

# frezza

## Fruit Freshness Detector

2502032125 – Alyza Rahima Pramudya

2501962851 – Dierta Pasific

2540133234 – Jacqueline Abyasa

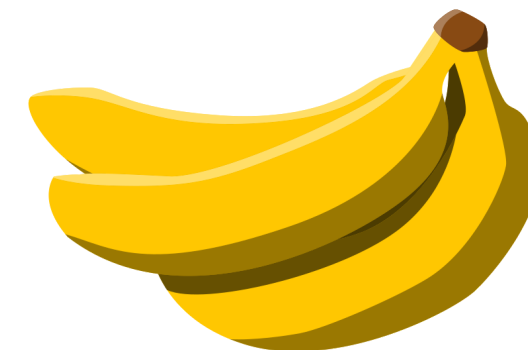
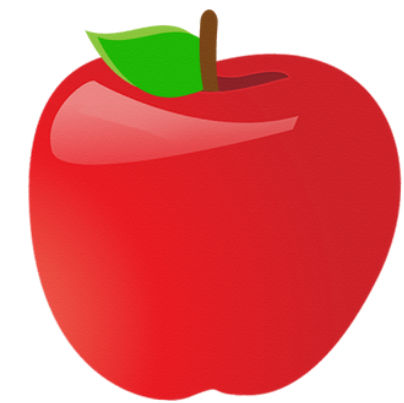
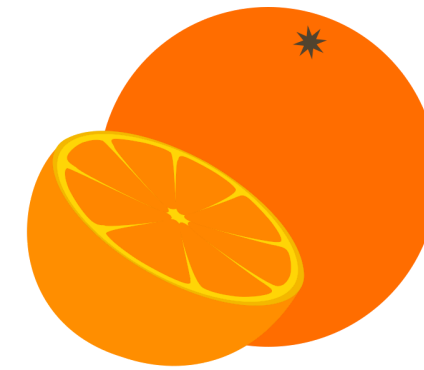
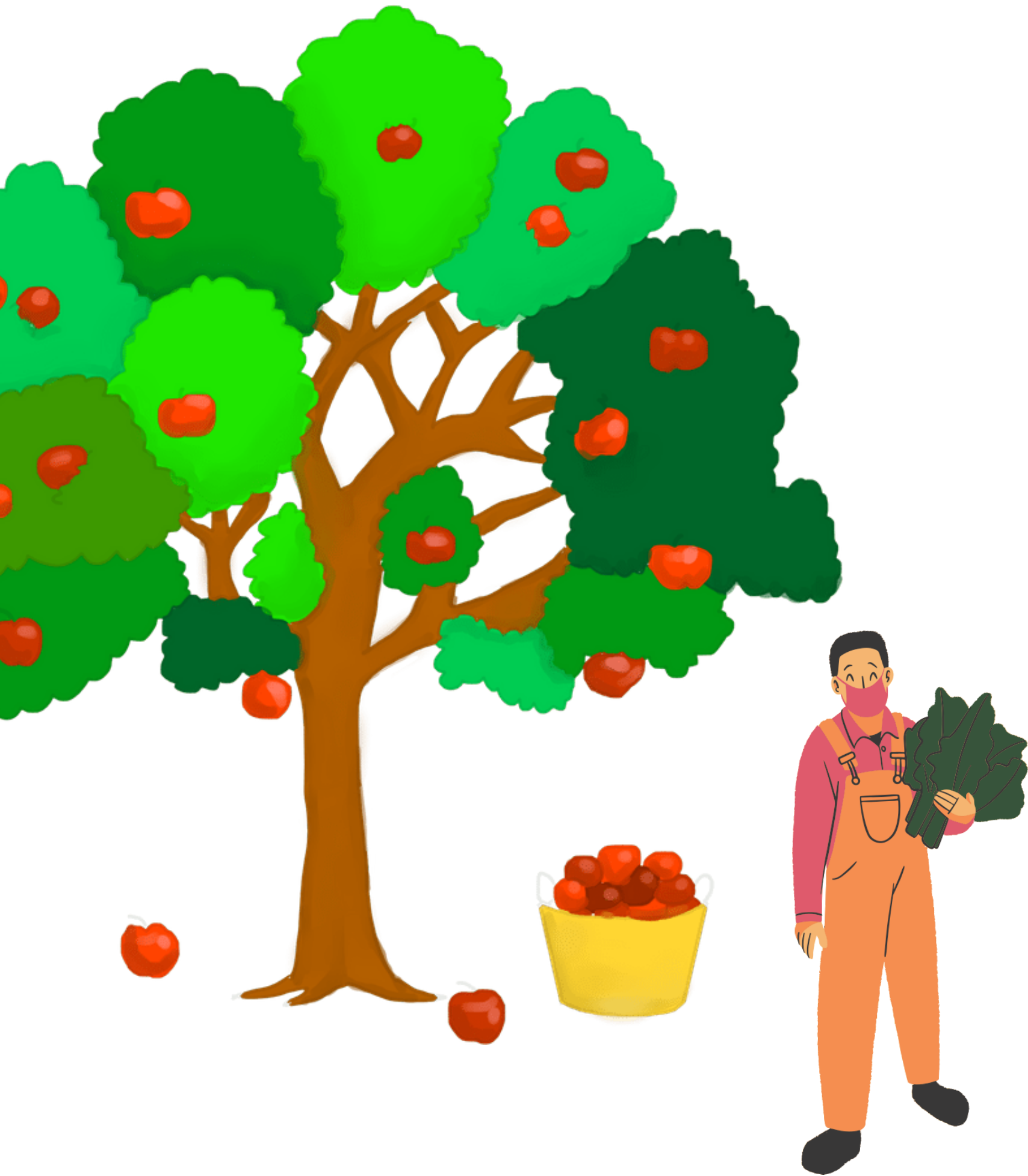
2502009173 – Shafa Amira Qonitatin

