Outlook

[CS 488T] Sprint 5 Report, Team 11 [stewartc]

From shelbyemailrelay@gmail.com <shelbyemailrelay@gmail.com>
Date Mon 3/10/2025 7:58 PM
To Stewart, Caleb <cstewart15@ewu.edu>

Caleb,

This report describes the activities of your EWU Senior Project team over the previous self-evaluation period (usually Saturday through Friday). It contains only public information. Private information and comments, etc. are available only to the instructor. If you notice any discrepancies or have questions, please contact Dan Tappan at dtappan@ewu.edu.

Sprint 5 Team Report

Team 11: Trademark ID & Analysis Engine

· Lane Keck
· Caleb Stewart
· Logan Taggart

Logged Hours

The team is generally free to work whenever they want during the sprint. The expectation for a team of three members is 45 hours total (15 per member) on average. However, this number will vary throughout the course.

Individual Hours:

		All Sprints							
Member	Hours	Total	Min	Max	Avg[1]	Avg[2]	Std[2]	Count[1]	Missed
Keck	8.0	37.0	3.0	9.0	7.4	7.4	2.2	5	0 (0%)
Stewart	8.5	45.0	2.0	16.0	9.0	9.0	4.5	5	0 (0%)
Taggart	9.0	32.0	3.0	9.0	6.4	6.4	2.6	5	0 (0%)
Team Total:	25.5								

[1]including and [2]excluding missed submissions for required sprints

Team Hours:

Sprint									
1	2	3	4	5	Total	Min	Max	Avg	Std
8.0	27.5	24.5	28.5	25.5	114.0	8.0	28.5	22.8	7.5

The following is optional descriptions of daily work that is not captured as activities below:

Taggart:

- Continued work on desktop app
- Implementing cosine similarity into app
- Continued work on desktop app
- Researching electron.js
- Began on working on altering existing app to become desktop app

Activities

Activities are member-defined units of work that are formally tracked from sprint to sprint (unlike the optional descriptions above). Every activity must be accounted for from its creation until it is completed or abandoned.

New Activities

These activities were created by during this sprint.

Keck

Activity 82: Revise Code in Backend

Implement new approach to comparing logos (two sprints expected)

Stewart

Activity 83: Proposing new Similarity Solution

Find a better way to find similarities between two images (two sprints expected)

Taggart

Activity 84: Changing web app to be a desktop app

Using electron.js to wrap React app (one sprint expected)

Continuing Activities

These activities were continued from the previous sprint.

Activity 76.1: Work on matching upload image

Opened in Sprint 3 by Keck; expected to take two sprints.

Original description: Match an uploaded logo with logo found by model

Progress in Sprint 4 (expected to take one more sprint): We have a way of matching a reference logo with logos in an image. We use cosine similarity and for the most part it works. Now we must test it on a variety of images

Progress in Current Sprint (expected to take one more sprint): We must do more testing and revise our code so we can reduce the amount of false positives

---

Activity 80.1: Create newly designed frontend

Opened in Sprint 4 by Keck; expected to take two sprints.

Original description: Frontend has recieved a new design, must implement it

Progress in Current Sprint (expected to take one more sprint): The new frontend has been revised, although we may need to make further adjustments

---

Activity 88.1: Research embedding algorithms

Opened in Sprint 4 by Stewart; expected to take one sprint.

Original description: Research the best embedding algorithms to use with cosine similarity

Progress in Current Sprint (expected to take one more sprint): Continuing. Still trying to figure out the best embedding algorithms to use and what thresholds should be set for cosine similarity and euclidean distance.

Activity 89.1: Get cosine similarity working

Opened in Sprint 4 by Stewart; expected to take one sprint.

Original description: Be able to detect similarities between to images by embedding the image into the vector, and then applying cosine similarity

Progress in Current Sprint (expected to take one more sprint): Continuing this. Trying to optimize and make this operation more accurate.

Activity 72.1: Setting up back end server

Opened in Sprint 2 by Taggart; expected to take two sprints.

Original description: Get Python and Flask back end to function

Progress in Sprint 3 (expected to take one more sprint): I have got the Python backend integrated and working with our trained model, and have began to setup our API routes with the Flask framework.

Progress in Sprint 4 (expected to take two more sprints): All of the API routes have been setup, and I am currently working on getting the cosine similarity feature to work on it.

