Project Proposal: Application of Computer Vision in Ecology

Fedor Chursin

220904

Table of contents

Value Proposition & Business Understanding

AI Canvas: provide high-level overview

AI canvas

Opportunity Why do it?	In the market of apps related to ecology, there is a need for a carbon footprint tracker that also gives tips on how to reduce it and is easy to use.
Solution What is it?	Mobile app with embedded computer vision application that will help people to track their personal CO2 footprint and will give tips on how to reduce it.
Users Who needs it?	People who are concerned about ecology and want to contribute to the improvement of the ecological state of our planet
Data What are the model inputs?	An image of the meal taken by an app user
Strategy Why us?	
Policy and process What else must change?	Ecological market should be adjusted to increase it capacities
Transfer learning How will we build it?	CNN model + transfer learning with VGG16 model and data augmentation
Success criteria How will we know it works?	A model with high performance + High-quality app

Market research & Target audience analysis

- Lack of high-quality tools for tracking and reducing the carbon footprint
- Recently there is a growth in interest towards ecology
- People want to contribute towards improving ecological state of the planet
- 'Eco-friendly trends'
- A new approach using computer vision to improve users' experience

A gap in the market for an advanced carbon footprint mobile application

Target audience analysis

12 – 65
Years Old

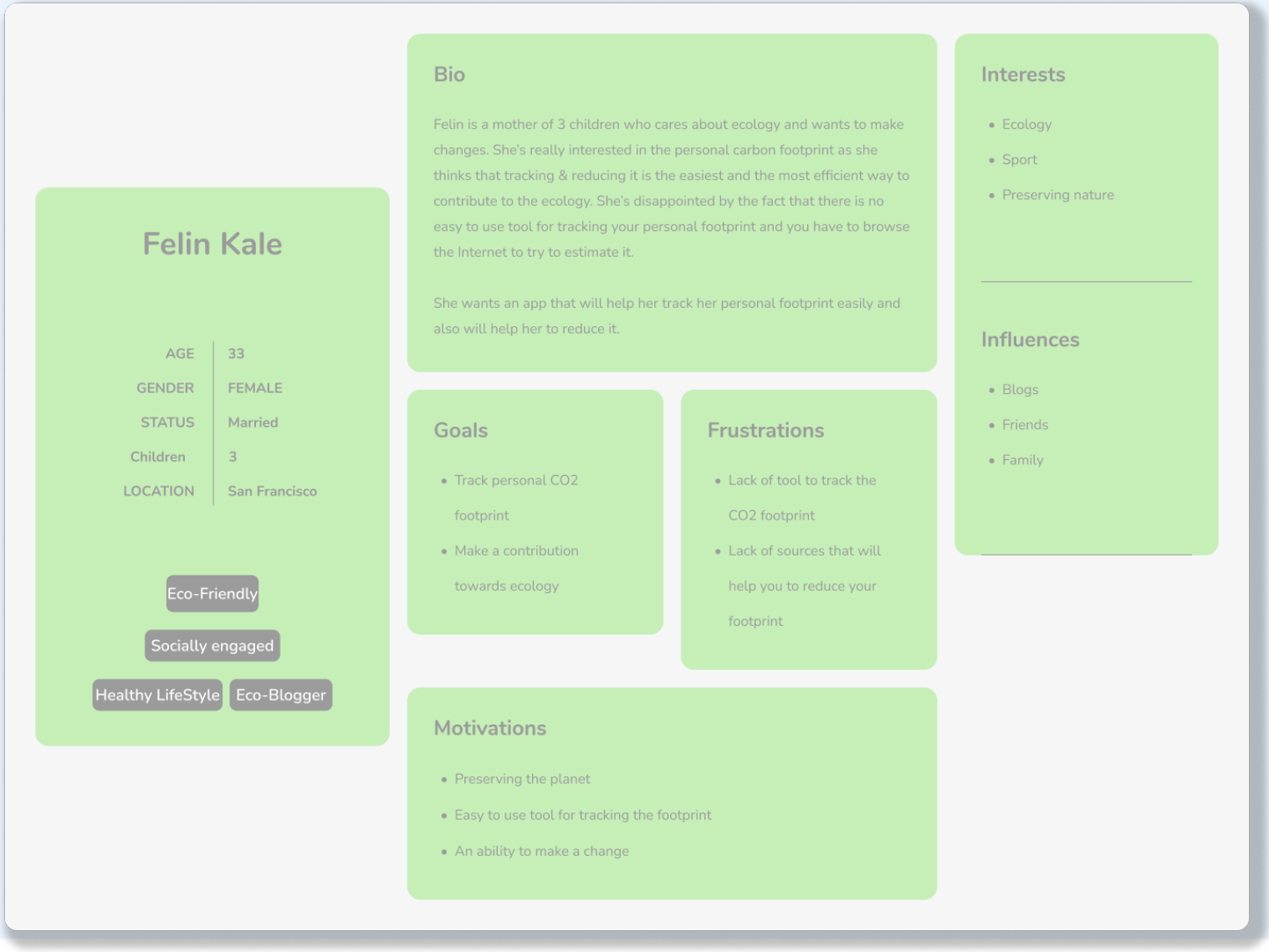
No geographic
segmentation

Eco-Friendly
Lifestyle

Environmentally
conscious
people

45-50% of consumers worldwide

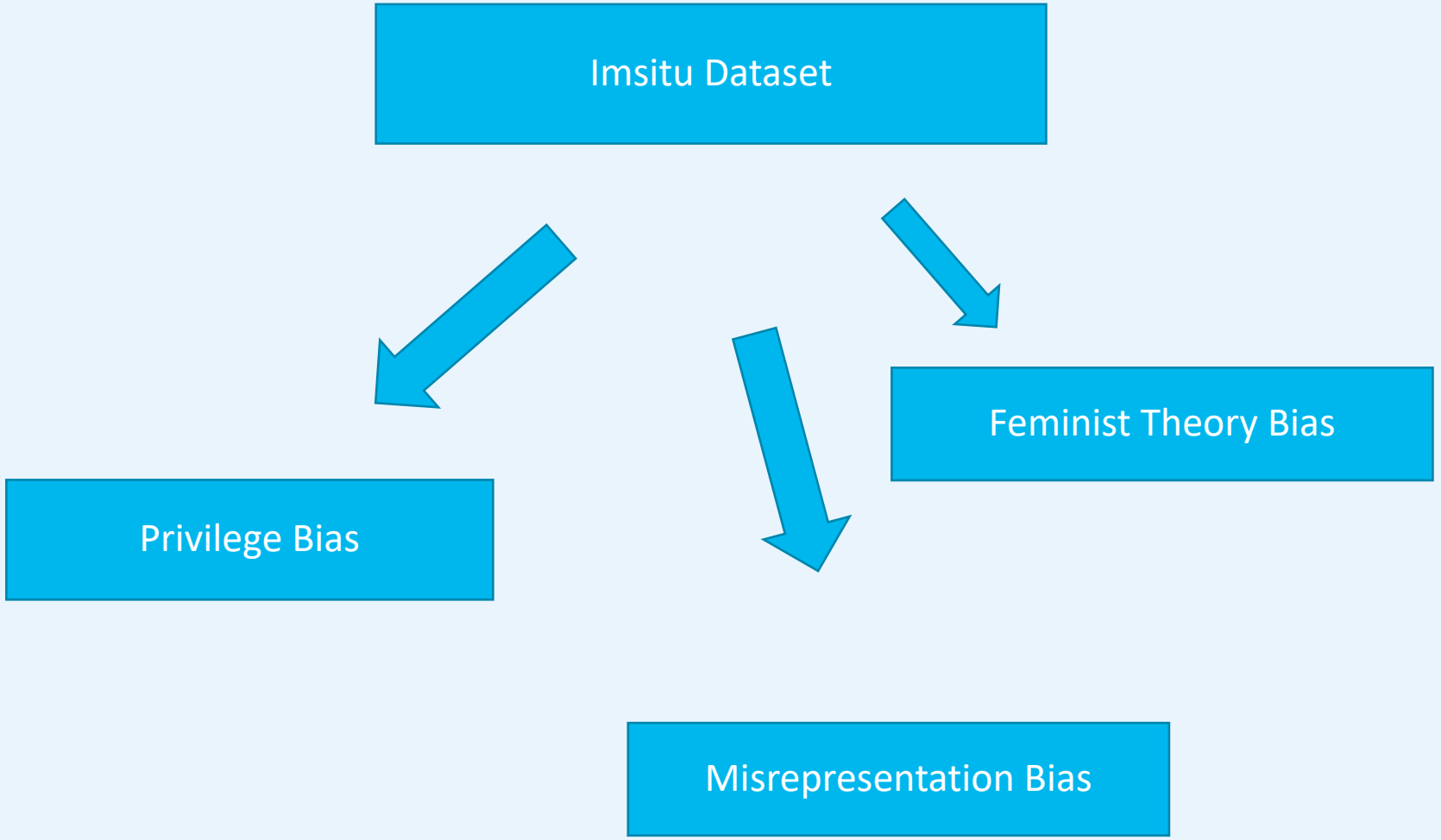
Target audience analysis



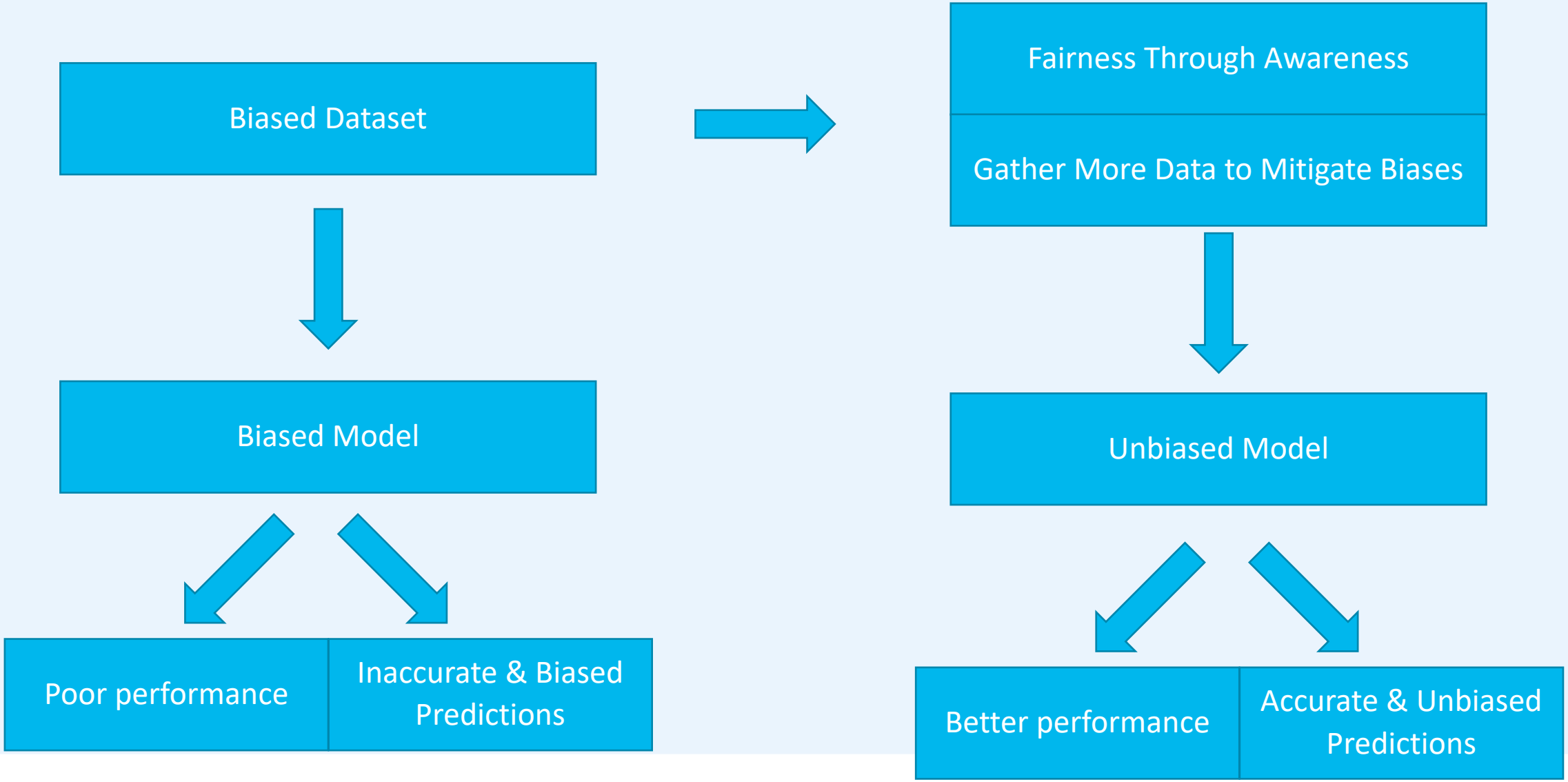
Proposal feasibility

Responsible AI

Responsible AI



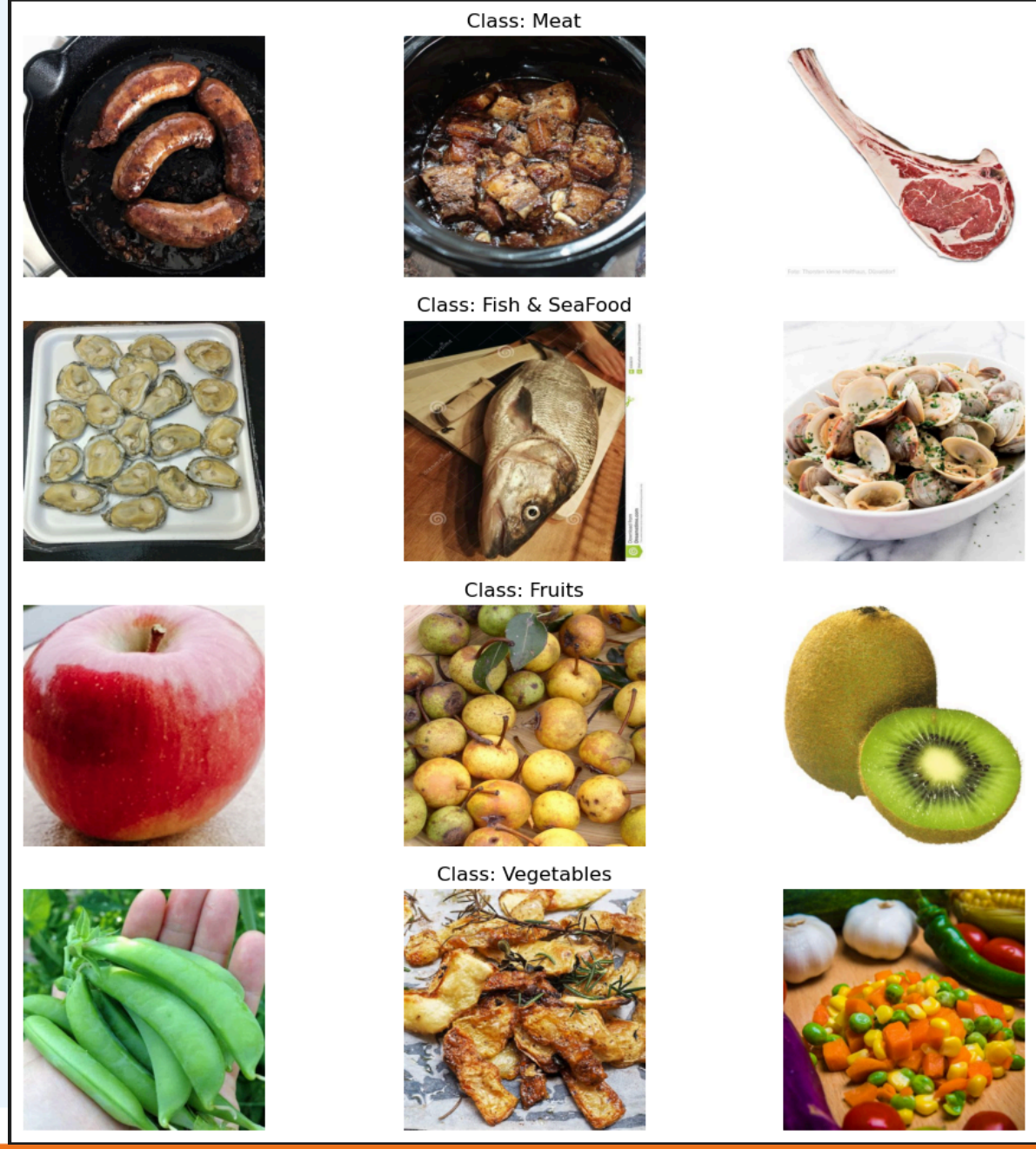
Responsible AI



Proof-of-Concept: Deep Learning

PoC: The dataset

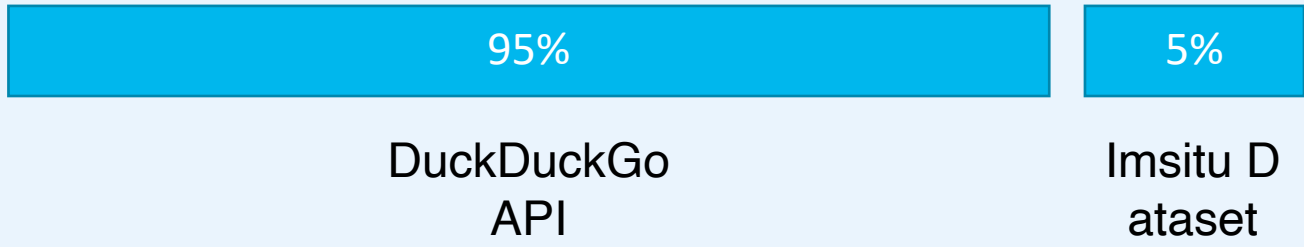
- Multi-class classification
- 4 classes:
  - Meat
  - Fish & Seafood
  - Fruits
  - Vegetables
- 184 images in each class
- Quality > Quantity



