**Deep Learning with Clarifai API**

CS 5551 Project Spring 2017

Team 4: Hunters

Shalin Patel (67), Jackie Batson (3), Sandeep Pabolu (63), Akhila Katta (40)

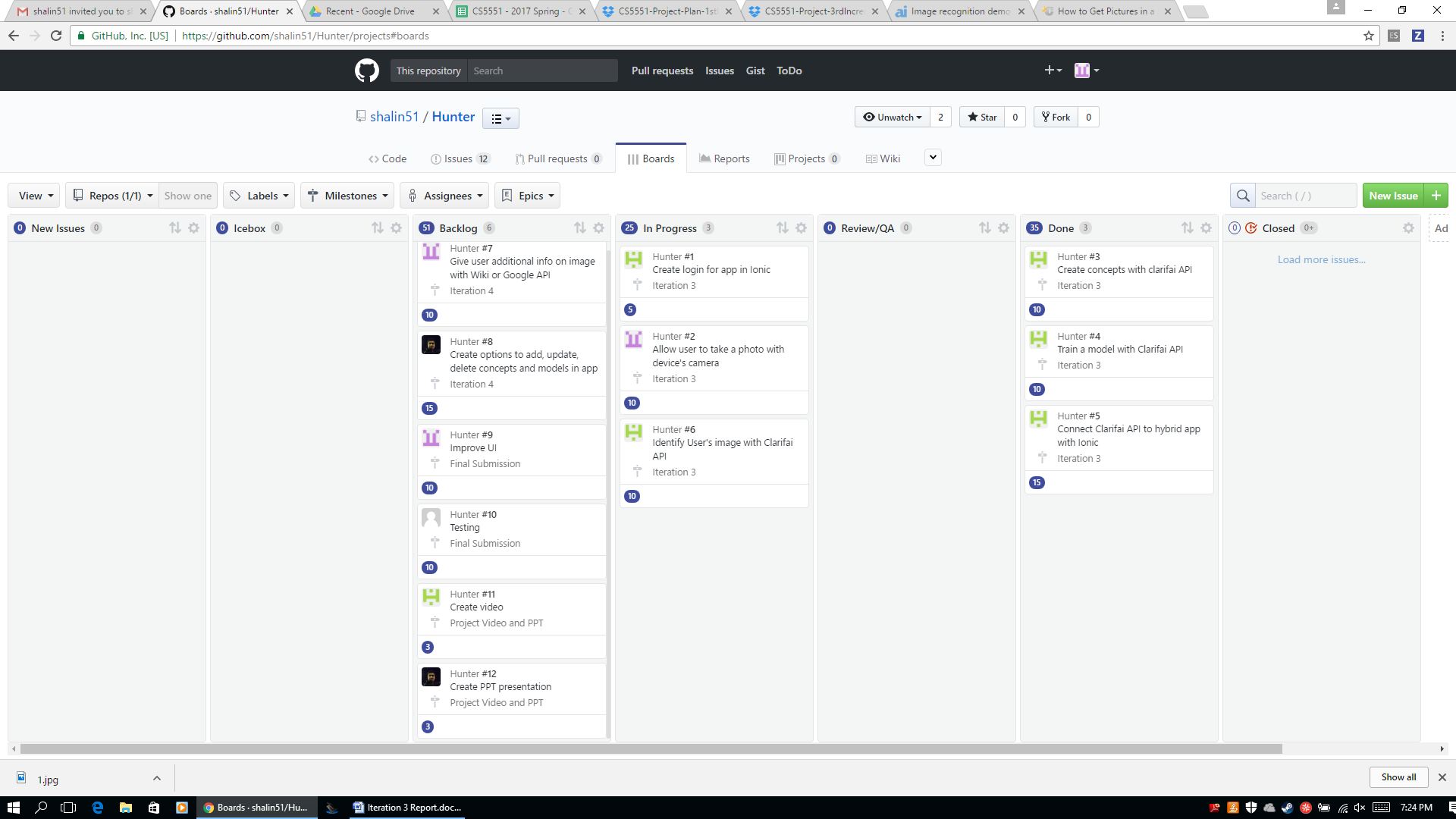
1. **INTRODUCTION**

This project will be to create a Native/Web hybrid mobile application in which a user can take a picture, and get an identification and information about what the image is. The project will involve training and testing of images using the Clarifai open API for deep learning purposes.