Progress in Current Sprint (expected to take one more sprint): Cosine similarity now works but we are figuring out a different approach to take to calculate logo similarity.

Activity 79.1: Implementing cosine similarity

Opened in Sprint 4 by Taggart; expected to take two sprints.

Original description: Getting cosine similarity to be able to recognize a specific logo in an image instead of just all logos.

Progress in Current Sprint (expected to take one more sprint): After implementation cosine similarity alone does not work as well as was expected to recognize a specific logo, so we are currently in the process of constructing a different process.

Completed Activities

These activities were completed during this sprint.

Activity 81.1: Connect frontend with backend

Opened in Sprint 4 by Keck; expected to take two sprints.

Original description: After design is complete, make it work with the backend

Progress in Current Sprint: The frontend design does work with the backend

---

Activity 78.1: Restructuring Python backend

Opened in Sprint 4 by Taggart; expected to take one sprint.

Original description: Getting back end file structure to be more organized and easy to understand.

Progress in Current Sprint: Backend file structure has been restructured and functionality has been made more modular.

Team Reflection

This section refers to the team's collective perception of and reflection on the project over this sprint.

The instructions are: Consider the following four pairs of questions hierarchically. They are not the same question. If you think they are, then you are likely not using an appropriate breadth and depth of software-engineering thought. This course is a practical application of the aspects of product, process, and people. We are trying to account for everything: not just to create a good product, but also to learn from the process to improve the people. Reflect on the experience of the entire team collectively over this sprint. You do not need to account for all work, just two examples that are most representative of easiest and hardest.

For reference, *understand* relates to the comprehension of what needs to be done; *approach* to how you think it should be solved; *solve* to implementing the actual solution; and *evaluate* to demonstrating to yourself and your team (if applicable) that the performance of your solution is consistent with everything else in the project. Remember The Cartoon from CS 350.

Understand

Easiest: The easiest aspect of the current work to understand is what our current goal is, and the requirements that we need to complete for it. It is clear to us that to achieve the goal we set for ourselves of finishing the still image portion of the project before the end of the quarter, that the remaining task that we still have to do is getting our program to correctly match images based upon a reference logo.

Hardest: The aspect that has been the hardest to understand is why our current approach of using a single embedding feature extractor alongside Cosine similarity is not working well for the purposes of matching images together, as they are using very complex algorithms that we do not 100% understand. This has made it difficult to debug/redesign as we are not sure where exactly it is failing to work as we intended.

Approach

Easiest: The easiest to approach aspect has been repurposing the code that we've been using to test out different embedding algorithms and similarity measures from our Jupyter Notebooks into working with our actual project. This has been easy to approach because we already had designed our application to be very modular, and easy to make feature changes to without breaking the functionality of the past methods that we have implemented.

Hardest: The aspect that has been the hardest to approach has been developing the front end as none of our team members have lots of experience/knowledge in do such, and we have had to quickly figure out how to implement the different components of it and where to start learning what we need to in order to be able to make it function and look good.

Solve

Easiest:	The aspect of the current work that has been the easiest to solve for us has been figuring out how we will handle the issue of the backend not working properly when hosted due to the limitations of the hosting service being free. We decided to instead pivot to a desktop application as when we were researching the issue we found that we could use Electron.js (which uses Chromium and is well-documented) to wrap our current application with very minimal changes to our current codebase.
Hardest:	The hardest aspect to solve has been how we are going to implement a method of matching logos that goes beyond just the use of Cosine similarity as that has not provided us with adequate enough results. As a result of this, we have had to complicate the solution much more than was originally anticipated to go from using 1 test to using 6 tests.

Evaluate

Easiest:	The easiest aspect to evaluate is whether or not we are getting a correct result of two logos matching. We are able to simply view whether or not the program has identified a logo that we know is or is not a match to the reference logo that we provided it.
Hardest:	The hardest aspect of the current work to evaluate is whether or not the similarity thresholds that we have set for each combination of an embedding algorithm (BEiT, CLIP, or ResNet) and a similarity measurement (Cosine or Euclidean) are providing accurate results and minimizing false positives. We need to perform lots of testing on different types of logos/images to find what will produce the best results for each pair.
Completion:	35%. We feel that with the current pace we are on and with the progress we are making each week, that success seems very likely.
Contact:	N/A
Comments:	We have no comments, concerns, or issues at this point in the project.

Report generated on Mon Mar 10 19:58:25 PDT 2025