PoC: The dataset

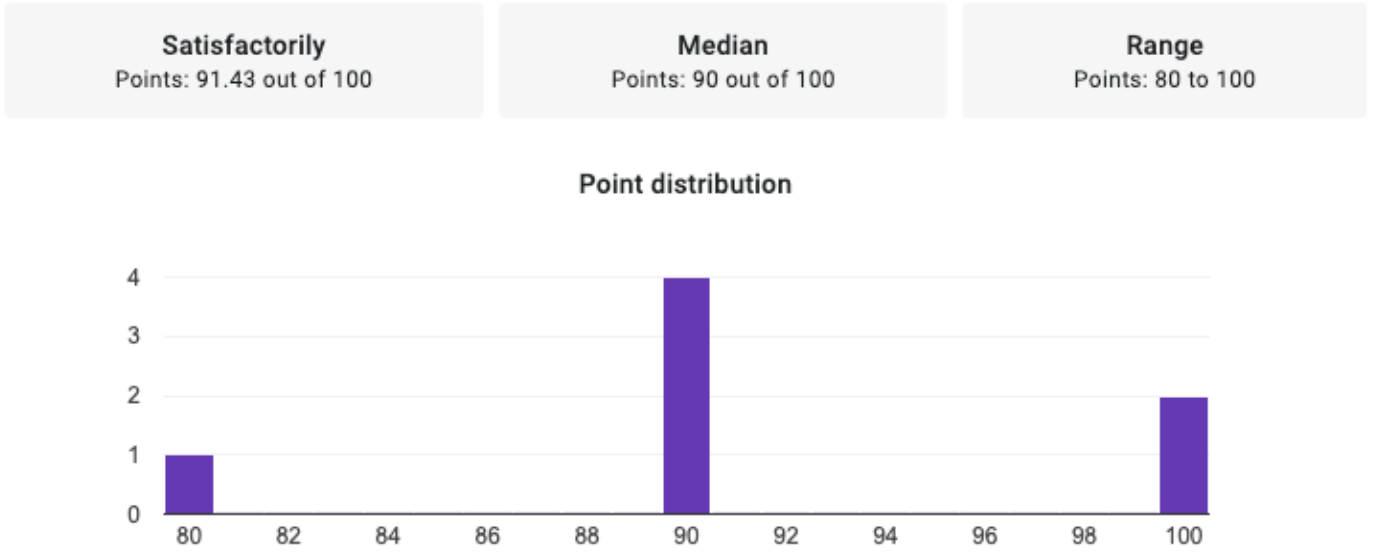
- 80% 10% 10% - Train Test Validation split
- Images are resized to 128 x 128
- Images are labelled accordingly to the file paths

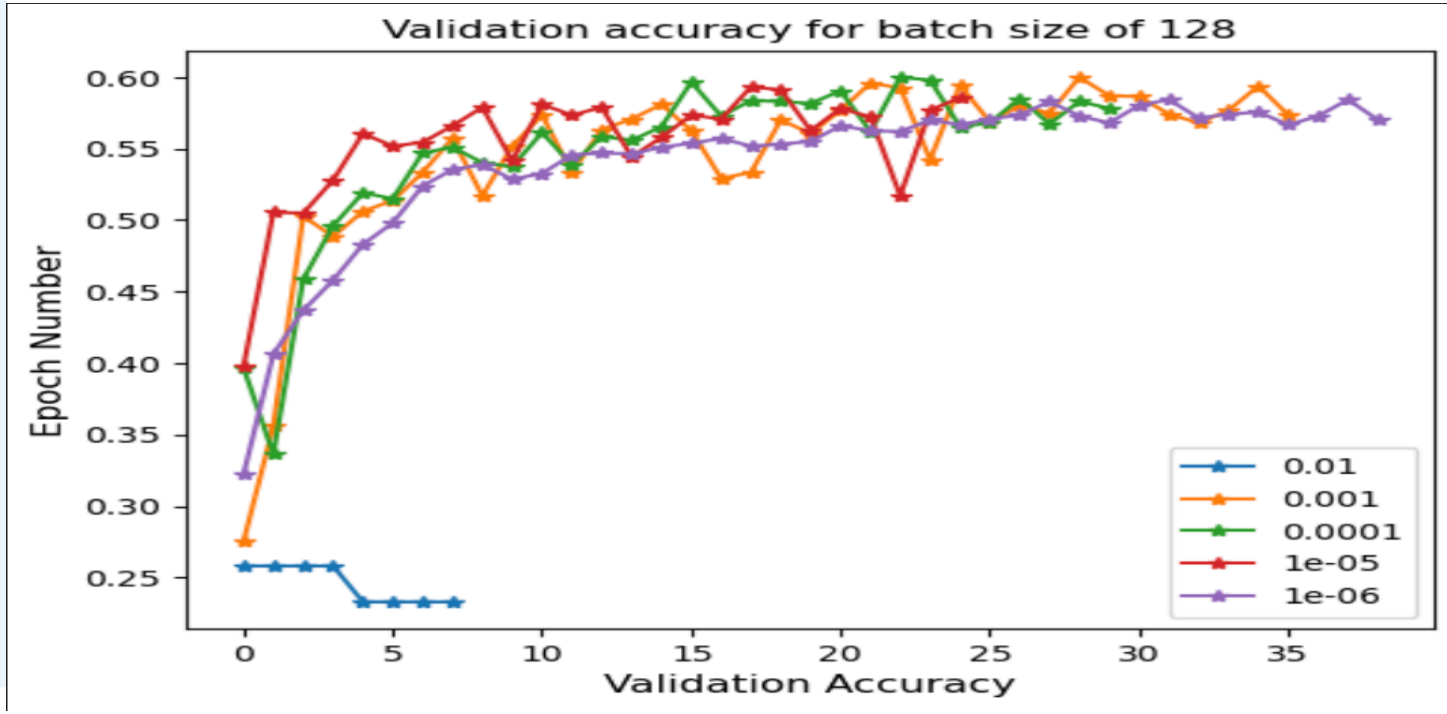
meat	fish	vegetables	fruits
0	1	0	0
0	0	0	1
1	0	0	0
0	0	0	1
1	0	0	0



PoC: Baselines

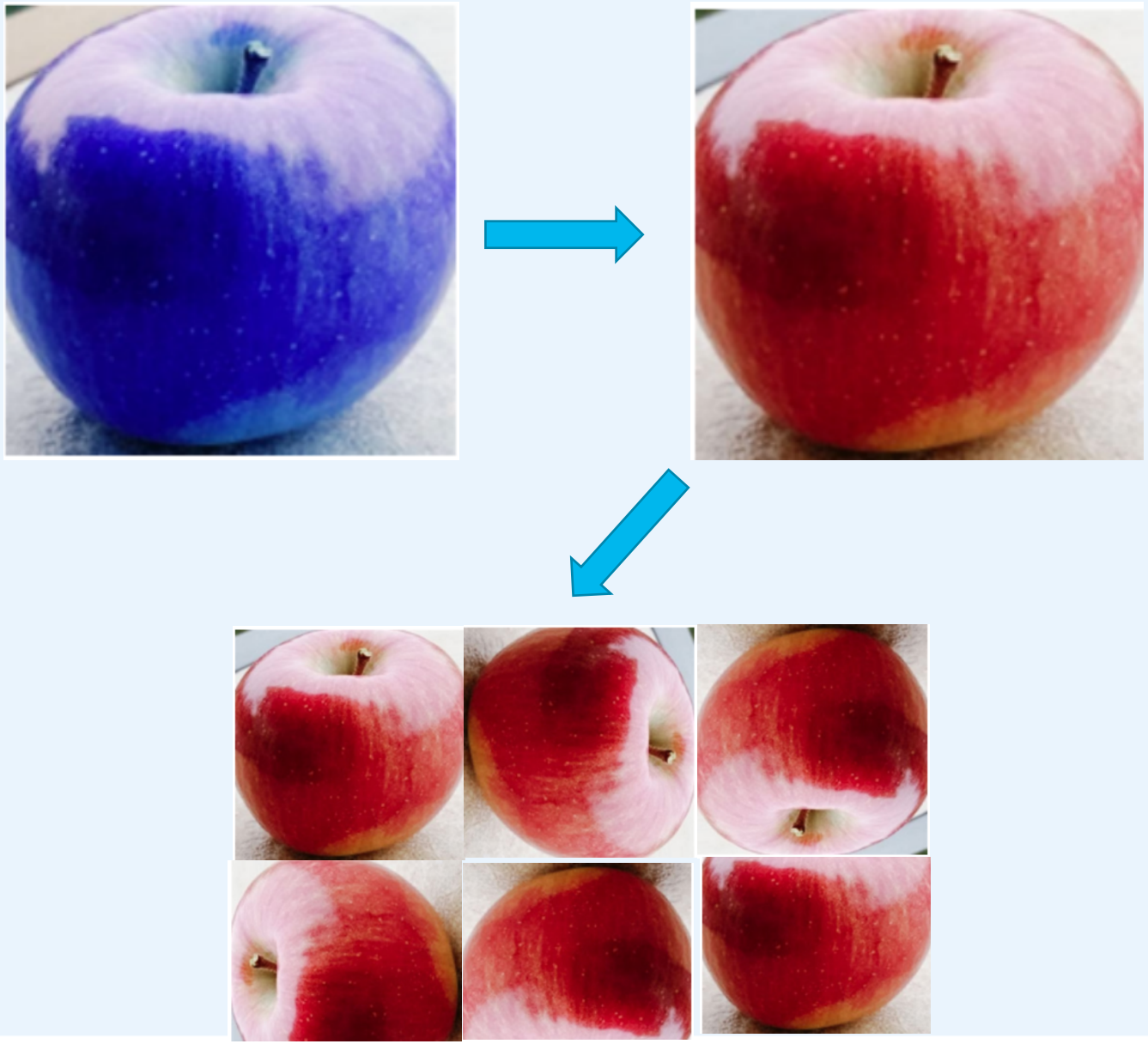
- Random Guess – 25%
- HLP – 90%
- Multilayer Perceptron Accuracy – 57% - 62%





