



# DriveSafe

Project Engineering

Year 4

Aleksander Kosiak

Bachelor of Engineering (Honours) in Software and  
Electronic Engineering

Galway-Mayo Institute of Technology

2021/2022

## Declaration

This project is presented in partial fulfilment of the requirements for the degree of Bachelor of Engineering (Honours) in Software and Electronic Engineering at Galway-Mayo Institute of Technology.

This project is my own work, except where otherwise accredited. Where the work of others has been used or incorporated during this project, this is acknowledged and referenced.

Aleksander Kosiak

## Acknowledgements

I would like to thank Simon Gaborit & Vladimir Mauer for the helpful advice along the way of creating this project.

# Table of Contents

- 1 Summary..... 5
- 2 Poster..... 6
- 3 Introduction ..... 7
- 4 Libraries – Backend - Models..... 8
- 5 Project Architecture ..... 9
- 6 Project Plan..... 10
- 7 Development ..... 11
  - 7.1 Full Stack..... 11
  - 7.2 Code Snippets..... 12
- 8 Ethics..... 13
- 9 Conclusion ..... 14
- 10 References ..... 15

## 1 Summary

During my time doing work experience in Xperi, I was helping develop and fix features of an overall face detection software. Getting more comfortable with the software and code, I decided to implement what I have learnt into my final year project. Especially being in an automotive oriented company, I also decided to pursue an in-car face detection software. The goal of the project is to grab emotion from a user driving and upload that onto a website, which saves the emotion and guides the user on actions they may take based on the emotion recorded. My intention was to create an emotion face detecting screen using my webcam first. Then creating a backend and frontend using Django & Next JS. The face detection is created from Python in the VS Code IDE with multiple libraries that are commonly used for face/image detection and manipulation. I was able to create the software itself, but the frontend and backend was a challenge itself, as I have not much experience in that field. In the end it was a good challenge to test myself to see what I can incorporate from what I've learned before and what it takes to get into such a working environment for the future.

## 2 Poster

# DriveSafe Poster



Department of  
Electronic and  
Electrical Engineering

Aleksander Kosiak – Software & Electronic Eng.

## Summary

Driver emotion feedback using face detection application. Gives recommendation based on emotion on style of driving. Uses a region of interest to draw a box around the face and displays the emotion above the face. Sends the emotion and image capture to a Next JS based website.

## Libraries Explained

OpenCV – Real time face detection  
Numpy – Image array manipulation  
Keras – Image information extraction, deep learning models



EMOTION DETECTION

## Hardware & Software

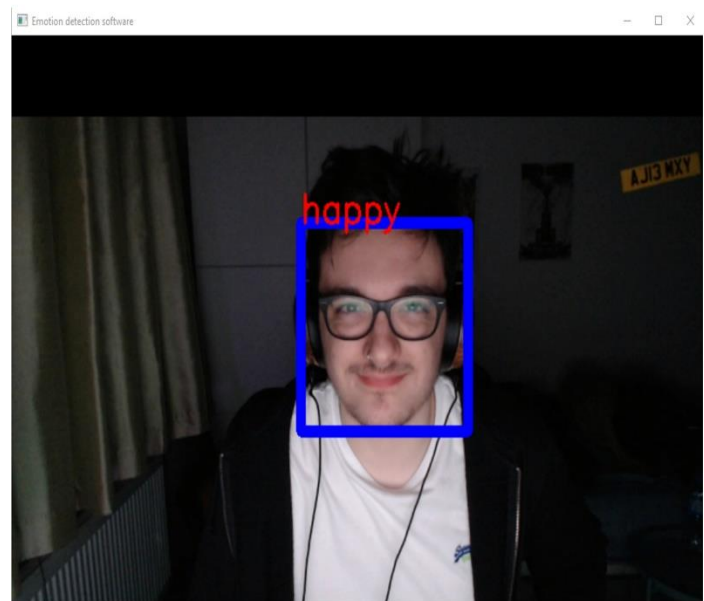
Logitech C920 HD Camera  
Logi Capture  
Python using VSCode  
Libraries:  
OpenCV  
Numpy  
Keras/Tensorflow

Frontend: NextJS  
Backend: Django

## Inspiration

Having previously done work experience in Xperi, based in Galway, taught me much about face detection. This inspired me to look for a missed opportunity in the market that could change the automotive market in the health and safety department that's always evolving.