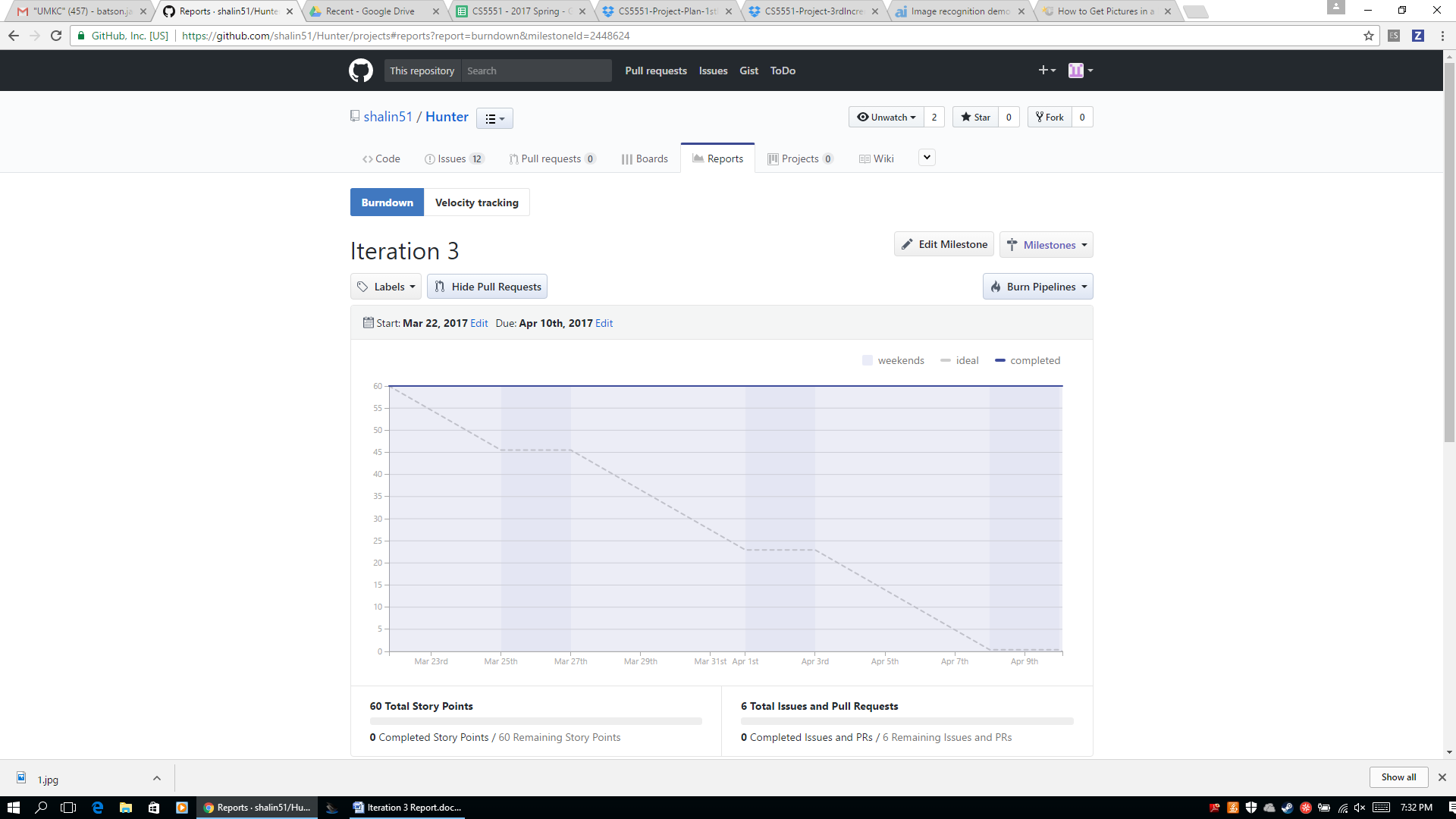
1. **PROJECT GOAL AND OBJECTIVES**
2. The overall goal of the project is to explore deep learning and hybrid mobile app development.
3. Problem statement: create a native/web hybrid mobile application in which a user can take a picture and get an identification and information about the image using the Clarifai open API.
4. Specific features of the app will be an interface with the device’s camera so the user can take a picture. The picture will then query the Clarifai API, which will return an identification of the picture. Using that identification, the app will return additional information about the picture to the user from its own database. The Clarifai API will be pre-trained using specific categories of images for the application.
5. Significance: machine learning and deep learning are considered the next wave in technological advancement. The combination of expanding storage and processing power, along with the use of the internet, has greatly increased the amount of data available for computer learning. A computer can be trained using available data to create a model that can be used to predict the output from new input. Our app will be significant because it will train a model for a specific image recognition task, detailed for our specific needs.
6. **PROJECT PLAN**
7. Scheduled use cases (Product Backlog):

|  |  |  |  |
| --- | --- | --- | --- |
| PRODUCT BACKLOG | | | |
| Story | Description | Story Points | Priority |
| 1 | As a user, I would like to be able to log in to the system via my phone | 5 | 3 |
| 2 | As a user, I would like to be able to take a photo with my phone from the app | 10 | 2 |
| 3 | As a user, I would like to have my captured image identified. | 10 | 1 |
| 4 | As a user, I would like to have additional information provided about the captured image. | 15 | 8 |
| 5 | As an administrator, I’d like to use the app to add categories of images (concepts) to train | 15 | 4 |
| 6 | As an administrator, I’d like to use the app to create models for image recognition | 20 | 5 |
| 7 | As an administrator, I’d like to use the app to update models | 15 | 6 |
| 8 | As an administrator, I’d like to use the app to delete models | 5 | 7 |

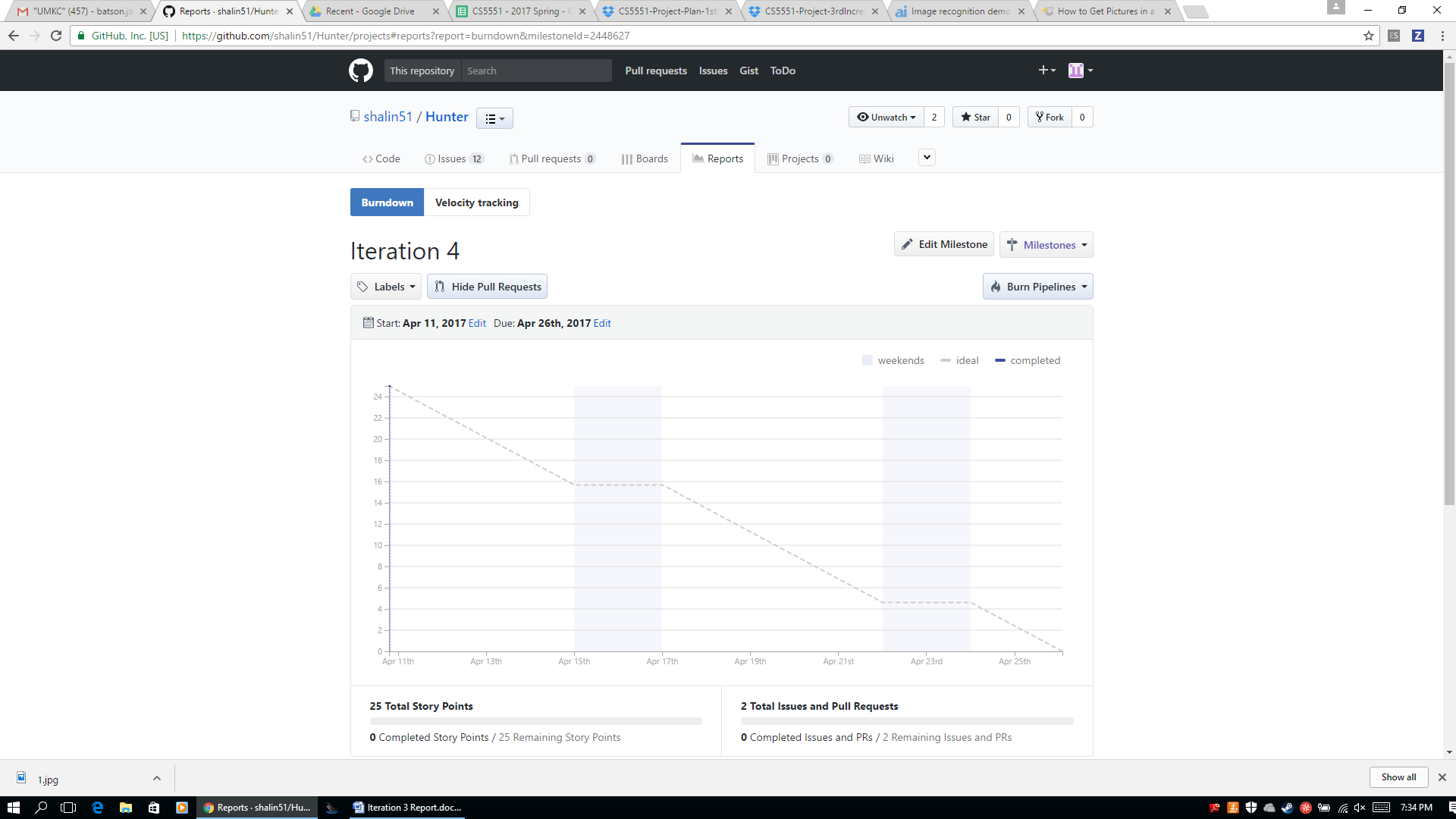
1. Timelines and responsibility:



