to the GUI phase. Swift will be used on our MacBook pro in all three phases of design and is one of two languages available to design any software for apple products. The application will be deployed to our iPhone 6 during the testing phase at first then to our handpicked test user group. The application is designed with Swift 2.2 for iPhones 5 or newer containing iOS 8 or newer. I will be using two Swift frameworks to design the mobile application: SwiftyJSON which will be used to parse and make JSON calls to the server and MWPhotoBrowserSwift a framework that allows for displaying photos as collections with the option of viewing or editing a photo that is a part of that collection. Mobile application will have a completion deadline of July 20th 2016.

The GUI for the mobile application consists of login page, main page, add story, view/edit story, account and log out. Login in page will contain a text field for username and password along with buttons for login, register, forgot password.

- Valid username and password submitted with the login button will grant the user access to the main page.
- Register page consists of input fields for name, username, email, password, storage type (either google drive or dropbox) along with the login information for the storage site and a button to submit information. Submit button returns user to login page.
- Forgot password page will have input field requesting email and a submit button returning you to the login page.

Main page contains a progress bar with storage information along with buttons labeled and granting access to add story, view/edit story, account, and logout.

- Add story page will display photos in a collection view style here there will be a button labeled select photos that enables the user to select that group of photos for uploading. Once the Photos are selected they will be brought to a separate page where they could be viewed in a collection with the option of being selected individually for editing or upload. Once upload is selected the user is asked if they want to add the photos to an existing story and moment or if they would like create a story and/or moment.
- View/Edit story page will allow the user to select any personal or shared story. A user can delete or comment on a given story. The user then selects moments, here they have the freedom of commenting or deleting the moment. When moment is selected a collection view of photos contained in that moment will be displayed, a group of photos can be selected to create a shareable link. Each photo can be selected for comment or deletion.
● The account page will have labels for user’s name, username, storage type, storage room remaining and an option to change password.
● Logout button that terminates user’s connection with the service and return them to the login page.
The GUI phase of the mobile application will be completed by June 4th. The completion of the GUI phase begins the coding phase.

The Coding functionality phase does as it's named and adds functionality and logic to the GUI and user input acquired from the GUI. The login page code will consist of:

● username and password is sent to the service when the login button is pressed. The service will respond with a key. The key will be received by the user and sent back to the service requesting access to the storage unit. If access is granted, then the user gains access to the main screen.
● Forgot password will send user input to service. Service checks if email of user exists and sends email to corresponding user if it does exist.
● Create account will take the information from the input and send to service along with request store this information in the database. Service responds with Boolean for success, if true, system sends user to login screen.

Functionality for the main page will be far more complex. Each command will require the application to make many logical decisions.

● Main page sends key along with int request for maximum user storage and current user storage from service. Service responds with requested int values for corresponding key. System adds values to progress bar.
● Add Story requests Photo assets (local iPhone photos) in the form of an array and passes them to the MWPhotoBrowserSwift framework to construct a collection view.
  ○ Selected array of photos will be passed back into the MWPhotoBrowserSwift framework for another collection view of selected photos.
    ■ Photos are individually edited via the MWPhotoBrowserSwift framework.
  ○ Application requests to upload photo array to service along with key. Service responds with an array of Story ID’s. User can respond with Story ID and key or request new Story ID. User sends key, Story ID to service and requests Moment ID. Service responds with array of Moment ID’s linked to Story ID and key. User can request new Moment ID.
    ■ User sends photo array along with valid ID’s and key, service sends boolean for success. If success user is returned to main menu
● View/Edit Story sends a key along with request for Valid Story ID’s linked to user Key. Service responds with array of Story ID’s linked to user key. User sends key and Story ID requesting Moment ID’s. Service Responds with Moment ID array corresponding to key and Story ID. User sends key, Story ID and Moment ID to Service. Service responds with corresponding photo array.
  ○ Photo array received is sent to MWPhotoBrowserSwift framework to construct a collection view of photos. Photos can be Selected.
- Selected Photos receive comments by user. System makes changes to photo array based off user input.
  - User sends key, Story ID, Moment ID, and changed array. Service registers changes and responds with boolean. If boolean is true, system sends user back to beginning view/edit story.
- Selected Photos are deleted by User. System makes changes to photo array based off user input.
  - User sends key, Story ID, Moment ID, and changed array. Service registers changes and responds with boolean. If boolean is true, system sends user back to beginning view/edit story.
- User selects photos and sends Story ID, Moment ID, key and array of selected photo ID’s with request for downloadable link in the form of a string from service. Service responds with string containing public link to previously selected photos.
  - Account User Requests account data from service by sending user key. Server responds with string data for all labels in the UIView.
    - User selects to change password and request is sent to service to update user password in database. Boolean is sent by the server to the user verifying success.
  - Logout requests to terminate connection from service, boolean is returned. If server returns true, user is returned to login screen.

The functionality and coding phase of the diagram has a completion deadline of June 30th. This is the heart of the mobile application and will prove to be the most difficult part. All communication with the service will be performed using the SwiftyJSON framework. The completion of this phase will mark the beginning of the testing phase.

The testing phase will consists of simulation testing performed on the MacBook Pro via Xcode, real world testing on a live Iphone 6 and finally testing performed by a group of handpicked beta testers who own qualified iPhones. The completed mobile application will be sent to apple where further testing will happen before it’s release in the application store. In order to perform the testing our team will purchase a developer account with apple which allows us to distribute the mobile application from the Macbook Pro to iPhones. The apple developer account will cost $99 for an entire year.

1. Simulation testing is performed on the Macbook Pro in Xcode. In this step we will test the ability for the mobile application to compile/execute on all iPhone models 5 and newer. The simple functionality of all the buttons and UIviews is tested.
2. Real world testing will be conducted after simulation, here the mobile application will be transferred to our personal iPhone through usb using Xcode. In this step we will perform all the functions of the application checking for fluidity and functionality of all features. Apple developer license must be purchased in order to conduct this step.
3. Beta testing is a complex phase that involves selecting beta testers with qualified Iphones and registering them to apple. Once the group of beta testers has been collected, each tester must submit their IPhone device ID’s to us. The device ID’s,
App ID and the developer certificate is submitted to apple in order to create an “ad hoc profile”. This profile gives permission to the IPhone of beta testers to download the mobile application. Once the Ad hoc profile is made, Xcode combines that ad hoc profile and the application. Xcode will request the html of the website which will host the mobile application. Once Xcode receives all necessary information a folder is created containing the mobile application and a manifest.plist file which contains the necessary information the iPhone needs to install the application via web server. Beta testers are instructed to download the mobile application from our mobile site and install it. Beta testers will be asked to use the mobile application in their daily lives and to report immediately by email to any group member incase terminal errors are encountered. After a few days the users will be asked to complete submit the mobile application evaluation form.

The team will evaluate the results of the form and determine if necessary changes need to be made to the mobile application. If changes need to be made the design process will restart with the implementing the new changes.

The mobile application testing has an estimated completion time of July 18th. When the team has reached a collectively decided the application is finished, it will be submitted to apple for testing before being added to the application store.

This design process implements a three step approach. Each step contains an estimated time of completion along with a detailed description of said step. The mobile application for Imageshare aims to achieve almost 100% similar functionality to the web application, although in some cases this will be impossible due to certain limitations in the Swift programming language.

Web

Development of the web application will be built in multiple stages first comprising of base system settings including user authentication and session management followed by implementing moment contains and finally story containers. The first stage will begin with the implementation of all system routing and the setting up of the login and session management controllers with the assistance of the the Devise ruby gem. Once devise has been installed all user session flow routes are generated including login, sign-up, and logout screens and page authentication. After Devise is fully incorporated in the new rails project apart of the first stage is implantation of the navigation system and initial UI design of all elements including web forms and navigation headers.

The second stage In the web application development is setting up moment’s within the application which standalone without an associated story. Including in this stage is complete implementation of all moment properties including title, description, location, and photos. Once moments view is complete then public sharing of moments will be setup.

The third stage in the development process is setting up of story view containers which groups multiple stories in a container view with their own title, description, and locations. Once story containers are complete public sharing of moments will be extended to support sharing stories.
Pending fourth stage is adding photo editing. Currently no pre-existing ruby gems are available for web editor views so sub-gems that offer minor editing components will be research pending time to constructing a custom web-image editor for Ruby of Rails.

**Image W.5.0**

Current overview of web application design stages.
Build, prototype, test, and evaluation plan

Backend

We decided to build our web server on localhost first hand. Apache web server will be installed along with Django web framework. All the services and APIs offered from backend will be implemented and fully tested with web and mobile application to ensure being fully functional.

Testing tools

Django provides a small set of tools that come in handy when writing tests:
- The test client

The test client is a Python class that acts as a dummy Web browser, allowing you to test your views and interact with your Django-powered application programmatically.

Some of the things you can do with the test client are:

- Simulate GET and POST requests on a URL and observe the response – everything from low-level HTTP (result headers and status codes) to page content.
- See the chain of redirects (if any) and check the URL and status code at each step.
- Test that a given request is rendered by a given Django template, with a template context that contains certain values.

Mobile

We plan on building the application using Swift on our own MacBook pro. We will compile/test the product on our MacBook pro using the simulator to perfect and fine tune the mobile application before constructing the prototype. Once a fully functional prototype is finally constructed, We will test its functionality on our own iPhone to check for bugs and ensure fluid functionality. Once we have completed testing of the prototype’s functionality on our own phone, we will go back and finalize cosmetic GUI changes to what the mobile application is meant to look like. Once more the final design will be tested for functionality on our phone. After all previous checkpoints have been passed a group of iPhone users will be selected and told to regularly use the application and report back any bugs or glitches they experience. All bugs will send the application back to design and personal testing until they have all been solved before releasing the update. Once all bugs and glitches have been worked out, the user base will be asked to fill out a short survey asking to name a few things they like and a few things they could change or add. All surveys will be taken into consideration when designing the final polished product. The design diagram visualizes this process.

Figure M14- Testing Design Flow
First a Ruby on Rails server is going to deployed on local vagrant boxes to allow local development. After successfully completion of the four stages in the web application design process the Rails application will be migrated to a remote Ubuntu server box on Google Cloud’s Platform service. Throughout the development of the web application the Zentest ruby gem will be utilized to verify system changes to ensure added features do not break existing use cases. After the web application has been deployed to a remote Ubuntu box a private beta will be opened to various members of the UCF community to test the application themselves before being open to a global open beta through the use of Reddit forums. After a participant has completed using the system they will be provided an online google form to fill out with suggestions or bugs found within the system.

Personnel and bibliography of related work
Backend

We have a backend developer who has been working for a couple years. With the experiences that I gained from my work, it has definitely helped me in designing the database, services and APIs for the backend to meet the requirements of the system.

Mobile

We have designed a few different applications for swift. Many have been of the simpler form such as calculators which used a simple view controller to perform simple arithmetic, a weather reader which used geolocation to locate the user and used JSON to communicate with an online resource that provided the current temperature for the user's location. Our most recent application that we’ve worked on was a video streaming application that used JSON to communicate with Twitch.tv, an online streaming site for gamers. Our twitch application displayed the top 25 most popular viewers at that hour as a thumbnail of their video in tableview format with a short description of the streamer. Selecting the thumbnail would open a mobile live stream of the selected viewer. Outside of this we have created countless application models using the GUI design for mobile applications provided by swift.

We have previously used JSON to communicate with servers on many different occasions from creating a personal website to a fully functioning interactive one. Our most recent work involved the design of a website that professors could select and review applications for new graduate teaching assistants for a databases class that involved using JSON to relay user and applicant information to the server.

Web

Our web developer has had over three years developing web applications using PHP frameworks for various web development firms. Skillsets in working with JSON objects, restful web services, amongst other web features will be able to be carried over to the current rails platform. With his experience in design and web implementations the overall design of the platform and mobile services will be handled by this member.

Facilities and Equipment

Figure M17-Macbook Pro-Public
The Sr. design lab on campus will be used for the team to collaborate on ideas and to perform testing on the entire system. To code, compile and test the mobile application we will be using a 15” MacBook Pro. An apple computer is required to compile and execute Swift and Objective C code for our mobile application.

**Figure M18-iPhone6-Public**

fortunately for us, we already own a Macbook Pro. After the mobile application is fully functional we will download it to our own Iphone. Current specifications of the mobile application require an iphone 5 or newer and iOS 8 and newer. Our personal Iphone is an Iphone 6 with iOS 9, which will suffice for testing. When searching for additional users for beta testing, we will require that they supply their own compatible Iphone with which to test the mobile application. All development for our backend and web application will also be completed on our Macbook pro although all backend and web services code can be programmed on any OS platform.

**Consultants, subcontractors, and suppliers**

All software frameworks used across all of our development arms including backend, web, and mobile all use open source well documented source code. With all source code being well documented all resources required to implement our platform is free and available online through all of the frameworks’ documentation and online tutorials throughout our learning and implementation process. All code outside of frameworks will be written by our team so there is no need for any subcontractors and the only external help we are receiving is Dr. Mark Heinrick for his insight in the photo sharing industry. The single supplier used in our platform is Google and it’s Cloud hosting services for hosting our databases and web servers.

**Budget and financing**
Backend

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Source</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google Cloud Platform</td>
<td>Hosting the web server application</td>
<td>Google</td>
<td>$0.03/Month Per GB Storage $4.09/Month Entry Level Linux Server</td>
</tr>
</tbody>
</table>

Mobile

The cost of building the mobile application will be composed of three separate parts: material, licensing and testing. The materials necessary for the project consist of a Mac computer/Laptop and IPhone 5 or newer. The required licensing for the mobile application is the apple developer license. The developer license allows for the distribution of constructed mobile applications by purchasing developer which also includes an apple developer account and access to a developer technical support team.

