

## Media Manipulator Lifecycle Objectives

### Vision

The Media Manipulator is a Facebook app which takes *images* and modifies them in an advanced artistic manner. It also takes *video files* and automates subtitles if the video file contains voice.

Most of today's image viewers offer either storage or editing capabilities, and sometimes both. However, we have observed that too often, applications offering image editing capabilities are either too hard to use, or only provide relatively simple editing features. Not many of them let you modify the image in a dramatic way with a single click. Our application allows any user with a Facebook account to take their existing photos and easily turn them into something expressive and artistic. It also enables the user to take their existing video files and make them more accessible, with the feature of automating subtitles, to a portion of users who may have low hearing. It may also appeal to users who just want to have subtitles attached to their video, who don't know how to do so otherwise.

Examples of existing image editing tools include Google's Picasa and Pixlr.com. Picasa lets you make simple adjustments to the image such as changing brightness and contrasts. On the other hand, Pixlr.com gives you full control over the image by providing you with Photoshop like tools kit.

### Image Manipulation

#### *Feature Set 1: Add Texture*

2-D art can be made with many different materials, such as charcoal, chalk, pastels, etc. Media Manipulator can modify images so that they look like they are drawn with one of these different materials. It basically lets the user choose a material they want the images to be drawn in, and the program will modify the image accordingly.

#### *Feature Set 2: Light Effects*

The manipulator can also change the apparent lighting of the image, causing light to focus on one object in the image while blurring others. This is contrary to traditional photography in that focus is a feature given to a photo after it is taken, not beforehand. This allows users with little photographing experience to have professional features without significant photographic/artistic experience.

#### *Feature Set 3: Basic*

In addition, the product will also retain basic editing capabilities such as contrasting, grey scaling, mosaic-izing, cropping, re-sizing, etc.

### Video Manipulation

The Media Manipulator also takes video files and automates subtitles. The goal of this is to allow almost anyone to create more professional videos from a basic video file with voice audio.

This is also helpful for people with low hearing, or hard time to understand the audio language if unclear, or in a setting with low quality audio.

## Software Architecture

### Facebook Application

Facebook's Graph API lets us, given a contributor license agreement, explore the user's social data. With this data and access to their platform, we can personalize the application to a specific user.

### HTML5 and Javascript

HTML5's canvas can be used with Javascript to extract pixel data from an image. With this pixel data, we can perform pixel manipulations to achieve desirable effect.

### Google Speech Recognition API

Google's Speech Recognition API will be used to get text from audio files, which will then be used to subtitle our videos.

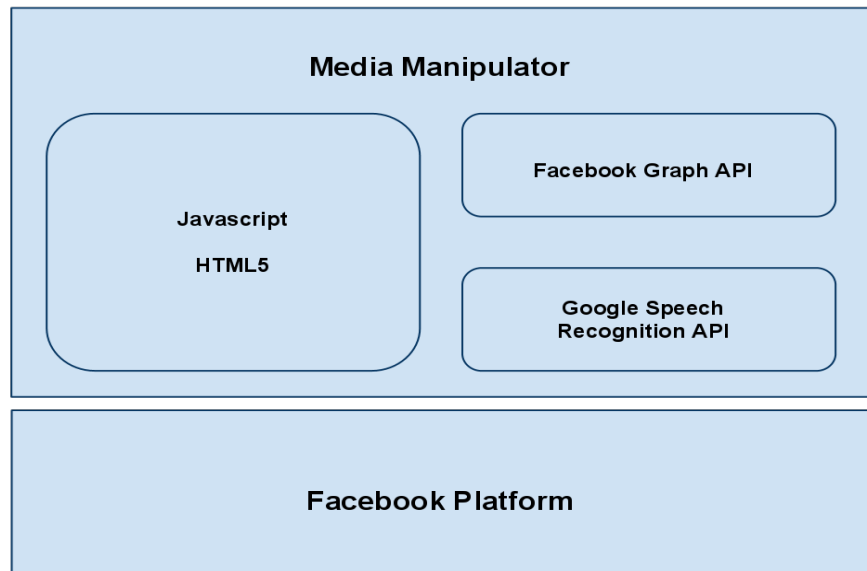


Figure 1. Software Architecture

### **Challenges and Risks**

The main challenge is in figuring out how to manipulate the pixels of the image so that it affects the image in desirable and artistic way. We are going to have to try out many different methods of pixel manipulations to see what effects are possible to realize. In addition, integrating the image manipulation software with Facebook's photo album on the Facebook platform also may present a challenge in our progress.

Note: Cost of storing image data is all on Facebook! All other system/software requirements are free, except for the labor required of the team.

### **Software Development Model**

We plan to implement our product under the Spiral model. We chose this model because the numerous iterations of design, risk analysis, engineering, and customer validation ease concerns on integration, along with taking care of bugs before next stage of development is entered. Reassessing risk and possibly better design decisions before it may be harder to change is also a plus, since we are not sure of the exact way the modules of our design will integrate with each other.