Introduction

Welcome to the start of our blog for CA326, this blog will give you an overview of our development process, decisions made and how things progress over the weeks and months of development.
We decided to start this blog now, at the beginning of the new year because we have moved into more of a structured and busy development cycle, in 2022 when we were just starting the project we had lots of things happening and were very busy so the work we could on our project was sporadic and mostly consisted of research, reading, design and small proof of concept scripts. Now as we move into 2023 we have all our exams, assignments, Christmas and new years behind us, we have decided to start off this year strong and begin working on the project more full time, and try and get a strong meaningful start to the work we have to do.
This blog will help you understand and see how we are getting along throughout our development of the project. We have decided to do weekly updates for the blog alternating between the two main parts of the project - Computer Vision (Gareth) and the Webapp (Josh). The entries will describe what we've done since our last update, any decisions we might have made and then what we plan to do next. This blog won't go into complex detail about the code, how things work or what the project is, it will be a broad overview of our thoughts and decisions throughout the development.

Week 1 (6/1/2023) - Gareth

For this first post, I'm just going to give you an overview of where I am right now with the computer vision component and how things are coming together.
For the computer vision component most of the research so far has been done using OpenCV, last year I focused a lot on this and how it works, I managed to get some simple scripts running using OpenCV that allowed me to either scan a still image for faces and then more recently I got a webcam stream with rudimentary facial recognition working, the script and size of the component we are designing should be pretty small, and the scripts so far have been quite small and easy to read, the complexity I have been facing is in the dependencies and linking things together such as the webcam, for a long time I was struggling with the graphical display components because I was developing on Windows-Subsystem for Linux, I have since moved to a Linux VM so I could get it working on Ubuntu.
After getting some of those proof of concept scripts working I have moved on to how we are going to assemble the component, after some discussion we believe that running the CV script on a flask app contained within a docker container is the way to go, it will allow us to make the script portable, and not have to worry about the dependencies associated with OpenCV. This has been where my focus has been this week, as I have been reading online and learning how to use both Docker and Flask, so I can combine them with OpenCV. So far the work has been going well, I have managed to combine flask and OpenCV so that I can display a webcam stream via the Flask app on localhost, I have also managed to combine flask and docker, running a simple flask app in a docker container. Combining all three has been the hardest part, and I am not quite there yet, I am having some trouble with

dependencies and getting OpenCV to work while in a docker container, but I have made promising progress and have lots of scripts and prototypes made, this will be my focus now for next week, to get the three different parts working together so I can use the camera with OpenCV via a Flask App, inside a Docker Container. Once I have that working I can start to flesh out and improve the Flask app, and the computer vision script to be better than my simple prototypes.

Week 2 (13/1/2023) - Josh

For this second post, I'm going to give an overview of where I am with the website component of the project and how it’s going.
For the website component, a lot of the research was based upon the process of designing prior projects we had done with Django as a baseline and then doing research on how to expand those to accommodate the current project. I began by reading through the old notes we had and readings we had been given with regard to Django and Node.JS so that I could familiarise myself with them as it had been a while since we used them. Once I felt comfortable with these last week, I began working on the Django models, doing some small tests to make sure that things were running smoothly as it went along.
Once the models were done, I got the serialisers and views for the Django part of the project running smoothly. This part of the project has been going smoothly overall but took slightly longer than anticipated, so has run over into this week. I plan to get the node frontend server up and running and working interactively with the Django backend this week, so that in future weeks I can move on to implementing a solid website design using principles we have learnt in the past, and also implementing the timetable into the site, as this is a key component of the website’s functionality.

Week 3 (20/1/2023) - Gareth

Development and Prototypes are coming along well for the computer vision microservice, after having our first meeting of the new year with our supervisor Michael we were a lot more clear on our focus of what needed to be done, we also agreed to aim for a simple demo of the system by the 30th of January. Since my last blog, a lot of exciting things have been happening. To solve the issue of a camera source I was able to use javascript to grab a media device from any browser, and this solved a lot of issues with the platform. Then I was successful in using WebSockets to link together the front and back of my flask app, sending captured pictures from the browser in the form of data URLs, and then running facial recognition on them and sending them back to the display. I was then able to further slice up and crop the image in the backend, identifying how many faces were detected, and making a small thumbnail image for each, using this small image I was then able to implement eye detection using the same method as our face detection but with a different haar cascade available from OpenCV. After some discussion with Michael, I also implemented using a pre-trained model the ability for the script to detect emotions from each small thumbnail, unfortunately, I am unsure if it is working correctly, it only detects surprise and fear, no matter what wacky facial expressions I make when testing, further discussion and testing is needed for that aspect. I then began to improve the code behind the scenes and also do some work on the current display, I added data trackers for things like the number of faces detected,

