

# Team 17 Project Backlog

## Annix

---

**Team Members:** Ethan Ling, Leo Gu, Matthew Sigit, Nick Song

**Project Title:** Annix, an easy way to recognize any character you find online

---

### Problem Statement

Facial recognition is a popular and effective tool employed by applications and search engines to fill many needs. However, current models for facial recognition have significantly lower efficacy when applied to drawn or animated formats. Our solution to this problem is Annix, a multi platform service, which aims to use Machine Learning and Convolutional Neural Networks trained specifically on animated characters to fulfill this need.

### Background Information

#### Audience

This tool is geared towards anime watchers. With the popularity of anime increasing, there subsequently follows the need for anime recognition software. Many anime watchers may find a clip of an anime online, but don't know what the name of the characters or anime are. This makes it much harder for them to find the anime online and watch on their own. That's where our tool comes in. Our tool aims to provide the user with both the name of the character and the name of the anime it's from, just from a single image.

#### Similar Platforms and Their Limitations

There exist several readily available and free-use image recognition softwares such as Google Lens, Bing Visual Search and Camfind. Some of these softwares are well integrated into search engines and are good at general purpose recognition for a host of items. However, while these image recognition softwares excel at recognizing images in real life, their accuracy drops significantly when prompted with animated characters. This is because they are trained on general and much more noisy images, meaning their accuracy on simply drawn animated characters is not as high. Our model will only be trained on animated characters, meaning that the accuracy when prompted anime characters in the future will be better.

## Functional Requirements

- 1) As a user, I would like to be able to register an account on Annix.
- 2) As a user, I would like to be able to login into my account on Annix.
- 3) As a user, I would like to be able to link and unlink my account to an email address.
- 4) As a user, I would like to be able to change critical information about my account like my password, username, email, and profile picture.
- 5) As a user, I would like to be able to change my profile picture to a recent search.
- 6) As a user, I would like to seamlessly upload an image and obtain a response according to the image that is accurate.
- 7) As a user, I would like to know if the application is unsure of my requested character.
- 8) As a user, I would like to add example images and descriptions of characters that I would like added.
- 9) As a user, I would like to be able to reset my password if I have forgotten it with security questions and an email sent to my inbox.
- 10) As a user, I would like to have the security to login using a 2-factor authentication service.
- 11) As a user, I would like to be able to access a guide to use this application.
- 12) As a user, I would like to see the functionality of each button when hovering the mouse over it.
- 13) As a user, I would like to be able to use the product without signing in.
- 14) As a user, I would like to have a method of exporting a backlog or history of previous images uploaded and their corresponding characters.
- 15) As a user, I would like to be able to know where to watch the show that a character I've searched appears in and have the links to the services where the show is hosted.
- 16) As a user, I would like to be able to tell the model if it has misidentified a character, as well as send the correct character if I know it.
- 17) As a user, I would like to be able to delete my account and all of the information Annix has stored in it.
- 18) As a user, I would like to be able to easily disable search history and other data collecting functions on Annix.
- 19) As a user, I would like to know generalized data about the user base such as most commonly recognized characters, shows, and hosting services.
- 20) As a user, I would like analytics specific to me like characters frequently searched.

- 21) As a user, I would like to be able to seamlessly go between profile, service and data pages as well as other function pages.
- 22) As a user, if the search function is unable to find my character, I would like to request a fellow user to recognize my character.
- 23) As a user, I would like to be able to help other users if they request the details of a character that I recognize.
- 24) As a user, I would like to be able to help improve the model by labeling data if I feel the service is lacking.
- 25) As a user, I would like to know statistics about my usage, such as characters searched, images recognized, people helped, etc.
- 26) As a user, I would like for the app to automatically switch from light and dark modes depending on my system settings, and I would like to be able to switch it back if I choose to do so.
- 27) As the user, I would like a button on the application to review it if I am satisfied with its performance.
- 28) As a user, I would like to be able to request new features.
- 29) As the user, I would like a button on the application to send constructive criticism to the developers if I would like a certain aspect of the application to be improved.
- 30) As a user, I would like to identify different characters by image with an accuracy of at least 80%.
- 31) As a user, I would like to not only identify characters from their source material but also fan-made versions of them as well.
- 32) As a user, I would like to be able to recognize characters, even if they are slightly edited with filters or Photoshop.
- 33) As a user, I would like to receive some form of rating or ranking system dependent on how many users I have accurately helped.
- 34) As a user, I would like to be able to see top ranked helpers based on time increments.
- 35) As a user, I would like the website to be friendly when opened on a phone or mobile device.
- 36) As a user, I would like to know the exact source of the image I uploaded displayed in the model.
- 37) As a user, I would like to know if I uploaded the wrong image i.e. a non animated character.

