VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

Multidisciplinary Project (CO3107)

Anti-aggressive system

Những người khó cười

Advisor: Băng Ngọc Bảo Tâm
Group Members:

Nguyễn Đình Khương Duy	1952207
Nguyễn Hải Đăng	1952652
Lưu Nguyễn Hoàng Minh	1952845
Nguyễn Phú Nghĩa	1952355
Nguyễn Lê Thảo Vy	1952536

HO CHI MINH CITY, FEBRUARY 2022

Contents

Member list

No.	Fullname	Student ID
1	Nguyễn Đình Khương Duy	1952207
2	Nguyễn Hải Đăng	1952652
3	Lưu Nguyễn Hoàng Minh	1952845
4	Nguyễn Phú Nghĩa	1952355
5	Nguyễn Lê Thảo Vy	1952536

Changelog

Date	Version	Changes	Changed by
02/10/2022	0.0	- Create report template	Nguyễn Hải Đăng
02/12/2022	1.0	- Write introduction	Lưu Nguyễn Hoàng Minh
02/14/2022	1.1	- Write functional requirements	Nguyễn Đình Khương Duy Nguyễn Hải Đăng
02/15/2022	1.2	- Write non-functional requirements	Nguyễn Phú Nghĩa
02/16/2022	1.3	- Write devices section	Nguyễn Lê Thảo Vy Nguyễn Đình Khương Duy
02/18/2022	2.0	- Change topic	
02/19/2022	2.1	- Write introduction	Lưu Nguyễn Hoàng Minh
02/21/2022	2.2	- Write functional requirements	Nguyễn Đình Khương Duy Nguyễn Hải Đăng
02/21/2022	2.3	- Write non-functional requirements	Nguyễn Phú Nghĩa
02/22/2022	2.4	- Write devices section	Nguyễn Lê Thảo Vy Nguyễn Đình Khương Duy
02/22/2022	3.0	- Draw use-case diagram	Nguyễn Lê Thảo Vy Nguyễn Đình Khương Duy Nguyễn Hải Đăng Nguyễn Phú Nghĩa Lưu Nguyễn Hoàng Minh
02/23/2022	3.1	- Write use-case table	Nguyễn Lê Thảo Vy Nguyễn Đình Khương Duy Nguyễn Hải Đăng Nguyễn Phú Nghĩa Lưu Nguyễn Hoàng Minh

1 Introduction

These days, when people are shifting from traditional working in the office to working from home, many workers from fields that need teamwork e.g. engineering, media, etc. suddenly felt disconnected from their co-workers. This would potentially cause people to feel irritated or angry, and in one way or another, would lead to the decline of work performance of some people. If not handled well, a vicious cycle would stem, the majority effect would take over to introduce the unproductive culture among peers, and damage for the company is inevitable.

Thus, we propose a solution to alleviate the problem by providing the physical bonds with the workers and giving them a gamified experience in controlling their emotion during working time.

This system is designed with the aid of micro:bit controller and Artificial Intelligence to track the emotion levels and amount of water drunk of a user during a working day. Then, the data of many users during many days will be organized in a form of monthly leaderboard for the users to compete with each others.

Companies can further incentivize their workers by having some kinds of monthly award for the ones who have the highest scores in our leaderboard.

With the idea, many problems are explored and discussed in this document.

Note that this is just a demonstration project for the idea. Real-life implementation of the idea should be accompanied with better equipment e.g. better motor for the fan.

2 Requirements

2.1 Functional requirements

- The AI system should be able to detect the emotion of the user.
- If the detected emotion is negative, the system should activate the fan, the water pump, and pop up a message on the user's laptop if needed.
- Users should be able to manually activate the water pump to fill up the cup.
- The system should provide the statistics about the user in the working day and provide a leaderboard for users in the network.

2.2 Non-functional requirements

- The response of the system (controlling the IoT devices and popping up messages) after detecting the negative emotion should be less than 1 second in regular time and less than 2 seconds in the internet rush hours.
- The AI model to detect emotions should have high accuracy, at least 95%. We do not want the system to be activated when the users is feeling good, or else our system would be the source of negative emotion.
- The software is cross-platform i.e. useable on Windows, Linux, macOS.
- Responsive web application since the app could be open from machines with different screen sizes.
- The web application is user-friendly. One should feel comfortable using it after a 15-minute tutorial.

- The web application should have adequate security by requiring users to log into their accounts to view their statistics.

3 Devices

3.1 General devices

Micro:bit

It serves as a controlling center for all sensors. Data from the sensors is sent through the extension circuit board to the Micro:bit and it will carry out several pre-processing sequence on the data such as converting units and calculating threshold, after which it will sent these data to server for further operations.

Extension circuit board

This device is used to connect additional sensors / devices and make it more convenient to build circuits with the micro:bit.

Adapter 5V

The adapter will act as a stable, long-term and sufficient power source that ensures the constant operation of our system.

3.2 Peripherals

DHT11

The DHT11 sensor's main functionality is to measure atmospheric temperature and humidity. In this system, only data of temperature is used.

- Input: Environment temperature and humidity.
- Output: Two integer values representing temperature and humidity level of the sensor's surrounding environment.

Mini-pump

A pump is needed to pour water into a glass. This pump operates on electrical power and will be attached to a water source.

Relay circuit

This circuit will act as a switch to control the operation of the mini-pump.

- Input: ON/OFF signals - these signals will be generated based on our system's analysis result.
- Output: The device let the current go though when received the ON signal and vice versa.

DRV power circuit

This is used to support a mini motor and a propeller, in order to create a mimic of an electric fan.

Mini motor and propeller

These devices will be connected to the above DRV power circuit.

Single push button

This is used to enable the user to manually activate the water pump.

3.3 External device

Webcam

A webcam on laptop will be used to capture the facial expression of user. This data can later be analyzed carefully. Our program will determine whether the user is not feeling well and produce an output signal. Base on this signal, the system will either carry out necessary steps or start this process over again.

4 Overall design

Our system can be separated into 2 parts: the AI-web application part and the IoT part.