**Figure M15- Apple Developer Licence Fee**

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Cost</th>
<th>99 USD*</th>
</tr>
</thead>
<tbody>
<tr>
<td>13+</td>
<td>Free</td>
<td></td>
</tr>
<tr>
<td>18+</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Lastly, in order to effectively test the application in action, a beta test group owning a compatible IPhone will be selected to use the mobile application and provide feedback. We estimate that the total cost of the mobile development process will cost $149. All funding will be provided by the team members. The table below represents a balance sheet for the mobile application.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Source</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laptop</td>
<td>MacBook Pro</td>
<td>Team Members</td>
<td>$0</td>
</tr>
<tr>
<td>Mobile Phone</td>
<td>IPhone 6</td>
<td>Team Members</td>
<td>$0</td>
</tr>
<tr>
<td>License</td>
<td>Apple Dev License</td>
<td>Team Members</td>
<td>Cost</td>
</tr>
<tr>
<td>----------</td>
<td>------------------</td>
<td>--------------</td>
<td>--------</td>
</tr>
<tr>
<td>Testing</td>
<td>Test Subjects</td>
<td>Team Members</td>
<td>~$50</td>
</tr>
</tbody>
</table>

Web

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Source</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google Cloud Platform</td>
<td>Hosting the web server application</td>
<td>Google</td>
<td>$4.09/ Month Entry Level Linux Server</td>
</tr>
</tbody>
</table>

Milestone chart for all activities related to the project

Backend

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase 1</td>
<td>Research APIs of the Google Drive and Dropbox</td>
<td>03/20/2016</td>
</tr>
<tr>
<td>Phase 2</td>
<td>Choosing the Web Framework</td>
<td>03/25/2016</td>
</tr>
<tr>
<td>Phase 3</td>
<td>Database design</td>
<td>04/01/2016</td>
</tr>
<tr>
<td>Phase 4</td>
<td>API implementation for Google Drive</td>
<td>05/05/2016</td>
</tr>
<tr>
<td>Phase 5</td>
<td>API testing for Google Drive</td>
<td>05/10/2016</td>
</tr>
<tr>
<td>Phase 6</td>
<td>API implementation for</td>
<td>05/20/2016</td>
</tr>
<tr>
<td>Phase</td>
<td>Description</td>
<td>Date</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>7</td>
<td>API testing for Dropbox</td>
<td>05/25/2016</td>
</tr>
<tr>
<td>8</td>
<td>Integration testing with web application</td>
<td>07/15/2016</td>
</tr>
<tr>
<td>9</td>
<td>Integration testing with mobile application</td>
<td>07/18/2016</td>
</tr>
<tr>
<td>10</td>
<td>Deployment on Google cloud host service</td>
<td>07/22/2016</td>
</tr>
</tbody>
</table>

**Mobile**

The mobile application will be composed of three major sections and varying subsections corresponding to each of the major sections. The three sections are GUI design, coding, and testing. The deadline for the completion of all three phases of the mobile application will be July 20th. We chose this deadline in order to have some breathing room for unexpected design changes and errors that might occur along the way.

First major section is composed of building the GUI for the mobile application. The GUI subsections are composed of the login screen screen design, main menu design, collection view design for uploading and downloading, and lastly the photo editing feature. All elements of the GUI design have a completion deadline of June 4th. We believe this will be one of the faster phases of application development.

Second Major section will be coding and implementing functionality to all the GUI sections designed in section 1. The coding subsection consists of establishing communication with the server via JSON, Login functionality/ user registration, main menu functionality, photo editing functionality, and lastly debugging the code. We have set a completion deadline for this section by June 30th. We believe that this will prove to be the most difficult and time consuming section of the mobile application development.

The last major section is the Testing and user feedback process. Once functionality is completed and debugged in section 2, the mobile application will be dispersed to a hand selected group of test users, their objective will be to use the application and report errors they encounter along with filling out a short feedback form describing the features they like and the features they would like to change and add. Simply this section is composed of functionality testing, evaluation of user experience and lastly implementing any design changes if any. This phase will have a completion deadline of July 18th.
Figure M16- Milestone Pie Chart

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase 1</td>
<td>GUI Design/ Creation</td>
<td>June 4th</td>
</tr>
<tr>
<td>Phase 2</td>
<td>Coding functionality</td>
<td>June 30th</td>
</tr>
<tr>
<td>Phase 3</td>
<td>Testing</td>
<td>July 18th</td>
</tr>
<tr>
<td>Completion</td>
<td>Mobile Completing</td>
<td>July 20th</td>
</tr>
</tbody>
</table>

Web
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Management</td>
<td>Registration, Session Management</td>
<td>May 16th</td>
</tr>
<tr>
<td>Moment Views</td>
<td>Create, Edit, View Moment Views</td>
<td>June 01st</td>
</tr>
<tr>
<td>Story Views</td>
<td>Create, Edit, View Stories</td>
<td>June 16th</td>
</tr>
<tr>
<td>Public Views</td>
<td>Public Viewing of Stories</td>
<td>June 30th</td>
</tr>
<tr>
<td>Closed Beta Testing</td>
<td></td>
<td>July 1st</td>
</tr>
<tr>
<td>Open Beta Testing</td>
<td></td>
<td>July 14th</td>
</tr>
</tbody>
</table>

Project Summary and conclusions.

Backend

Imageshare’s backend will primarily be used to handle the storage layers and management of all story and moment views alongside user accounts. The back end storage layers unique ability to store photos across multiple cloud services including google drive and dropbox allow the user to maintain control of their photos while still having access to the power of all of Imageshare’s services.

Mobile

Imageshare will support a mobile application for the iPhone that will be designed with Swift using Xcode on a MacBook Pro. The mobile application will require the purchase of an apple developer license in order to distribute. The mobile application will be recommended for download to all iPhone users attempting to access the site via web portal. Therefore, the mobile application will attempt to utilize all features of the Imageshare platform with the given Swift libraries supplied by Apple Inc. In some cases the mobile application might not have every ability offered by the web service due to limitations in the Swift API, although this will be a rare occurrence if an occurrence at all. The mobile application will allow users to retrieve lost passwords, create an Imageshare account and login. A logged in user will be presented with their storage
capacity and allowed to create, view/edit/delete and comment on stories and moments. Additionally, the logged in user will have the ability to perform simple photo editing and view/edit certain account credentials. The mobile application will be tested in 3 different ways: simulation, real time single user, and eventually real time multi-user by hand selected beta testers that will be required to fill out an evaluation form based on their experience. Based on the beta tester feedback the team will make real time adjustments to the mobile application and repeat this process until perfected. It is our goal as a team to create every element of this platform with the same basic qualities: Ease of use, powerful, comprehensive, and fun. Therefore, the mobile application was designed with those 4 goals in mind. The estimated total cost for designing the mobile application will be $149. The estimated completion date for the mobile application is July 20th. All design data presented in this document concerning the mobile application’s functionality will in all likelihood remain the same, however its method of implementation is subject to change based on feedback from beta testers.

Web

The web application of Image Share is designed to service as our cross-platform solution in lieu of native desktop development. The web application will be available to all modern browsers including Safari, Firefox, Chrome, and Edge allowing the user to access to their content the way they want to. The web application is to serve as the staging platform for any new services added to the Imageshare platform with all accepted features and designs eventually being deployed to mobile platforms. For many users it is expected that the web application will be the first time they experience our platform and as such extra attention is going to be given to it to help the user’s experience.

References

Mobile Matrix, Comcourse 20 April 2016, Mon. 25 April 2016
Get on with building your app, not duplicating code. Cocoapods, Sun. 24 April 2016
Tangplin, SwiftlyJson, GitHub, Tues. 26 April 2016
MWPhotoBrowserSwift, GitHub, Tues. 26 April 2016

Appendices

Copyright permissions
A. Copyright permissions

Figure M15-

Ownership of Site; Agreement to Terms of Use

These Terms and Conditions of Use (the “Terms of Use”) apply to the Apple web site located at www.apple.com, and all associated sites linked to www.apple.com by Apple, its subsidiaries and affiliates, including Apple sites around the world (collectively, the “Site”). The Site is the property of Apple Inc. (“Apple”) and its licensors. BY USING THE SITE, YOU AGREE TO THESE TERMS OF USE; IF YOU DO NOT AGREE, DO NOT USE THE SITE.

Apple reserves the right, at its sole discretion, to change, modify, add or remove portions of these Terms of Use, at any time. It is your responsibility to check these Terms of Use periodically for changes. Your continued use of the Site following the posting of changes will mean that you accept and agree to the changes. As long as you comply with these Terms of Use, Apple grants you a personal, non-exclusive, non-transferable, limited privilege to enter and use the Site.

Content

All text, graphics, user interfaces, visual interfaces, photographs, trademarks, logos, sounds, music, artwork and computer code (collectively, “Content”), including but not limited to the design, structure, selection, coordination, expression, “look and feel” and arrangement of such Content, contained on the Site is owned, controlled or licensed by or to Apple, and is protected by trade dress, copyright, patent and trademark laws, and various other intellectual property rights and unfair competition laws.

Except as expressly provided in these Terms of Use, no part of the Site and no Content may be copied, reproduced, republished, uploaded, posted, publicly displayed, encoded, translated, transmitted or distributed in any way (including “mirroring”) to any other computer, server, Web site or other medium for publication or distribution or for any commercial enterprise, without Apple’s express prior written consent.

You may use information on Apple products and services (such as data sheets, knowledge base articles, and similar materials) purposely made available by Apple for downloading from the Site, provided that you (1) not remove any proprietary notice language in all copies of such documents, (2) use such information only for your personal, non-commercial informational purpose and do not make any additional representations or warranties relating to such documents.

Your Use of the Site

You may not use any “deep-link”, “page-scrape”, “robot”, “spider” or other automatic device, program, algorithm or methodology, or any similar or
equivalent manual process, to access, acquire, copy or monitor any portion of the Site or any Content, or in any way reproduce or circumvent the navigational structure or presentation of the Site or any Content, to obtain or attempt to obtain any materials, documents or information through any means not purposely made available through the Site. Apple reserves the right to bar any such activity.

You may not attempt to gain unauthorized access to any portion or feature of the Site, or any other systems or networks connected to the Site or to any Apple server, or to any of the services offered on or through the Site, by hacking, password "mining" or any other illegitimate means.

You may not probe, scan or test the vulnerability of the Site or any network connected to the Site, nor breach the security or authentication measures on the Site or any network connected to the Site. You may not reverse look-up, trace or seek to trace any information on any other user of or visitor to the Site, or any other customer of Apple, including any Apple account not owned by you, to its source, or exploit the Site or any service or information made available or offered by or through the Site, in any way where the purpose is to reveal any information, including but not limited to personal identification or information, other than your own information, as provided for by the Site.

You agree that you will not take any action that imposes an unreasonable or disproportionately large load on the infrastructure of the Site or Apple's systems or networks, or any systems or networks connected to the Site or to Apple.

You agree not to use any device, software or routine to interfere or attempt to interfere with the proper working of the Site or any transaction being conducted on the Site, or with any other person's use of the Site.

You may not forge headers or otherwise manipulate identifiers in order to disguise the origin of any message or transmittal you send to Apple on or through the Site or any service offered on or through the Site. You may not pretend that you are, or that you represent, someone else, or impersonate any other individual or entity.

You may not use the Site or any Content for any purpose that is unlawful or prohibited by these Terms of Use, or to solicit the performance of any illegal activity or other activity which infringes the rights of Apple or others.

Purchases; Other Terms and Conditions
Additional terms and conditions may apply to purchases of goods or services and to specific portions or features of the Site, including contests, promotions or other similar features, all of which terms are made a part of these Terms of Use by this reference. You agree to abide by such other terms and conditions, including where applicable representing that you are of sufficient legal age to use or participate in such service or feature. If there is a conflict between these Terms of
Use and the terms posted for or applicable to a specific portion of the Site or for any service offered on or through the Site, the latter terms shall control with respect to your use of that portion of the Site or the specific service.

Apple’s obligations, if any, with regard to its products and services are governed solely by the agreements pursuant to which they are provided, and nothing on this Site should be construed to alter such agreements.

Apple may make changes to any products or services offered on the Site, or to the applicable prices for any such products or services, at any time, without notice. The materials on the Site with respect to products and services may be out of date, and Apple makes no commitment to update the materials on the Site with respect to such products and services.

The following terms also govern and apply to your use of the Site, and they are incorporated herein by this reference:

- Guidelines for Using Apple Trademarks & Copyrights
- Rights & Permissions
- Trademarks
- Claims of Copyright Infringement
- Piracy
- Counterfeit Products
- Apple’s Unsolicited Idea Submission Policy
- Software License Information
- Legal Contacts

Each of these policies may be changed from time to time and are effective immediately upon posting such changes on the Site.

Accounts, Passwords and Security
Certain features or services offered on or through the Site may require you to open an account (including setting up an Apple ID and password). You are entirely responsible for maintaining the confidentiality of the information you hold for your account, including your password, and for any and all activity that occurs under your account as a result of your failing to keep this information secure and confidential. You agree to notify Apple immediately of any unauthorized use of your account or password, or any other breach of security. You may be held liable for losses incurred by Apple or any other user of or visitor to the Site due to someone else using your Apple ID, password or account as a result of your failing to keep your account information secure and confidential.

You may not use anyone else’s Apple ID, password or account at any time without the express permission and consent of the holder of that Apple ID, password or account. Apple cannot and will not be liable for any loss or damage arising from your failure to comply with these obligations.
Privacy
Apple’s Privacy Policy applies to use of this Site, and its terms are made a part of these Terms of Use by this reference. To view Apple’s Privacy Policy, click here. Additionally, by using the Site, you acknowledge and agree that Internet transmissions are never completely private or secure. You understand that any message or information you send to the Site may be read or intercepted by others, even if there is a special notice that a particular transmission (for example, credit card information) is encrypted.

Links to Other Sites and to the Apple Site
This Site may contain links to other independent third-party Web sites (“Linked Sites”). These Linked Sites are provided solely as a convenience to our visitors. Such Linked Sites are not under Apple’s control, and Apple is not responsible for and does not endorse the content of such Linked Sites, including any information or materials contained on such Linked Sites. You will need to make your own independent judgment regarding your interaction with these Linked Sites.