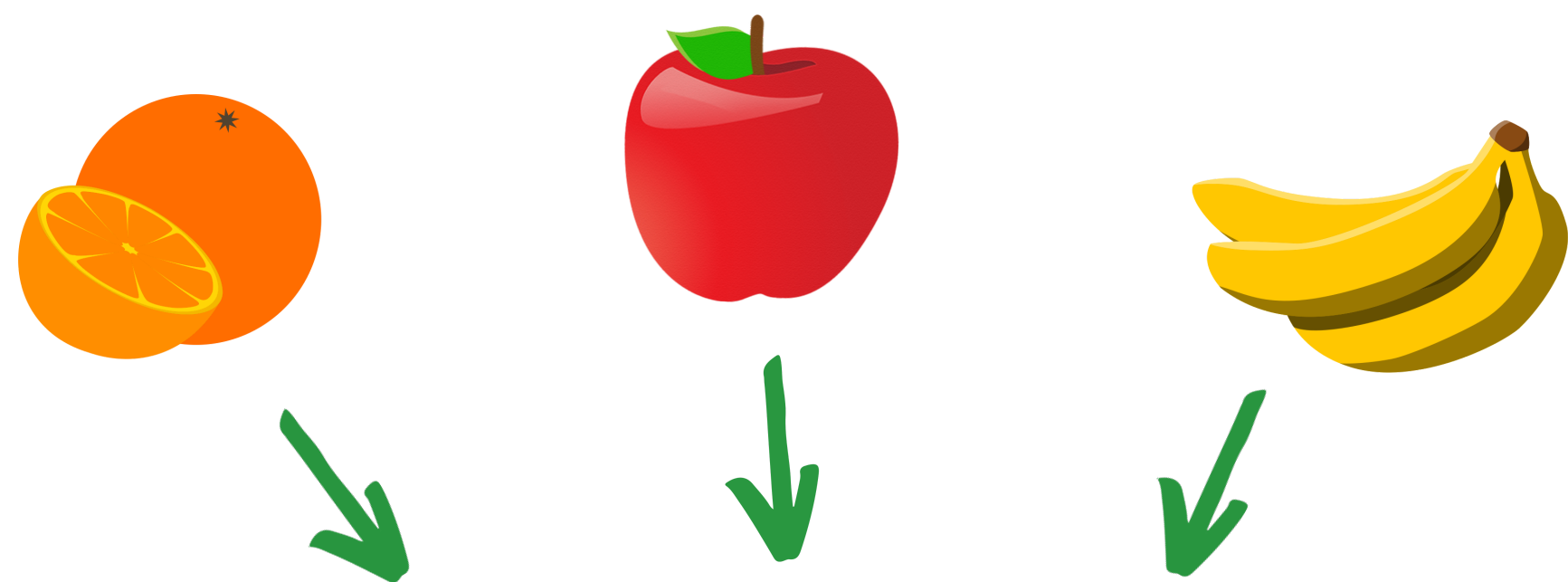
**Kelompok 3**

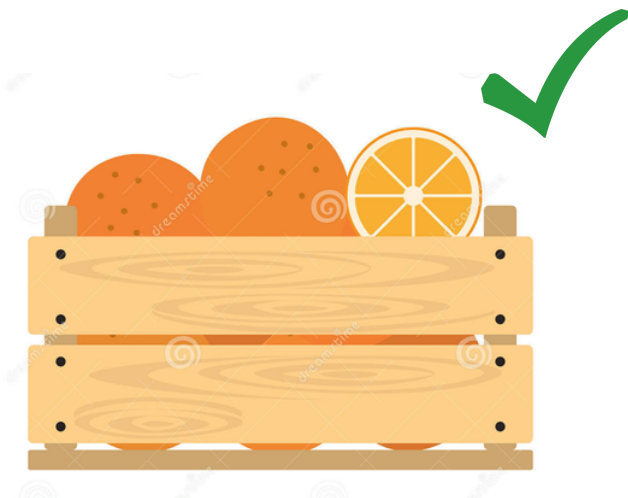
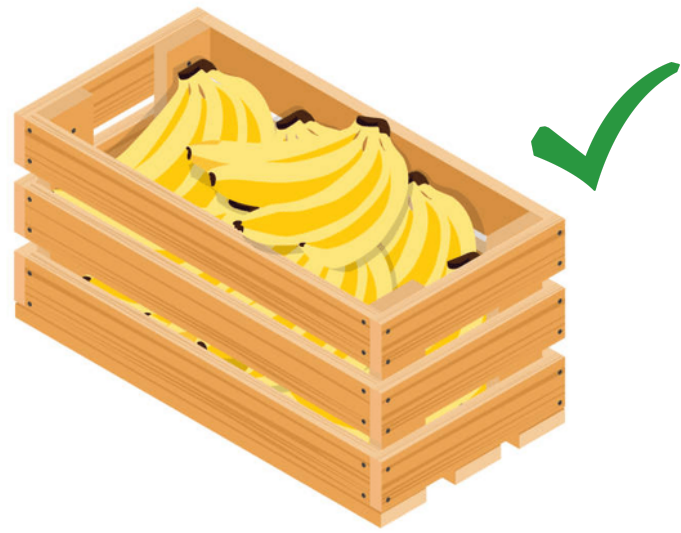
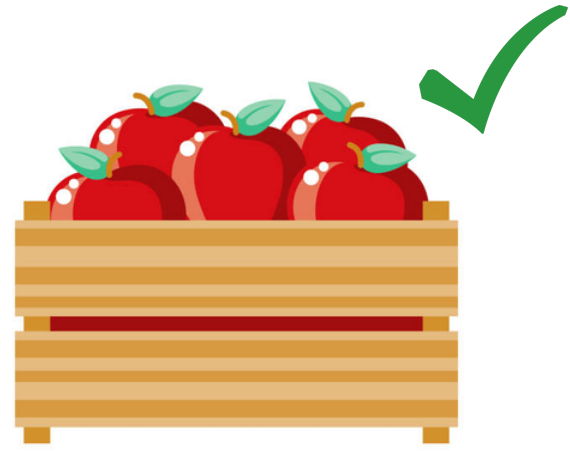
**LC75**