4.1 AI - web application

4.1.1 AI

Facial expression recognition (FER) has been an active research area over the past few decades. There are some traditional approach based on hand-crafted features such as SIFT, HOG and LBP which are perform reasonably well on the datasets of images captured in a controlled condition, but fail to perform as good on more challenging datasets with more image variation and partial faces. In this application, we use the model presented in paper [1] based on attentional convolutional network, which is able to focus on important parts of the face and in variate condition. They believe that not all parts of the face are important in detecting a specific emotion, and in many cases, we only need to attend to the specific regions to get a sense of the underlying emotion. Below is the model architecture:

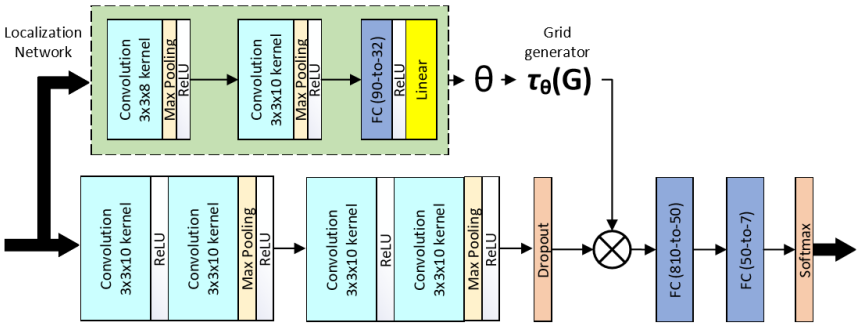

Figure 1: Overview of model architecture.

The above branch is the spatial transformer module which help to focus on the most relevant part of the image. The spatial transformer consists of two convolution layers followed by max pooling layer and ReLU activation and two fully-connected layers, then feed to the sampling grid [2] for producing the warped data. Figure 2 shows the result of utilizing a spatial transformer. (a) An picture of an MNIST digit that has been distorted by random translation, scaling, rotation, and clutter which is fed into the spatial transformer network. (b) The spatial transformer's localization network predicts a transformation to apply to the input picture. (c) After applying the transformation, the output of the spatial transformer. (d) The categorization prediction is generated by a fully-connected network on the spatial transformer's output. The spatial transformer network (a CNN with a spatial transformer module) is trained end-to-end using just class labels; the system is not provided any knowledge of the ground truth transformations.

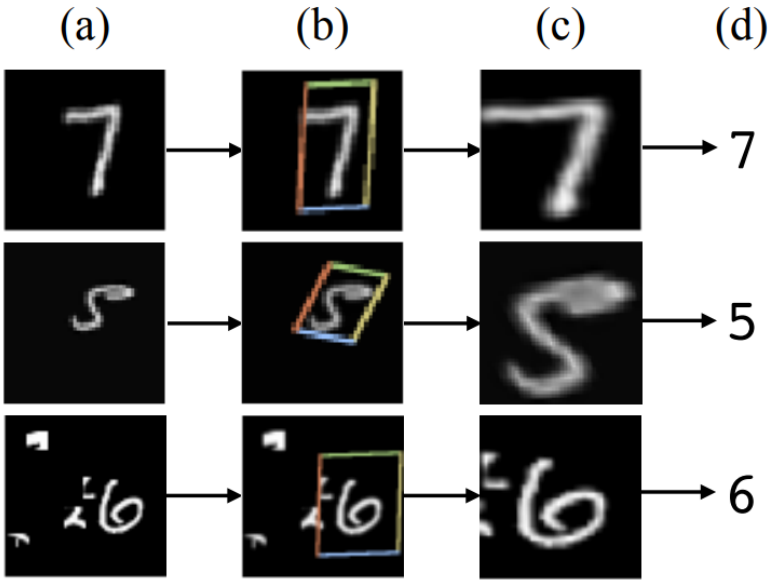

Figure 2: Spatial module

The below branch in Figure 1 is the feature extraction part which consist of four convolutional layers, each two followed by a max-pooling layer and rectified linear unit (ReLU) activation function and then a dropout layer followed by two fully-connected layers. The loss function is just simple the summation of two terms, the classification loss (cross-entropy) and the regularization term.

$$\mathcal{L}_{\text{overall}} = \mathcal{L}_{\text{classifier}} + \lambda \left\| w_{(fc)} \right\|_2^2$$

The validation set is used to tweak the regularization weight λ. We can train our models from scratch using both dropout and *l2* regularization, even on relatively tiny datasets like **JAFFE** and **CK+**. It's worth noting that for each of the datasets utilized in this study, we train a distinct model. We also attempted a network design with over 50 layers, but the accuracy remained the same. As a result, the simplified model depicted above was ultimately selected.

FER2013 [3], the extended Cohn Kanade (CK+) [4], and the Facial Expression Research Group Database (FERG) [5] are some of the most popular facial expression

identification datasets. The model's performance on the preceding datasets is shown below. The model was trained on a part of the dataset, validated on the validation set, and the accuracy was reported over the test set in each case.

Method	**Accuracy Rate**
Bag of Words [6]	67.4%
VGG+SVM [7]	66.31%
GoogleNet [8]	65.2%
Mollahosseini et al [9]	66.4%
The used model	**70.02%**

Table 1: Classification Accuracy on FER2013

Method	**Accuracy Rate**
MSR [10]	91.4%
3DCNN-DAP [11]	92.4%
Inception [8]	93.2%
IB-CNN [12]	95.1%
IACNN [13]	95.37%
DTAGN [14]	97.2%
ST-RNN [15]	97.2%
PPDN [16]	97.3%
The used model	**98.0%**

Table 2: Classification Accuracy on CK+

Method	**Accuracy Rate**
DeepExpr [5]	89.02%
Ensemble Multi-feature [17]	97%
Adversarial NN [18]	98.2%
The used model	**99.3%**

Table 3: Classification Accuracy on FER2013

We chose to use an attentional convolutional network as a new framework for facial expression recognition because we believe that attention will become a more important part of detecting facial expressions in the future, allowing neural networks with less than 10 layers to compete with (and even outperform) much deeper networks for emotion recognition.

4.1.2 Web application

Framework and Platform

Currently, we are using Django as our web framework to develop the web. We choose

it since Django is one of the first choice when it come to development speed, which is a priority consider our project's scope.

After successfully developed the web locally, we deployed it using Heroku - a free hosting platform that is really suitable for our need.

Authentication

For authentication, users can choose between two methods, authenticate using the accounts registered specifically for our app or by using social accounts from Google or Facebook. Either way, they first has to register an account on our register page with the process described in section 6.2.3. After successfully register, the user's account details will be stored in our database and used for future log in.

For modelling the user in the database, we use a custom model called User that extends from the Django default user model AbstractUser, and each account is stored as an User instance. For normal users, common information about their account will be stored such as first name, last name, username, email and password while user ID and extra data needed to communicate with the social account provider is stored for social account users.