Disclaimers
APPLE DOES NOT PROMISE THAT THE SITE OR ANY CONTENT, SERVICE OR FEATURE OF THE SITE WILL BE ERROR-FREE OR UNINTERRUPTED, OR THAT ANY DEFECTS WILL BE CORRECTED, OR THAT YOUR USE OF THE SITE WILL PROVIDE SPECIFIC RESULTS. THE SITE AND ITS CONTENT ARE DELIVERED ON AN “AS-IS” AND “AS-AVAILABLE” BASIS. ALL INFORMATION PROVIDED ON THE SITE IS SUBJECT TO CHANGE WITHOUT NOTICE. APPLE CANNOT ENSURE THAT ANY FILES OR OTHER DATA YOU DOWNLOAD FROM THE SITE WILL BE FREE OF VIRUSES OR CONTAMINATION OR DESTRUCTIVE FEATURES. APPLE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING ANY WARRANTIES OF ACCURACY, NON-INFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. APPLE DISCLAIMS ANY AND ALL LIABILITY FOR THE ACTS, OMISSIONS AND CONDUCT OF ANY THIRD PARTIES IN CONNECTION WITH OR RELATED TO YOUR USE OF THE SITE AND/OR ANY APPLE SERVICES. YOU ASSUME TOTAL RESPONSIBILITY FOR YOUR USE OF THE SITE AND ANY LINKED SITES. YOUR SOLE REMEDY AGAINST APPLE FOR DISSATISFACTION WITH THE SITE OR ANY CONTENT IS TO STOP USING THE SITE OR ANY SUCH CONTENT. THIS LIMITATION OF RELIEF IS A PART OF THE BARGAIN BETWEEN THE PARTIES.

The above disclaimer applies to any damages, liability or injuries caused by any failure of performance, error, omission, interruption, deletion, defect, delay in operation or transmission, computer virus, communication line failure, theft or destruction of or unauthorized access to, alteration of, or use, whether for breach of contract, tort, negligence or any other cause of action.

Apple reserves the right to do any of the following, at any time, without notice: (1) to modify, suspend or terminate operation of or access to the Site, or any portion
of the Site, for any reason; (2) to modify or change the Site, or any portion of the Site, and any applicable policies or terms; and (3) to interrupt the operation of the Site, or any portion of the Site, as necessary to perform routine or non-routine maintenance, error correction, or other changes.

Limitation of Liability
Except where prohibited by law, in no event will Apple be liable to you for any indirect, consequential, exemplary, incidental or punitive damages, including lost profits, even if Apple has been advised of the possibility of such damages.

If, notwithstanding the other provisions of these Terms of Use, Apple is found to be liable to you for any damage or loss which arises out of or is in any way connected with your use of the Site or any Content, Apple's liability shall in no event exceed the greater of (1) the total of any subscription or similar fees with respect to any service or feature of or on the Site paid in the six months prior to the date of the initial claim made against Apple (but not including the purchase price for any Apple hardware or software products or any AppleCare or similar support program), or (2) US$100.00. Some jurisdictions do not allow limitations of liability, so the foregoing limitation may not apply to you.

Indemnity
You agree to indemnify and hold Apple, its officers, directors, shareholders, predecessors, successors in interest, employees, agents, subsidiaries and affiliates, harmless from any demands, loss, liability, claims or expenses (including attorneys’ fees), made against Apple by any third party due to or arising out of or in connection with your use of the Site.

Violation of These Terms of Use
Apple may disclose any information we have about you (including your identity) if we determine that such disclosure is necessary in connection with any investigation or complaint regarding your use of the Site, or to identify, contact or bring legal action against someone who may be causing injury to or interference with (either intentionally or unintentionally) Apple’s rights or property, or the rights or property of visitors to or users of the Site, including Apple’s customers. Apple reserves the right at all times to disclose any information that Apple deems necessary to comply with any applicable law, regulation, legal process or governmental request. Apple also may disclose your information when Apple determines that applicable law requires or permits such disclosure, including exchanging information with other companies and organizations for fraud protection purposes.

You acknowledge and agree that Apple may preserve any transmittal or communication by you with Apple through the Site or any service offered on or through the Site, and may also disclose such data if required to do so by law or Apple determines that such preservation or disclosure is reasonably necessary to (1) comply with legal process, (2) enforce these Terms of Use, (3) respond to
claims that any such data violates the rights of others, or (4) protect the rights, property or personal safety of Apple, its employees, users of or visitors to the Site, and the public.

You agree that Apple may, in its sole discretion and without prior notice, terminate your access to the Site and/or block your future access to the Site if we determine that you have violated these Terms of Use or other agreements or guidelines which may be associated with your use of the Site. You also agree that any violation by you of these Terms of Use will constitute an unlawful and unfair business practice, and will cause irreparable harm to Apple, for which monetary damages would be inadequate, and you consent to Apple obtaining any injunctive or equitable relief that Apple deems necessary or appropriate in such circumstances. These remedies are in addition to any other remedies Apple may have at law or in equity.

You agree that Apple may, in its sole discretion and without prior notice, terminate your access to the Site, for cause, which includes (but is not limited to) (1) requests by law enforcement or other government agencies, (2) a request by you (self-initiated account deletions), (3) discontinuance or material modification of the Site or any service offered on or through the Site, or (4) unexpected technical issues or problems.

If Apple does take any legal action against you as a result of your violation of these Terms of Use, Apple will be entitled to recover from you, and you agree to pay, all reasonable attorneys’ fees and costs of such action, in addition to any other relief granted to Apple. You agree that Apple will not be liable to you or to any third party for termination of your access to the Site as a result of any violation of these Terms of Use.

Governing Law; Dispute Resolution
You agree that all matters relating to your access to or use of the Site, including all disputes, will be governed by the laws of the United States and by the laws of the State of California without regard to its conflicts of laws provisions. You agree to the personal jurisdiction by and venue in the state and federal courts in Santa Clara County, California, and waive any objection to such jurisdiction or venue. The preceding provision regarding venue does not apply if you are a consumer based in the European Union. If you are a consumer based in the European Union, you may make a claim in the courts of the country where you reside. Any claim under these Terms of Use must be brought within one (1) year after the cause of action arises, or such claim or cause of action is barred. Claims made under the separate terms and conditions of purchase for goods and services are not subject to this limitation. No recovery may be sought or received for damages other than out-of-pocket expenses, except that the prevailing party will be entitled to costs and attorneys’ fees. In the event of any controversy or dispute between Apple and you arising out of or in connection with your use of the Site, the parties shall attempt, promptly and in good faith, to resolve any such dispute. If we are
unable to resolve any such dispute within a reasonable time (not to exceed thirty (30) days), then either party may submit such controversy or dispute to mediation. If the dispute cannot be resolved through mediation, then the parties shall be free to pursue any right or remedy available to them under applicable law.

Void Where Prohibited
Apple administers and operates the www.apple.com Site from its location in Cupertino, California USA; other Apple sites may be administered and operated from various locations outside the United States. Although the Site is accessible worldwide, not all features, products or services discussed, referenced, provided or offered through or on the Site are available to all persons or in all geographic locations, or appropriate or available for use outside the United States. Apple reserves the right to limit, in its sole discretion, the provision and quantity of any feature, product or service to any person or geographic area. Any offer for any feature, product or service made on the Site is void where prohibited. If you choose to access the Site from outside the United States, you do so on your own initiative and you are solely responsible for complying with applicable local laws.

Miscellaneous
You may not use or export or re-export any Content or any copy or adaptation of such Content, or any product or service offered on the Site, in violation of any applicable laws or regulations, including without limitation United States export laws and regulations.

If any of the provisions of these Terms of Use are held by a court or other tribunal of competent jurisdiction to be void or unenforceable, such provisions shall be limited or eliminated to the minimum extent necessary and replaced with a valid provision that best embodies the intent of these Terms of Use, so that these Terms of Use shall remain in full force and effect. These Terms of Use constitute the entire agreement between you and Apple with regard to your use of the Site, and any and all other written or oral agreements or understandings previously existing between you and Apple with respect to such use are hereby superseded and cancelled. Other than as provided in a purchase agreement you enter into with Apple, Apple will not accept any counter-offers to these Terms of Use, and all such offers are hereby categorically rejected. Apple’s failure to insist on or enforce strict performance of these Terms of Use shall not be construed as a waiver by Apple of any provision or any right it has to enforce these Terms of Use, nor shall any course of conduct between Apple and you or any other party be deemed to modify any provision of these Terms of Use. These Terms of Use shall not be interpreted or construed to confer any rights or remedies on any third parties.

Apple provides access to Apple international data and, therefore, may contain references or cross references to Apple products, programs and services that are
not announced in your country. Such reference does not imply that Apple in your
country intends to announce such products, programs or services.

Feedback and Information
Any feedback you provide at this site shall be deemed to be non-confidential.
Apple shall be free to use such information on an unrestricted basis.

The information contained in this web site is subject to change without notice.

Copyright © 1997-2009 Apple Inc. All rights reserved.

Apple Inc., 1 Infinite Loop, Cupertino, CA 95014, USA.

Updated by The Apple Legal Team on Nov. 20, 2009

Figure M17 & M18-
Photo License
It’s hard to understand complex licenses that is why all photos on Pexels are
licensed under the Creative Commons Zero (CC0) license. This means the
pictures are completely free to be used for any legal purpose.

• The pictures are free for personal and even for commercial use.
• You can modify, copy and distribute the photos.
• All without asking for permission or setting a link to the source. So
  that attribution is not required.

The CC0 license was released by the non-profit organization Creative Commons
(CC). Get more information about the license on the official license page.

Restriction
Identifiable people may not appear in a bad light or in a way that they may find
offensive, unless they give their consent.

About Us
Currently we cover over 10,000 free stock photos. And every month at least
1500 new high resolution photos will be added. All photos are hand-picked from
photos uploaded by our users or sourced from free image websites. We make
sure all published pictures are high-quality and licensed under CC0. Only
creative commons images from reliable photographers and sources like
Unsplash, Gratisography, Little Visuals and many more are added to our photo
database.

To support us you can donate or submit your own photos to Pexels and share
them with the community: Start uploading
Public Citation of Data

Companies using comScore data for promotional purposes must have purchased said data from comScore ("clients") unless said data have already been placed into the public domain by an authorized party. In this context, public domain includes comScore websites, comScore press releases and other public communications, and/or articles in the media. Non-clients who use comScore data are required to comply with the relevant usage guidelines as if they were clients.

Clients must notify and seek approval from their Client Service contact(s), who will then work with the comScore marketing team to review the request before providing clients with approval to publish any comScore data.

comScore reserves the right to prohibit Clients from publishing data about specific entities or categories pertaining to objectionable content.

If a Client publishes incorrect data and does not comply with comScore’s request to correct the data in a timely fashion, comScore reserves the right to provide the correct data in the public domain, or to provide the correct data to parties that inquire about it.

comScore employees may not be quoted making promotional comments in client press releases. comScore must maintain its standing as an objective, third party measurement standard supplier and cannot appear to endorse one party over another. There may be specific exceptions in which inclusion does not jeopardize comScore’s third-party standing (e.g. industry associations such as the OPA, IAB or NAA that commission comScore research for the benefit of an entire industry or sector).

Regarding comScore Marketing Solutions data, the following additional requirements apply: The use of such data must be reviewed and approved by the appropriate CMS team, and must specifically be attributed to the industry team that produced it.

Style & Formatting Guidelines

If a client refers to a specific, standard comScore report, whether custom or syndicated, the report must be appropriately cited using the proper name of the report.

The comScore product/service, data periods, and geography must be properly sourced (e.g. Source: comScore Media Metrix, June 2013, U.S.). Appropriate
reference to the specific reporting metrics being used (e.g. “unique visitors”) should also be cited.

comScore syndicated data should be referenced as provided by a digital audience measurement or ratings service, not a “survey” or “study”.

Clients who quote a comScore data set alongside another data source or a different comScore data set (such as standard and custom categories) must clearly distinguish the different data sets. comScore data may not be used in direct comparison to another audience measurement or web analytics platform.

Clients who wish to include a description of comScore in press releases may do so. The most recent comScore company information may be found in press section of our website at: www.comscore.com/companyinfo.

comScore corporate and business unit names are trademarks in various markets and must be referenced accordingly. This includes the proper use of capitalization, even at the beginning of a sentence or paragraph, and use of the trademark symbol “®” for comScore’s registered trademarks (e.g., "Media Metrix", "Digital Analytix", "MMX", "vCE", "UDM", "Unified Digital Measurement", and the "comScore" logo). Important: “Media Metrix®” should not be used in any content intended for a European audience; “MMX®” must be used instead. For example:

comScore, Inc.

comScore Media Metrix® (comScore MMX® in Europe)

comScore Digital Analytix®

comScore Mobile Metrix

comScore AdEffx

**Reporting Entity Guidelines**

Specific and accurate entity descriptions in the citation must be consistent with the labels used in comScore syndicated reports – e.g. “Match.com” vs. “Match.com Sites”, whichever appears in the report. If the client wishes to use an abbreviated version of the entity for the sake of marketing style/brevity, it may do
so as long as there is an accompanying footnote indicating the specific entity being used for the claim.

Any custom (whether syndicated or proprietary) reports from which data are quoted must be appropriately referenced as such through footnotes. Example: *Custom research conducted for (Client Name) by comScore (Month, Year).

Custom Entities (denoted with an [E] in comScore’s syndicated reports) are entities featuring an aggregation of URLs or an alternate view of a property, that may be determined either by comScore or by the client. While clients may cite their own custom entities in marketing materials, it is recommended that the specific URLs falling into this aggregation be sourced when it is reasonable to do so.

If a client does not wish to source all URLs that compose a Custom Entity being cited, then comScore reserves the right to share the definition with inquiring clients and news media. Data transparency is essential for publicly cited information, and interested parties should be able to vet the validity behind such public claims.

Because Custom Entities may be client-defined and do not necessarily meet the same reporting standards required for the Ranked Category view, they may be used only if they are pure domains as opposed to a collection of properties. Otherwise, Custom Entities should not be used for competitive comparisons unless comScore approves such use on the basis of the entity being an appropriate and best available comparison.

Clients may not cite data from an Alternate Roll-Up in public communications materials because these entities are, by definition, private to the client and cannot be viewed by other users of MyMetrix.

Clients may publicly cite information from their Trust Profiles in MyMetrix, such as their viewability and non-human traffic categorizations. Clients may publicly report on their own entities, but may not provide competitive Trust Profile information.