captures taken and average faces detected over time, the flask app is now looking very good and promising for the simple demonstration we are aiming for soon. At the start of next week, we will meet again with Michael and discuss progress and how we will continue. My priority is next to integrate my flask app into the larger Django app that Josh has been making. Once we can integrate these two main components together, the rest of the project will be styling and fine-tuning, adding features to our webpage, and hopefully tuning and improving the CV scripts.

Week 4 (27/1/2023) - Josh

I've made quite a lot of progress over the last 2 weeks on the frontend, which I'm content with going into our demo with our supervisor Michael next week on the 30th of January. I got our basic node frontend server set up soon after my last blog, and have been adding functionality to it since. I set up pages for our basic homepage, logging in and registering an account, viewing modules, and viewing lectures for a module. The next thing I was working on was setting up the lecture page to only show lectures for a specified module, which I was able to figure out pretty quickly using a fetch request and comparing the module IDs of the lectures being checked with the ID being passed to the fetch to ensure only lectures from that module would be displayed. Next up was filtering only modules assigned to the current user, which was a bit more difficult to set up as I needed a way to check the user. I decided on saving the username into local storage when the user logs in to their account, which would then be deleted once they had logged out, and filtering the module list using this username.
My most recent testing has been to see if I could get a simple flask application to work when called from my frontend, as this would then set the foundation for Gareth to be able to link his app and pass along any data he may need to either side of the transmission. I got a simple one working yesterday which allows for some simple data to be passed back and forth to the flask app, which should be a solid base for Gareth to work from. Next, I'm going to be attempting to implement adding lectures and modules to our system from the node page over the weekend before the demo, as this would be a really nice proof of concept to be able to show as we prepare to add more functionality.

Week 5 (3/2/2023) - Gareth

Week 5, development has accelerated and is going really well. We are very happy with the progress and want to keep up the pace so we have plenty of time near the finish to clean up and wrap the project up. Development for the Computer Vision component has gone very well, I have a full flask app up and running that can detect faces, eyes, and side profiles (turned heads), and can also now analyse what emotion a face has via a pre-trained model. All these parts of the app have been turned into data points that will in turn feed the reports we will generate for each lecture analysed. I worked on the flask app to have it polished enough for a demo we did for Michael, he said it was great to work and we are well on track, since that demo I have paused development and refinement on the app and have moved to help Josh on the web app side, doing styling and integration. Before the demo Josh and I worked together to integrate the flask app I had developed independently into his Django Node framework, we were successfully able to host it in the backend via a subprocess and

link it into the frontend web page via an endpoint leading to the backend. The project is looking very good now. Josh and I have moved into working closely together on the system as a whole, Josh continues to focus on the models and data backend, whereas I have been working on the frontend styling and javascript. We have boilerplate features like login and registration working as well as a lot of the core functionality working. We are now working on some of the final and most important parts of the web app including the timetable view, lecture management and generating reports from the data we get from flask. Overall I am happy with the pace we are working and think we are looking good for finishing in time and without much stress or time pressure.

Week 6 (10/2/2023) - Josh

Since my last blog post, I haven't gotten as much progress as I'd like due to focusing on the in-person module we've been attending for the past 2 weeks. This took up a considerable amount of time, but I was still able to get a fair amount of work done and am preparing to ramp up over the next week to make sure we have enough time at the end for testing and reviewing what we have done, along with writing up all our documentation.
I managed to implement adding new lectures to a module and adding new modules to a user through the node frontend right before our demo with our supervisor Michael, which proved to be a good demonstration of how our frontend and backend come together, along with helping Gareth integrate calling the flask app into the frontend.
The next thing I worked on was researching a way to handle JSON data in a model field, which led to me discovering a package known as 'django-jsonfield'. Unfortunately, this package turned out to only be supported up to Django 3.7, while we used Django version 4.0.4, so I had to find another method. I settled on simply having a text field that could take in the assortment of data from the flask app in a single field, and then it could be manually parsed by the frontend fetch request. I've been doing some searching into how to add the functionality for removing lectures and modules from their respective models. So far I think I'll need to use a DELETE request, as it appears to accomplish what I'm looking for, but I haven't started development yet so I will have to see how that works. I've also been looking into developing the timetable, and how I would go about filtering all the lectures a user has, and then sorting them in order. This seems like it would be a rather straightforward process, albeit with a lot of nested fetch requests, so I'll need to see how it works out in practice.