PoC: Preprocessing

- Change colour scheme
- Resize the image by 128 x 128
- Apply data augmentation 1 => 10
- Load image into numpy array



PoC: CNN architecture

Base-line model

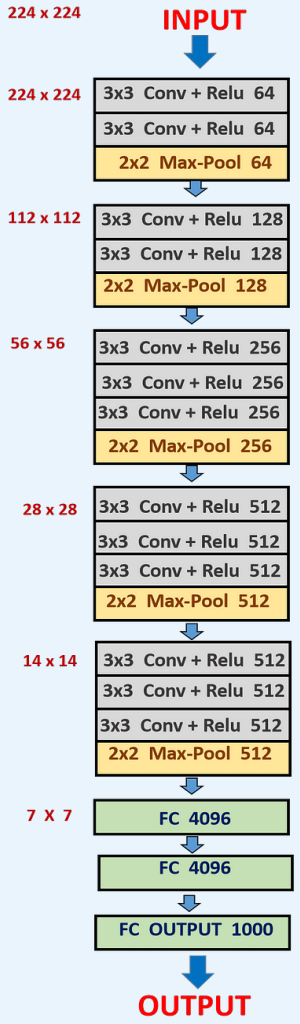
conv2d (Conv2D)	(None, 128, 128, 16)	208
max_pooling2d (MaxPooling2D)	(None, 64, 64, 16)	0
conv2d_1 (Conv2D)	(None, 64, 64, 16)	1040
max_pooling2d_1 (MaxPooling2D)	(None, 32, 32, 16)	0
dropout (Dropout)	(None, 32, 32, 16)	0
conv2d_2 (Conv2D)	(None, 32, 32, 16)	272
max_pooling2d_2 (MaxPooling2D)	(None, 16, 16, 16)	0
conv2d_3 (Conv2D)	(None, 16, 16, 8)	136
max_pooling2d_3 (MaxPooling2D)	(None, 8, 8, 8)	0
dropout_1 (Dropout)	(None, 8, 8, 8)	0
conv2d_4 (Conv2D)	(None, 8, 8, 4)	36
max_pooling2d_4 (MaxPooling2D)	(None, 4, 4, 4)	0
flatten (Flatten)	(None, 64)	0
dropout_2 (Dropout)	(None, 64)	0
dense (Dense)	(None, 32)	2080
dense_1 (Dense)	(None, 16)	528
dense_2 (Dense)	(None, 8)	136
dense_3 (Dense)	(None, 4)	36

Tuned base-line model

conv2d_5 (Conv2D)	(None, 128, 128, 16)	64
max_pooling2d_5 (MaxPooling2D)	(None, 64, 64, 16)	0
conv2d_6 (Conv2D)	(None, 64, 64, 32)	544
max_pooling2d_6 (MaxPooling2D)	(None, 32, 32, 32)	0
conv2d_7 (Conv2D)	(None, 32, 32, 8)	264
max_pooling2d_7 (MaxPooling2D)	(None, 16, 16, 8)	0
flatten_1 (Flatten)	(None, 2048)	0
dropout_3 (Dropout)	(None, 2048)	0
dense_4 (Dense)	(None, 4096)	8392704
dense_5 (Dense)	(None, 2048)	8390656
dense_6 (Dense)	(None, 1024)	2098176
dense_7 (Dense)	(None, 512)	524800
dense_8 (Dense)	(None, 256)	131328
dense_9 (Dense)	(None, 128)	32896
dense_10 (Dense)	(None, 32)	4128
dense_11 (Dense)	(None, 4)	132

PoC: CNN architecture

- Transfer learning model
- VGG16
- 4 Dense Layers



vgg16 (Functional)	(None, 4, 4, 512)	14714688
flatten_4 (Flatten)	(None, 8192)	0
dense_24 (Dense)	(None, 512)	4194816
dense_25 (Dense)	(None, 512)	262656
dense_26 (Dense)	(None, 512)	262656
dense_27 (Dense)	(None, 4)	2052

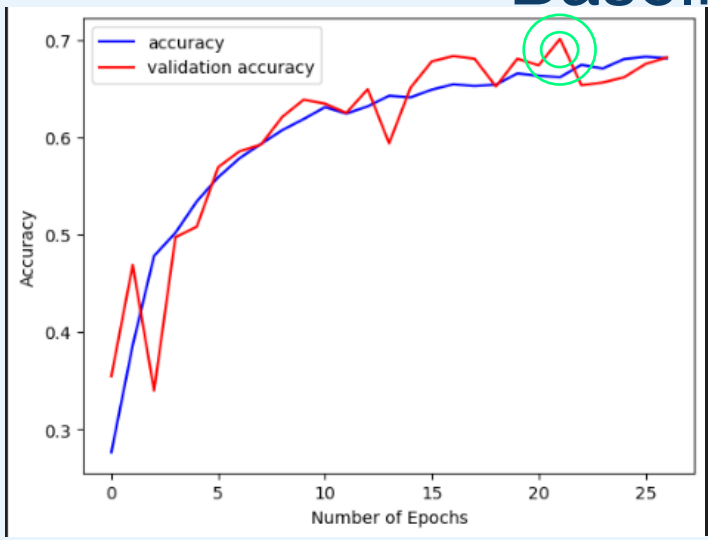
PoC: CNN training

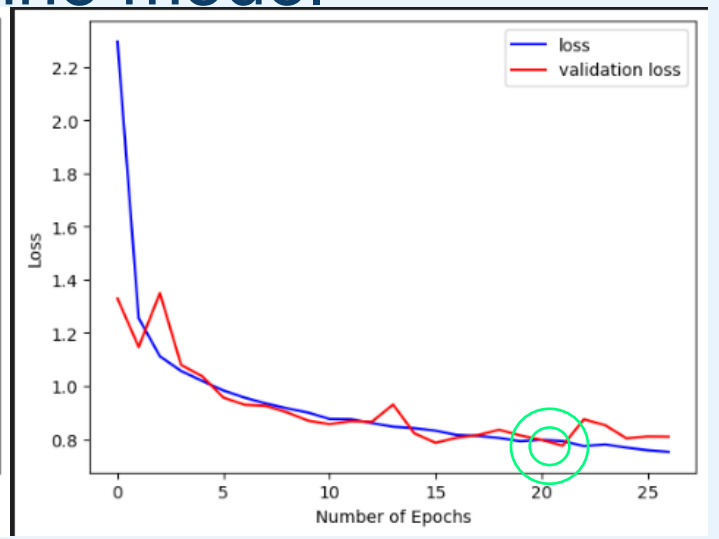
	Loss function	Optimizer	Learning rate	Metrics
Base-line model	Categorical cross-entropy	Adam	0.001	Loss & Accuracy
Improved base-line model	Categorical cross-entropy	Rmsprop	0.001	Loss & Accuracy
Transfer-learning model	Categorical cross-entropy	Adam	0.001	Loss & Accuracy

PoC: CNN training

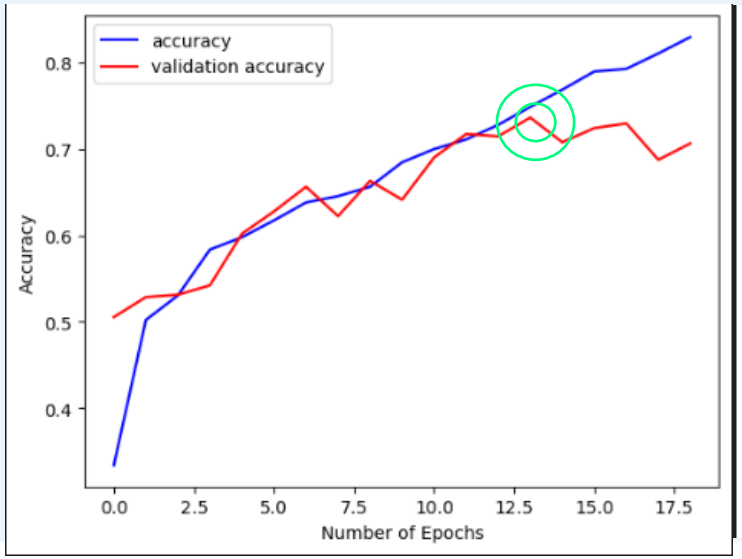
- 150 Epochs
- EarlyStopping monitoring val_accuracy with patience of 5
- Restore_best_weights = true
- Prevents overfitting