If comScore Digital Analytix clients wish to source their analytics data, they must include “Digital Analytix” sourcing and they must refer to the reporting metrics in accordance with their nomenclature in that product. More specifically, clients should not refer to “unique visitor” numbers for Digital Analytix data, as comScore uses the term “unique browser” in order to differentiate from “unique visitor,” which is commonly used for audience measurement purposes. Since the two metrics are measured and calculated differently, it is important to avoid conflating the terms.
Ranking Guidelines

Citations of rank must mention the ranking in some context, such as a category, according to comScore’s Ranked Category view. If the client refers to a category other than as defined by comScore, the ranking must have an accompanying footnote indicating that the category is a custom-defined category by the client, and must indicate the entities included in that view.

Any marketing or promotional language citing a client’s custom ranking must, in addition to indicating that it is according to their own custom ranking, not be directly attributed to comScore, though it should mention that the custom ranking was created using comScore data. (e.g. “PQR is the leading entity in XYZ category”; footnote: *According to PQR’s custom-defined ranking based on comScore data.)

It is generally advised that any custom client rankings based on comScore data use ranked entities from the comScore Ranked Category view. There may be select instances in which clients wish to make competitive claims based on the best available apples-to-apples definitions at the brand level, content level or other appropriate view of the market. Such instances will be allowed provided that comScore vets the comparability of these entities and the context in which it’s being used.

Unified Digital Measurement (UDM) reporting is the official public source of record for individual site data and category rankings in all markets.

Rankings may not be based on Unified Digital Measurement (UDM) or non-UDM sub-sets of a category (e.g. “XYZ is the #1 ranked UDM site in the Sports category” when it is ranked #5 overall in the category).

Clients may publicly refer to their syndicated Multi-Platform audience estimates and rankings. Data may be reported either at the aggregate level or for individual media channels (i.e. Total Digital Population or Desktop, Mobile), with appropriate sourcing of the applicable comScore product.

Data Trending Guidelines

Methodological enhancements are an ongoing reality in audience measurement that may have an impact on trended information. In most cases, these changes have minimal impact, but in some cases they may be more significant. Since comScore does not re-state previously released data, any published data from a prior reporting period should be considered the data of record.

Clients may refer to trends that include some element of change that is inorganic; however, they are cautioned to be judicious with such claims when the impact may be significant. If asked by news media as to the validity of the claims,
comScore will be transparent about factors that may have caused an inorganic increase or decrease.

Clients should avoid making claims of inorganic growth (whether referring to rates of growth or moves within a ranking) that are primarily attributable to the adoption of the UDM methodology, wherever possible. However, in some cases clients will only be able to compare UDM to non-UDM data when referring to growth trends. Such instances are permitted, provided that the growth trend is accompanied by a footnote indicating the following: “[Entity] transitioned to comScore’s UDM methodology in [date]. Data trend is based on a comparison of UDM to [non/partial] UDM data and may include inorganic growth due to the change in methodology.”

Clients are cautioned to be careful when reporting trends going back to any applicable Beta period as the comparisons may not be methodologically consistent. Accordingly, such references should include sourcing indicating the referenced time periods that are from Beta reporting.

Figure M4-

plotly
Terms of Service

Last Revised: July 17, 2014

Welcome to Plotly! Our product goal is for anyone to be able to make and share a beautiful graph for free. Crafting graphs and communicating data is empowering. We believe it’s also good for the world when data is free and available.

When you make something on Plotly (graphs, a dataset, etc.), you retain the rights to your content (see “User Content” section). You also control whether your graphs are public or private. Public plotting is free; for a lot of private use, you can get a Premium or Organizational plan (see http://plot.ly/plans). Happy plotting!

The following Terms of Service (“Terms”) govern your access to and use of the related websites, networks, and other services provided by Plotly (collectively, the “Service”). The Service is operated by Plotly (“us”, “we” or “Plotly”).

PLEASE REVIEW THIS ENTIRE AGREEMENT CAREFULLY. THE SERVICE IS OFFERED TO YOU SUBJECT TO THESE TERMS. BY CLICKING “ACCEPT” OR BY ACCESSING THE SERVICE IN ANY MANNER, YOU SIGNIFY THAT YOU HAVE READ, UNDERSTAND AND AGREE TO BE BOUND BY THESE TERMS, INCLUDING ALL PLOTLY POLICIES THAT HAVE BEEN MADE AVAILABLE TO YOU AND GOVERN YOUR USE OF THE SERVICE. IF YOU DO NOT AGREE TO ABIDE BY THESE TERMS, YOUR SOLE REMEDY IS TO NOT USE OR ACCESS THE SERVICE
Please note that if you are accessing any Plotly service in your capacity as a government entity, this Amendment to Plotly Terms of Service shall apply to you.

**Eligibility**

In order to use the Service, you must be the age of majority in your jurisdiction and fully able and competent to enter into these Terms. You represent and warrant that you meet these eligibility requirements. If you are using or opening an account on the Service on behalf of a company, entity, or organization, then you represent and warrant that you are an authorized representative of that entity with the authority to bind such entity to these Terms.

**Account Registration**

To access and use the Service, you must register for an account. When registering, you must fill in all mandatory fields with accurate, current and complete information about yourself as prompted in the registration form and keep this information up to date. Plotly has the right to suspend or terminate your account and refuse any and all use of the Service if it suspects that your account information is inaccurate, not current or incomplete.

**Privacy**

We may collect registration and other information about you through the Service. Our collection, use, and disclosure of this information is described in the Plotly Privacy Policy available at plot.ly/privacy.

**Account Security**

You are responsible for maintaining the confidentiality of the password and username you provided during the registration process, and you are fully responsible for all activities that occur under your password or account. You agree to immediately notify Plotly of any unauthorized use of your password or username or any other breach of security.

**Paid Services**

Registration for a Plotly account and access to selected features of the Service are currently provided to you free of charge. Plotly may charge fees for access to certain portions of the Service ("Paid Services"), and will notify you of those fees at the time that we offer you a Paid Service. You may not share access to a Paid Service with any other individual. By electing to access a Paid Service, you authorize Plotly to direct its third party payment processor to charge the credit card identified by you (which you represent and warrant that you are authorized to use) all applicable fees for that Paid Service in U.S. dollars, including all applicable taxes. If Plotly does not receive payment from your credit card provider, you agree to pay all amounts due upon demand and Plotly may suspend your access to the Paid Services until you have paid all outstanding amounts. You agree to reimburse any costs or expenses (including, but not limited to, reasonable attorneys’ fees) incurred by Plotly to collect any amounts that are not paid when due. You may cancel your use of a Paid Service or your account at any time. All sales are final and Plotly will not refund any amounts paid, except as expressly provided in these Terms. You acknowledge that Plotly reserves the right to change its fees at any time upon prior notice to you. At no time will you be charged without your prior knowledge and consent.
Account Termination
Plotly may terminate your account and your access to the Service (including, any Paid Service) at any time, for any reason, without warning. You may terminate your account and your access to the Service at any time, for any reason, by contacting us by email at accounts@plot.ly. Requests for account termination may also be sent to us at the mailing address included at the end of these Terms. Even after your account is terminated, these Terms will remain in effect. In the event that you or Plotly terminate your access to a Paid Service, Plotly will not refund any amounts paid for cancelled Paid Services, unless Plotly terminates your access to a Paid Service solely for its convenience at which point Plotly will refund any amounts paid by you for use of the Paid Service after the effective termination date.

Storage; Inactive Accounts
You acknowledge that Plotly may establish general practices and limits concerning use of the Service, including without limitation limits on the periods for which data or other uploaded content will be retained by Plotly. You agree that Plotly has no responsibility or liability for the deletion or failure to store any data or other content maintained by or transmitted to Plotly. You acknowledge that Plotly reserves the right to deactivate your account if it is inactive for longer than one (1) year.

Ownership; Proprietary Rights
The Service is owned by Plotly. All text, images, designs, computer code, software, graphics, user interface, and all other elements of the Service (collectively, "Content"), and all intellectual property and proprietary rights therein, are owned by Plotly or its third-party licensors to the full extent permitted by United States and international law. All rights in the product names, company names, trade names, logos, and designs of all Plotly or third-party products or services, belong exclusively to Plotly or their respective owners. The use of these trademarks or any Content, is expressly prohibited, and nothing stated or implied on the Service confers on you any license or right other than those expressly granted in these Terms. Plotly may update the Service from time to time, in its sole discretion.

User Content; License to User Content
You retain all rights to your User Content. User content means data, graphs, code, commentary, and derivative works you import, create, share, modify, and distribute through Plotly's platform or products.
By setting your work to be viewed publicly, you agree to allow others to view, copy, and re-use your work and the underlying data. For any User Content that you upload to or generate through the Service and publicly share, you grant us a worldwide, perpetual, royalty-free, transferable, right to use, reproduce, modify, adapt, publish, prepare derivative works of, distribute, and publicly display that User Content. You may modify or remove your User Content via your Plotly account or by terminating your Plotly account.
You agree not to upload to the Service or otherwise post, transmit, make available, distribute, or disseminate through the Service any material that: (a) breaches any duty toward or rights of any person or entity, including rights of
publicity, privacy rights, or similar rights; (b) contains corrupted data or any other harmful, disruptive, or destructive files or computer code; (c) is offensive or otherwise objectionable; (d) restricts or inhibits any person or entity from using or enjoying any portion of the Service; (e) may expose Plotly, its affiliates, or other users to harm or liability of any nature; (f) constitutes unsolicited or unauthorized advertising, promotional materials, spam, or any other form of solicitation; or (g) contains the private or confidential information of another person (including personal health information) or entity without that person’s or entity’s authorization, or (h) violates any law or regulatory requirement.

Although Plotly has no obligation to screen, edit, or monitor any User Content, Plotly reserves the right, and has absolute discretion, to screen, edit, or remove any User Content at any time and for any reason without notice. You further understand and agree that Plotly may use or disclose any information related to you (including User Content) in order to investigate, prevent, or take action regarding activities that may be prohibited or unlawful, to exercise our legal rights under the Terms, or to otherwise protect the rights and property of Plotly or any third party.

WE CAN TAKE NO RESPONSIBILITY AND ASSUME NO LIABILITY FOR ANY USER CONTENT (INCLUDING, WITHOUT LIMITATION, ANY LOSS OR DAMAGE TO YOUR USER CONTENT, YOUR INABILITY TO ACCESS YOUR USER CONTENT, AND ANY THIRD PARTY’S UNAUTHORIZED ACCESS TO YOUR USER CONTENT).

Use of the Service; Applicable Laws
You are required to comply with all applicable laws in connection with your access to and use of the Service. As a condition of your access to and use of the Service, you warrant that you will not use the Service for any purpose that is unlawful or prohibited by these Terms. Use of the Service (including, without limitation, any graphs, charts or other output generated by the Service) is for the use described in these Terms and Plotly does not grant you any express or implied rights to access or use the Service for any other purpose.

Health Data
You shall comply with the obligations under the Health Insurance Portability and Accountability Act (“HIPAA”) in use of the Service. Plotly does not have, and will not accept, any obligations of confidentiality with respect to any communications other than those expressly stated in these Terms or the Privacy Policy.

You agree to fully de-identified all health data pursuant to the HIPAA privacy rule standards for de-identification set forth in 45 CFR 164.514(b), prior to being uploaded to Plotly. You agree to remove any information that would enable Plotly or a third party to associate any health data with a specific individual or group of individual.

You declare that you have the right to communicate or broadcast all health data and that you have obtained all necessary authorizations and, where legally required, prior specific, informed and freely given consents from the individuals from which the data is obtained or derived prior to submitting the data to Plotly.

Prohibited Uses Generally
Without limiting any other section of these Terms, you agree not to:
• reverse engineer or otherwise attempt to derive source code or other trade secrets from the Service;
• access the Service to build a competitive product or service or monitor its availability, performance or functionality, or for any other benchmarking or competitive purposes;
• sell, resell, distribute, lease or rent your access to the Service;
• remove, circumvent, disable, damage or otherwise interfere with any security-related features or measures employed to prevent or enforce limitations on the use of the Service or any Content;
• copy, "frame," or "mirror," any part of the Service;
• use the Service in any other manner that could damage, disable, overburden or impair the Service;
• use automated scripts to collect information from or otherwise interact with the Service;
• impersonate any person or entity, or falsely state or otherwise misrepresent yourself, your age or your affiliation with any person or entity;
• solicit personal information from anyone under 18 or solicit passwords or personally identifying information for commercial or unlawful purposes;
• intimidate or harass another; or
• use or attempt to use another's account, service or system without authorization from Plotly.

Violations of system or network security may result in civil or criminal liability. In accordance with these Terms, Plotly will investigate and work with law enforcement authorities to prosecute users who are involved in such violations.

Third-Party Services
To the extent that the Service interacts with third party services, if a third party ceases providing a relevant service, it may affect Plotly's ability to provide part or all of the Service. Under such circumstances, Plotly may cease providing the Service without any refund to you. In addition, the Service may contain links to websites of third parties as a service to those interested in this information. We do not monitor or endorse any such websites and can make no guarantee as to their accuracy or completeness. You should review the applicable terms and policies, including privacy and data gathering practices, of any third party websites accessible through the Service. In addition, the Service may contain advertisements from third parties. Any terms, conditions, warranties, or representations associated with such advertisements or any products or services provided by third party advertisers to you are solely between you and such third party.

Member Disputes
You are solely responsible for your interactions with other members of the Plotly community. Plotly reserves the right, but has no obligation, to monitor disputes between you and other members.

Notice of Copyright Infringement
Plotly respects the intellectual property rights of others, and asks you to do the same. It is Plotly’s policy to terminate the access privileges of those who repeatedly infringe the copyright rights of others. If you believe that your work has been posted on the Service in a way that constitutes copyright infringement, please contact Plotly’s copyright agent at the address below and provide the following information: (a) an electronic or physical signature of the person authorized to act on behalf of the owner of an exclusive right that is allegedly infringed; (b) a description of the copyright-protected work that you claim has been infringed; (c) the location on the Service of the material that you claim is infringing; (d) your address, telephone number, and email address; (e) a statement by you regarding your good faith belief that the disputed use is not authorized by the copyright owner, its agent, or the law; and (f) a statement by you, made under penalty of perjury, that the information in your notice is accurate and that you are the copyright owner or authorized to act on the copyright owner’s behalf. By submitting a copyright infringement notice, you acknowledge and agree that Plotly or its copyright agent may forward the information you provide in this notice to the person who uploaded the allegedly infringing material.

