# Project Barrett

November 4, 2014

# 1 Project Barrett

Project Barrett is the given project name for the first prototype of an offline advertisement analytics engine. It's goal is to serve two main functions:

1. Provide users with analytics on how many people view the advertisement at a given time, and demographics (age, gender).

2. Provide a means to change the content of the currently displayed advertisement in real time, in response to changing environment characteristics. This will only be done on a dynamic display such as a TV, LED array, or other 'screen'-like device.

# 2 Product Components

Project Barrett will have four main modules:

1. Computer vision system. This will be a software implementation of face detection, finding gender and age of each face.

2. Embedded device. This will be temporary for this prototype. The device will have to be able to communicate with our servers via wifi, and/or usb with a computer and gui application.

3. Front end web page with authentication. This will be for users to view analytics, and upload images or videos to change the 'advertising package' that the pi will display. This will be our main view.

4. The backend server. This will be our model and controller, which will serve our front end authentication, and provide communication between user and pi.

# 3 Proposed Technologies

The intent of this section is to provide easy to use technologies that are available off the shelf. This will help with rapid prototyping of Project Barrett. In order of product components:

1. Opencv. Half the code is already written, and can easily be ported over to the proposed embedded device. Optimizations will need to be made to achieve a minimum of two samples per second.

2. Raspberry pi with 3d printed case. This is the easiest off the shelf device to use with an integrated low power camera. The pi will need to be modified to only boot the single application everytime at startup. The pi will need to be paired with a desktop program to upload analytics, and transfer advertising packages.

3. ASP.NET, or comparable web technology (javascript, html, css). They only need to be able to communicate with our user endpoint, and our server.

4. ASP.NET backend. This is to be hosted with azure. This recommendation is made because of current familiarity with the technology, and because a $150 per month credit will be given for free to us.

# 4 Recommended Breakdown of Teams

It is recommended two teams be formed. The first being a hardware team, an the second being a software team. At least one person must maintain status with both teams to enable effective communication.

The hardware team would be responsible with integrating Product Component (1) with Product Component(2), and interfacing the device with the user's current computer system, and our own backend server.

The software team would be responsible for implementing Product Component (1), and building Product Components (3) and (4).