# Project Background

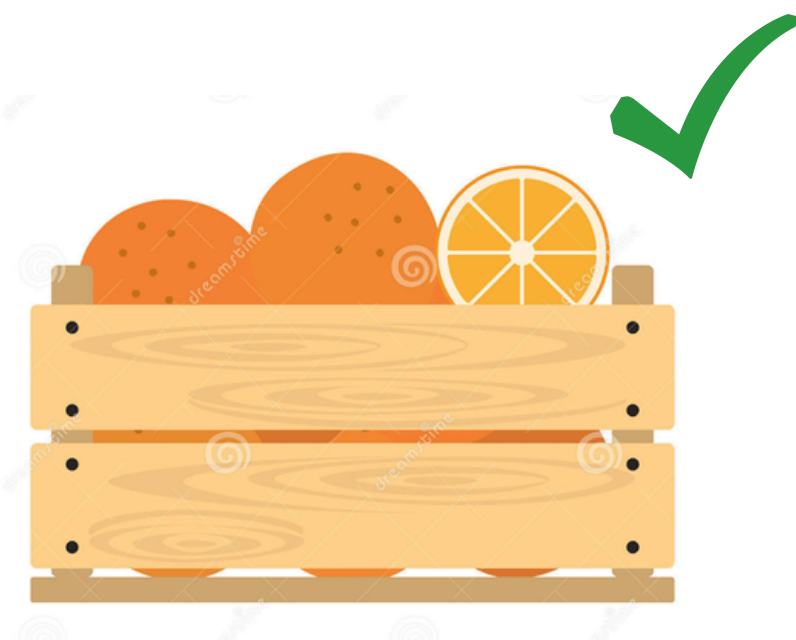
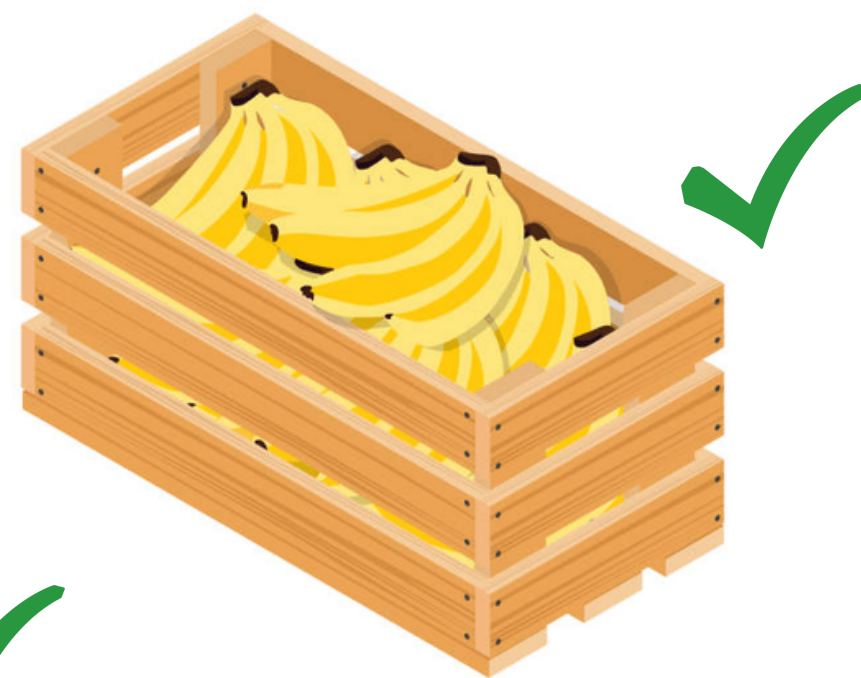