1. Burndown chart:
2. Iteration 3:



1. Iteration 4:



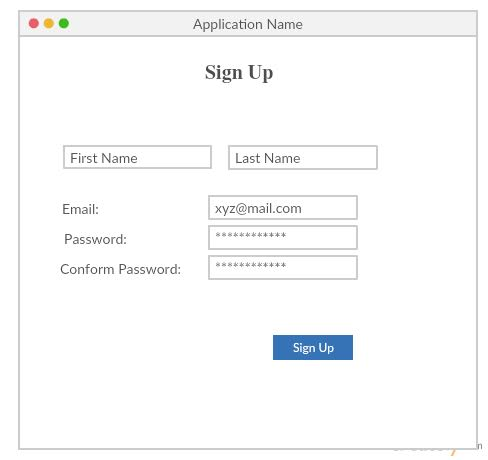
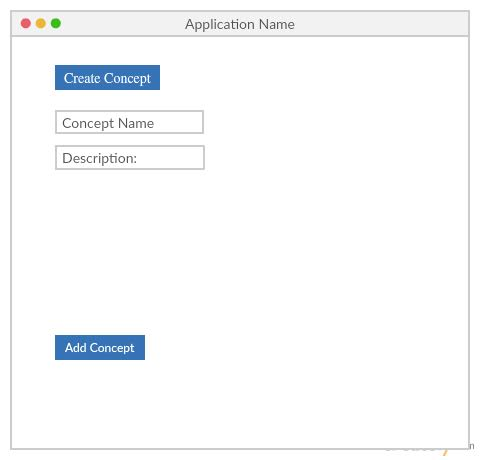
1. **ITERATION 3 REPORT**
2. Services/APIs used:

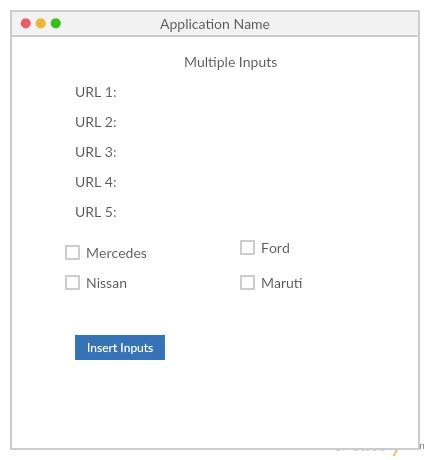
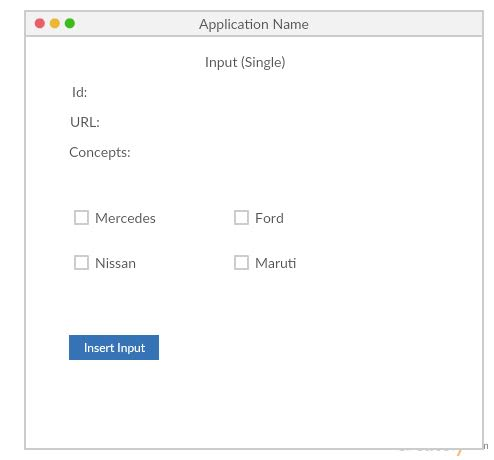
We used the Clarifai API (https://clarifai.com) . With Clarifai, we could upload our own categories of training images (called concepts by Clarifai). Using those images, Clarifai creates a model specific to our unique case for image recognition. It returns a tag for the image along with a percent confidence in that label.