## Results



### 3 Introduction

This report will outline all the technologies used for the project to work. The goal of the project is to showcase emotion detection on a face, using a modern code language as Python. Also, the use of libraries for image manipulation, and how they can be used for future applications in various industries, such as automotive or health and safety. The idea and motivation came from working on automotive face & object recognition in my work placement, having learned the common software and use of libraries.

## 4 Libraries – Backend - Models

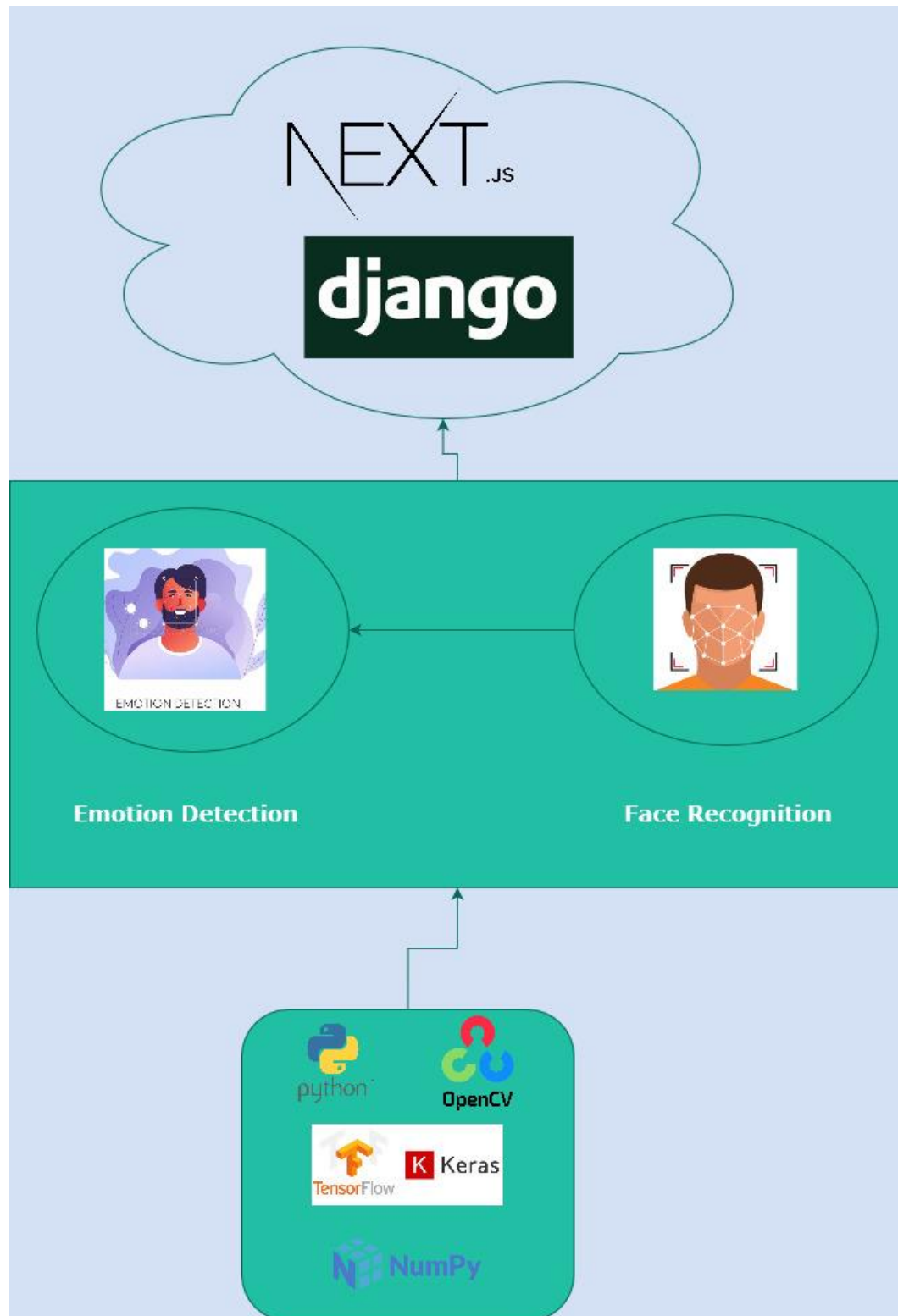
There were multiple libraries and a backend used in my project that needed extra research as they were either not taught in class or were not covered extensively. These were:

- OpenCV – Open CV contains pre-trained classifiers for faces so it's ready to use out of the box, as face detection uses Haar cascades which relates to machine learning, that need to be trained to with a set of input data. [4]
- Numpy – Library used to manipulate arrays of images.
- Keras/Tensorflow – Used to import a pre trained model. Converts real time image to array so it can be manipulated. Emotion is then predicted at the end using the manipulated array of the image.
- Django – Open-source backend framework used to create a secure and maintainable website.

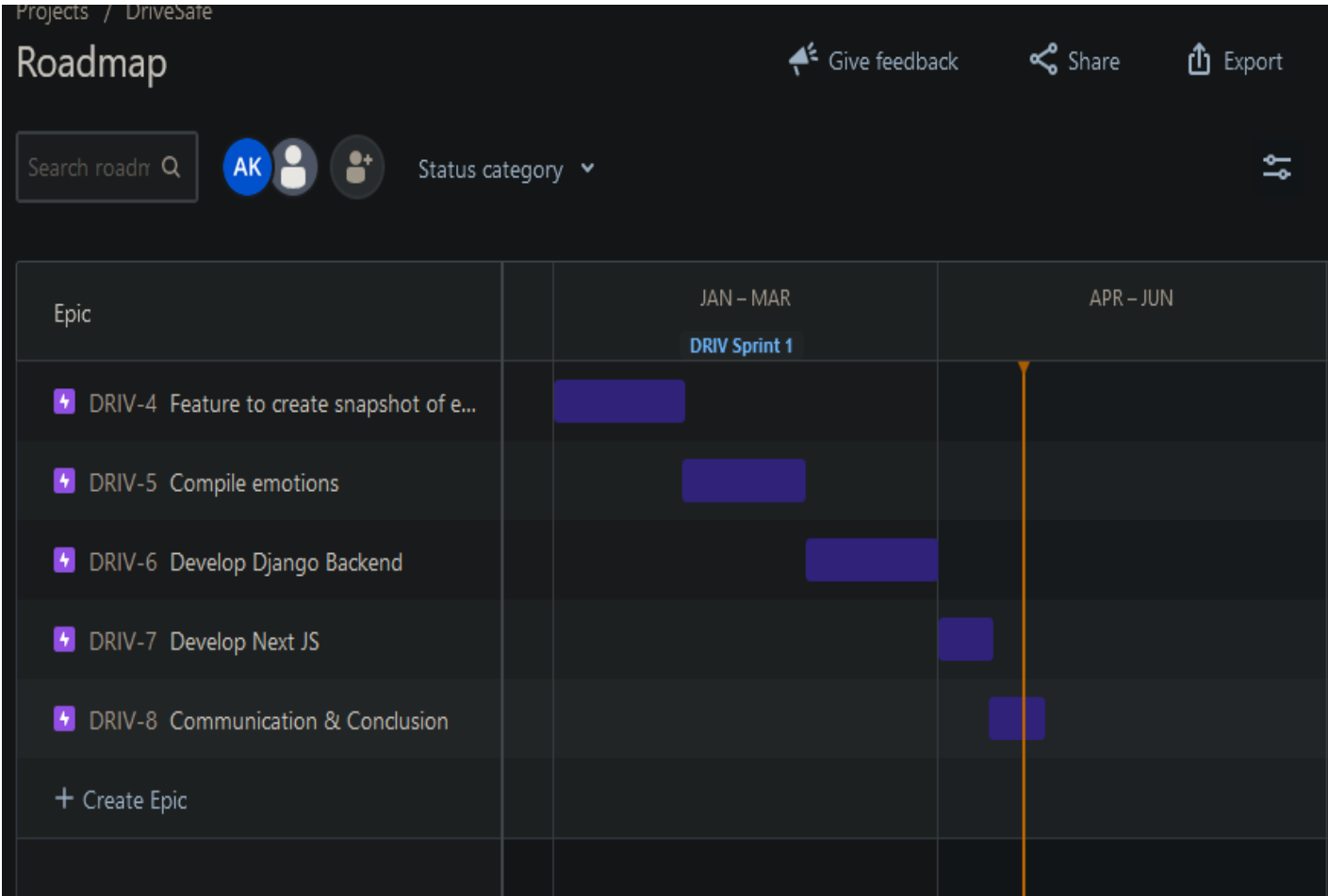
I also needed to research the pre trained model that was going to be used to predict an emotion from. A pre trained model is a file that holds structured data that was previously trained on a large dataset. This helps recognise an emotion on a face an easier process.