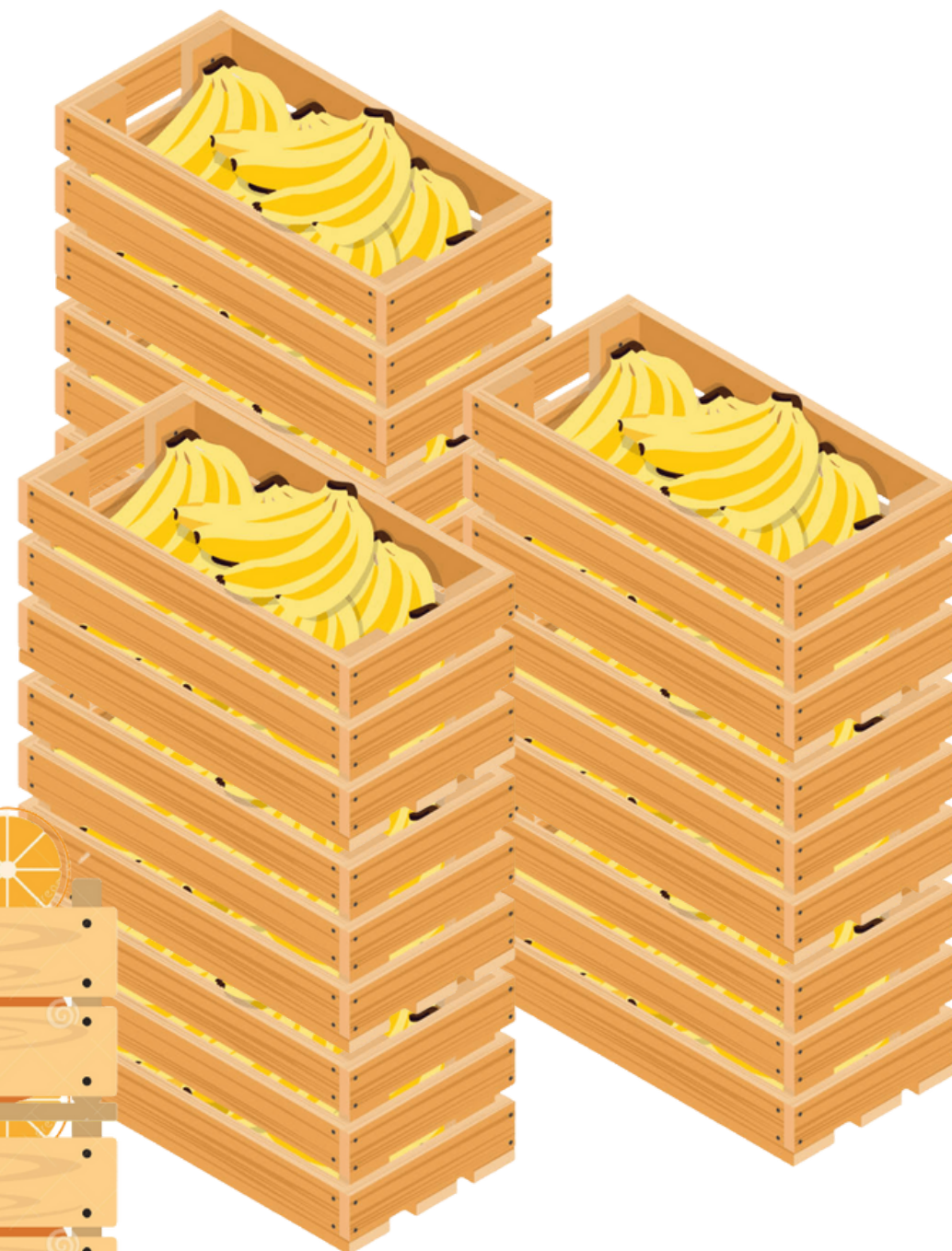
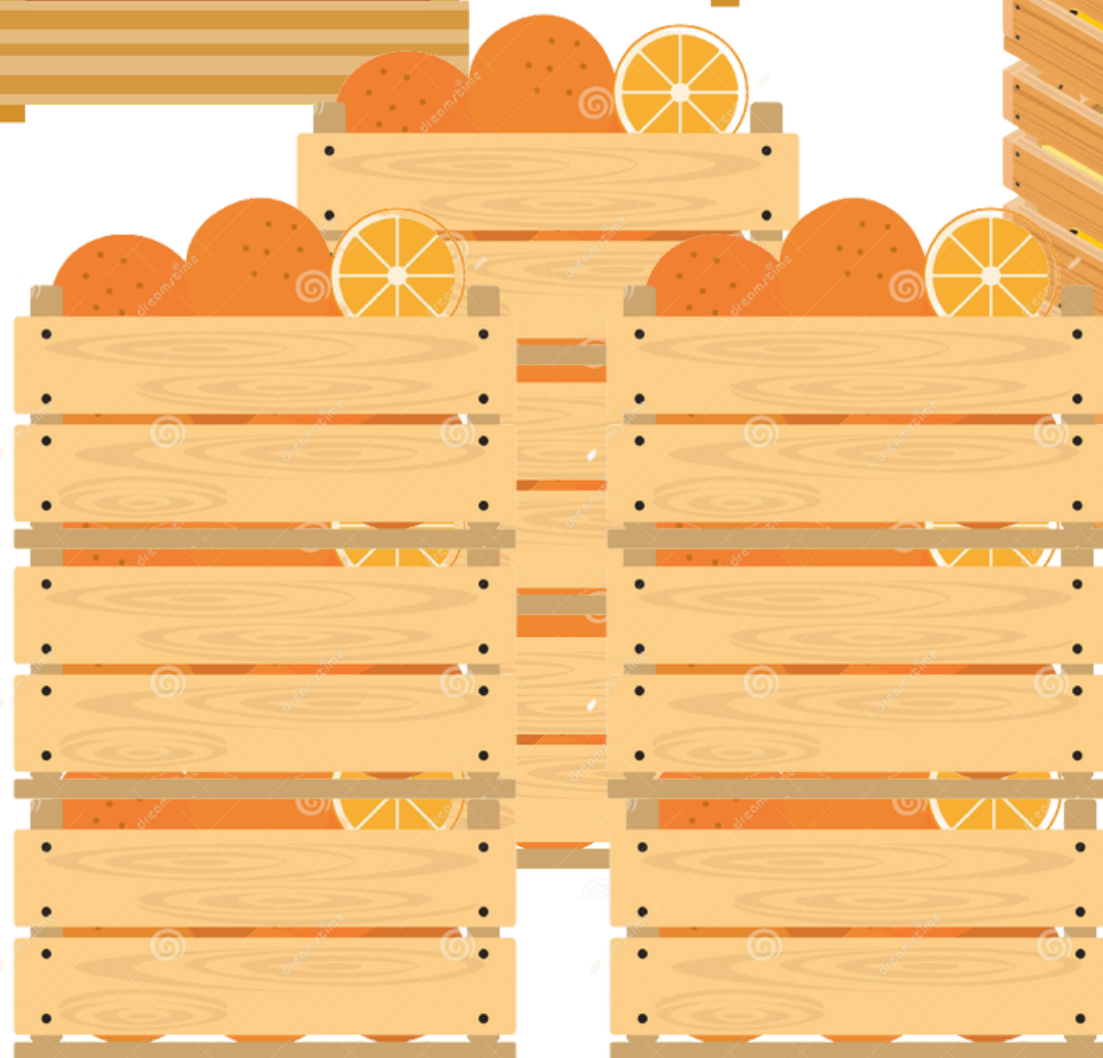