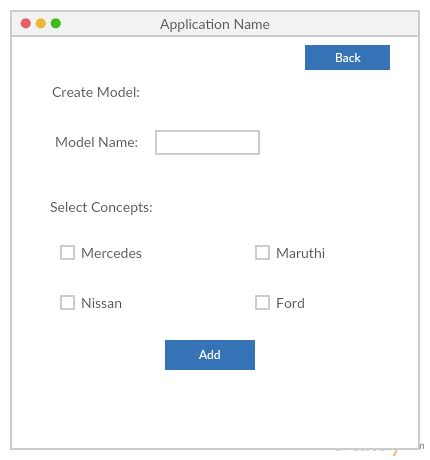
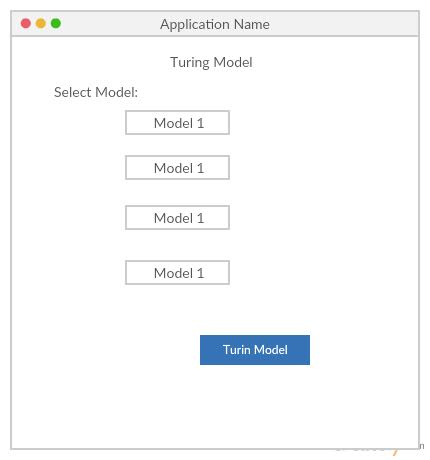
Other technologies:

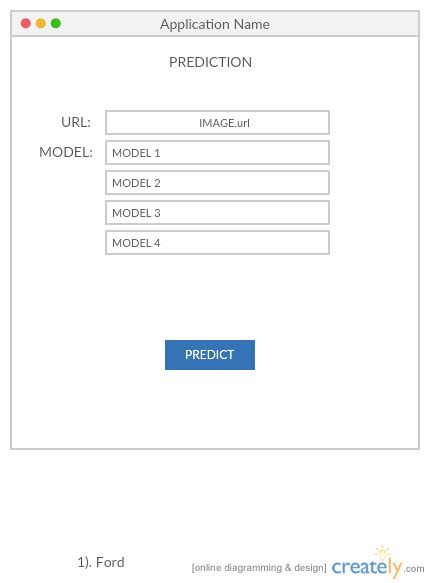
We also will use the Ionic framework for creating a hybrid web/Android application. It will be written in Java with Angular.js, HTML, CSS, Bootstrap.