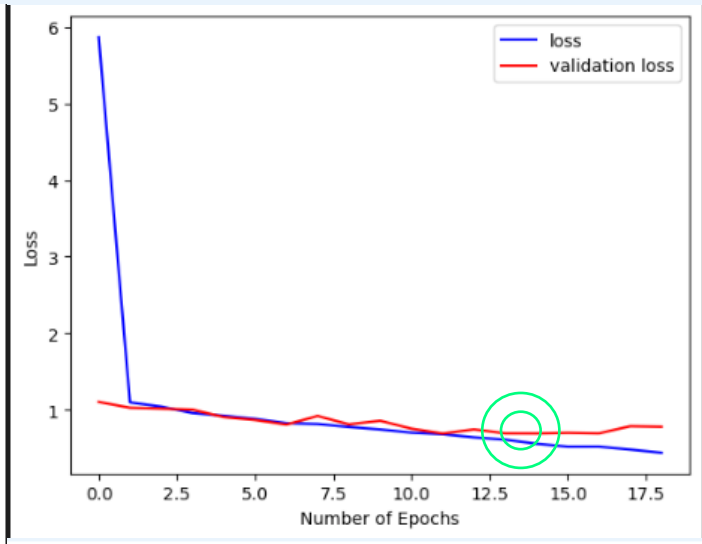
Baseline model





Improved baseline model



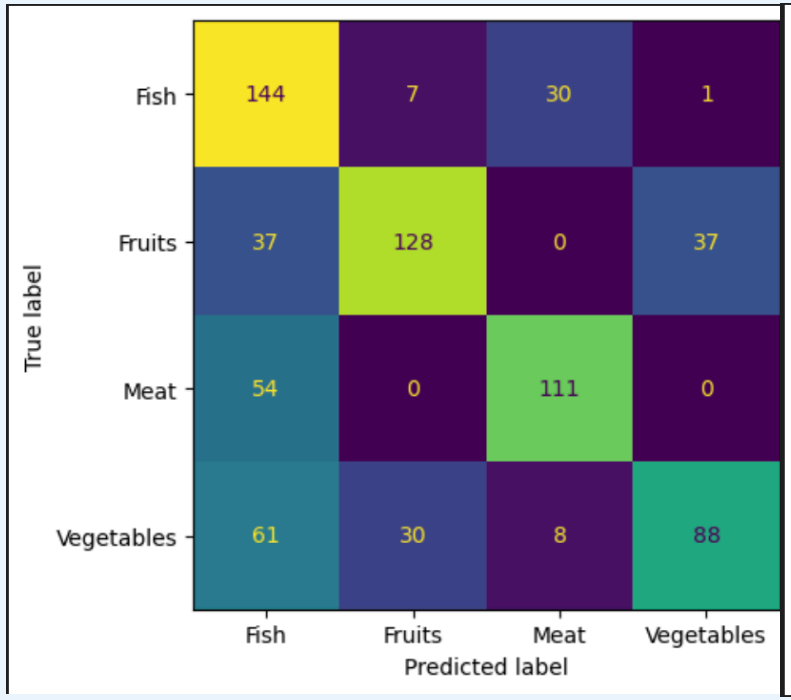


PoC: Model performance

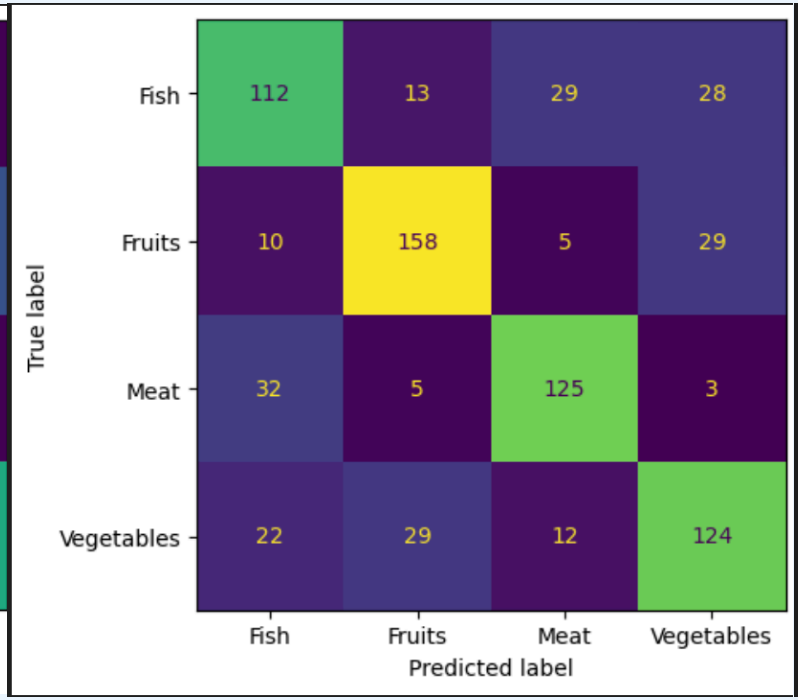
	Precision	Recall	F1-score	Accuracy on unseen data
Base-line model	0.67	0.64	0.64	0.65
Improved base-line model	0.70	0.71	0.7	0.72
Transfer-learning model	0.79	0.79	0.8	0.82

PoC: Model performance

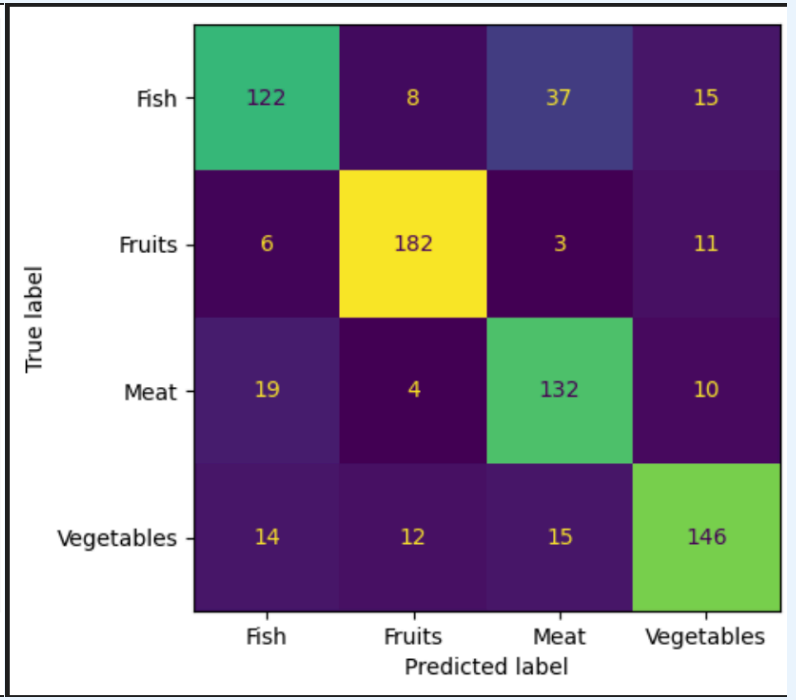
Base-line model



Tuned base-line model

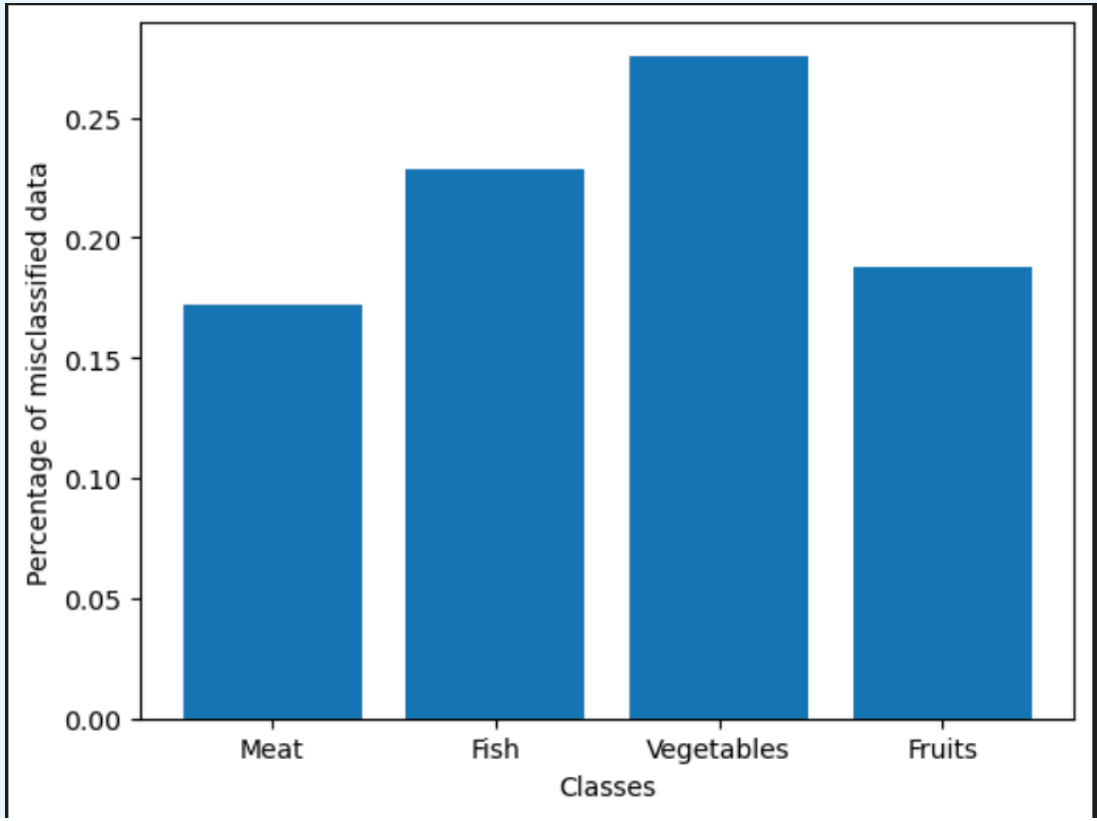


Transfer-learning model



PoC: Error analysis

- Most misclassified classes:
  - Vegetables – 26% of the data
  - Fish – 23% of the data
- 3 main reasons



