Milestone 2: Image Processing

Softwin Enginwinning

CSC 340

Isaac Wayne, Joseph DeJong, Cheryn Lindsay, and Tyler McGrath

Introduction

We are constructing a computer vision model to recognize EVERY image. Our goal is to create a program to respond with a word to recognize the image.

Executive Summary

We are building an image recognition application. The purpose of this project is to provide a robust image processing service to our users. Our estimated time is 25.75 hours. This project actually took us 26.25 hours. The cost of this project is based on our $25/hour (not including tips) is $656.25.

Project Organization

Our team is made up of four of the top 10 programmers at Concordia College (numbers 1, 3, 6, and 9). Isaac is the team leader, Joseph is the project manager, Tyler is our lead developer, and Cheryn is head of UX/UI.

Risk Analysis

There are several risks associated with project development. It could be difficult to amass the number of photos needed to make a workable project. Thousands of photos are needed for just one animal.

Hardware and Software Resource Requirements

As developers. For a computer program, we will need computers. We will additionally need software that we can use to code this project.

Work Breakdown
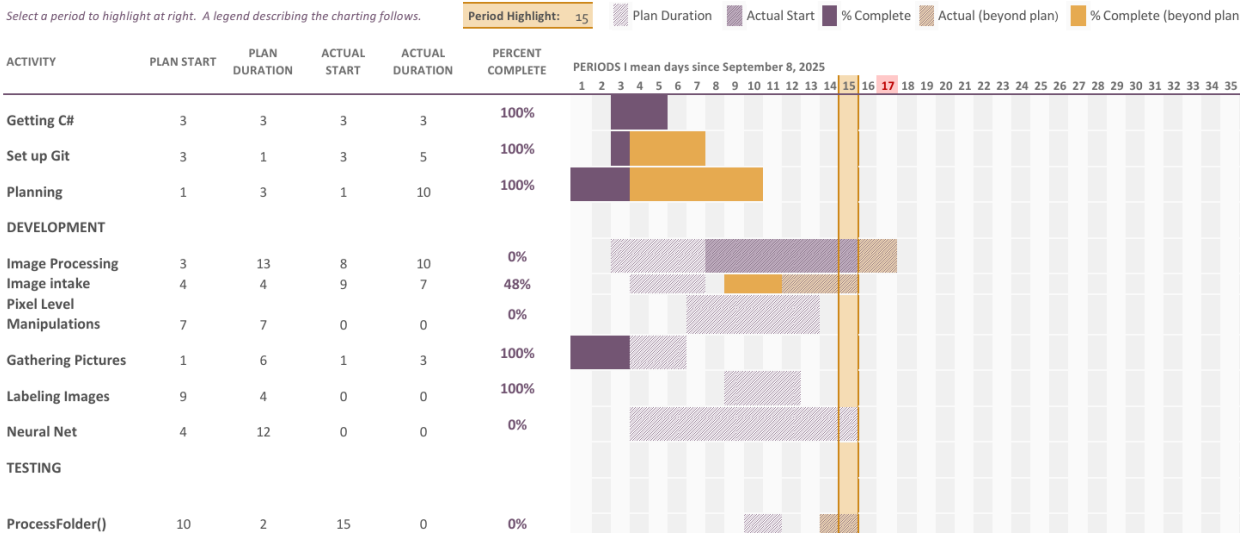
Overall time and price estimate:

Task Name	Description	Estimated Hours	Actual Hours
Milestone 2			
Step One	Complete Planning Document	8	6
Step Two	Time and Budget Estimates	4	4
Step Three	Good Software and Ethics	2	2
Step Four	Configuration Management	3	3.5
Step Five	Software Processes	2	4
Step Six	Component Implementation	6	6
Step Seven	Presentation	0.75	0.75
Total			26.25

Project Schedule

We will be splitting up tasks between the group members by their respective roles.

Milestone 2: Computer Vision

Period Highlight: 15

Plan Duration · Actual Start · % Complete · Actual (beyond plan) · % Complete (beyond plan)

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Getting C#	3	3	3	3	100%
Set up Git	3	1	3	5	100%
Planning	1	3	1	10	100%
DEVELOPMENT					
Image Processing	3	13	8	10	0%
Image intake	4	4	9	7	48%
Pixel Level Manipulations	7	7	0	0	0%
Gathering Pictures	1	6	1	3	100%
Labeling Images	9	4	0	0	100%
Neural Net	4	12	0	0	0%
TESTING					
ProcessFolder()	10	2	15	0	0%

PERIODS I mean days since September 8, 2025

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35

Good Software and Ethics

This section contains the notes from a group discussion about good software, ethics, and best practices.

Good Software:

- Maintainability Joseph

Our features should be well documented. It should be easy to understand and should be bug free. We will make sure the program is able to be understood by any developer.

- Dependability and Security Joseph

Our image processor should be able to work with large numbers of images. It should be able to handle differing folder structures and image types. Security should be to the highest standard. Images should be processed and handled locally, never leaving the user's device. This ensures no data leakage.

- Efficiency Joseph

Our code should be efficient to require the least possible amount of time. Time complexity should be limited and nested loops should be avoided when possible.

- Acceptability Joseph

Our software will be acceptable if it reaches a state that performs the desired task well.

- Dependability and Security

The features we develop should be dependable, meaning our image processing users should be able to use our program to complete the desired task without running into issues. As for security, our team must be aware of the image libraries we use to train our image processing program and ensure they are suited for the task. Additionally, we must have a way to ensure that files submitted by users for identification do not compromise the utility of our program.

- Efficiency

Since we have created our features from scratch, we can choose to implement only what we find necessary, unlike in a library made by others. Our tradeoff for convenience in the programming is the promise of making something better ourselves.

- Acceptability

Our project and features are acceptable as they function as designed. Furthermore, there should be no moral or ethical issues in the use and creation of this project.

Ethics:

- Confidentiality

Our project will be publicly available. We maintain a transparent view of our project, and do not intend to hide anything within sensible means.

- Competence

Our project is very competent as it fulfills the requirement of working correctly and without any major issues. Furthermore, each one of us in the group has demonstrated our competence by contributing to this project in strong and meaningful ways.

- Intellectual Property Rights

One of the challenging parts of making this project entirely from scratch is gathering enough images to create the training libraries for image identification. Luckily, each of us has taken a deep learning course that involved creating our own training set for Niblet identification images. Without this, we would have to be very certain that the images and code we use do not infringe upon someone else's intellectual property and avoid any potential legal disputes that may arise from the development of our program.

- Confidentiality Joseph

Our project should protect the confidentiality of all users. Images submitted by the users should remain on device and should never be stored on a server. Additionally, there is no reason to collect any data about our potential user; that would risk their privacy.

- Competence Joseph

Competence in the scope of this project is really just about acknowledging what we know and don't know. If we don't know how to do something yet, we should ask group members rather than simply not doing it.

- Intellectual Property Rights Joseph

When training any AI things, it is super important to recognize the existing IP rights of any data we use. This won't be a problem for ours specifically because we are using images we've used in another class and were all taken by Concordia students.

- Computer Misuse Joseph

Computer Misuse is common with software not prepared for it. To prevent computer misuse, our group should focus on filtering images that come into our training model and images that are submitted by users.

Configuration Management

This section was completed on GitHub. After changing the scope to be completely team-based, this will have a minimal impact on the future development of our project.

Software Processes

Screenshots of our project's GUI will go here, implementation will be through GitHub.

■ ComputerVision   ⬓ ⬜ ✖

Welcome to Computer Vision!

Upload your image here: Submit

Output:

Computer vision will identify the user's image and name it here!