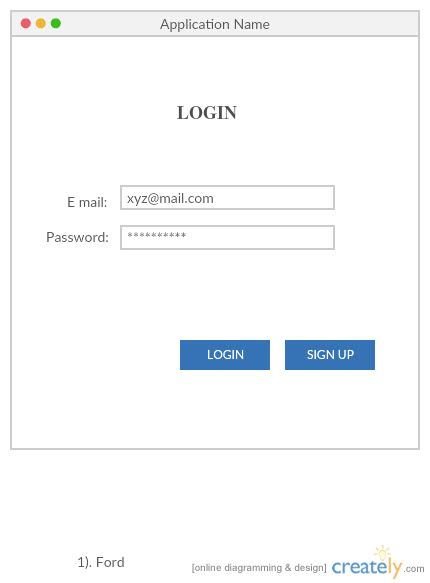
1. Detailed design
2. Wireframes

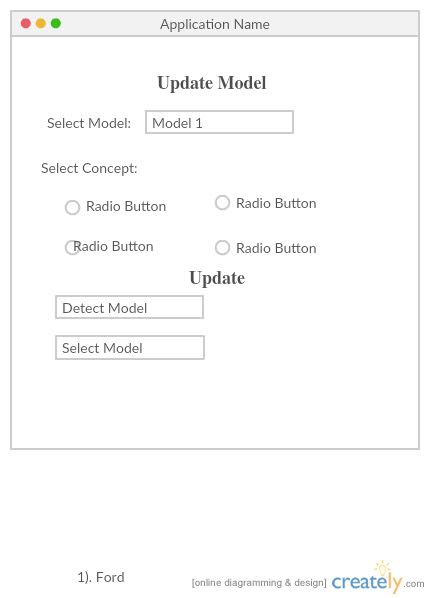




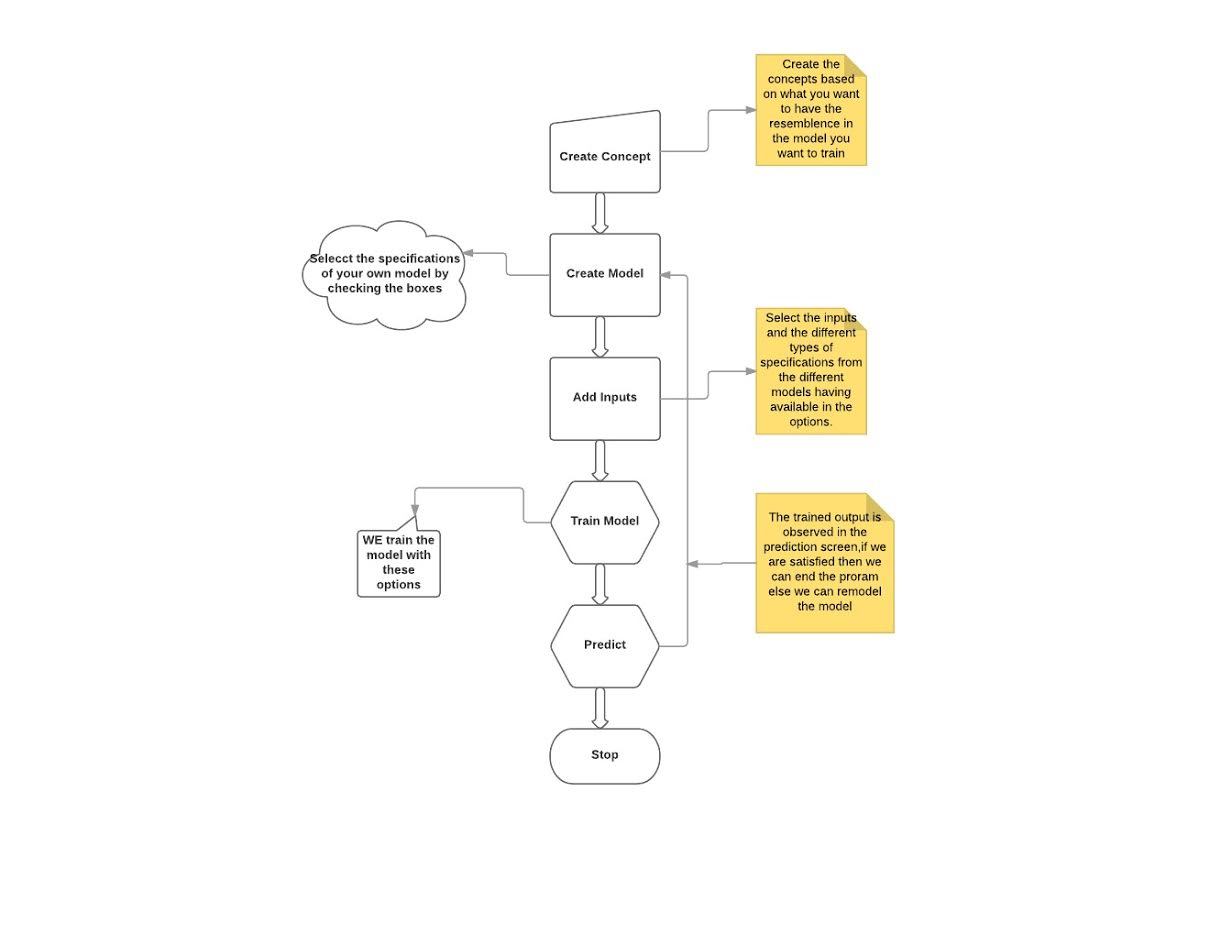








1. UML diagrams
2. Flow chart:



1. User stories

S1. Authentication

The user logs into the system

Cost: low

Risk: high

Value: high  
**Actors:** Clarifai app user

**Precondition:** the user has not yet entered the system

**Description**

1. This use case begins when a user opens up the application and is not logged in.
2. The user is prompted to enter a user id and password.
3. If the user's credentials are authenticated, the use case ends in success with the user reaching the application's functions.

Alternative: an invalid user id and password are entered at step 2. An error message is displayed and step 2 is repeated until the use case succeeds to step 3.

**Open issues:** how many times should they be allowed to enter invalid credentials before being denied access? How would access be reinstated? Logout when the app is closed or after a certain amount of time?

S2. The user takes a photo with their device’s camera

Cost: low

Risk: high

Value: high

Actors: Clarifai app user

Precondition: The user has successfully logged in to the app.

Description:

1. This use case begins when a user has successfully logged in to the app.
2. The user is given an option to take a photo with their device’s camera.
3. The camera opens up and the user takes a picture.
4. The use case ends when the image is successfully captured.

Alternative: The user could potentially skip taking a photo and instead select an item from the device’s image gallery.

Open issues: Add access to camera, storage, and gallery?

S3. The user’s image is identified

Cost: high

Risk: high

Value: high

Actors: Clarifai app user

Precondition: The user has chosen an image to identify, either from the camera or gallery

Description:

1. This use case begins when a user has selected an image to identify
2. The image is sent to the Clarifai API and a prediction is returned.
3. The use case ends when the prediction is returned

Open issues: What happens if the image is identified incorrectly? Does the user have the option to give feedback?

S4. The user is given additional info about the image

Cost: high

Risk: high

Value: low

Actors: Clarifai app user

Precondition: the user’s image has been identified

Description:

1. This use case begins when a user’s image has been identified with a tag.
2. The user is then given additional information with the tag, e.g. a description of the item from Wikipedia or Google Knowledge Graph.
3. The use case ends when the information has been returned. The user is then asked if they would like to try another image.

Open issues: What if the image was identified incorrectly? Do we get this far or is the user asked for a different image?

S5: An administrator adds a new concept to train.

Cost: high

Risk: medium

Value: high

Actors: an administrative user

Precondition: The administrator has selected an option to add a concept

Description:

1. The use case begins when an admin selects an option to add a new concept to train
2. The admin enters a name or label for this concept
3. The admin enters multiple URLs for different images that match the label
4. The use case ends when the images have been used for training and the model is complete

Open issues: is this option available to mobile app users? For any user or just an admin? Or is it only available in a web app?

S6: An administrator creates a model for image recognition

Cost: high

Risk: medium

Value: medium

Actors: an administrative user

Precondition: The admin has selected an option to create a model. The admin has already chosen the concepts for the model.

Description:

1. The use case begins when an admin selects an option to add a model.
2. The admin selects any concepts to be used for the model.
3. The admin selects any configuration options for the model.
4. The use case ends in success when the API develops a model based on the concepts provided.

Open issues: is this option available to mobile app users? For any user or just an admin? Or is it only available in a web app?

S7. An administrator updates a model

Cost: medium

Risk: medium

Value: medium

Actors: an administrative user

Precondition: the admin has selected an option to update an existing model. The admin has a new concept to add or configuration parameters to modify.

Description:

1. The use case begins when an admin selects an option to update a model.
2. The admin selects the existing model to modify.
3. The admin selects to add a concept to the model or alter configurations
4. The use case ends in success when the API updates the model based on the selections.

Open issues: is this option available to mobile app users? For any user or just an admin? Or is it only available in a web app?