PoC: Error analysis

Lack of data

Predicted Class: Meat

Noise in data

Predicted Class: Fish

Low-quality ('Garbage') data

Predicted Class: Fish

PoC: Iteration

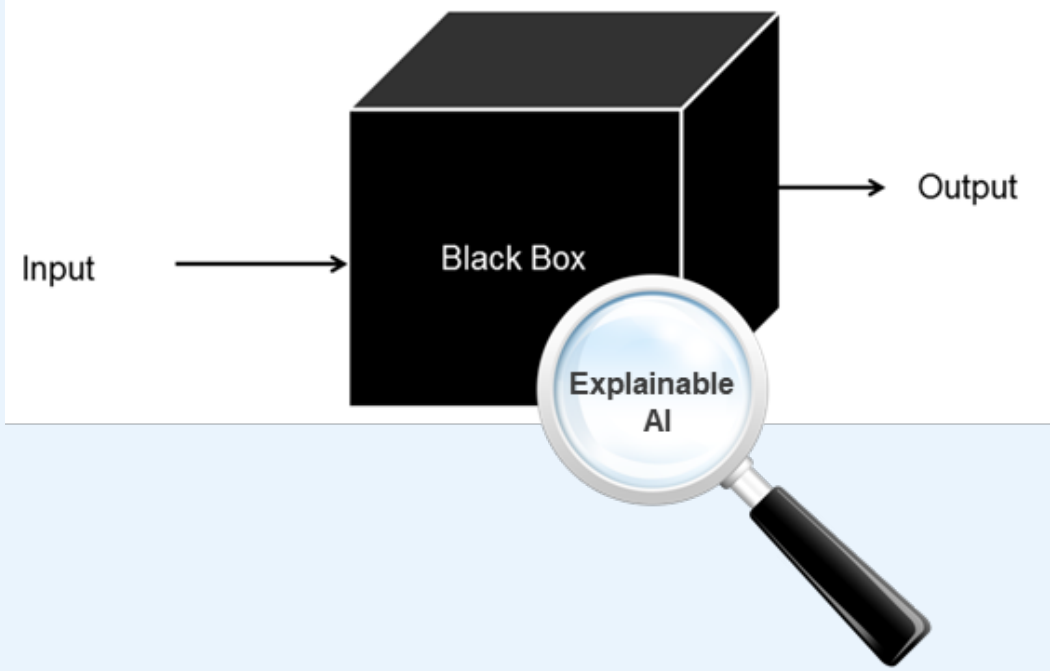
- Data Augmentation
- Transfer Learning
- Improve Data
- Model tunning

```
data_augmnetation = Sequential([
    layers.RandomFlip('horizontal_and_vertical'),
    layers.RandomRotation(0.45),
    layers.RandomTranslation(
        height_factor = (-0.2,0.2),
        width_factor = (-0.2,0.2),
        fill_mode='reflect',
        interpolation='nearest'
    )
])

def augment_image(image):

    with tf.device('/cpu:0'):

        return  data_augmnetation(image)
```

Responsible AI (Transparency & Interpretability)

Responsible AI (Transparency & Interpretability)

- Deep Learning models are hard to interpret
- Sometimes you have to sacrifice accuracy to explain the model
- Several methods are needed to understand what is under the hood
- Accuracy > Explainability

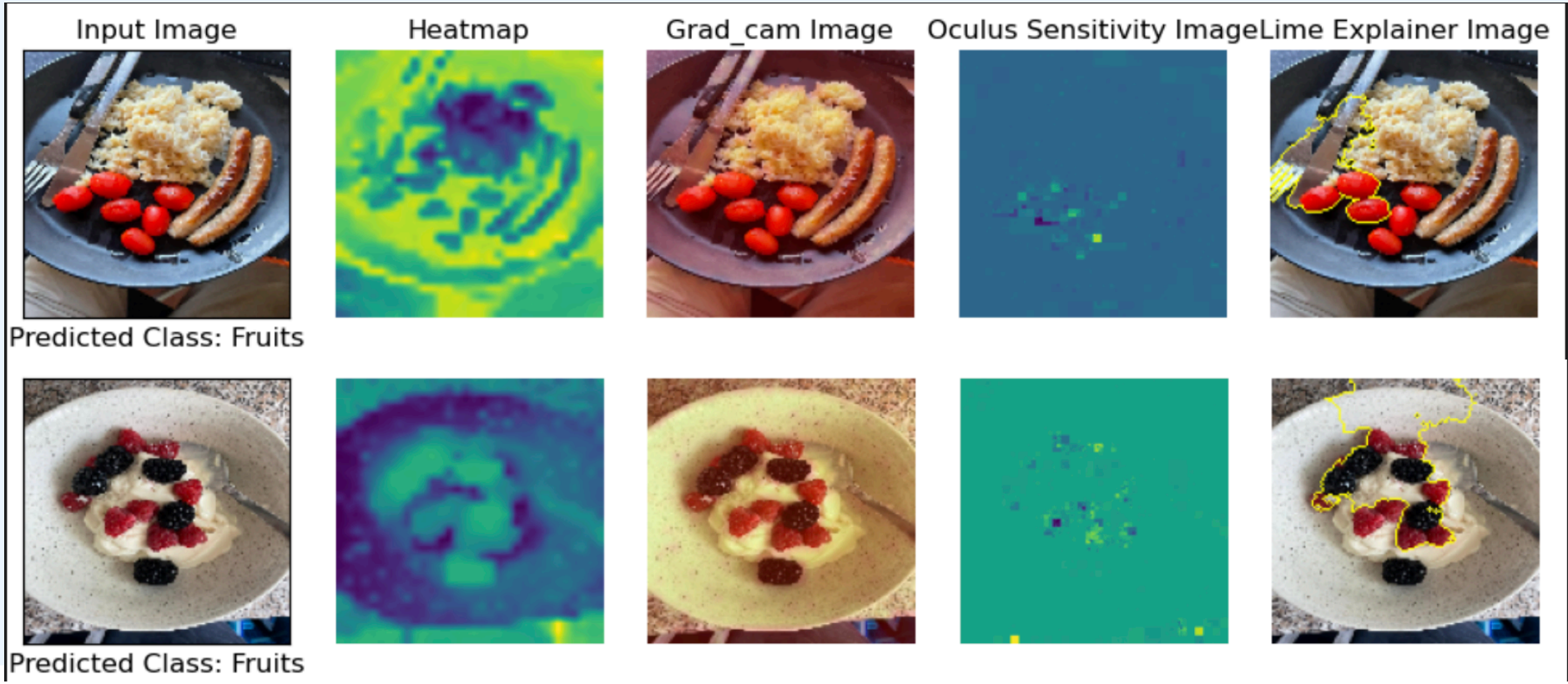


Responsible AI

- Accuracy – 75%

- Outputs are explained with 3 different methods:
  - Grad_cams
  - Oculus sensitivity
  - Lime explainer

Tuned base-line model explained by XAI methods



Disruptive Technology Risks

Disruptive Technology Risks

Risks

- Inaccurate model => Inaccurate carbon footprint
- Inaccurate carbon footprint measurement data for different actions => Useless tips on how to reduce personal footprint
- The growth in the interest in ecology and in the number of people who want to make a difference => Market is not ready

Contingency Plan

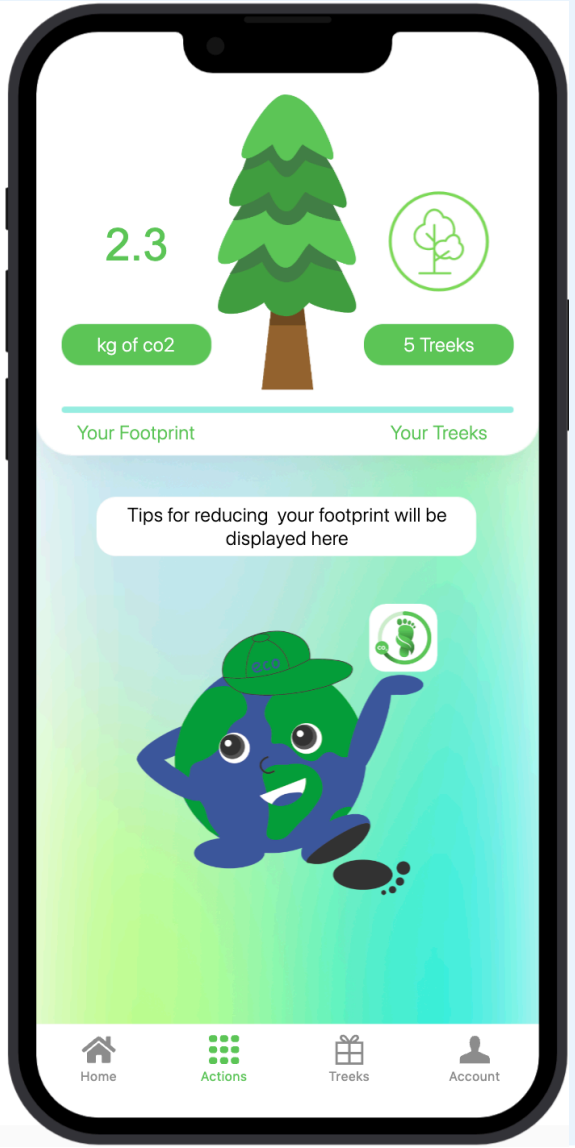
- Constant improvement of the model used for food labeling
- Conduct research on a carbon footprint emissions caused by a broad range of different actions
- Integrate & promote eco-friendly habits step by step to promote investments in this area of the market and increase it capacities