- 38) As a user, I would like to be able to easily replace an image if I realize I uploaded the wrong one.
- 39) As a user, I would like to know what personal data Annix has stored and how it is being used.
- 40) As a user, I would like to be able to see Frequently Asked Questions for any features I don't understand.
- 41) As a user, I would like to be able to see a Contact Us page to contact the developers if I find something wrong.
- 42) As a user, I would like to be able to see a quick summary of how the model works.
- 43) As a user, I would like to be able to apply to be a moderator.
- 44) As a user, I would like to be able to join a discord of like minded users.
- 45) As a user, I would like to be able to search for characters and see details about the characters.
- 46) As a user, I would like to be able to search for other users and see details relating to ranking, account status and more.
- 47) As a user, I would like to be able to filter my search depending on use time, ranking, or date of profile creation.
- 48) As a user, I would like to be able to see related items to the characters I've searched.
- 49) As a moderator, I would like my moderator status to show on my profile.
- 50) As a moderator, I would like to be able to verify any responses in the help section.

# **Non-Functional**

## **Machine Learning Application**

At this project's core is the image recognition machine learning model. This model has two main hurdles to its success: getting proper training data and high accuracy.

### **Data**

To train a high accuracy model, we would like a minimum of 100,000 images that covers the most popular animated characters in the last 15 years. In this project, our training data will be obtained from a few major sources. First are publicly available online datasets. These datasets don't perfectly fit the data wanted for our model type, in this case being triplet loss data, but can be reformatted through python packages or through a data pipeline made in tensorflow. The other major source of data is going to come from web scraping. This is going to require us to write a web scraping script to extract images from sources like memes, show review sites, and other pages and deposit them in a database. To conduct this efficiently, we will require a way to obtain clean and labeled triplet loss data.

Our main solution for obtaining this is a proper data pipeline. Triplet loss data comprises of three images: an anchor, a positive, and a negative. The anchor and positive have the same label, which allows the model to learn to differentiate between different labels. We can build a pipeline which inserts triplets into our model from our dataset in the appropriate format (anchor, positive, negative) such that the anchor and positive images have the same label and the anchor and the negative have different labels. This only requires us to have properly labeled images, and many of the open source datasets are clean and have accurately labeled images. However, these datasets are often not up to date and don't include more modern characters or renditions of characters.

This is where web scraping comes in, as the internet has plenty of images with the latest characters and shows. However, these images are hard to label from every context and the data source may not have the most accurate metadata or information to label the image. For this we propose manual labeling, either through us or by creating a system such that other people can help us label. This would be a lot of work, but it is necessary to keep our labels properly diverse, relevant and accurate. All of these solutions would allow us to get great training data for our model.

## **Model Accuracy and Other Problems**

For our model to be deployed on a publicly available service, we would like our model to have an accuracy above 80 percent. To achieve this, we selected one of the foremost models for facial recognition, triplet loss. We plan to implement a triplet loss model using Tensorflow and Keras, which we are going to follow the many papers written on the subject as well as see many of the pretrained triplet loss models that exists such as FaceNet, ResNet 50, or other models that we can find on TFHub and other services. Ideally, we would like to build this model from the ground up and we will attempt to, but if the accuracy on real world images does not match a high standard (e.g. 80 percent for us), then we would have a fall back.

Other problems that could face our model include overfitting, underfitting, and poor generalization, but these can be solved through tweaking the training process, dataset, or the layers of the model itself. Through this, we know given enough work, we can make a proper image recognition model with a good accuracy.

## **Web Application**

For our Web Application, we are planning to use a REST API model of communication, in which clients' devices will send requests to a server and receive back information in a standardized representational state. For our back-end, we plan to use MongoDB and Flask for the purpose of storing information in a relational database and coding the functions to utilize and distribute said information.

In terms of our front-end, we plan to use ReactJS to hold a framework in HTML and CSS. The user will have a button to upload an image file and a button to generate the information required. There will also be a login page and a statistics page.

## **Systems and Architecture**

The goals of a good system is scalability, data consistency, and timeliness. The primary host of most of our services will be through AWS. AWS allows us to have accurate responses in a timely manner and also has a great auto-scaling feature that allows for our system to handle differing amounts of users with a CDN. The overall architecture would consist of Route 53 or some other CDN paired with a EC2 layer that contains the frontend. That frontend would route to another EC2 that contains our backend written in Flask that can communicate with our DB, S3 Buckets, or our ML App. First of all this adds an extra layer of security because all of our data will be on a virtual network and also allows us to decentralize our model and handle thousands of users. The cost of this won't even be too bad with student credits. Overall, AWS allows us to pair a bunch of features with our application which can allow us to reach our functional goals.