Week 7 (17/2/2023) - Gareth

We are just entering the final week of development now before submission. Over the last two weeks, I have been doing a lot of styling, formatting and generating of the website's front end. Doing a lot of Javascript and CSS to get the website looking nice, and generating elements from the data in our backend. As I am writing this I have pretty much finished my part of the development, there is only 1 more major feature to implement which is the timetable that Josh is working on. I have successfully implemented the data reports and aggregated module statistics which I am very happy with, the data we extract from our flask app now is processed and presented nicely for the end user to review in reports. Also during the last two weeks we submitted the ethics form, we had to go back and forth with Michael to make sure it was all correct, we have a plan in place to do anonymous surveys on the

design of the website and also user testing of the camera functionality. Unfortunately, I am now concerned we submitted it too late and won't have time to do any testing with users before the due date, this was an oversight on our part and we should have really submitted it earlier. Other than that we are looking good for the deadline on Friday, I am about to begin our documentation, outlining what we must do for the technical specification and user manual. Then during this week, we will do some final tests, bug fixes, cleaning up and populating the database with dummy data for our demo. Our time schedule is looking good, we will be working hard all week but it is achievable to get everything done that we hoped we could.

Week 8 (21/2/2023) - Josh

It's the start of the final week, and a lot of progress has been made since then. I implemented the deletion of modules and lectures through a DELETE request like I was theorising in my last blog post, which at first seemed to work well. Unfortunately, we soon discovered that this did not work for non-superuser accounts, and we were getting errors when we tried on other accounts with fewer permissions. I did some more searching around online and found some examples of people using post requests that simply specified the particular object and then removed them. This implementation worked on all accounts I tested with (our superuser, an old non-superuser account, and a freshly registered account), so I've determined that this bug has been resolved.
While I was working on this, Gareth noticed that the add module request was failing occasionally, and it was determined it was only failing on newly registered users, rather than older accounts. I realised that I was only saving the username to local storage when the user logged in, and not when they registered the account, so adding that to the registration process quickly resolved this. I'm glad we sorted out these bugs now, rather than them potentially causing us issues during our final demo or presentation.
Over the past few days I've gotten the timetable page working, and have the lectures displayed in order and sorted by days. The final functionality that needs to be added is simply being able to start our computer vision application for a particular lecture from the timetable, which simply requires me to pass enough information through to it in the fetch request for the lectures to allow the flask app to narrow down which lecture it should assign the record to. I think I should be able to get this up and running by early tomorrow, which would leave some minor styling before our website's functional programming is completed in time for us to do some reviewing and documentation.

Final Blog (23/2/2023) - Gareth

Well, here we are at the end of the road. We are very close to completing our project and submitting it. Since last week there have been a lot of long days and many commits to our repo, but as I am writing this on the day before submission I think we have it all under control. Last weekend I completed my last points of development on the website, finishing up the aggregated reports and styling of the UI. Josh was finished with his major development by Tuesday and wrapped it all up yesterday. The last week has been a lot of documentation and writing. We are finished now with both the Technical Spec and the User Manual and have sent both off for review by Michael. We also got some good insight on Monday during

our last meeting with Michael on the best way to make and what to include in those documents. Other than documentation we have been doing a lot of cleaning up and finishing touches, changing READMEs, clearing out unnecessary files and structuring the repo how it is meant to be. We also were very happy to receive full ethical approval on Tuesday evening, because of that we were able to get some User testing done, we managed to organise and complete two sessions of testing on Wednesday evening, one with family to test the camera functionality, seeing how it dealt with multiple faces and participants being further from the camera, the second session was with friends and was focused more on the design and UI of the site, we got some good feedback there and managed to implement it into our system this morning. Our tests didn't have the widest sample size or the most ideal conditions but we are happy that we were able to complete them, and get some testing done because we know how important it is for big systems to go through testing. All that is left now is a lot of clean-up work and final touches; commenting on important code, populating the database, recording our video walkthrough and then making sure everything is in the repo for the deadline, we are very close to the finish line now and will be very happy when all our hard work is finally submitted and finished.