HCAI - Application Design:

HCAI - Application Design:

High-level concept design:

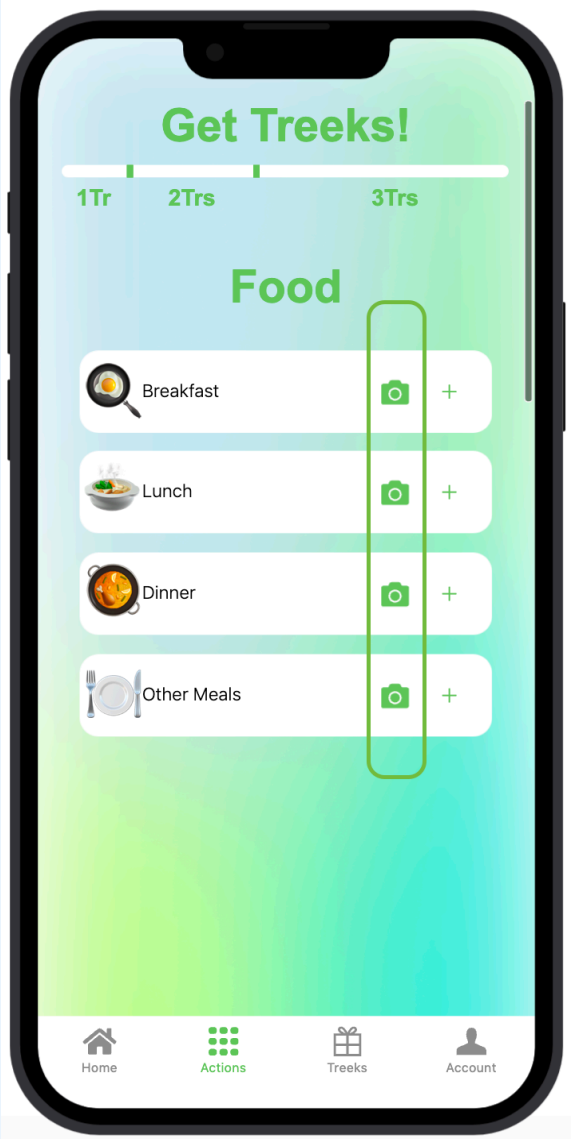
- Users - Environmentally conscious people
- Problem – lack of tools to track personal carbon footprint
- Mobile app – is a personal carbon footprint tracker, that also helps to reduce your footprint
- Daily interaction
- Colour scheme – eco-related colours

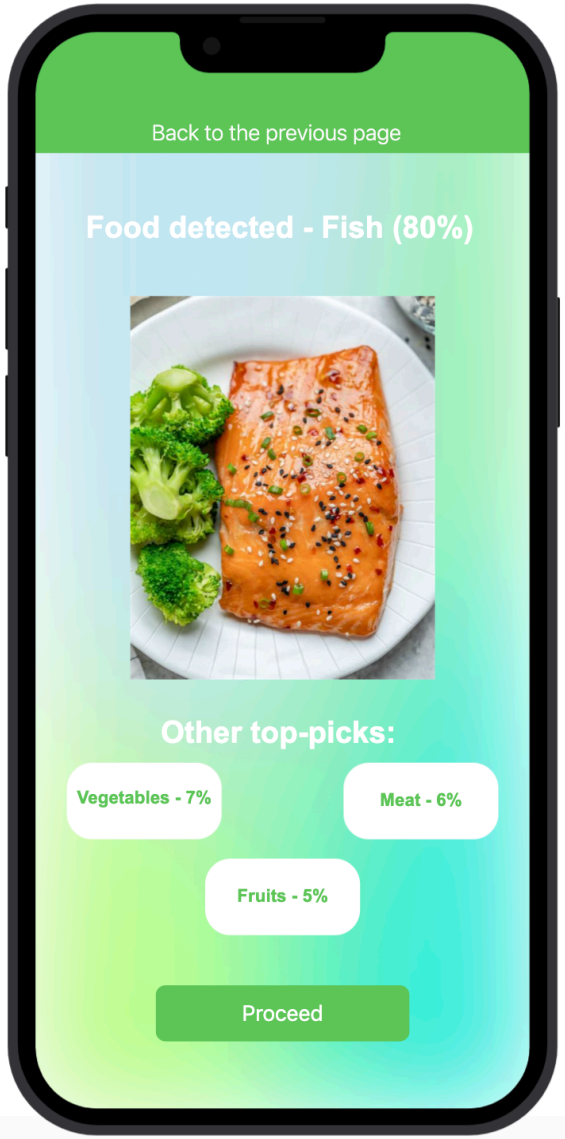


HCAI - Application Design

Algorithm Embedding:

- The algorithm is embedded in one of the app screens, that can be accessed from the actions screen
- The algorithm will label food on the meal picture taken by a user
- Object detection and weight assumption

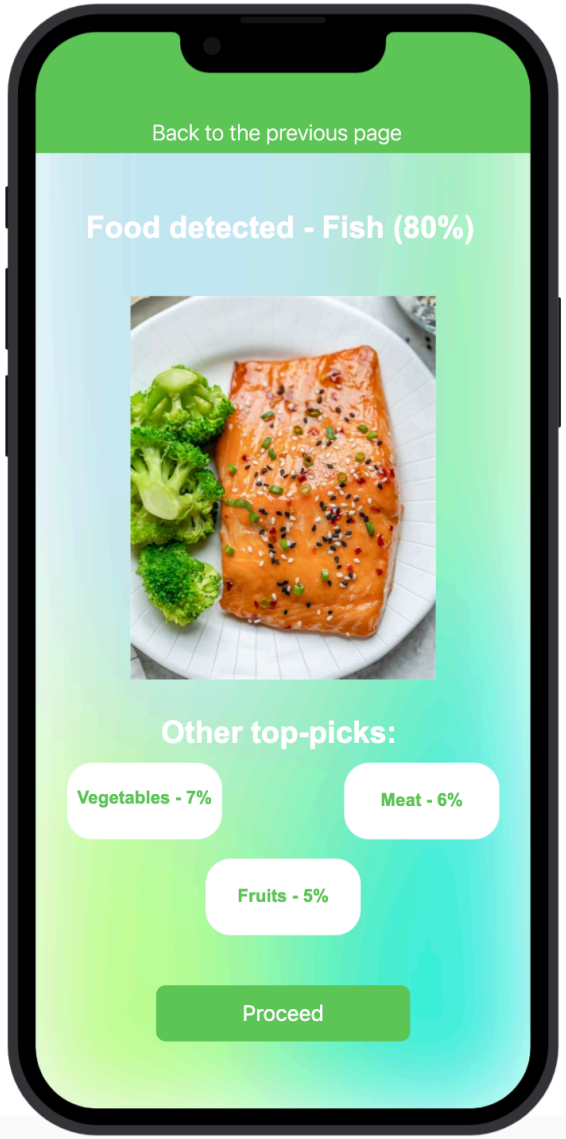




HCAI - Application Design:

Continuous Learning

- Users can correct the model's predictions by simply clicking on one of 3 buttons to change the food label
- Buttons are sorted in descending order from highest probability to lowest
- If the label is changed => Algorithm will adjust its predictions based on the feedback from the user



HCAI - Application Design:

- Several user testing methodologies:
  - A/B Testing
  - Paper prototyping
  - Users feedback
  - Samples T-test

#	Field	Very much agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Very much disagree	Total
1	I understood what I could use the app for	50.00% 4	0.00% 0	12.50% 1	37.50% 3	0.00% 0	0.00% 0	0.00% 0	8
2	I found the application intuitive to use	25.00% 2	37.50% 3	12.50% 1	25.00% 2	0.00% 0	0.00% 0	0.00% 0	8
3	I thought the application was useful	37.50% 3	12.50% 1	12.50% 1	25.00% 2	12.50% 1	0.00% 0	0.00% 0	8
4	I enjoyed using the application	25.00% 2	25.00% 2	0.00% 0	25.00% 2	12.50% 1	0.00% 0	12.50% 1	8
5	The computer vision function was intuitive and easy to use	37.50% 3	37.50% 3	0.00% 0	12.50% 1	12.50% 1	0.00% 0	0.00% 0	8