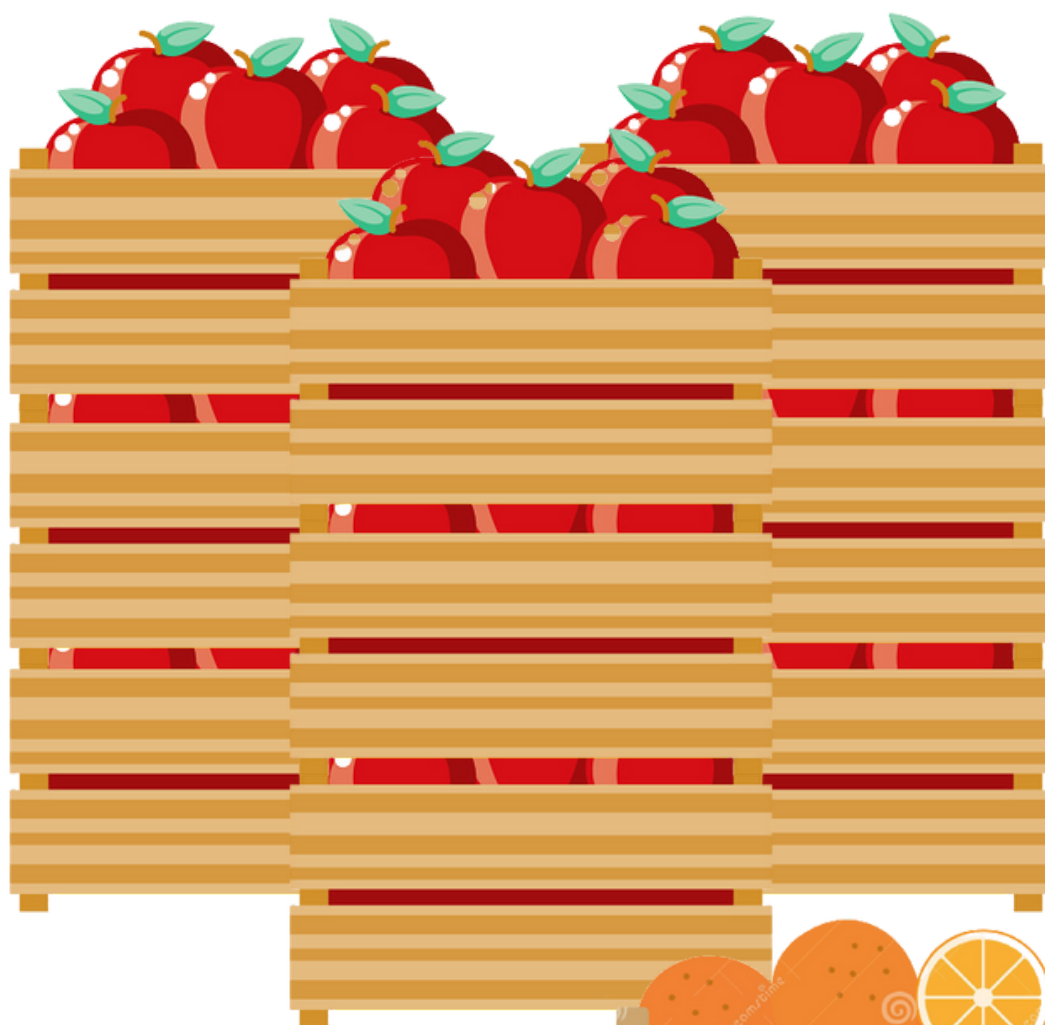