4.2 IoT

The IoT subsystem deploys a set of hardware devices, including a DHT11 sensor, push button, a combination of motor-propeller-DRV that mimics a fan, and a mini-pump with relay circuit.

The functionality of this subsystem can be described as follow: Upon receiving data of user's emotion from Adafruit server, the system will determine whether that is positive (happy / neutral) or negative (scare / fear / angry / disgust). If it is a positive feeling, nothing should be done after that. Otherwise, the mini-pump should be activated and stayed on for a pre-determined amount of time. Besides, the system would send a request for the room temperature to be read from DHT11, if this temperature is above 25 degree Celsius, the mimic fan would be turned on. The same process of pumping water and activating fan can also be triggered manually through pushing the button.

5 Use case details

5.1 Use case 1: Capture image

Use case ID	1
Use case name	Capture image
Actor	Camera
Description	The camera continuously captures images and send them to the AI system
Preconditions	The camera has to be turned on
Normal flow	1. AI System continuously requests the camera to capture the images 2. Camera captures images 3. Camera sends captured images back to the AI system
Postconditions	AI system successfully receives the images from the camera
Exceptions	
Alternative Flow	

5.2 Use case 2: Detect emotion

Use case ID	2
Use case name	Detect emotion
Actor	
Description	After receiving captured images from the camera, the AI system detects the trait of the user in the images and sends control signals if there is an anger from the user
Preconditions	AI system successfully receives captured images from the camera
Normal flow	1. The AI system recognizes the user is in the frame 2. The AI system analyzes the trait of the user 3. The AI system detects there is anger from the user
Postconditions	AI system outputs a signals indicating the user is angry
Exceptions	
Alternative Flow	At step 1, if the user is not in the frame: (1.1) The AI system ignores the image At step 3, if there is no anger from the user: (3.1) The AI system ignores the image

5.3 Use case 3: Pop up message

Use case ID	3
Use case name	Pop up message
Actor	
Description	When the AI system detects that the user is angry, it immediately pop up a happy message on user's monitor
Preconditions	User is recognized being angry
Normal flow	1. System picks a random message 2. System pop up the message as a notification on the monitor of the user
Postconditions	The message is successfully pop up on the screen
Exceptions	
Alternative Flow	

5.4 Use case 4: Send control signals

Use case ID	4
Use case name	Send control signals
Actor	Adafruit server
Description	When the AI system dectects that the user is angry, system sends a control signal indicating the user is angry to Adafruit server
Preconditions	User is recognized being angry
Normal flow	1. System sends a control signal about the anger of the user to Adafruit server
Postconditions	Adafruit successfully receives the control signal the system sent
Exceptions	
Alternative Flow	

5.5 Use case 5: Collect information

Use case ID	5
Use case name	Collect information
Actor	Adafruit server and Firebase
Description	System collects timestamps whenever the user is angry from AI system and pumped water capacity from Adafruit server and then pushes them on Firebase
Preconditions	There are data in either the AI system or Adafruit server
Normal flow	1. System collects timestamps whenever the user is angry from AI system 2. System collects pumped water capacity from Adafruit server 3. System pushes collected data on Firebase
Postconditions	Firebase successfully receives data from the system
Exceptions	
Alternative Flow	

5.6 Use case 6: Push button

Use case ID	6
Use case name	Push button
Actor	User and Adafruit server
Description	User can manually push a button to let the system pump water into the cup even if the user is not angry
Preconditions	User is not angry but wants to drink water
Normal flow	1. User pushes the button 2. The button sends a signal to Adafruit server
Postconditions	Adafruit server successfully receives the signal
Exceptions	
Alternative Flow	

5.7 Use case 7: Read light sensor

Use case ID	7
Use case name	Read light sensor
Actor	Adafruit server
Description	System use light sensor to control the pump. Water is not pumped if there is no cup in the base or the water is at a significant level.
Preconditions	Light sensor can work well
Normal flow	1. System tries to detect light 2. System sends a value from light sensor to Adafruit server
Postconditions	Adafruit server successfully receives the value
Exceptions	
Alternative Flow	

5.8 Use case 8: Read temperature

Use case ID	8
Use case name	Read temperature
Actor	Adafruit server
Description	System use DHT11 to control the fan. The fan is not turned on if the temperature of the room is low
Preconditions	DHT11 can work well
Normal flow	1. System senses the temperature of the room 2. System sends a temperature value to Adafruit server
Postconditions	Adafruit server successfully receives the value
Exceptions	
Alternative Flow	

5.9 Use case 9: Turn on fan

Use case ID	9
Use case name	Turn on fan
Actor	Adafruit server
Description	System turns on the fan when it receives a control signal from Adafruit server
Preconditions	System receives a control signal that is turning the fan from Adafruit server
Normal flow	1. System turns on the fan
Postconditions	
Exceptions	
Alternative Flow	

5.10 Use case 10: Pump water

Use case ID	10
Use case name	Pump water
Actor	Adafruit server
Description	System pumps water into the cup when it receives a control signal from Adafruit server
Preconditions	System receives a control signal that is pumping water from Adafruit server
Normal flow	1. System pumps water into the cup
Postconditions	
Exceptions	
Alternative Flow	

5.11 Use case 11: Stop pumping water

Use case ID	11
Use case name	Stop pumping water
Actor	Adafruit server
Description	System stops pumping water into the cup when it receives a control signal from Adafruit server
Preconditions	System receives a control signal that is stopping pumping water from Adafruit server
Normal flow	1. System stops pumping water into the cup
Postconditions	There is no water pumped into the cup
Exceptions	
Alternative Flow	

5.12 Use case 12: View webpage

Use case ID	12
Use case name	View webpage
Actor	User and Vercel
Description	User can login the webpage with a account to know the number of time that the user was angry in a day, the total water the user drank in a day or view these values visualized as graphs. There is also a leaderboard to know who can keep patient and drink enough water
Preconditions	User access the right URL of the webpage
Normal flow	1. Vercel hosts the webpage 2. User login into the webpage 3. User views relative indexes, graphs, and leaderboard
Postconditions	User can access his/her data on the webpage
Exceptions	
Alternative Flow	At step 2, if the user has not have an account: (2.1) User sign-up an account (2.2) User login by the new account