You may write to Plotly’s designated agent for notice of copyright infringement at:

Plotly, Inc
5555 avenue de Gaspé, Suite 201
Montréal, Québec H2T 2A3
Canada

By submitting a copyright infringement notice, you acknowledge and agree that Plotly or its copyright agent may forward the information you provide in this notice to the person who uploaded the allegedly infringing material. Plotly will remove any content that allegedly infringes upon the copyright of any person under the laws of the United States upon receipt of such a statement (or, more specifically, any statement in conformance with 17 U.S.C. § 512(c)(3)). United States law provides significant penalties for submitting such a statement falsely.

If you believe that your removed or disabled User Content is not infringing, or that you have the authorization or right to post and use that User Content from the copyright owner, the copyright owner’s agent, or pursuant to law, you may send a counter-notice containing the information required by Section 512(g)(3) of the Digital Millennium Copyright Act (17 USC § 512(g)(3)). Plotly or its copyright agent will forward your counter-notification to the party who submitted the original copyright infringement claim. If the original claimant does not file an action seeking a court order to restrain you from engaging in infringing activity related to the removed or disabled User Content within ten (10) business days of receiving the counter-notice from Plotly, then Plotly may, in its sole discretion, reinstate the removed or disabled material. Please note that under Section 512(f) of the Digital Millennium Copyright Act, any person who knowingly materially misrepresents that material or activity was removed or disabled by mistake or misidentification may be subject to liability.

Disclaimer of Warranties
THE SERVICE IS PROVIDED "AS IS." PLOTLY MAKES NO REPRESENTATION OR WARRANTY OF ANY KIND WHATSOEVER RELATING TO THE SERVICE, LINKED SITES, OR OTHER CONTENT THAT MAY BE ACCESSIBLE DIRECTLY OR INDIRECTLY THROUGH THE SERVICE. PLOTLY EXPRESSLY DISCLAIMS TO THE MAXIMUM EXTENT PERMITTED BY LAW ANY AND ALL WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND WARRANTIES AGAINST INFRINGEMENT OF ANY THIRD-PARTY INTELLECTUAL PROPERTY OR PROPRIETARY RIGHTS. FURTHER, THERE IS NO WARRANTY THAT THE SERVICE WILL MEET YOUR NEEDS OR REQUIREMENTS OR THE NEEDS OR REQUIREMENTS OF ANY OTHER PERSON.

WITHOUT LIMITING THE GENERALITY OF THE FOREGOING, PLOTLY MAKES NO WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED: (A) THAT THE INFORMATION, GRAPHS, CHARTS, DATA AND OTHER CONTENT PROVIDED THROUGH THE SERVICE WILL BE ACCURATE, RELIABLE, AND COMPLETE AND FREE FROM ERRORS, OMISSIONS, DEFECTS, OR DELAYS IN OPERATION; (B) THAT THE SERVICE WILL BE AVAILABLE AT ANY PARTICULAR TIME OR LOCATION; (C) THAT DEFECTS OR ERRORS IN THE SERVICE WILL BE CORRECTED; OR (D) THAT THE CONTENT ON THE SERVICE IS FREE OF VIRUSES OR OTHER HARMFUL COMPONENTS.

Limitation of Liability
IN NO EVENT WILL PLOTLY OR ITS AFFILIATES, OR ANY PARTY INVOLVED IN CREATING, PRODUCING, OR DELIVERING THE SERVICE BE LIABLE FOR ANY DIRECT, INCIDENTAL, CONSEQUENTIAL, INDIRECT, SPECIAL, OR PUNITIVE DAMAGES ARISING OUT OF YOUR ACCESS, USE, MISUSE, OR INABILITY TO USE THE CONTENT, SERVICE OR ANY LINKED SITES, OR IN CONNECTION WITH ANY LOST OR CORRUPTED DATA OR USER CONTENT OR UNAUTHORIZED THIRD PARTY ACCESS TO ANY DATA OR USER CONTENT, FAILURE OF PERFORMANCE, DEFECT, DELAY IN OPERATION OR TRANSMISSION, COMPUTER VIRUS, OR LINE OR SYSTEM FAILURE. IN THE EVENT THAT YOU HAVE A DISPUTE WITH ANOTHER USER IN ANY WAY CONNECTED WITH YOUR USE OF THE SERVICE, YOU RELEASE PLOTLY FROM ANY CLAIMS, DEMANDS, AND DAMAGES OF EVERY KIND AND NATURE ARISING OUT OF OR IN ANY WAY CONNECTED WITH SUCH A DISPUTE. THE MAXIMUM TOTAL LIABILITY OF PLOTLY, ITS AFFILIATES AND LICENSORS TO YOU FOR ANY CLAIM UNDER THIS AGREEMENT, IS THE LESSER OF: (A) THE AMOUNT RECEIVED BY PLOTLY FROM YOU DURING THE TWELVE (12) MONTHS PRECEDING THE CLAIM OR (B) $100.

EACH PROVISION OF THIS AGREEMENT THAT PROVIDES FOR A LIMITATION OF LIABILITY, DISCLAIMER OF WARRANTIES, OR EXCLUSION OF DAMAGES IS TO ALLOCATE THE RISKS UNDER THIS AGREEMENT BETWEEN THE PARTIES. THIS ALLOCATION IS AN ESSENTIAL ELEMENT OF THE BASIS OF THE BARGAIN BETWEEN YOU AND PLOTLY. EACH OF
THESE PROVISIONS IS SEVERABLE AND INDEPENDENT OF ALL OTHER PROVISIONS OF THIS AGREEMENT.

THESE LIMITATIONS APPLY WHETHER THE ALLEGED LIABILITY IS BASED ON CONTRACT, TORT, NEGLIGENCE, STRICT LIABILITY, OR ANY OTHER BASIS, EVEN IF PLOTLY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE, AND EVEN IF ANY LIMITED REMEDY FAILS OF ITS ESSENTIAL PURPOSE. BECAUSE SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF DAMAGES, PLOTLY’S LIABILITY IN SUCH JURISDICTIONS WILL BE LIMITED TO THE EXTENT PERMITTED BY LAW.

Indemnification
You will indemnify, defend and hold Plotly and its licensors harmless (and our respective employees, directors, agents, affiliates and representatives) from and against any and all claims, costs, losses, damages, judgments, tax assessments, penalties, interest, and expenses (including without limitation reasonable attorneys’ fees) arising out of any claim, action, audit, investigation, inquiry, or other proceeding instituted by a person or entity that arises out of or relates to: (i) your use of the Service or any activity on the Service via your account, (ii) your violation of any provision of these Terms including any of your covenants, warranties or representations hereunder, (iii) any User Content uploaded to Plotly, or (iv) the collection, storage, or disclosure of personal or health data or disclosure of personal information or health data in violation of applicable law, or any failure to obtain consents or authorizations for such data required by applicable law. Plotly reserves the right, at your expense, to assume the exclusive defense and control of any matter for which you are required to indemnify us and you agree to cooperate with our defense of these claims. You may not settle any matter without the prior written consent from Plotly.

Choice of Law and Forum
These Terms are governed by the laws of the State of California. The provisions of the United Nations Convention on the International Sale of Goods and the Uniform Computer Information Transactions Act, however designated, are excluded and will not apply to these Terms or any transactions under these Terms. If there is any dispute between us concerning the Terms or your access to or use of the Service, we both agree to submit to the personal and exclusive jurisdiction of the state courts and federal courts located in Santa Clara, CA for the purpose of litigating all such claims or disputes.

Feedback
If you provide feedback to Plotly regarding the Service or Content (“Feedback”), you authorize Plotly to use that Feedback without restriction and without payment to you. Accordingly, you hereby grant to us a fully paid up, perpetual, irrevocable, worldwide, fully transferable and sublicensable license to use the Feedback in any manner and for any purpose.

No Waiver
The failure of Plotly to enforce any provisions of these Terms or respond to a breach by you or other parties will not in any way waive Plotly’s right to enforce
subsequently any terms or conditions of these Terms or to act with respect to similar breaches.

**Notice**
Plotly may provide you with notices by electronic mail, regular mail or postings on the Service. If notice is provided by electronic mail, notice will be deemed given twenty-four hours after electronic mail is sent, unless Plotly is notified that the electronic mail address is invalid. If notice is provided by posting through the Service, then notice will be deemed given upon such posting. Alternatively, we may give you legal notice by mail to a postal address, if provided by you through the Service. In such case, notice will be deemed given three days after the date of mailing.

**Miscellaneous**
You agree that no joint venture, partnership, employment, or agency relationship exists between you and Plotly as a result of these Terms or your access to and use of the Service. A printed version of these Terms and of any notice given in electronic form will be admissible in judicial or administrative proceedings based upon or relating to these Terms to the same extent and subject to the same conditions as other business documents and records originally generated and maintained in printed form. This Terms, and any rights and licenses granted hereunder, may not be transferred or assigned by you, but may be assigned by Plotly without restriction, including by operation of law, merger, or sale of all or substantially all of the assets to which these Terms pertains. Any assignment attempted to be made in violation of these Terms shall be void. Unless otherwise specified herein, these Terms constitutes the entire agreement between you and Plotly and governs your use of the Service. If any portion of these Terms is held invalid or unenforceable, that portion will be construed in a manner consistent with applicable law to reflect, as nearly as possible, the original intention of the parties, and the remaining portions will remain in full force and effect.

**Figure M3**

**Disclaimers/Disclosures**

**Disclaimers, Terms and Conditions of Use**

The information on this website is provided solely for general information purposes and does not create a business or professional services relationship. Transmission of such information to any other person without the written permission of ACSI is prohibited.

Nothing on this website should be construed as investment, tax, legal or other advice nor is it to be relied upon in making any investment decisions. ACSI
accepts no liability in respect of any losses or damages arising out of errors or omissions contained in this website.

Past performance with respect to the indices, securities or economic sectors presented on this website is not indicative of future results; no representation is being made regarding any such index, security or economic sector.

The materials contained on this website are provided “as is.” ACSI and its affiliates cannot and do not warrant the accuracy, completeness, currentness, timeliness, non-infringement, title, merchantability or fitness for a particular purpose of the materials, and they hereby disclaim any such express or implied warranties.

ACSI assumes no responsibility for, and makes no warranties that, functions contained at or on this website will be uninterrupted or error-free, that defects will be corrected, or that the website or the server that makes it available will be free of viruses or other harmful components. ACSI shall not be liable for any damages to, viruses that may infect, or services, repairs or corrections that may need to be performed on your computer on account of your accessing this website.

All works of authorship contained on this website, including, reports, articles, text, graphics, designs, and other materials (the “Content”) are, unless otherwise noted, owned by ACSI. ACSI and/or other such owners of the Content are protected by copyright and other applicable laws worldwide. You are permitted to review, store and/or print Content for your personal, non-commercial use. Otherwise, all other use of the Content, including any distribution, circulation, transmission, reproduction, publication, display or modification, in whole or in part, in any medium, is strictly prohibited. All rights to and in the Content are expressly reserved.

In no event shall ACSI, its affiliates, agents or licensors or their respective directors, officers, employees or agents have any liability to you or any other person or entity for direct, special, incidental, indirect or consequential damages (including, without limitation, lost profits or revenues or other economic loss of yours or any third party, loss of data, loss of use or claims of third parties), arising out of or in any manner in connection with use of or access to this website, including, without limitation, the content, the subject matter of this website or your or any other person's or entity's use of, or inability to use, the content or any other materials provided to you hereunder regardless of the form of action (including negligence or strict liability), whether or not ACSI has been advised of, or otherwise might have anticipated the possibility of, such damages.
This website contains links to other websites maintained by third parties. Such linked websites are not under the control of ACSI, and ACSI makes no warranties or representations of any kind with respect to such third-party websites and ACSI is not responsible for the content available on other websites or their terms of use. ACSI provides such third-party links solely for your convenience and does not endorse such third-parties or the content contained on their websites.

The substantive laws of the State of New York shall govern these Terms and Conditions of use as though these Terms and Conditions of Use were entered into, and were to be entirely performed within, the State of New York. The parties expressly disclaim the applicability of, and waive any rights based upon, the Uniform Computer Information Transactions Act and the Convention on the Use of Electronic Communications in International Contracts. All claims or disputes arising out of or in connection with these Terms and Conditions of Use shall be heard exclusively by any of the federal or state courts of competent jurisdiction located in New York, New York, USA.

Figure M2-

Terms of Service

PLEASE READ THESE TERMS OF SERVICE (“AGREEMENT” OR “TERMS OF SERVICE”) CAREFULLY BEFORE USING THE SERVICES OFFERED BY CHAR SOFTWARE, INC. D/B/A LOCALYTICS℠ (“LOCALYTICS”). THIS AGREEMENT SETS FORTH THE LEGALLY BINDING TERMS AND CONDITIONS FOR YOUR USE OF THE SITES AND THE SERVICE (EACH AS DEFINED BELOW). BY USING THE SUBSCRIPTION SERVICE, YOU AGREE TO BE BOUND BY THIS AGREEMENT.

1. DEFINITIONS

1.1 “Client Libraries” means the client libraries provided by Localytics and installed within Customer’s application(s) for the purpose of collecting Customer Data and sending such Customer Data to Localytics’ servers.

1.2 “Customer Data” means all information, data and other content provided by Customer in connection with its authorized use of the Service, including, without limitation, all data and information transmitted to the Service via the Client Libraries.

1.3 “Data Point” means a session start, session stop or Event Call.

1.4 “Documentation” means the implementation guides; help information and other user documentation regarding the Service that is provided by Localytics to Customer in electronic or other form.

1.5 “Event Call” means an individual recording of a single event.
1.6 “Localytics Property” means all ideas, concepts, inventions, systems, platforms, software, interfaces, tools, utilities, templates, forms, techniques, methods, processes, algorithms, know-how, trade secrets and other technologies and information acquired, created, developed or licensed by Localytics prior to or outside the scope of this Agreement and any improvement, modification, extension or other derivative works thereof and all intellectual property rights thereto including without limitation the Service, Client Libraries, Documentation, Report Templates, and Aggregate Data (as defined in Section 6.4). Localytics Property excludes Customer Data and Results.

1.7 “Monthly Active User” means each individual installation of a Customer application that transmits Customer Data during the billing period.

1.8 “Order Form” shall mean an order form referencing this Agreement that has been mutually agreed to and executed by the parties.