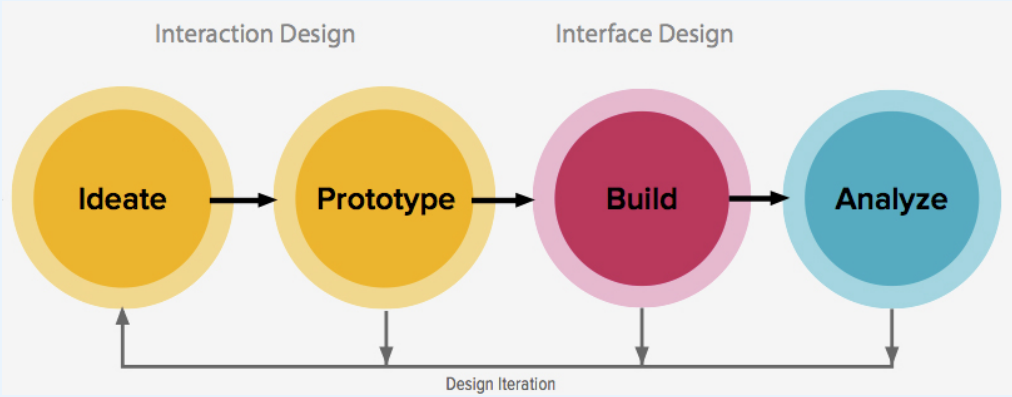
#	Field	Very much agree	Agree	Somewhat agree	Neutral	Somewhat disagree	Disagree	Very much disagree	Total
5	The computer vision function was intuitive and easy to use	50.00% 5	40.00% 4	10.00% 1	0.00% 0	0.00% 0	0.00% 0	0.00% 0	10
1	I understood what I could use the app for	80.00% 8	20.00% 2	0.00% 0	0.00% 0	0.00% 0	0.00% 0	0.00% 0	10
3	I thought the application was useful	50.00% 5	40.00% 4	10.00% 1	0.00% 0	0.00% 0	0.00% 0	0.00% 0	10
2	I found the application intuitive to use	70.00% 7	20.00% 2	10.00% 1	0.00% 0	0.00% 0	0.00% 0	0.00% 0	10
4	I enjoyed using the application	60.00% 6	20.00% 2	10.00% 1	10.00% 1	0.00% 0	0.00% 0	0.00% 0	10

HCAI - Application Design:

- Several UI design iterations

- Improved aspects:

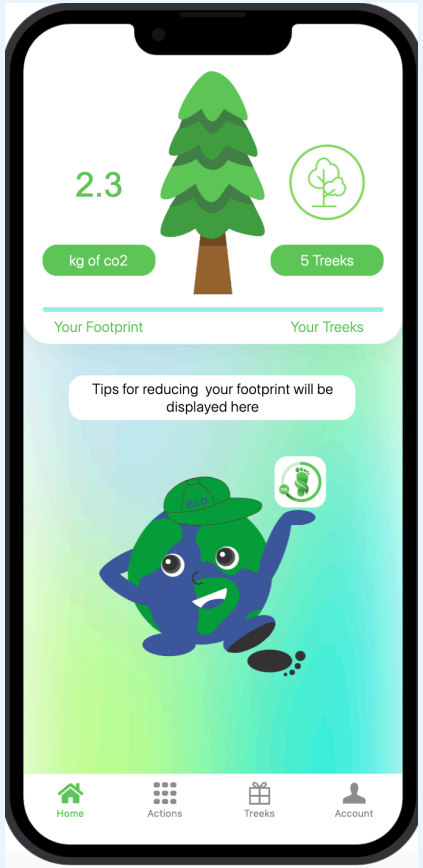
  - UI design
  - User Experience
  - Algorithm implementation
  - Recommendations and continuous learning

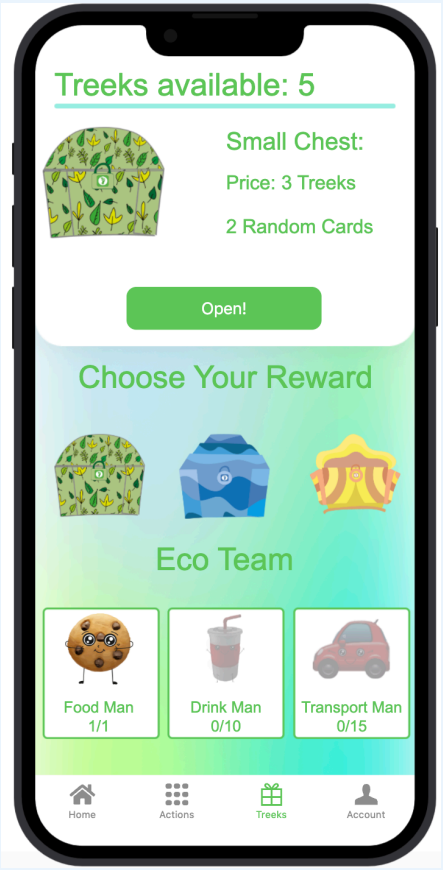


HCAI - Application Design:

Low-level design:

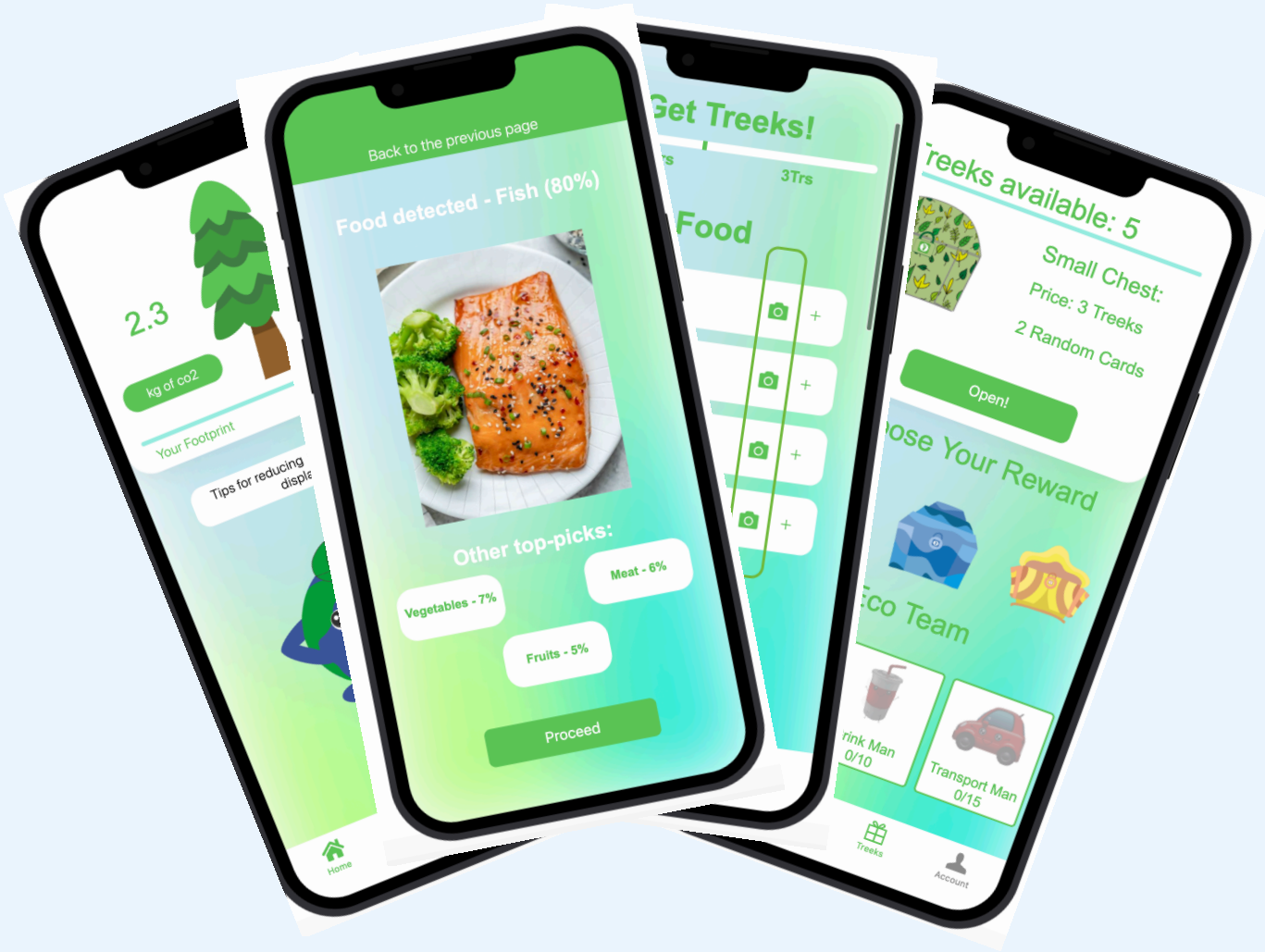
- The application provides personal recommendations on how to reduce personal carbon footprint
- Users will get notifications twice a day with a reminder to track their footprint
- Rewards system is integrated in the app to increase engagement rate
- To show the impact on the planet there is a tree indicator that slowly dies as the CO2 footprint increases





HCAI - Application Design:

- App Prototype Zip
- App Prototype Html File
- App Prototype Folder Unzipped
- Prototype Showcase
- Proto.io prototype



Breda University OF APPLIED SCIENCES

Thank you!

Any questions?