5.13 Use case diagram

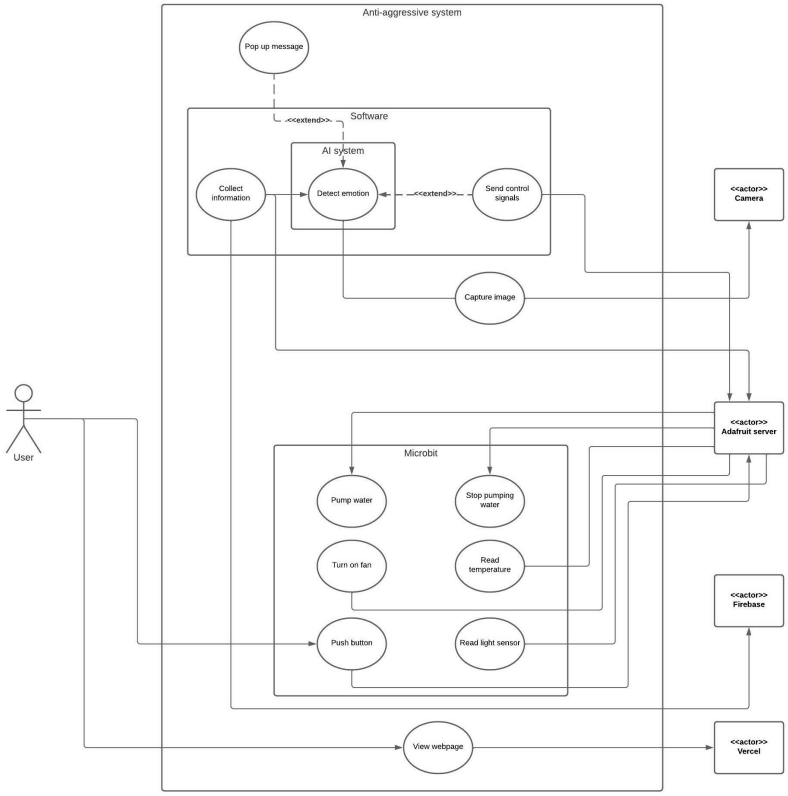

Figure 3: Use-case diagram of Anti-aggressive system

6 UI / UX Design

For a larger view of attached images, please follow this link.

6.1 Homepage

This interface is displayed by default when people access the website. From this page, they can proceed to log into an existing account, register a new one or simply scroll through some introductory paragraphs about our team and our project.

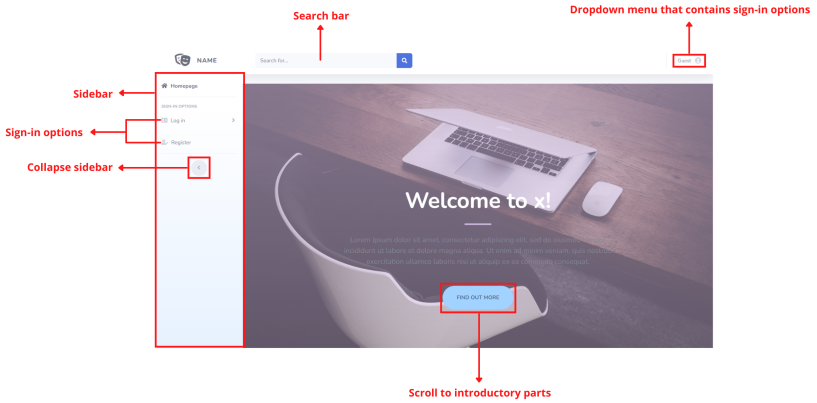

Figure 4: Homepage interface for Guest

After going through the authentication process, users will be presented with a slightly different Homepage interface. In this interface, the sidebar will contain more navigation items, and users now have access to their notification center (for ordinary users) and message center.

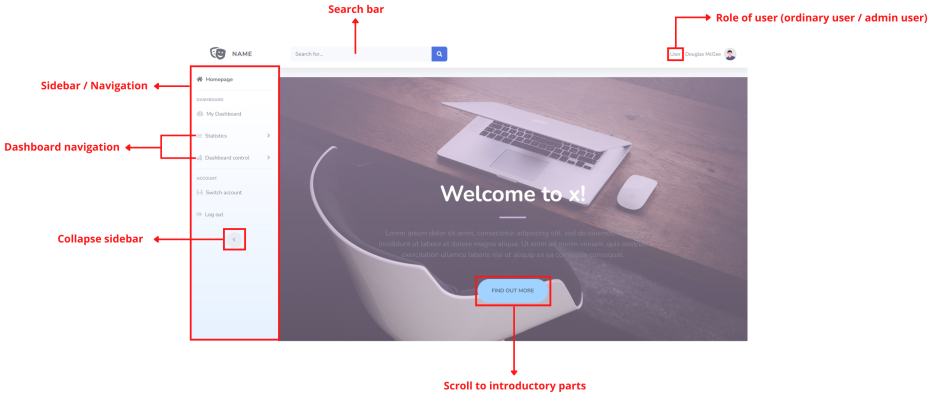

Figure 5: Homepage interface for User

If logged in as Admin, users have the authorization to edit data displayed on the Homepage. Any changes made by Admin will be applied to the Homepage of all ordinary users.

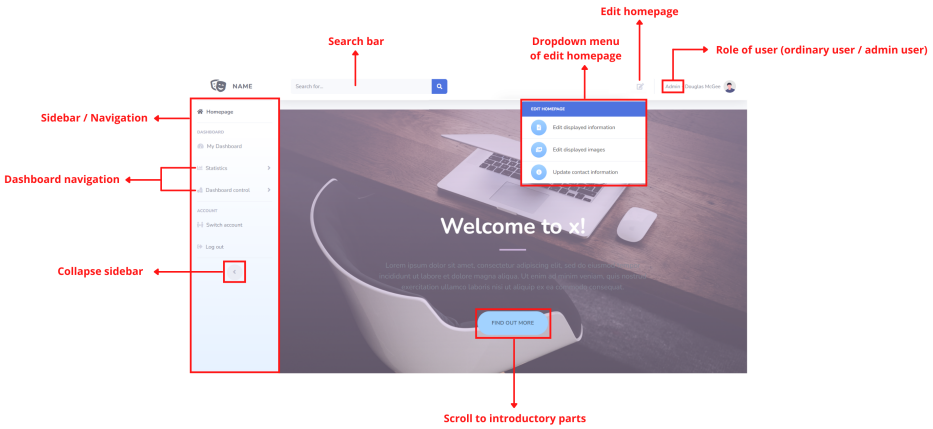

Figure 6: Homepage interface for Administrator

A section that contains contact information is positioned at the end of the website.