1.9 “Report Template” means the content, formatting, look and feel of the templates used for the reports, charts, graphs and other presentation in which the Results are presented to Customer.

1.10 “Results” means the work product resulting from the Service delivered to Customer by Localytics through the Service, to the extent based on the Customer Data. Results expressly exclude all Localytics Property.

1.11 “Service” means Localytics’ mobile application analytics and marketing services (including any Localytics Property used to provide such service), as hosted by or on behalf of Localytics and provided to Customer under this Agreement, as identified in an Order Form, as updated from time to time by Localytics in its sole discretion.

1.12 “Sites” means websites operated by Localytics that are made accessible to Customer under this Agreement.

2. SERVICE

2.1 Service Start Up. Each party will provide the other with reasonable cooperation, assistance, information and access as may be necessary to initiate Customer’s use of the Service. Localytics will provide implementation services to Customer to the extent set forth in the Order Form.

2.2 Service Access and Availability.

(a) Customer Systems. Customer is responsible for providing (i) all equipment, subscriptions and credentials necessary for Localytics to receive the Customer Data and (ii) all servers, devices, storage, software (other than the Client Libraries), databases, network and communications equipment and ancillary services needed to connect to, access or otherwise use the Service at its facility (collectively, “Customer Systems”). Customer shall ensure that all Customer Systems are compatible with the Service and comply with all configurations and specifications described in the Documentation.
(b) Service Access. As part of the implementation process, Customer will identify a primary administrative username and password that will be used to set up Customer’s account. Customer may use the administrative user name and password to create subaccounts for its employee users (each with unique login IDs and passwords). Customer shall be responsible for the acts or omissions of any person who accesses the Service using passwords or access procedures provided to or created by Customer. Localytics reserves the right to refuse registration of, or to cancel, login IDs that violate the terms and conditions set forth in this Agreement. Customer agrees to notify Localytics immediately upon learning of any unauthorized use of Customer’s account or any other breach of security. From time to time, Localytics’ personnel may log in to the Service under Customer’s account in order to maintain or improve the Service, including providing Customer assistance with technical or billing issues. Customer hereby acknowledges and consents to such access.

2.3 Service Availability. Localytics will use commercially reasonable efforts to maintain the Service availability to send and receive data, subject to downtimes resulting from maintenance, repairs and upgrades. Localytics will attempt to notify Customer electronically via the Service in advance of any planned downtime. Notwithstanding the foregoing, Localytics will not be liable for any failures in the Service or any other problems which are related to (a) the Customer Data or Customer Systems or (b) outages to any telecommunications or public Internet backbones, networks or servers, or other equipment or service outside of Localytics’ facilities or control.

2.4 Service License. Localytics grants Customer a nonexclusive, nontransferable right and license (without right to sublicense) during the Service term specified in an Order Form, to (a) access and use the Service as described in an Order Form, solely for Customer’s internal business purposes and (b) download, install and use the Client Libraries in connection with Customer’s authorized use of the Service. The Service is made available to Customer solely as hosted by or on behalf of Localytics, and nothing in this Agreement shall be construed to grant Customer any right to receive any copy of the Service or any software (other than the Client Libraries). Customer’s access and use of the Service shall comply with all other conditions set forth in the Order Form and Documentation (such as, for example, any requirements regarding data formats, number of permitted users or prohibited uses).

2.5 Service Support. Localytics will provide Customer with telephone and e-mail support for Customer’s use of the Service during Localytics’ regular business hours. Customer agrees that Localytics is not responsible to provide support for any issues resulting from problems, errors or inquiries related to Customer Systems.

2.6 Professional Services. From time to time, Customer may request and Localytics may agree to provide certain custom development, consulting, training or other professional services as mutually agreed to by the parties (“Professional Services”). The terms and conditions of any such arrangement for Professional
Services shall be set forth in a separate statement of work executed by the parties that specifically references this Agreement (each, a “Statement of Work”), each of which shall be governed by the terms of this Agreement. Unless otherwise agreed to by the parties and set forth in the applicable Statement of Work, all intellectual property, and all rights embodied therein, that are created by Localytics during the performance of Professional Services shall be owned solely and exclusively by Localytics.

3. LICENSE RESTRICTIONS AND CUSTOMER OBLIGATIONS

3.1 Service License Restrictions. Customer shall not directly or indirectly: (i) use the Service or any of Localytics Property or Confidential Information to create any service, software or documentation that performs substantially the same functionality as the Service, (ii) disassemble, decompile, reverse engineer or use any other means to attempt to discover any source code, algorithms or trade secrets underlying the Service (except and only to the extent these restrictions are expressly prohibited by applicable statutory law), (iii) encumber, sublicense, transfer, distribute, rent, lease, time-share or use any Localytics Property in any service bureau arrangement or otherwise for the benefit of any third party, (iv) adapt, combine, create derivative works of or otherwise modify any Localytics Property, or (v) use or allow the transmission, transfer, export, re-export or other transfer of any software, technology or information it obtains or learns pursuant to this Agreement in violation of any export control or other laws and regulations of the United States or any other relevant jurisdiction.

3.2 Unauthorized Use of Service. Customer shall not directly or indirectly: (i) exceed an average of 100 Data Points (Professional Edition) or 150 Data Points (Enterprise Edition) per Monthly Active User in a month, or a fee of $100 per one million additional Data Points will be applied (it being agreed that in the event a different Data Point limit and/or fee for exceeding it is set forth in an Order Form, the limit/fee set forth on the Order Form will apply); (ii) use unique values or a continuous set of values as inputs to event attributes, (iii) interfere or attempt to interfere with the proper working of the Service or any activities conducted on the Service; (iv) bypass any privacy settings or measures Localytics may use to prevent or restrict access to the Service (or other accounts, computer systems or networks connected to the Service); (v) run mail list, Listserv, any form of auto-responder or “spam” on the Service; or (vi) use manual or automated software, devices, or other processes to “crawl” or “spider” any page of the Sites.

3.3 Privacy and Data Protection. Customer shall not provide any personally identifiable information relating to individual persons (“PII”) in connection with its use of the Service, unless Localytics expressly agrees to receive such information and then solely as necessary in connection with Customer’s use of the Service. Customer is responsible for any such PII that Customer provides to Localytics. Customer will have in a privacy policy regarding its use of PII and shall comply with all applicable laws relating to the collection and use of PII and Customer Data.

4. ORDERS, FEES AND PAYMENTS
4.1 Order Form. The parties may enter into one or more Order Forms pursuant to this Agreement. Each Order Form shall specify the Service to be provided, Service Fees, the term during which the Service is to be provided, invoicing terms and any other terms mutually agreed to by the parties. Localytics shall only be responsible to provide the Service identified in an Order Form for the term specified in the Order Form.

4.2 Service Fees. Customer shall pay a fee for the right to use the Service ("Service Fees") in the amount and pursuant to the invoicing schedule set forth in the Order Form. Unless otherwise set forth in the Order Form, Localytics shall invoice the Service Fees monthly in arrears.

4.3 Professional Services Fees. Customer shall pay for Professional Services at the rates and pursuant to the invoicing schedule set forth in the applicable Statement of Work.

4.4 Payment Terms. Payments shall be made in US dollars at Localytics’ address (or to an account specified by Localytics), in full without set-off, counterclaim or deduction by the due date as indicated on the applicable Order Form. If no specific date or payment term is indicated on an Order Form, the due date will be thirty (30) days from the date of the invoice. Past due amounts which are not subject to a good faith dispute shall bear a late payment charge, until paid, at the rate of one and one half percent (1.5%) per month or the maximum amount permitted by law, whichever is less. Customer agrees to reimburse Localytics for all costs (including attorneys’ fees) incurred by Localytics in collecting late payments. In addition to its other rights and remedies, Localytics may, at its option, suspend Customer’s access to the Service or terminate this Agreement in the event that Customer is not current in the payment of fees owed to Localytics.

4.5 Taxes. All payments required by this Agreement are exclusive of federal, state, local and foreign taxes, duties, tariffs, levies, withholdings and similar assessments (including without limitation, sales taxes, use taxes and value added taxes), and Customer agrees to bear and be responsible for the payment of all such charges, excluding taxes based upon Localytics’ net income. All amounts payable by Customer hereunder, including all Service Fees, shall be grossed-up for any withholding taxes imposed by any foreign government on Customer’s payment of such amounts to Localytics.

5. CONFIDENTIALITY

5.1 Scope. “Confidential Information” means all financial, business, operational, marketing or technical information disclosed by or for a party in relation to this Agreement whether disclosed in tangible, written, oral or electronic form, that is of a nature that should reasonably be considered to be confidential and proprietary. Without limitation, (a) the Service, Localytics Property and pricing information are Localytics’ Confidential Information and (b) all Customer Data (including any PII) shall be deemed Customer’s Confidential Information for purposes of this Section 5. Confidential Information expressly excludes any
information (other than PII) to the extent that a recipient can demonstrate such information is (a) already known by it prior to receipt for the disclosing party without restriction, (b) rightfully furnished to it without restriction by a third party not in breach of any obligation to the disclosing party, (c) generally available to the public without breach of this Agreement or (d) independently developed by the recipient without reference to or use of any of the disclosing party’s Confidential Information.

5.2 Confidentiality. Except for the specific rights expressly granted by this Agreement, the receiving party shall not use, copy or disclose any of the disclosing party’s Confidential Information without disclosing party’s prior written consent. The receiving party shall use the disclosing party’s Confidential Information solely for the purpose of exercising its rights and performing its obligations hereunder. The receiving party shall only disclose Confidential Information to its employees, contractors and agents (“Representatives”) who have a need to know for the purposes of this Agreement and are bound by substantially similar confidentiality obligations and shall use reasonable care to safeguard the disclosing party’s Confidential Information. Each party shall be responsible for any breach of confidentiality by its Representatives, as applicable. Promptly upon the disclosing party’s request at any time, the receiving party shall return all of the disclosing party’s tangible Confidential Information, permanently erase all Confidential Information in electronic form and destroy all information, records, copies, summaries, analyses and materials developed therefrom. Each party may disclose the general nature, but not the specific terms, of this Agreement without the prior consent of the other party; provided, however, that either party may provide a copy of this Agreement or otherwise disclose its terms on a confidential basis in connection with any financing transaction or due diligence inquiry.

5.3 Compelled Disclosure. Nothing herein shall prevent a party from disclosing this Agreement or any of the other’s Confidential Information as necessary pursuant to any court order or any legal, regulatory, law enforcement or similar requirement or investigation; provided, prior to any such disclosure, the receiving party shall use reasonable efforts to (a) promptly notify the disclosing party in writing of such requirement to disclose and (b) cooperate with the disclosing party in protecting against or minimizing any such disclosure or obtaining a protective order.

6. PROPRIETARY RIGHTS

6.1 Results and Customer Data. Customer shall own all right, title and interest (including all intellectual property and other proprietary rights) in and to the Results, Customer Data and Customer Systems. Customer hereby grants Localytics a nonexclusive, royalty-free right and license to access, use, copy, process and store the Customer Data solely for the purpose of providing the Service. Customer agrees that the Service depends on the availability of the Customer Data. Customer will be solely responsible for all Customer Data collected from end users as a result of Customer’s use of the Service including
the accuracy and completeness of such information. Unless otherwise set forth in the Order Form, Localytics shall not have any obligation to store any Customer Data or Results. Except for the limited rights and licenses expressly granted hereunder, no other license is granted, no other use is permitted and Customer shall retain all right, title and interest (including all intellectual property and proprietary rights embodied therein) in and to the Results, Customer Data and Customer Systems.

6.2 Service Feedback. Customer may, from time to time and in its sole discretion, make suggestions for changes, modifications or improvements to the Service ("Feedback"). All Feedback shall be solely owned by Localytics (including all intellectual property rights therein and thereto) and shall also be Localytics' Confidential Information. Customer shall and hereby does make all assignments necessary to achieve such ownership.

6.3 Localytics Property. To the extent that Localytics includes any Localytics Property in the Results (including the Report Template), then subject to all terms and conditions of this Agreement, Localytics agrees to grant Customer (without the right to sublicense) a nonexclusive, nontransferable, royalty-free right and license to use such Localytics Property as combined with or embodied in the applicable Results, solely for Customer's internal business purposes in connection with its use of the Results. Except for the limited rights and licenses expressly granted hereunder, no other license is granted, no other use is permitted and Localytics (and its licensors) shall retain all right, title and interest (including all intellectual property and proprietary rights embodied therein) in and to the Service and the Localytics Properties.

6.4 General Learning; Aggregate Data. Customer agrees that Localytics is free to (i) collect, use and create derivative works of data regarding Service usage and performance derived from the Results; (ii) aggregate such data with other data to create compilations and analysis of such data (the “Aggregated Data”); and (iii) use, copy, modify, create derivative works of, publish and disclose such Aggregated Data in a manner that does not directly or indirectly identify Customer or any individual person. Localytics shall own all right, title and interest to the Aggregated Data and any derivative works thereof. In addition, Localytics shall be free to reuse all general knowledge, experience, know-how, works and technologies (including ideas, concepts, processes and techniques) related to the Results or acquired during provision of the Service (including without limitation, that which it could have acquired performing the same or similar services for another customer).

7. WARRANTIES AND DISCLAIMERS

7.1 Customer Data. Customer represents and warrants that it owns all right, title and interest, or possesses sufficient license rights, in and to the Customer Data and PII as may be necessary to permit the use contemplated under this Agreement.
7.2 Service. Localytics represents and warrants to Customer that the Service will be provided in a professional manner in accordance with the terms of this Agreement and the Documentation. Any warranty claim under this Section 7.2 must be made in writing within thirty (30) days after performance of the portion of the Services giving rise to the claim. Localytics’ sole liability and Customer’s exclusive right and remedy for a breach of such warranty is for Localytics to correct or re-perform the nonconforming Service.

7.3 Disclaimers. EXCEPT AS SPECIFICALLY PROVIDED FOR IN THIS SECTION 7, TO THE FULLEST EXTENT PERMITTED BY LAW, EACH PARTY HEREBY DISCLAIMS (FOR ITSELF, ITS AFFILIATES AND THEIR SUPPLIERS) ALL REPRESENTATIONS AND WARRANTIES, WHETHER EXPRESS OR IMPLIED, ORAL OR WRITTEN, INCLUDING WITHOUT LIMITATION, ALL IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE AND ALL WARRANTIES ARISING FROM ANY COURSE OF DEALING, COURSE OF PERFORMANCE OR USAGE OF TRADE. WITHOUT LIMITING THE FOREGOING, LOCALYTICS MAKES NO WARRANTY THAT THE SERVICE WILL MEET CUSTOMER’S REQUIREMENTS OR BE UNINTERRUPTED, ERROR-FREE OR BUG-FREE.