## 5 Project Architecture



6 Project Plan



## 7 Development

Some development aspects of the project & complications that I ran into while developing this application.

### 7.1 Full Stack

I tried to include a website for this project using Django and Next JS. While researching the best way to have Python code communicate between a local instance and an online interface, I found that Django is the most well known and used back-end in web development specifically for Python [1]. I ran into complications trying to get Django set up as I have no previous experience in it. Linux comes pre-installed with Python and easier to install such frameworks on. I got advice that there is a new technology on Windows called Windows Subsystem for Linux (WSL) [2], that creates an instance of Linux on a Windows machine that can run a normal Linux system. Everything worked fine until I needed to use my webcam to test the software. WSL2 does not natively support local USB hardware from Windows, so it could not detect my webcam. There are solutions to this problem [3] but it's too time consuming and new errors develop so I went back to Windows development of Django, trying to clear the errors that came up there. To communicate with Next JS, Django Framework Rest API is needed.

## 7.2 Code Snippets

Using the Keras library, I load in a pre-trained model for face recognition. This helps detecting where the face is on the stream.

```
model = load_model("DriveSafe/best_model.h5")
```

Video capture of the webcam is stored in the variable 'cap' to be used later.

```
cap = cv2.VideoCapture(1)
```

The region of interest (ROI) of the face is cropped from the image. This will leave only the face to manipulate. Using OpenCV, the ROI is resized to 224x224 pixels. Then with Keras, the ROI image is converted into a Numpy array, as that's how images are manipulated, by turning them into arrays.

```
roi_gray = gray_img[y:y + w, x:x + h] # crop  
roi_gray = cv2.resize(rois_gray, (224, 224))  
img_pixels = image.img_to_array(rois_gray) #
```

At the end of the code, Keras predicts the emotion using the pre-trained model from before.

Numpy then grabs the maximum value along the list, and from a list of strings outlining the different emotions, the most common emotion is stored in a variable and displayed above the ROI box on the face.

```
predictions = model.predict(img_pixels)  
  
max_index = np.argmax(predictions[0]) #  
  
emotions = ('angry', 'disgust', 'fear',  
predicted_emotion = emotions[max_index]
```

## 8 Ethics

Ethical considerations for face recognition bring many questions to the table on how this will impact the future, and how reliable the software can be.

Racial concerns – *“ More than half of American adults, or nearly 117 million people, have photos on law enforcement's facial recognition network. However, it's disturbing those errors detected in the face recognition system were more common on dark-skinned faces, but fewer errors when matching light-skinned faces. ” [5]*

Data Privacy – Unsecure data storage can expose facial recognition data that can lead to many dangerous situations for a human. Also, the public monitoring of people's faces with or without consent is an infringement on human rights, especially in Europe following the GDPR laws.

## 9 Conclusion

The outcome of the project shows, how the software is displaying a window that detects the face. With that, the ROI box is shown around the face, acknowledging that a face has been found. An emotion is predicted above the box in a real time operating web cam.

Overall, it was a challenging project, but a good experience for the future, that interests me what the future of face detection has in store for many applications.

## 10 References

- [1] <https://docs.djangoproject.com/en/4.0/intro/install/>
- [2] <https://docs.microsoft.com/en-us/windows/wsl/install>
- [3] <https://docs.microsoft.com/en-us/windows/wsl/connect-usb>
- [4] <https://pyimagesearch.com/2021/04/12/opencv-haar-cascades/>
- [5] <https://learn.g2.com/ethics-of-facial-recognition>