S8. An administrator deletes a model

Cost: low

Risk: low

Value: low

Actors: an administrative user

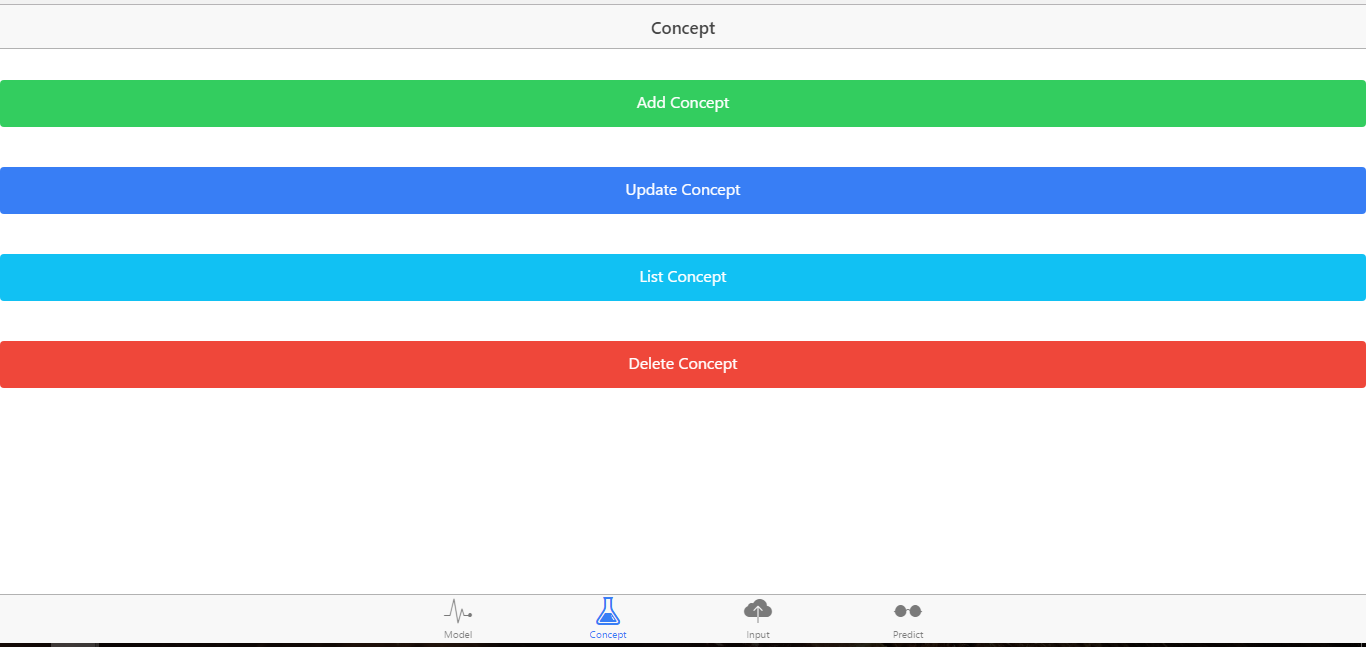
Precondition: a model exists that the admin wishes to delete.

Description:

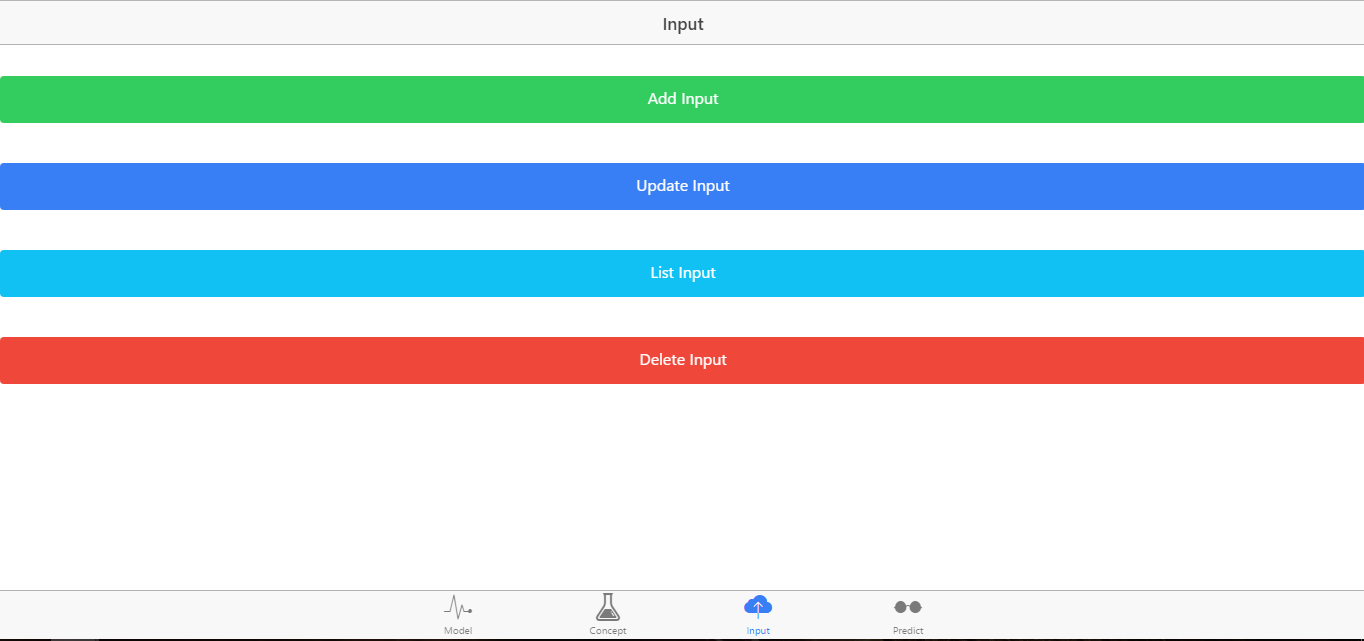
1. The use case begins when an admin selects an option to delete a model.
2. The admin selects the existing model to delete.
3. The use case ends when the model is deleted from the app. The model can no longer be used to identify user images.

Open issues: is this option available to mobile app users? For any user or just an admin? Or is it only available in a web app?