8. INDEMNIFICATION

8.1 Localytics Infringement Indemnity. Except as provided below, Localytics agrees to (a) defend Customer against any allegation demand, claim, action, proceeding or suit (each, a “Claim”) by a third party that Customer’s authorized use of the Service infringes any US patent or copyright or misappropriates any trade secret of such third party and (b) indemnify Customer for settlement amounts or damages, liabilities, costs and expenses (including reasonable attorneys’ fees, “Loss(es)”) awarded to such third party by a court of competent jurisdiction or agreed to as part of a monetary settlement arising out of such Claim; provided, that (i) Customer promptly provides Localytics with written notice thereof and reasonable cooperation, information, and assistance in connection therewith, and (ii) Localytics shall have sole control and authority to defend, settle or compromise such Claim. If the Service becomes or, in Localytics’ opinion, is likely to become, the subject of any injunction preventing its use as contemplated herein, Localytics may, at its option (1) obtain for Customer the right to continue using the Service or (2) replace or modify the Service so that it becomes non-infringing without substantially compromising its principal functions. If (1) and (2) are not reasonably available to Localytics, then it may terminate this Agreement upon written notice to Customer and refund to Customer any unused prepaid Service Fees, pro-rated for the remainder of the prepaid period. Localytics shall have no liability or obligation to Customer hereunder with respect to any Claim or Loss to the extent based upon (a) any unauthorized use of the Service, (b) any modification or combination of the Service with data, software, hardware, or systems not provided by Localytics, (c) any portion of the Service that implements Customer’s specific requirements, (d) Customer’s continuing allegedly infringing activity after being notified to cease.
use as provided for herein or (e) Customer’s continuing use of any version of the Localytics Properties after being provided modifications that would have avoided the alleged infringement. The foregoing states the sole and exclusive liability of Localytics, and Customer’s sole and exclusive remedy, with respect to any actual or alleged violation of intellectual property rights by the Service or any part thereof or by its use or operation.

8.2 Customer Indemnity. Customer agrees to (i) defend Localytics against any Claim by a third party that results from or arises out of (a) any breach by Customer of any of its obligations in Section 3.3 hereof, (b) any breach by Customer of its representations and warranties set forth in Section 7.1 hereof, or (c) any violation of any third party’s (including any of Customer’s end users) privacy right or PII arising out of Customer’s use of the Service; and (ii) indemnify Localytics for any Losses awarded by a court of competent jurisdiction or agreed to as part of a monetary settlement and arising out of such Claim; provided, that (i) Localytics promptly provides Customers with written notice thereof and reasonable cooperation, information, and assistance in connection therewith, and (ii) Customer shall have sole control and authority to defend, settle or compromise such Claim.

9. LIMITATION OF LIABILITY.

IN NO EVENT SHALL EITHER PARTY BE LIABLE TO THE OTHER PARTY CONCERNING THE SUBJECT MATTER OF THIS AGREEMENT, REGARDLESS OF THE FORM OF ANY CLAIM OR ACTION (WHETHER IN CONTRACT, NEGLIGENCE, STRICT LIABILITY OR OTHERWISE), FOR ANY (A) LOSS OR INACCURACY OF OR DAMAGE TO DATA, LOSS OR INTERRUPTION OF USE, OR COST OF PROCUREMENT SUBSTITUTE TECHNOLOGY, GOODS OR SERVICES, (B) INDIRECT, PUNITIVE, INCIDENTAL, RELIANCE, SPECIAL, EXEMPLARY OR CONSEQUENTIAL DAMAGES INCLUDING, BUT NOT LIMITED TO, LOSS OF BUSINESS, REVENUES, PROFITS AND GOODWILL, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, OR (C) DAMAGES, IN THE AGGREGATE, IN EXCESS OF THE TOTAL OF THE AMOUNTS PAID TO IT (IN THE CASE OF LOCALYTICS) OR THE TOTAL OF THE AMOUNTS PAID AND PAYABLE HEREUNDER (IN THE CASE OF CUSTOMER) DURING THE THEN CURRENT TERM OF THE ORDER FORM GIVING RISE TO THE CLAIM. THESE LIMITATIONS ARE INDEPENDENT FROM ALL OTHER PROVISIONS OF THIS AGREEMENT AND SHALL APPLY NOTWITHSTANDING THE FAILURE OF ANY REMEDY PROVIDED HEREIN. THE LIMITATIONS SET FORTH IN THIS SECTION SHALL NOT APPLY TO ANY BREACH OF CONFIDENTIALITY OR PROPRIETARY RIGHTS OR FOR COST OF DEFENSE OR LIABILITIES ARISING OUT OF A PARTY’S INDEMNIFICATION OBLIGATIONS HEREUNDER.

10. TERM AND TERMINATION

10.1 Term of Agreement. This Agreement shall commence on the Effective Date and continue in effect thereafter until the later of (i) such time as a party provides
thirty (30) days prior written notice of termination to the other party if there is no Order Form(s) then in effect or (ii) if there is one or more existing Order Form(s) then in effect, upon the expiration of the last to expire of such existing Order Forms.

10.2 Term of Order Form. This term of each Order Form shall be the term stated in the Order Form (the “Initial Term”). Thereafter, the Order Form will be extended automatically for additional terms of one (1) year each (each “Renewal Term”) at the end of the Initial Term and each Renewal Term (the Initial Term and all Renewal Term, collectively, as the “Term”), unless either party gives written notice of non-renewal at least ninety (90) days prior to the end of the then current Term. An Order Form shall not be subject to termination except as provided in Section 10.3 hereof.

10.3 Termination of Order Form. An Order Form may be earlier terminated by either party (a) if the other party materially breaches a material term of this Agreement or the Order Form and fails to cure such breach within thirty (30) days after receiving written notice of such breach from the other party, or (b) immediately upon written notice, if the other party makes any assignment for the benefit of creditors, or a receiver, trustee in bankruptcy or similar officer is appointed to take charge of any or all of the other party’s property, or the other party seeks protection under any bankruptcy, receivership, trust deed, creditors arrangement, composition or comparable proceeding or such a proceeding is instituted against the other party and is not dismissed within ninety (90) days, or the other party becomes insolvent or, without a successor, dissolves, liquidates or otherwise fails to operate in the ordinary course.

10.4 Effects of Termination. Upon any expiration or termination of this Agreement, all rights, obligations and licenses of the parties shall cease, except that (a) all obligations that accrued prior to the effective date of termination (including without limitation, all payment obligations) and all remedies for breach of this Agreement shall survive, confidentiality, proprietary rights provisions, warranties and disclaimers), indemnification, limitation of liability, and general provisions shall survive. Localytics has no obligation to retain any Customer Data or Results after the Term and will destroy all Customer Data and Results in its possession within ninety (90) days after the end of the Term; provided, upon Enterprise Customer’s written request received within thirty (30) days after termination, Localytics will deliver to Enterprise Customer a copy of the Customer Data then currently stored by Localytics (in the same format maintained by Localytics).

11. GENERAL PROVISIONS

11.1 Entire Agreement. This Agreement (including the Order Form and all Statements of Work) and the CL License constitutes the entire agreement, and supersedes all prior negotiations, understandings or agreements (oral or written), between the parties about the subject matter of this Agreement. Any inconsistent or additional terms on any related purchase order, confirmation or similar form, even if signed by the parties after the date hereof, shall have no force or effect
under this Agreement. No waiver, consent or modification of this Agreement shall bind either party unless in writing and signed by the party against which enforcement is sought. The failure of either party to enforce its rights under this Agreement at any time for any period will not be construed as a waiver of such rights. If any provision of this Agreement is determined to be illegal or unenforceable, that provision will be limited or eliminated to the minimum extent necessary so that this Agreement will otherwise remain in full force and effect and enforceable. Any use of the terms “include,” “included” or “including” shall also be deemed to mean “but not limited to” or “without limitation.”

11.2 Governing Law and Dispute Resolution. This Agreement shall be governed by and construed in accordance with the laws of the Commonwealth of Massachusetts, without regard to its conflicts of law provisions. Neither the United Nations Convention on Contracts for the International Sale of Goods nor any implementation of the Uniform Computer Information Transactions Act in any jurisdiction shall apply to this Agreement. Except with respect to claims for injunctive or other equitable relief, which may be brought at any time before any court of competent jurisdiction, in the event of any dispute arising from or relating to the subject matter of this Agreement, the senior executive officers of the parties shall meet within thirty (30) days following receipt of notice of such dispute, and shall use good faith efforts to attempt to amicably resolve the dispute within such thirty (30) day period. In the event the parties are unable to amicably resolve the dispute within such thirty (30) day period, the dispute shall be finally settled by arbitration in Boston, Massachusetts, in accordance with the American Arbitration Association Rules and Procedures then in effect, by a single arbitrator selected in accordance with said AAA rules. Judgment upon the award so rendered may be entered in a court having jurisdiction. In any action or proceeding to enforce or interpret this Agreement, the prevailing party will be entitled to recover from the other party its costs and expenses (including reasonable attorneys’ fees) incurred in connection with such action or proceeding and enforcing any judgment or order obtained.

11.3 Compliance with Laws. Each party shall comply with all applicable, laws and regulations in connection with the performance of its obligations and the exercise of its rights under this Agreement.

11.4 Remedies. Except as specifically provided otherwise, each right and remedy in this Agreement is in addition to any other right or remedy, which may be available at law or in equity. Each party agrees that, in the event of any breach or threatened breach of Section 5 (Confidentiality) or 6 (Proprietary Rights) or the scope of any license granted hereunder, the non-breaching party will suffer irreparable damage for which it will have no adequate remedy at law. Accordingly, the non-breaching party shall be entitled to seek injunctive and other equitable remedies to prevent or restrain such breach or threatened breach, without the necessity of posting any bond.

11.5 Force Majeure. In the event that either party is prevented from performing, or is unable to perform, any of its obligations under this Agreement (except
payment obligations) due to any cause beyond its reasonable control, the affected party shall give written notice thereof to the other party and its performance shall be extended for the period of delay or inability to perform due to such occurrence.

11.6 Publicity. Localytics will not use Customer’s trademarks, service marks and logos ("Customer Marks") in press releases or as a testimonial without obtaining Customer’s prior written approval. Customer hereby consents to inclusion of its name and logos in customer lists that may be published as part of Localytics’ marketing and promotional efforts.

11.7 Notices. Any notice or communication hereunder shall be in writing and either personally delivered or recognized express delivery courier or certified or registered mail, prepaid and return receipt requested, addressed to the other party at its address specified herein, or at such other address designated in a subsequent notice. All notices shall be in English, effective upon receipt.

11.8 Assignment. Except as expressly provided herein, this Agreement and the rights and obligations hereunder may not be assigned, in whole or in part, by either party without the other party’s written consent. However, without consent, each party may assign this Agreement to any successor to all or substantially all of its business that concerns this Agreement (whether by sale of assets or equity, merger, consolidation or otherwise). Any assignment in violation of this Section 11.8 shall be deemed null and void ab initio. Localytics may use contractors and other third party service providers in performing the Service provided that Localytics shall be liable for the acts and omissions of its subcontractors to the same extent as for its own acts and omissions. This Agreement shall be binding upon, and inure to the benefit of, the successors, representatives and permitted assigns of the parties hereto.

11.9 Independent Contractors. The parties shall be independent contractors under this Agreement, and nothing herein will constitute either party as the employer, employee, agent or representative of the other party, or both parties as joint venturers or partners for any purpose.

11.10 Miscellaneous. This Agreement may be executed in counterparts, which, taken together, will constitute one and the same instrument. The exchange of a fully executed Agreement (in counterparts or otherwise) by electronic means or in writing shall be sufficient to bind the parties to the terms and conditions of this Agreement.

Last Updated: March 29, 2016.

Figure M1-

Terms of Use
Welcome to a website operated by Primate Labs (the “Site”). This Site is intended to provide information about our company, products and services. It allows you to interact with us, provides you the opportunity to contact us directly and may provide you with access to and use of certain products and services. While we want you to enjoy the experience of visiting our Site, we also want you to understand the terms to which you agree when visiting the Site. References to “we,” “our,” “us” or “Primate Labs” herein refer to Primate Labs Inc., a Canadian corporation, and our service providers and designees as deemed appropriate by us.

Compliance with Terms

By accessing this Site, you are agreeing to these Terms of Use, our Privacy Policy and any other legal notices, terms and policies on this Site (together referred to as “Terms”), all of which are expressly incorporated herein by this reference. If you are entering into this agreement on behalf of a company or other legal entity, you represent that you have the authority to bind such entity to the Terms, in which case the terms “you” or “your” shall refer to such entity. For clarity, if you elect to purchase or otherwise obtain access to any of Primate Lab’s software or mobile applications, you acknowledge and agree that your access to and use of all such products and services shall be subject to and governed solely by the terms of the accompanying such software or mobile application.

You agree to use the Site only in accordance with the Terms, whether you are a “Visitor” (which means that you simply browse the Site) or you are a “Member” (which means you have registered with us for the Site). The term “User” means a “Visitor” or a “Member.” Please read and save all of the Terms. If you do not agree with the Terms, do not use this Site or any of its features. If you register to become a Member you may be required to indicate your acceptance to these Terms during the registration process.

Amendment

We may amend or terminate any Terms at any time and such amendment or termination will be effective at the time we post the revised Terms on the Site. Each time you use the Site you should visit and review the then current Terms that apply to your transactions and use of this Site. Your continued use of the Site after we have posted revised Terms signifies your acceptance of such revised Terms. If you are dissatisfied with the Site, its content or any Terms (including as modified), you agree that your sole and exclusive remedy is to discontinue using the Site. The Terms are the entire agreement between you and Primate Labs with respect to your use of the Site.