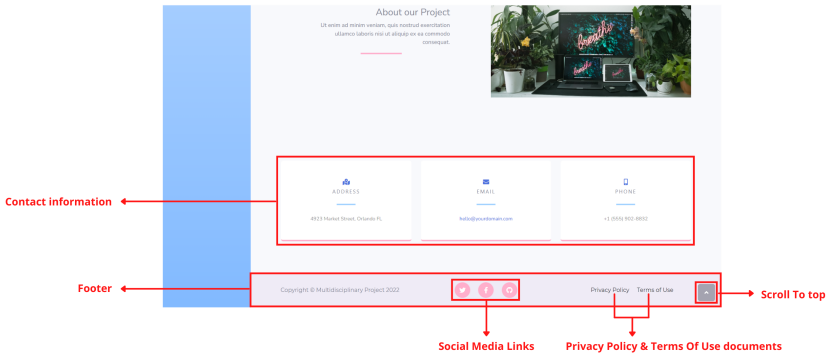

Figure 7: Contact information

6.2 Authentication

6.2.1 Login interface

If the user choose to log into an existing account from the homepage, they will be presented with this interface. In this page, user will enter their registered information such as email address and password, then press the Login button to get into their account. There is a "Remember me" button to remember the account if the user want to have more convenience for later use. The user also have the option to login with their Google or Facebook account. If the user forget their password, they can click on the "Forgot password?" text to redirect themselves to the forgot password page where they can reset the password. Lastly, user can choose to create a new account by clicking on the "Create an account" link.

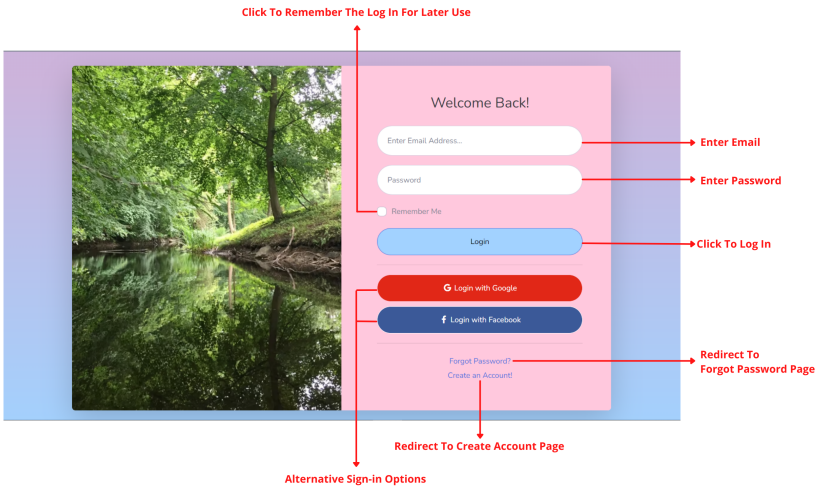

Figure 8: Log in interface

6.2.2 Forget password interface

This interface will appear when user forget their password and clicked on the "Forgot password?" link. Here, user will enter the email of the account that they have registered, then an email will be sent to that email address and the user can choose to reset the password of their account. During the reset process, user would also have the choice to redirect to either the register page or the login interface.

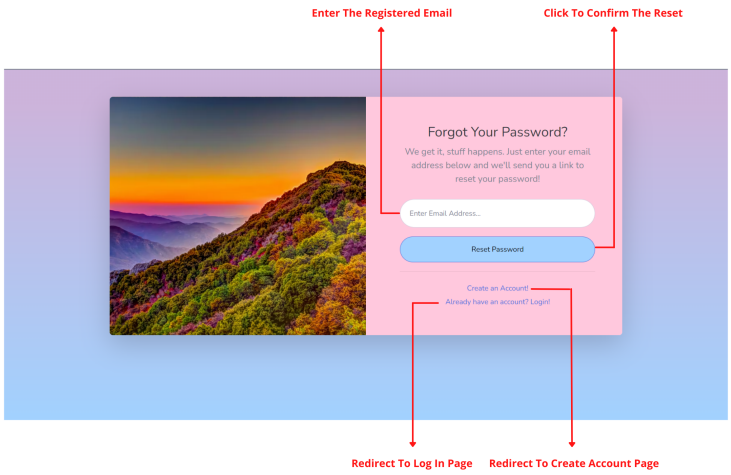

Figure 9: Forget password interface

6.2.3 Create account interface

The creation of any user account will be taking place here. The user can choose to create a new account by providing basic information such as name, email address and their password or they can skill all of that and register with their Google or Facebook account. This page would also provide the link to the login and the forgot password page.

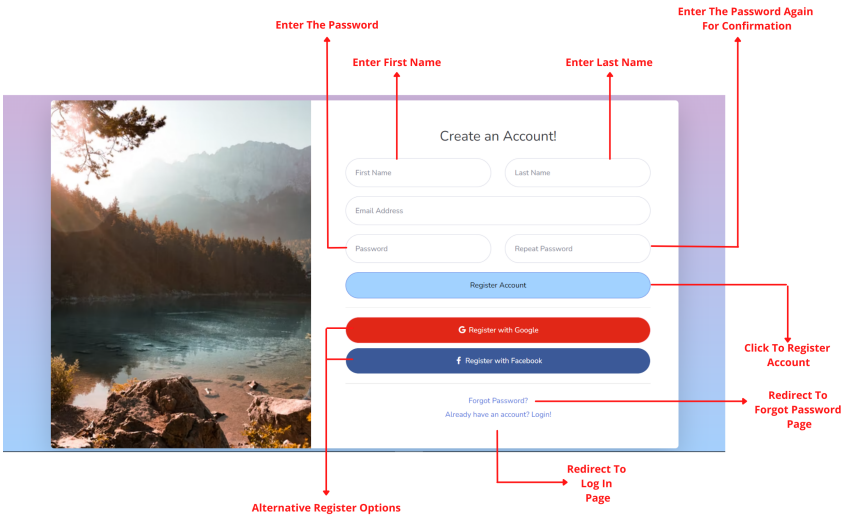

Figure 10: Create account interface

6.2.4 Log out interface

If users click on "Log out", a confirmation message will pop up. Users can either choose to continue to log out or return to their current session.