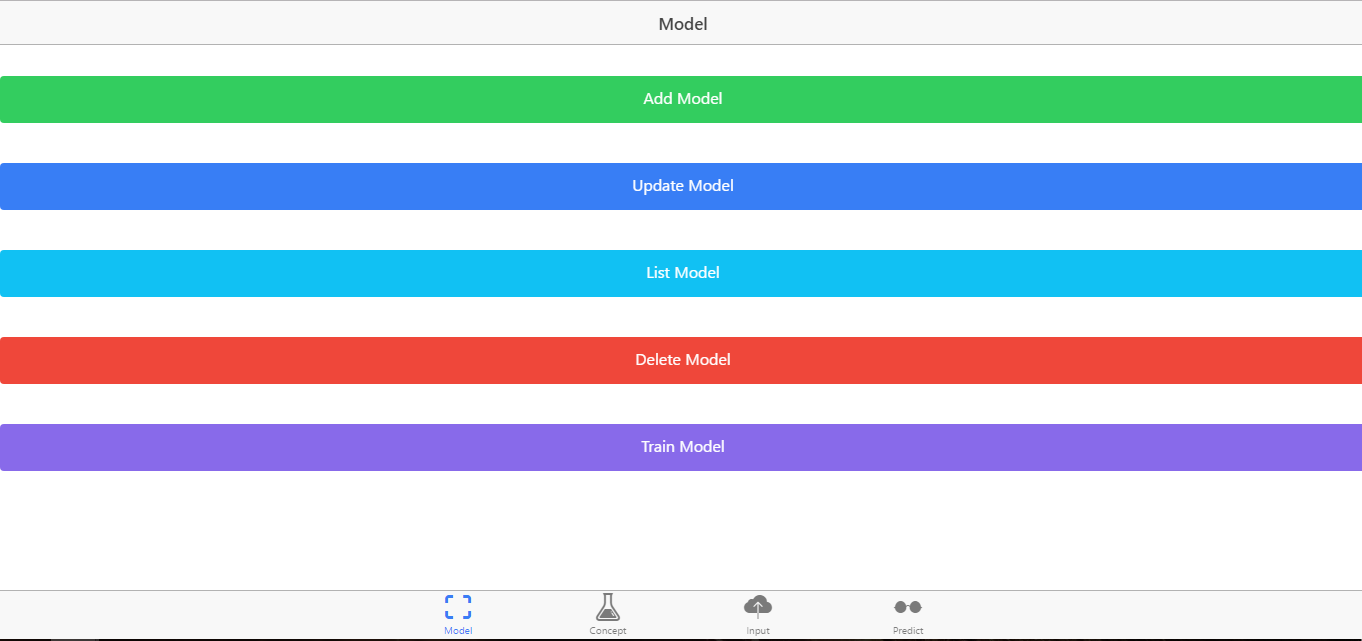
1. Testing: TBD
2. Implementation and Deployment
3. Concept Screen