Eligibility

Use of the Site and its features and registration to be a Member (“Membership”) is void where prohibited. By using the Site and its features, you represent and
warrant that (a) you are not a minor and may otherwise enter into and form binding contracts under applicable law; (b) all registration information you submit is truthful and accurate; (c) you will maintain the accuracy of such information; and (d) your use of the Site and its features does not violate any applicable law or regulation.

Term

The Terms, as we may revise them from time to time, shall remain in full force and effect while you use the Site features or are a Member. Except with respect to any offerings governed by a separate agreement, you may terminate your Membership at any time, for any reason, by providing written notice to us. We may terminate your Membership at any time, for any or no reason, with or without prior notice or explanation, and without liability. Even after Membership is terminated, the Terms will remain in effect and you will remain bound by them except that your right to use the Site as a Member will terminate.

Passwords

If you register to become a Member, you will be asked to choose a user name (or email address) and password. You are solely responsible for maintaining the confidentiality of this information. You are not permitted to allow anyone else to use your user name and password to log into the Site. You must notify us immediately if you suspect any unauthorized use of your account or access to your password. You are solely responsible for any and all use of your account so please be vigilant in protecting its confidentiality.

Your Use of the Site

Users have a non-transferable, non-exclusive license to access this Site, to view information contained at this Site and to interact with the Site solely for their own personal use and not for any commercial purpose. You agree not to use the Site for any unlawful purpose. You agree not to rent, retransmit, disclose, publish, sell, assign, lease, sublicense, market, or transfer the Site or any portion thereof or use it in any manner not expressly authorized by the Terms. You agree not to copy, reverse engineer, translate, port, modify or make derivative works of any portion of the Site.

Tampering with the Site, misrepresenting the identity or age of a user, using buying agents or conducting fraudulent activities on the Site are prohibited.

Content

Proprietary Rights

Primate Labs does not claim any ownership rights in the text, files, images, photos, video, sounds, musical works, works of authorship, applications, or any other materials (collectively, “Content”) that you post on the Site (if applicable). After posting your Content to the Site, you continue to retain any such rights that you may have in your Content, subject to the limited license herein. The Site may contain Content of Users and other licensors. Except as
provided within the Terms, you may not copy, modify, translate, publish, broadcast, transmit, distribute, perform, display, or sell any Content appearing on or through the Site.

By displaying, publishing Content on the Site, or otherwise submitting Content to us (collectively, “posting”), you hereby grant to Primate Labs an irrevocable, perpetual, worldwide, royalty-free, non-exclusive license to use, modify, delete from, add to, create derivative works of, publicly perform, publicly display, reproduce and distribute (and to sublicense the foregoing rights through multiple tiers of licensees) such Content on or through the Site. From time to time, we may remove Content from the Site, permanently or temporarily. For clarity, Content includes Benchmark Results. “Benchmark Results” means any data and related information, including but not limited to technical information about a device, system and application software, and peripherals, that is gathered periodically by a Primate Labs’ service or product which you may license under a separate agreement. You represent and warrant that: (i) you own the Content posted by you on or through the Site or otherwise have the right to grant the licenses set forth above, and (ii) the posting of your Content on or through the Site does not violate the privacy rights, publicity rights, contract rights, intellectual property or any other rights of any person or entity. You agree to be fully responsible for and to pay any and all royalties, fees, and any other monies owing any person or entity by reason of any Content posted by you.

The Site also contains Content provided by Primate Labs, including, without limitation, text, images and logos (“Primate Labs Content”). Primate Labs Content is protected by copyright, trademark, patent, trade secret and other laws, and Primate Labs owns and retains all rights in the Primate Labs Content and the features and functionality of the Site. Primate Labs hereby grants you a limited, revocable, nonsublicensable license to reproduce and display the Primate Labs Content (excluding any software code) solely for your personal use in connection with utilizing the products and services available on the Site.

Content Posted

You agree to not use the Site to: a. upload, post, email, transmit or otherwise make available any Content that is unlawful, harmful, threatening, abusive, harassing, tortuous, defamatory, vulgar, obscene, libelous, invasive of another’s privacy, hateful, or racially, ethnically or otherwise objectionable; b. harm minors in any way; c. forge headers or otherwise manipulate identifiers in order to disguise the origin of any Content transmitted through the Site; d. upload, post, transmit or otherwise make available any Content that you do not have a right to make available under any law or under contractual or fiduciary relationships; e. upload, post, transmit or otherwise make available any unsolicited or unauthorized advertising, promotional materials or any other form of solicitation; f. upload, post, transmit or otherwise make available any material that contains software viruses or any other computer code, files or programs designed to interrupt, destroy or limit the functionality of any computer software or hardware or telecommunications equipment; g. interfere with or disrupt the Site or servers
or networks connected to the Site, or disobey any requirements, procedures, policies or regulations of networks connected to the Site; or h. intentionally or unintentionally violate any applicable local, state, national or international law; and/or collect or store personal data about other users in connection with the prohibited conduct and activities set forth in paragraphs a through g above.

Primate Labs may reject, refuse to post or delete any Content for any or no reason, including, without limitation, Content that in the sole judgment of Primate Labs violates these Terms, including our Privacy Policy. We assume no responsibility for monitoring the Site for inappropriate Content or conduct. If at any time Primate Labs chooses, in its sole discretion, to monitor the Site, we nonetheless assume no responsibility for the Content, no obligation to modify or remove any inappropriate Content, and no responsibility for the conduct of the User submitting any such Content.

You are solely responsible for your use of the Site, the Content that you post on or through the Site, and any material or information that you transmit to other Members and for your interactions with other Users.

Disclaimers

Primate Labs is not responsible for and makes no warranties, express or implied, as to any content on the Site, including, without limitation with respect to the accuracy and reliability of the Primate Labs Content, Content or other Content posted on or through the Site, whether caused by us, by Users, by any of the equipment or programming associated with or utilized by the Site, or otherwise. The User Content does not necessarily reflect the opinions or policies of Primate Labs. Profiles and third party applications created and posted by Members on the Site may contain links to other websites. Primate Labs is not responsible for the content, accuracy or opinions expressed on such websites, and such websites are not necessarily investigated, monitored or checked for accuracy or completeness by Primate Labs. Inclusion of any linked website on the Site does not imply approval or endorsement of the linked website by Primate Labs. When you access these third party sites, you do so at your own risk. Primate Labs takes no responsibility for third party advertisements or third party applications that are posted on or through the Site, nor does it take any responsibility for the goods or services provided by its advertisers. Primate Labs is not responsible for the conduct, whether online or offline, of any User of the Site including, without limitation, any Content posted by any User. Primate Labs assumes no responsibility for any error, omission, interruption, deletion, defect, delay in operation or transmission, communications line failure, theft or destruction or unauthorized access to, or alteration of, any User or Member communication. Primate Labs is not responsible for any problems or technical malfunction of any telephone network or lines, computer online systems, servers or providers, computer equipment, software, failure of any email or players due to technical problems or traffic congestion on the Internet or on any portion of the Site or combination thereof, including any injury or damage to Users or to any person’s computer related to or resulting from participation or downloading
materials in connection with the Site. Under no circumstances shall Primate Labs be responsible for any loss or damage, including personal injury or death, resulting from use of the Site, from any User Content posted on or through the Site or from the conduct of any Users, whether online or offline. Additionally, Primate Labs shall have no liability for any viruses or anything beyond our control. We are not responsible for any damage to your computer, software, modem, telephone or other property resulting from your use of the Site. We will not be liable to you if you are unable to access information through the Site. Some states do not allow the exclusion or limitation of certain warranties and/or liabilities, so certain of the above limitations or exclusions may not apply to you. The Site is provided “AS-IS” and as available and Primate Labs expressly disclaims any warranty of fitness for a particular purpose or non-infringement. Primate Labs cannot guarantee and does not promise any specific results from use of the Site.

Limitations of Our Liability

IN NO EVENT SHALL WE BE LIABLE FOR ANY DAMAGE, CLAIM OR LOSS INCURRED BY YOU, INCLUDING WITHOUT LIMITATION COMPENSATORY, INCIDENTAL, DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES, IRRESPECTIVE OF WHETHER WE HAVE BEEN INFORMED OF, KNEW OF, OR SHOULD HAVE KNOWN OF THE LIKELIHOOD OF SUCH DAMAGES. THIS LIMITATION APPLIES TO ALL CAUSES OF ACTION IN THE AGGREGATE INCLUDING WITHOUT LIMITATION BREACH OF CONTRACT, BREACH OF WARRANTY, DEFAMATION, NEGLIGENCE, STRICT LIABILITY, MISREPRESENTATION, AND OTHER TORTS, AS WELL AS THIRD-PARTY CLAIMS. IF THE WARRANTY EXCLUSIONS OR LIMITATIONS OF LIABILITY SET FORTH IN THIS USE AGREEMENT ARE FOR ANY REASON HELD UNENFORCEABLE OR INAPPLICABLE, YOU AGREE THAT OUR AGGREGATE LIABILITY SHALL NOT EXCEED ONE HUNDRED U.S. DOLLARS ($100).

Indemnification

You agree to indemnify and hold Primate Labs, its subsidiaries, and affiliates, and their respective officers, agents, partners and employees, harmless from any loss, liability, cost, expense, claim, or demand, including without limitation, reasonable attorneys’ fees, due or relating to or arising out of your use of the Site in violation of the Terms and/or arising from a breach of the Terms and/or any breach of your representations and warranties set forth in the Terms and/or arising out of or relating to any Content that you post or third party transaction that you enter or attempt to enter in connection with the Site.

Miscellaneous

The Terms will be construed, and their performance enforced, under the laws of Oregon without reference to choice of law principles. Any dispute relating to the Terms or the Site may be litigated only in a court having jurisdiction and venue in Multnomah County for state court causes of action and in the District of Oregon
for federal court causes of action. The United Nations Convention on Contracts for the International Sale of Goods does not apply to the Terms. We may assign the Terms, in whole or in part, to a related entity or to a third party.

EACH OF THE PARTIES HEREBY KNOWINGLY, VOLUNTARILY AND INTENTIONALLY WAIVES ANY RIGHT IT MAY HAVE TO A TRIAL BY JURY IN RESPECT OF ANY LITIGATION (INCLUDING BUT NOT LIMITED TO ANY CLAIMS, COUNTERCLAIMS, CROSS-CLAIMS, OR THIRD PARTY CLAIMS) ARISING OUT OF, UNDER OR IN CONNECTION WITH THIS AGREEMENT. FURTHER, EACH PARTY HERETO CERTIFIES THAT NO REPRESENTATIVE OR AGENT OF EITHER PARTY HAS REPRESENTED, EXPRESSLY OR OTHERWISE, THAT SUCH PARTY WOULD NOT IN THE EVENT OF SUCH LITIGATION, SEEK TO ENFORCE THIS WAIVER OF RIGHT TO JURY TRIAL PROVISION. EACH OF THE PARTIES ACKNOWLEDGES THAT THIS SECTION IS A MATERIAL INDUCEMENT FOR THE OTHER PARTY ENTERING INTO THE TERMS.

The Terms are accepted upon your use of the Site or any of its features and is further affirmed upon you becoming a Member (if applicable). The Terms constitute the entire agreement between you and Primate Labs regarding the use of Site and its services and features. The failure of Primate Labs to exercise or enforce any right or provision of the Terms shall not operate as a waiver of such right or provision. The section titles in the Terms are for convenience only and have no legal or contractual effect. The Terms operate to the fullest extent permissible by law. If any provision of the Terms is unlawful, void or unenforceable, that provision is deemed severable from the Terms and does not affect the validity and enforceability of any remaining provisions.

SwiftyJSON-

The MIT License (MIT)

Copyright (c) 2014 Ruoyu Fu

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY
KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Cocoapods- 
Copyright and Content Ownership

1. You retain your rights to any Content you submit, post or display on or through the Services. By submitting, posting or displaying Content on or through the Services, you grant us a worldwide, non-exclusive, royalty-free license (with the right to sub-license) to use, copy, reproduce, process, adapt, modify, publish, transmit, display and distribute such Content in any and all media or distribution methods (now known or later developed).

2. We do not moderate or monitor Content (and we are not responsible for doing so), but we and our designee have the right (but not the obligation), at any time in our sole discretion to monitor, refuse or remove any Content on or available via the Service.

3. You agree to indemnify, defend, and hold harmless CocoaPods (and, as applicable, our officers, directors, agents, and employees) from and against any and all claims, demands, suits, and proceedings brought by any third party, and all related damages, judgments, losses, liabilities, and expenses (including reasonable attorneys’ fees), arising or resulting from any allegation that your Content, or your use of the Service in violation of this Agreement, violates any applicable law or any intellectual property right or other right of any third party; provided that we (a) promptly give you written notice of such claim, demand, suit or proceeding; (b) give you sole control of the defence and settlement of such claim, demand, suit or proceeding (provided you may not settle any such claim, demand, suit or proceeding unless the settlement unconditionally releases us (and, as applicable, our officers, directors, agents, and employees) of all liability); and (c) provide you with such cooperation as you may reasonably require in such defense, at your expense.
4. The look and feel of the Service is copyright ©2015 CocoaPods. All rights reserved. You may not duplicate, copy, or reuse any portion of the HTML/CSS, JavaScript, or visual design elements or concepts without our express prior written permission.

MWPhotoBrowserSwift Licence-

Copyright (c) 2010 Michael Waterfall <michaelwaterfall@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Google developers

Site Policies
We are pleased to license much of the documentation on Google Developers under terms that explicitly encourage people to take, modify, reuse, re-purpose, and remix our work as they see fit.

You will find the following notice at the bottom of many pages on the Google Developers website:

*Except as otherwise noted, the content of this page is licensed under the Creative Commons Attribution 3.0 License. For details, see our Site Policies.*

When you see a page with this notice you are free to use *nearly everything* on the page in your own creations. For example, you could quote the text in a book, cut-and-paste sections to your blog, record it as an audiobook for the visually impaired, or even translate it into Swahili. Really. That's what open content licenses are all about. We just ask that you give us attribution when you reuse our work.

You may also find the following notice on the bottom of some pages:

*Except as otherwise noted, the content of this page is licensed under the Creative Commons Attribution 3.0 License, and code samples are licensed under the Apache 2.0 License. For details, see our Site Policies.*

When you see this notice, you are free to use the content, and in addition you are free to use the computer source code that appears in the content (such as in examples) in the code of your own projects.