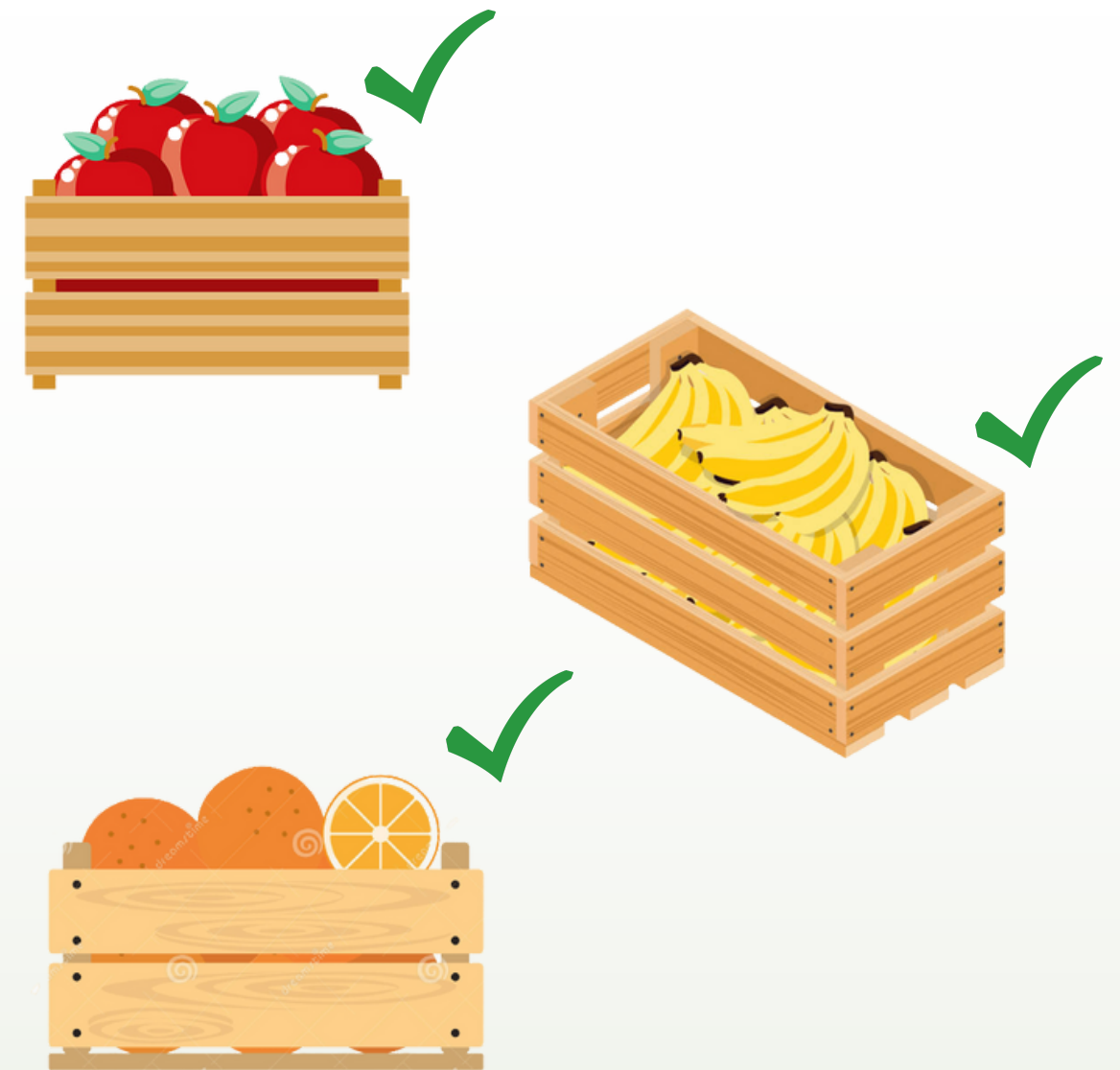
**SOLUTION**

**frezz**

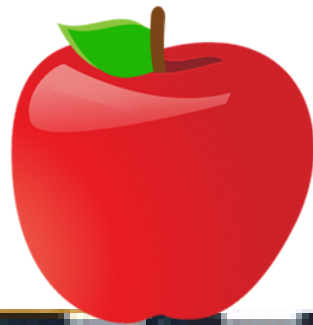
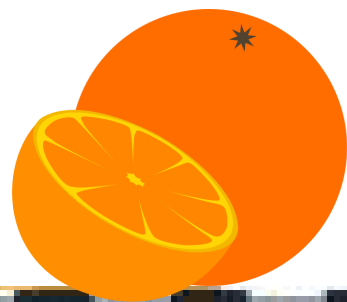
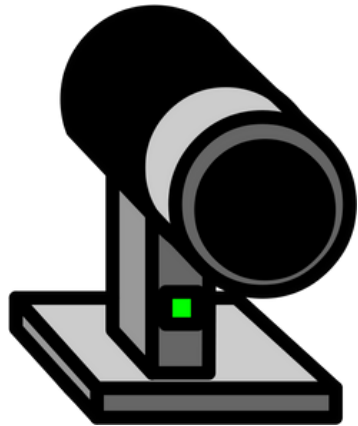
**Memudahkan supermarket atau  
petani buah dalam melakukan  
Quality Control terhadap kesegaran  
buah.**

## Kenapa ada Quality Control?

- Meningkatkan nilai dan kualitas produk yang dijual
- Menjamin kesegaran buah yang diterima
- Mencegah transportasi buah yang tidak layak (biaya transportasi tidak terbuang pada buah yang tidak layak dijual)



camera





## PEAS

**Performance:** Accuracy, Ease of use

**Environment:** Conveyor belt

**Actuators:** Software/Website

**Sensors:** Kamera

## **Key points**

**Efisien**

**Mudah digunakan**

**Akurat**

# TARGET MARKET





# Business Model



Mempermudah Petani buah untuk  
quality control sebelum jual ke  
supermarket

or

Mempermudah Supermarket untuk  
quality control sebelum jual ke  
customers

# PROJECT FLOW

# Featured Technologies:



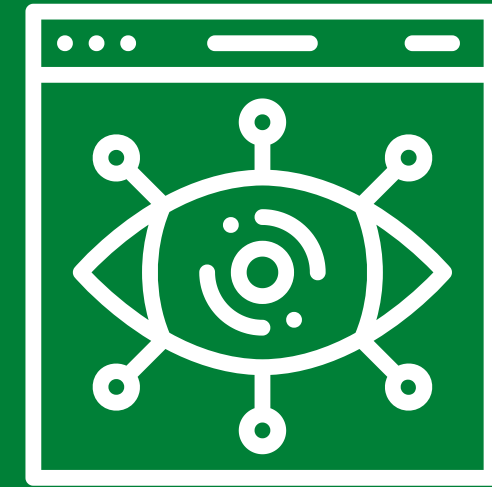
## Deep Learning

A Subset of AI that uses multi-layers neural networks that learn from lots of data.



## Web Development

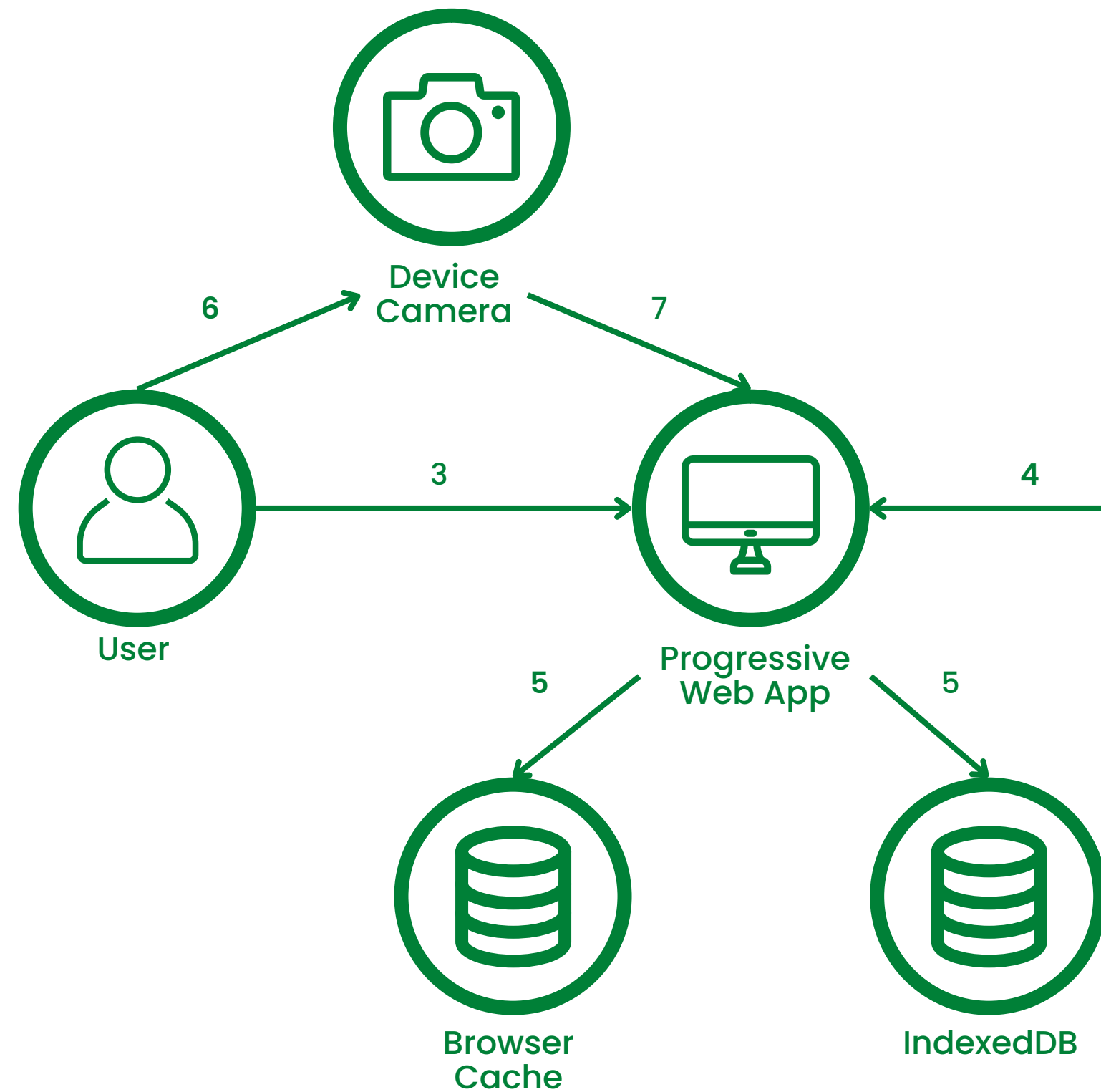
The construction of modern web apps using open-standards technologies