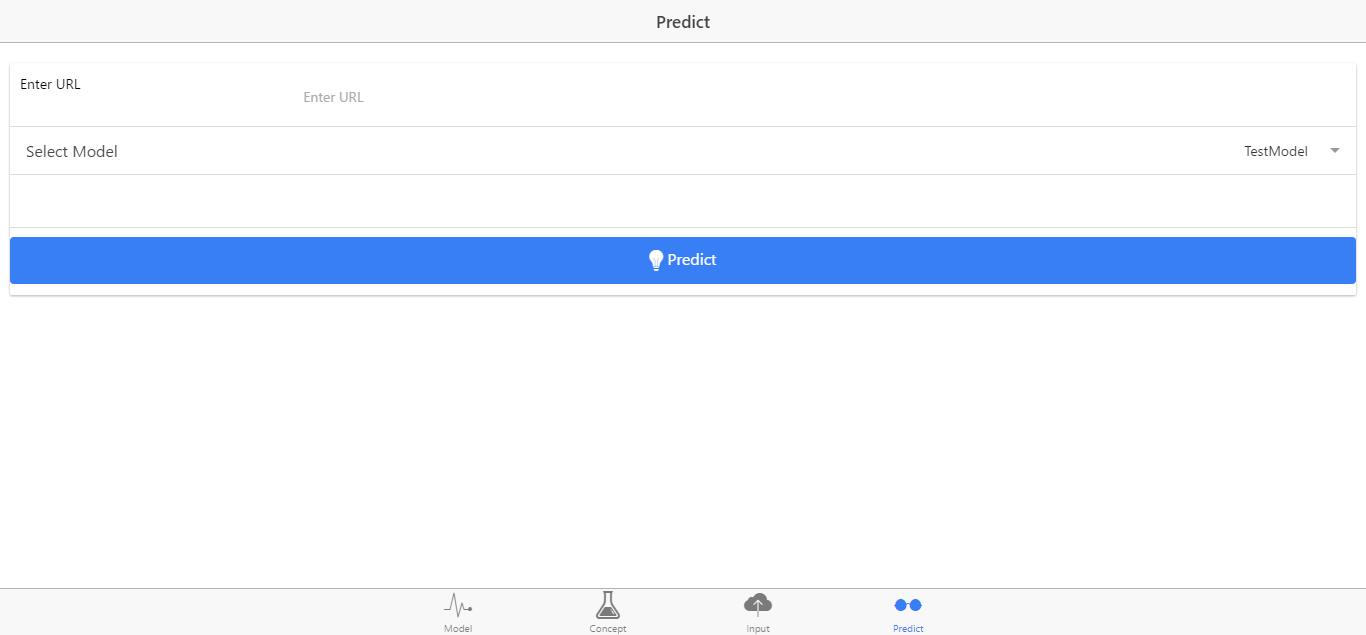
1. Input Screen



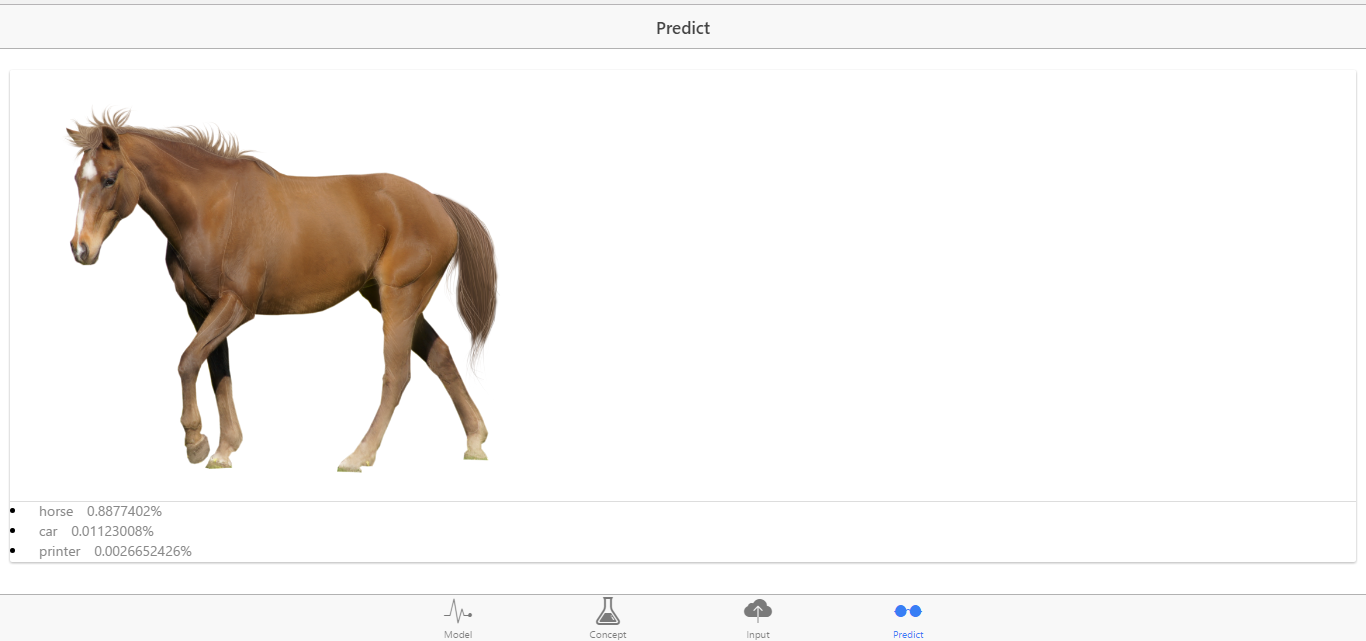
1. Model Screen



1. Prediction screen



1. Final Results



1. Project Management
2. Work completed:
3. The hybrid app was developed using Ionic framework and Clarifai API
4. Three concepts were trained: horse, car, and printer
5. A model was created with Clarifai API
6. Results are returned in the app for the entered image
7. Work to be completed:
8. Add login to app
9. Let users submit own photos through camera or gallery
10. Add new concepts to match user theme
11. Improve overall UI
12. Testing
13. Concerns: getting the app finished in time after starting project from scratch each iteration.
14. **ITERATION 4 REPORT**
15. Services/APIs used

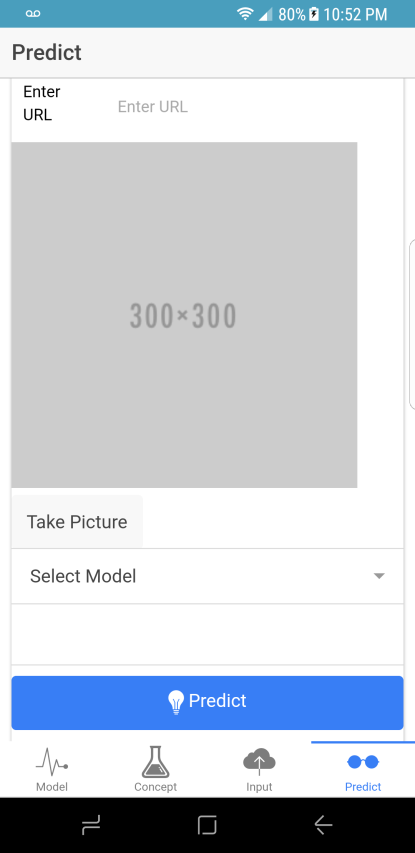
We added functionality that uses the camera for the device to send an image to the Clarifai API. We also utilized the Google Charts API to generate a bar chart showing the confidence level of the results.

1. Implementation: uses Ionic

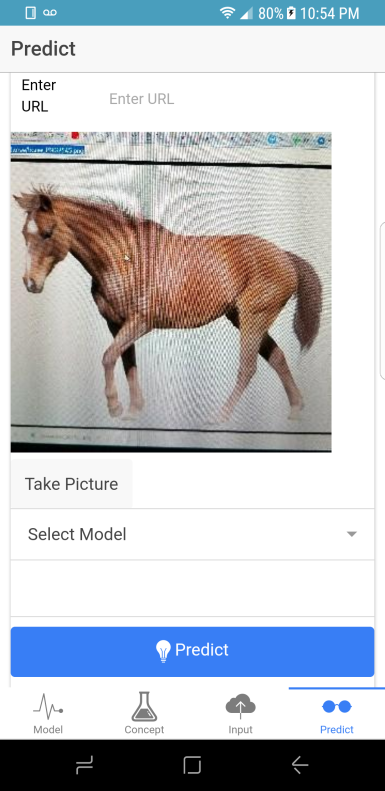
Source code: <https://github.com/shalin51/Hunter/tree/master/Source>

1. Deployment: Updated screenshots of deployment to mobile device

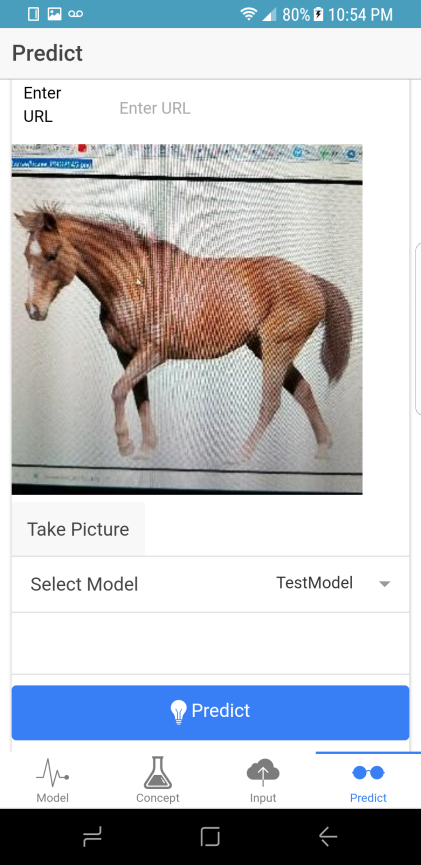
There is now an option to either enter a URL or take a photo under the prediction tab:



After the Take Picture button is pressed, the devices camera will open. On my device, I took a photo of a horse on my computer screen. It is then displayed in the box:



The model is then selected:



After the predict button is pressed, we see the image followed by a prediction of what the image is. Then we see a graph showing the confidence level of the prediction relative to other options:



1. Project Management
2. Work completed:
3. Users can enter their own photo for prediction
4. The top prediction is shown
5. A chart showing confidence levels is shown
6. Troubleshooting with the model/concept functionality
7. Work to be completed:
8. Finish bug fixes
9. Improve UI
10. Make video
11. Make ppt presentation