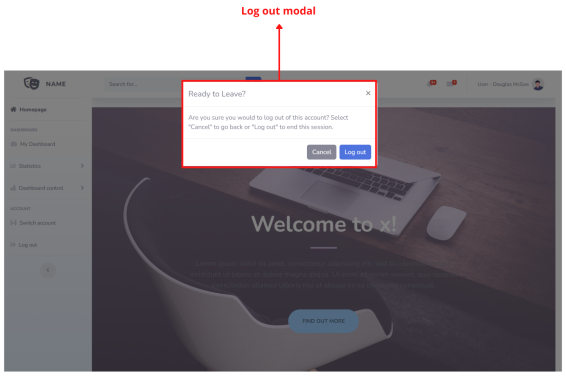

Figure 11: Log out confirmation

6.3 Dashboard

6.3.1 Statistic

After logging in as a User, the client can browse for his/her statistics by navigating to the "My Dashboard" tab on the sidebar. From this view, they can filter the requested data by a specific date of a month.

With the selected time range, the user can see their overall score for the day, which in turn is calculated based on the amount of water drunk and the emotion score in that time range. For a better visualization, the user can refer to the graph at any time, which is drawn based on the data collected from the user (water consumed, emotion score) at different work hours during the day.

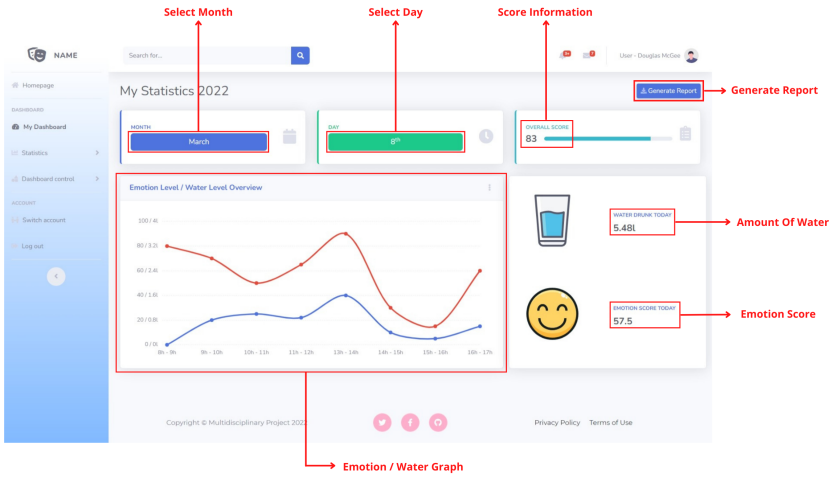

Figure 12: Personal Statistics

6.3.2 Leaderboard

The User can also view his/her score ranked on a leaderboard. The leaderboard has 3 columns: No., User and Score. The higher score the User get, the higher rank he/she is. The User can change the number of visible entries among 10, 25, 50 and 100. User can sort the leaderboard by No., User or Score. The "Search box" is where the user can search for particular records. Furthermore, the user can navigate through pages to view more records on the leaderboard.

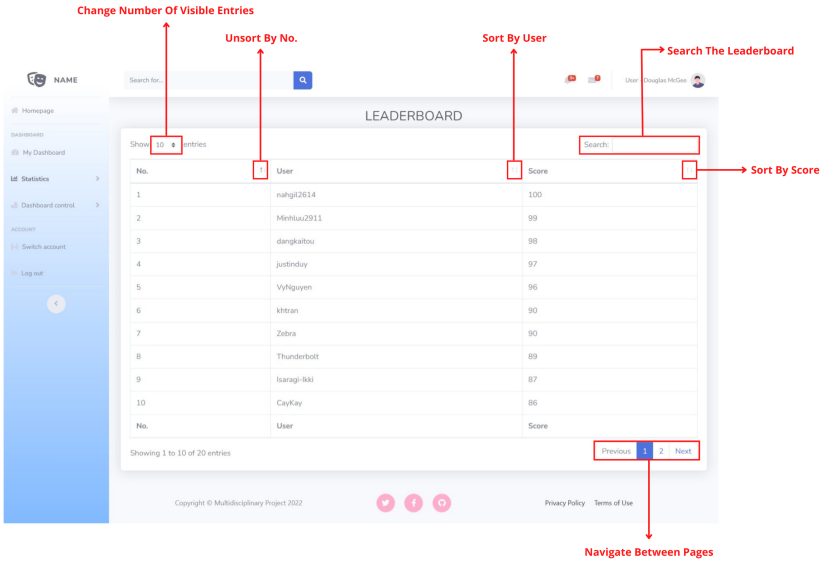

Figure 13: Leaderboard - User mode

While having a fairly similar interface to that of a User's role, the Admin can also edit other users' data by clicking on a small button on the corresponding row in the leaderboard. The edit include insertion, deletion, and modification of the data.

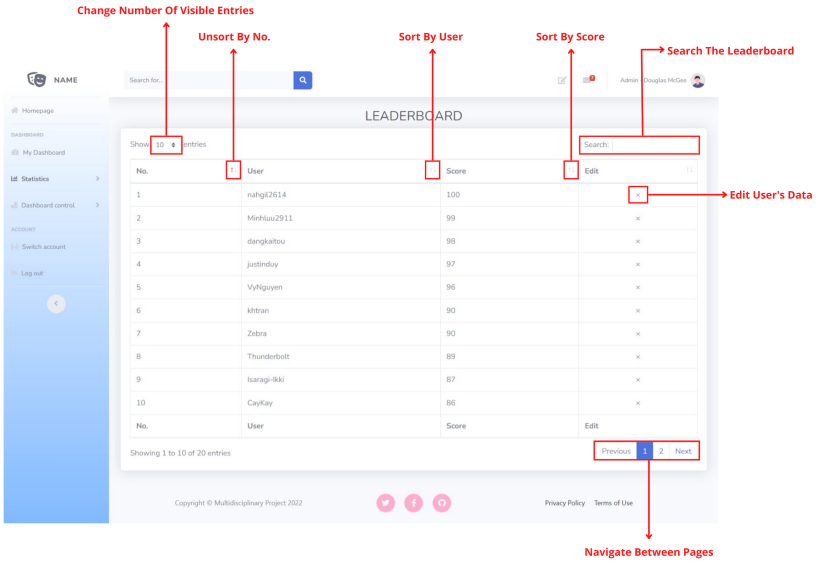

Figure 14: Leaderboard - Admin mode

7 Database Overview

In order to run our system effectively, we need to have a database to store necessary information about it. We have done the conceptual design of our database in the EER diagram as follows:

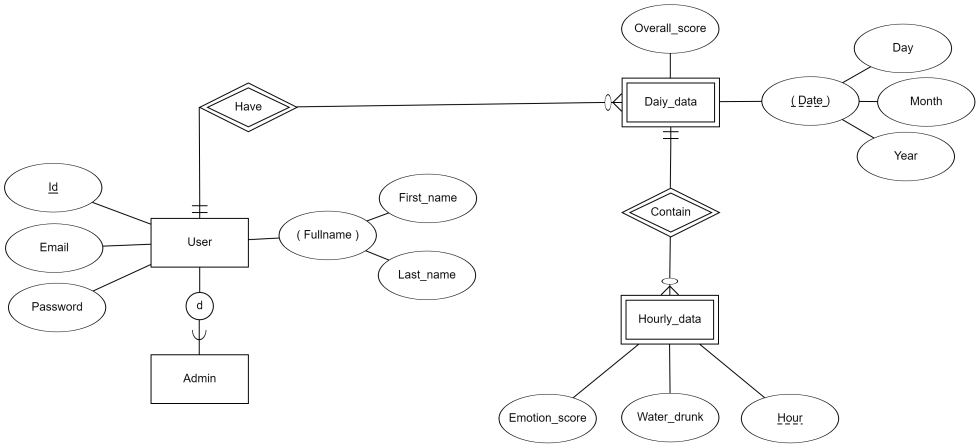

Figure 15: Enhanced Entity-Relationship Diagram of Anti-aggressive system

As you can see, our only strong entity type is **User**, which represents a generic

User (not a Guest) with basic information like email, password, and name. Among the User, there are some special entities called **Admin** representing the admins of our web application. The Admin has more rights to the database than the User, which include inserting, modifying, and deleting the data.

Note that currently, our decision is to control the user's role in the application layer, so we need to store the information of User and Admin in our database explicitly.

Besides, there is a weak entity type identified by **User** called **Daily_data** to store information about the water drunk and emotion score of the user in a day. Each Daily_data entity contains its corresponding date and the user's overall score for that day.

Furthermore, to represent more information about a day, Daily_data also identifies another weak entity type called **Hourly_data**, which store the information about each hour of the day of a user, including the exact hour, the water drunk, and the emotion score.