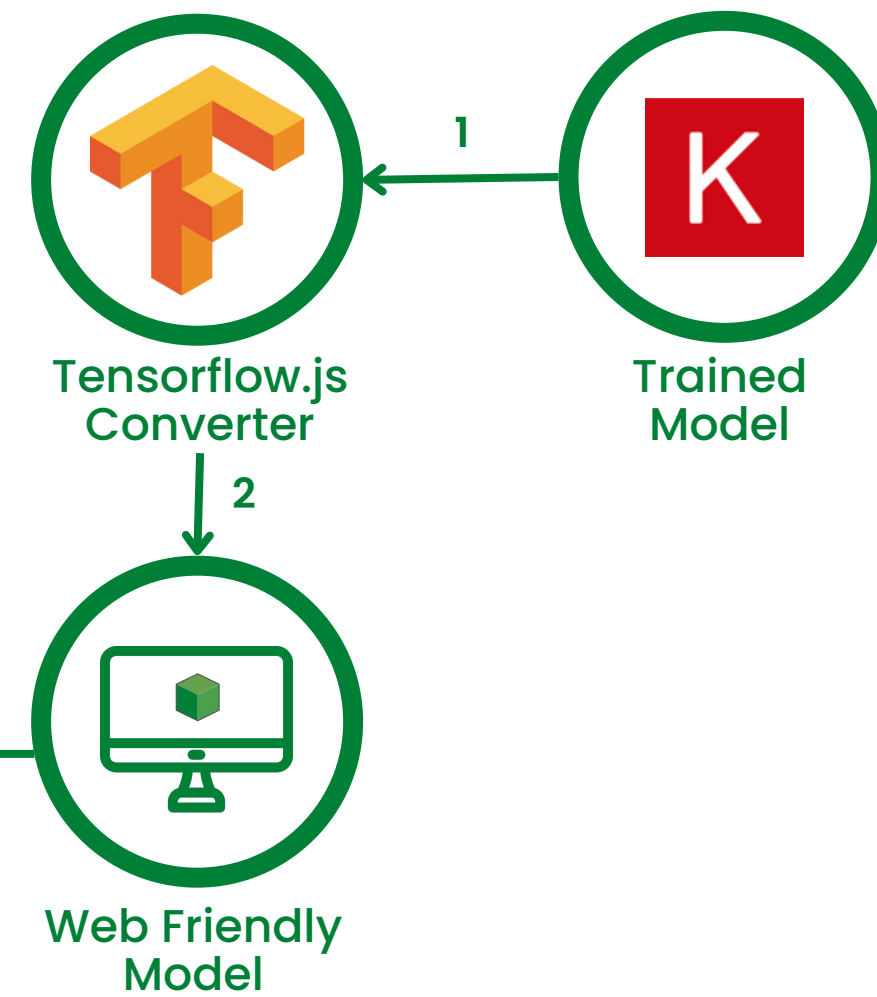
## Visual Recognition

Tag, classify, and train visual content using machine learning.

## Browser Environment



## Python Environment





# Dataset (Kaggle)

≡ kaggle

+ Create

🏠 Home

🏆 Competitions

📊 Datasets

<> Code

💬 Discussions

🎓 Learn

▼ More

🔍 Search



SRIRAM REDDY KALLURI · UPDATED 4 YEARS AGO



242

New Notebook

## Fruits fresh and rotten for classification

Apples Oranges Bananas

Data Card

Code (50)

Discussion (1)

<https://www.kaggle.com/datasets/sriramr/fruits-fresh-and-rotten-for-classification>

# Model Training (Google Colab)

```
[ ] from google.colab import files
files.upload() # Run and select the kaggle.json file downloaded
!ls -lha kaggle.json
!pip install -q kaggle
!mkdir -p ~/.kaggle
!cp kaggle.json ~/.kaggle/
!chmod 600 ~/.kaggle/kaggle.json
! kaggle datasets download -d sriramr/fruits-fresh-and-rotten-for-classification
```

No file chosen      Upload widget is only available when the cell has been executed and the file has been shared on Google Drive.

Saving kaggle.json to kaggle.json  
-rw-r--r-- 1 root root 65 Dec 12 01:01 kaggle.json

Downloading fruits-fresh-and-rotten-for-classification.zip to /content  
100% 3.58G/3.58G [00:35<00:00, 114MB/s]  
100% 3.58G/3.58G [00:35<00:00, 109MB/s]

```
[ ] import zipfile
import os

curDir = os.getcwd()
print(curDir)

local_zip = '/content/fruits-fresh-and-rotten-for-classification.zip'
zip_ref = zipfile.ZipFile(local_zip, 'r')
zip_ref.extractall('/content')

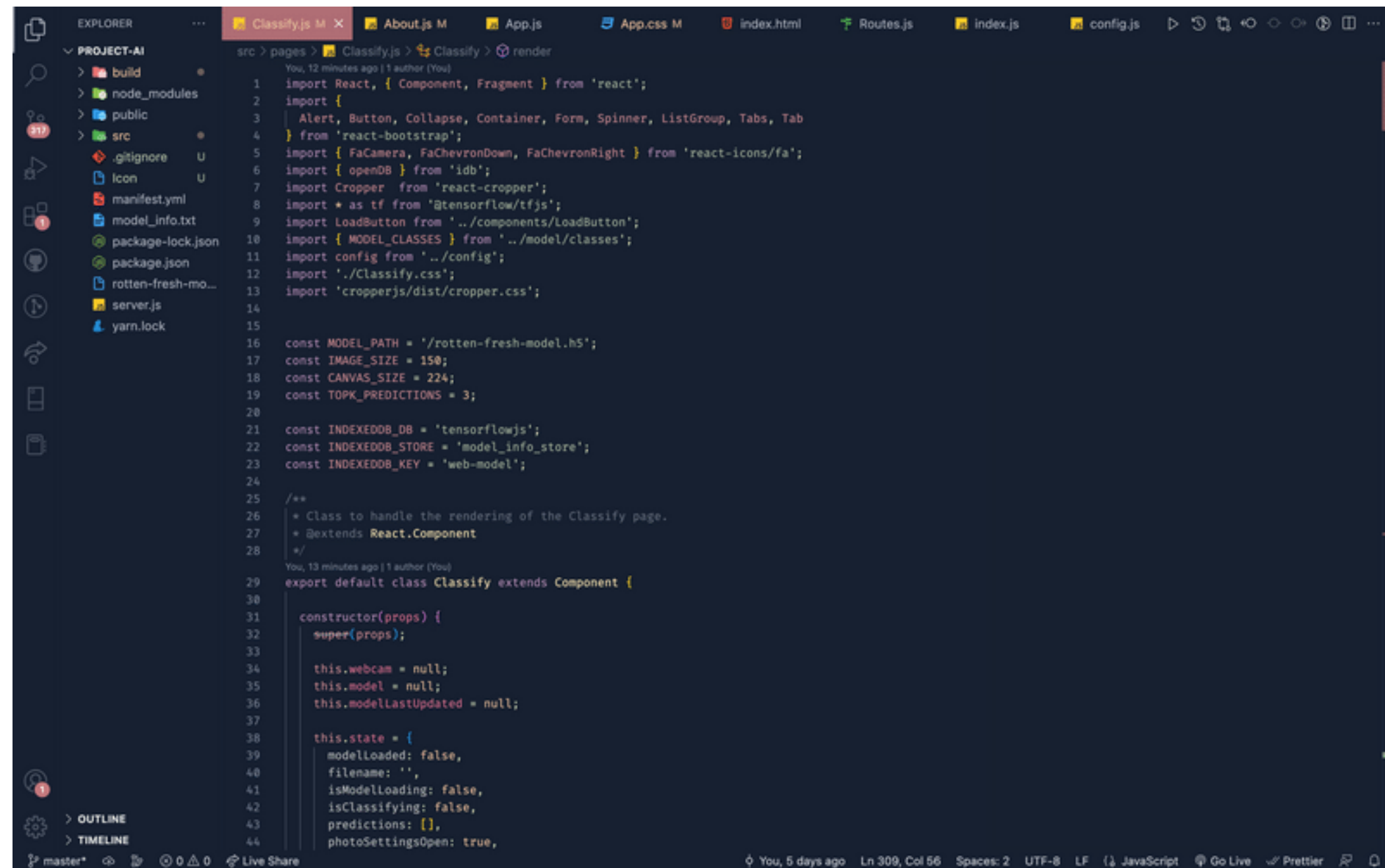
zip_ref.close()

/content
```

```
[ ] model.save('rotten-fresh-model.h5')
```

[https://colab.research.google.com/drive/1Y88SiWiZlsz0tSIlkj0HJMkA3\\_D9NXSe?usp=sharing](https://colab.research.google.com/drive/1Y88SiWiZlsz0tSIlkj0HJMkA3_D9NXSe?usp=sharing)

# Integrating Model into Web (Tensorflow.js)



```
EXPLOER  ...  Classify.js M X  About.js M  App.js  App.css M  index.html  Routes.js  index.js  config.js  ...
PROJECT-AI
  build
  node_modules
  public
  src
    .gitignore
    icon
    manifest.yml
    model_info.txt
    package-lock.json
    package.json
    rotten-fresh-mo...
    server.js
    yarn.lock
src > pages > Classify.js > Classify > render
You, 12 minutes ago | 1 author (You)
1  import React, { Component, Fragment } from 'react';
2  import {
3    | Alert, Button, Collapse, Container, Form, Spinner, ListGroup, Tabs, Tab
4    | } from 'react-bootstrap';
5  import { FaCamera, FaChevronDown, FaChevronRight } from 'react-icons/fa';
6  import { openDB } from 'idb';
7  import Cropper from 'react-cropper';
8  import * as tf from '@tensorflow/tfjs';
9  import LoadButton from '../components/LoadButton';
10 import { MODEL_CLASSES } from '../model/classes';
11 import config from '../config';
12 import './Classify.css';
13 import 'cropperjs/dist/cropper.css';
14
15
16 const MODEL_PATH = '/rotten-fresh-model.h5';
17 const IMAGE_SIZE = 150;
18 const CANVAS_SIZE = 224;
19 const TOPK_PREDICTIONS = 3;
20
21 const INDEXEDDB_DB = 'tensorflowjs';
22 const INDEXEDDB_STORE = 'model_info_store';
23 const INDEXEDDB_KEY = 'web-model';
24
25 /**
26  * Class to handle the rendering of the Classify page.
27  * @extends React.Component
28  */
29 export default class Classify extends Component {
30
31   constructor(props) {
32     super(props);
33
34     this.webcam = null;
35     this.model = null;
36     this.modelLastUpdated = null;
37
38     this.state = {
39       modelLoaded: false,
40       filename: '',
41       isModelLoading: false,
42       isClassifying: false,
43       predictions: [],
44       photoSettingsOpen: true,
```

**DEMO**



**Terima  
Kasih**