8 Technical diagrams

8.1 Deployment Diagram

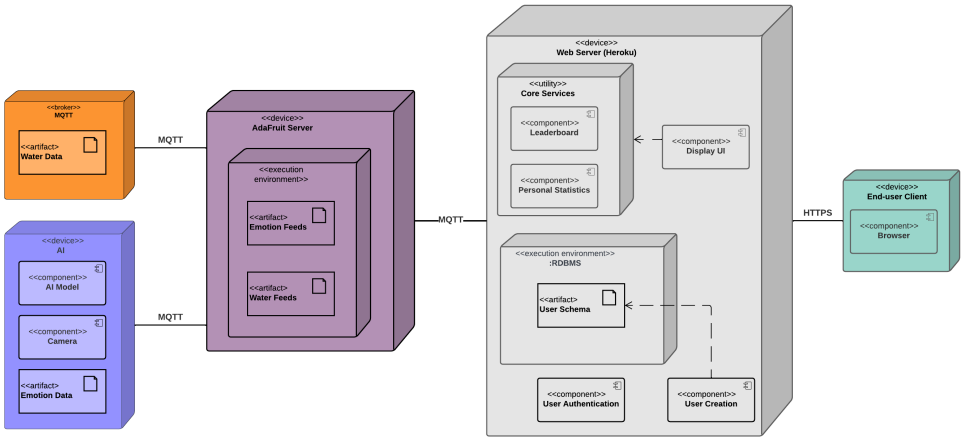

Figure 16: Deployment Diagram of Anti-aggressive system

8.2 Activity Diagram

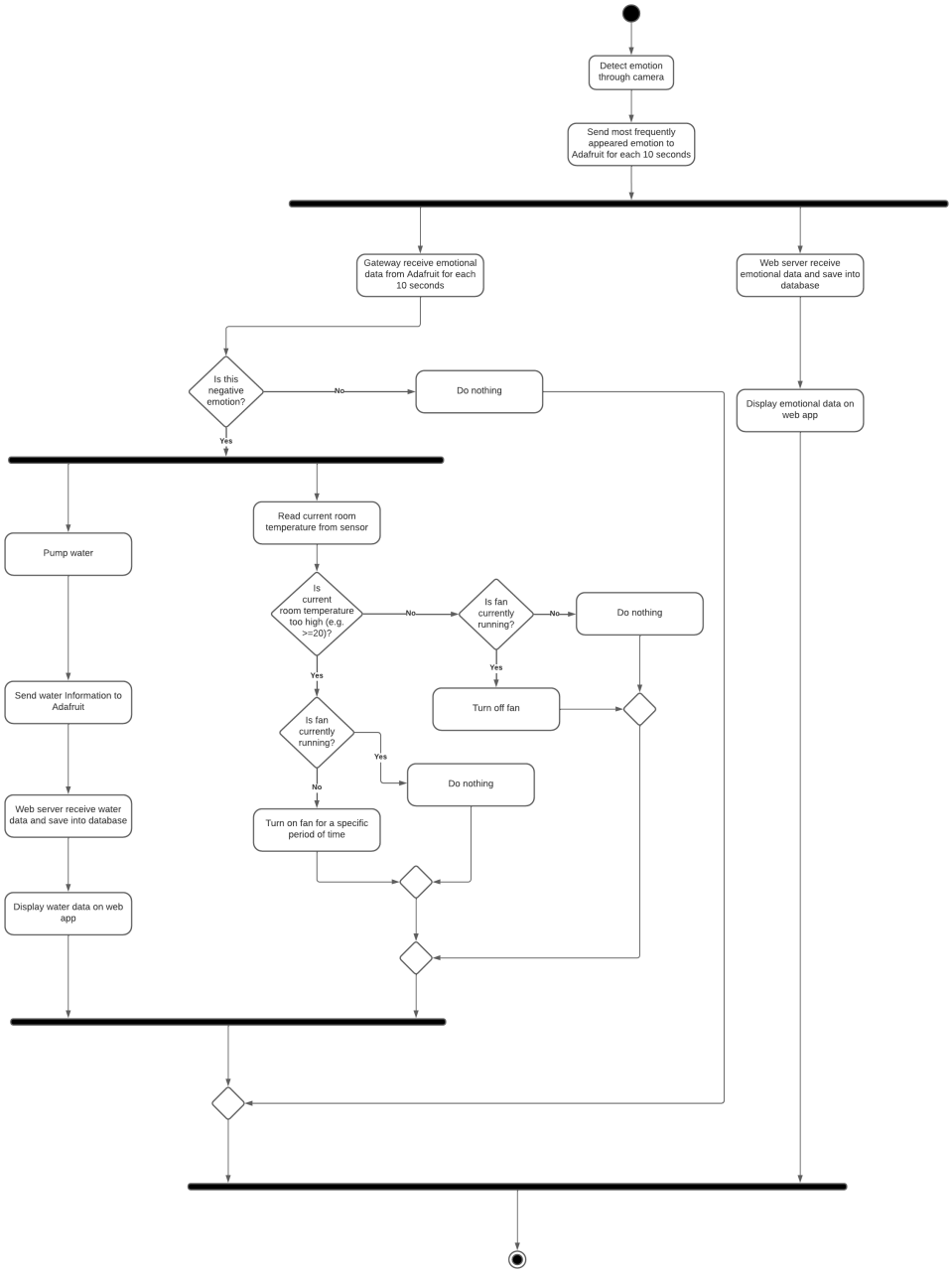

Figure 17: Activity Diagram of Anti-aggressive system

9 Final workflow

So now, after having all of the components for our product, we need to connect them together: IoT, AI, web server, database.

Firstly, we use SQLite as our final database due to the time constraint, given that it does not change the application's behaviours.

Secondly, for the IoT part, our microbit is in charge of controlling the devices and connect to the world via the gateway. In its turn, the gateway will connect to AdaFruit server via their API to connect to every other part of our application.

Next, for the authentication part, we store users' information in the database and check that to authenticate the user whenever he/she logs in.

After that, Django will hold a session for the user and we use that to know which user is using the web app. The web server also subscribe to the emotion and water feed of the current user at the login time. Whenever something is changed, it is reflected on the user's statistics page. The leaderboard is therefore just a comprehensive compilation of all the users' data in a sorted order.

Being quite important, the AI system will stand in the middle of everything. It reads the emotion from the user and send the corresponding signal to the gateway through AdaFruit. In its turn, the gateway will ask microbit to give some more information like temperature etc. to give the final decision about whether to turn on the fan and open the water pump. If it decides to do so, it will publish that information to the feeds on AdaFruit and the web server will update itself on the publish.

Remember that our web server is from Heroku as this is our choice of hosting server.

10 Meeting Minute

The folder holding Meeting Minute is here.

11 Git History

The Git History is here. You can also visit the web app deployed on Heroku here.

References

[1] Shervin Minaee, Mehdi Minaei, and Amirali Abdolrashidi. Deep-emotion: Facial expression recognition using attentional convolutional network. *Sensors*, 21(9):3046, 2021.

[2] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *Advances in neural information processing systems*, 28, 2015.

[3] Ian J Goodfellow, Dumitru Erhan, Pierre Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, et al. Challenges in representation learning: A report on three machine learning contests. In *International conference on neural information processing*, pages 117–124. Springer, 2013.

[4] Patrick Lucey, Jeffrey F Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *2010 ieee computer society conference on computer vision and pattern recognition-workshops*, pages 94–101. IEEE, 2010.

[5] Deepali Aneja, Alex Colburn, Gary Faigin, Linda Shapiro, and Barbara Mones. Modeling stylized character expressions via deep learning. In *Asian conference on computer vision*, pages 136–153. Springer, 2016.

[6] Radu Tudor Ionescu, Marius Popescu, and Cristian Grozea. Local learning to improve bag of visual words model for facial expression recognition. In *Workshop on challenges in representation learning, ICML*. Citeseer, 2013.

[7] Mariana-Iuliana Georgescu, Radu Tudor Ionescu, and Marius Popescu. Local learning with deep and handcrafted features for facial expression recognition. *IEEE Access*, 7:64827–64836, 2019.

[8] Panagiotis Giannopoulos, Isidoros Perikos, and Ioannis Hatzilygeroudis. Deep learning approaches for facial emotion recognition: A case study on fer-2013. In *Advances in hybridization of intelligent methods*, pages 1–16. Springer, 2018.

[9] Ali Mollahosseini, David Chan, and Mohammad H Mahoor. Going deeper in facial expression recognition using deep neural networks. In *2016 IEEE Winter conference on applications of computer vision (WACV)*, pages 1–10. IEEE, 2016.

[10] Salah Rifai, Yoshua Bengio, Aaron Courville, Pascal Vincent, and Mehdi Mirza. Disentangling factors of variation for facial expression recognition. In *European Conference on Computer Vision*, pages 808–822. Springer, 2012.

[11] Mengyi Liu, Shaoxin Li, Shiguang Shan, Ruiping Wang, and Xilin Chen. Deeply learning deformable facial action parts model for dynamic expression analysis. In *Asian conference on computer vision*, pages 143–157. Springer, 2014.

[12] Shizhong Han, Zibo Meng, Ahmed-Shehab Khan, and Yan Tong. Incremental boosting convolutional neural network for facial action unit recognition. *Advances in neural information processing systems*, 29, 2016.

[13] Mengchao Bai, Weicheng Xie, and Linlin Shen. Disentangled feature based adversarial learning for facial expression recognition. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 31–35. IEEE, 2019.

[14] Heechul Jung, Sihaeng Lee, Junho Yim, Sunjeong Park, and Junmo Kim. Joint fine-tuning in deep neural networks for facial expression recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 2983–2991, 2015.

[15] Tong Zhang, Wenming Zheng, Zhen Cui, Yuan Zong, and Yang Li. Spatial–temporal recurrent neural network for emotion recognition. *IEEE transactions on cybernetics*, 49(3):839–847, 2018.

[16] Xiangyun Zhao, Xiaodan Liang, Luoqi Liu, Teng Li, Yugang Han, Nuno Vasconcelos, and Shuicheng Yan. Peak-piloted deep network for facial expression recognition. In *European conference on computer vision*, pages 425–442. Springer, 2016.

[17] Hang Zhao, Qing Liu, and Yun Yang. Transfer learning with ensemble of multiple feature representations. In *2018 IEEE 16th international conference on software engineering research, management and applications (SERA)*, pages 54–61. IEEE, 2018.

[18] Clément Feutry, Pablo Piantanida, Yoshua Bengio, and Pierre Duhamel. Learning anonymized representations with adversarial neural networks. *arXiv preprint arXiv:1802.09386*, 2018.