A D Patel Institute of Technology

(A Constituent College of CVM University)

New V. V. Nagar

INFORMATION TECHNOLOGY DEPARTMENT

Mini Project Report on

FruitHub

Submitted By

Name of Student : Divyesh Kuchhadia

Enrolment Number : 12102080601035

Name of Student : HARSH DADIYA

Enrolment Number : 12102080601040

Name of Student : Jay Vekariya

Enrolment Number : 12102080601048

Guided By

Prof. Khushali Patel

MINI PROJECT (102040601)

A.Y. 2022-23 EVEN TERM

CERTIFICATE

This is to certify that the Mini Project Report submitted entitled **"FruitHub"** has been carried out by **Divyesh Kuchhadia** (12102080601035) under guidance in partial fulfilment for the Degree of Bachelor of Engineering in Computer Engineering, 6th Semester of A D Patel Institute of Technology, CVM University, New Vallabh Vidyanagar during the academic year 2022-23.

Prof. Khushali Patel
Internal Guide

Dr. Narendrasinh Chauhan
Head of Department

CERTIFICATE

This is to certify that the Mini Project Report submitted entitled **"FruitHub"** has been carried out by **HARSH DADIYA** (12102080601040) under guidance in partial fulfilment for the Degree of Bachelor of Engineering in Computer Engineering, 6th Semester of A D Patel Institute of Technology, CVM University, New Vallabh Vidyanagar during the academic year 2022-23.

Prof. Khushali Patel	Dr. Narendrasinh Chauhan
Internal Guide	Head of Department

CERTIFICATE

This is to certify that the Mini Project Report submitted entitled **"FruitHub"** has been carried out by **Jay Vekariya** (12102080601048) under guidance in partial fulfilment for the Degree of Bachelor of Engineering in Computer Engineering, 6th Semester of A D Patel Institute of Technology, CVM University, New Vallabh Vidyanagar during the academic year 2022-23.

Prof. Khushali Patel	Dr. Narendrasinh Chauhan
Internal Guide	Head of Department

ACKNOWLEDGEMENT

It’s our great pleasure to present the project report on “FruitHub” We are Thankful to our project guide to Prof. Khushali Patel who inspiring us by spending his valuable time for our project and being supportive at every stage of the project work. We give our deep sense of respect and gratitude to Prof. Khushali Patel for providing us an opportunity to carry out the project. We are grateful to him for sharing his valuable experience, management expertise and knowledge in the field of computer engineering. We are thankful to our college – A D Patel Institute of Technology to allow us to carry out the Training Project Work at their esteemed organization and utilizing their provided facilities. Finally, yet importantly, we would like to express our heartfelt thanks to our beloved parents for their blessings, my friends and classmates for their help and wishes for the successful completion of this project.

Abstract

In response to the growing demand for accurate information about fruits and vegetables in the post-COVID era, we present a Fruit Recognition App designed to address the need for reliable nutrition data and guidance on natural cultivation methods. Leveraging deep learning techniques, particularly convolutional neural networks (CNNs), our app enables users to simply capture images of fruits, which are then classified with high accuracy. The app provides comprehensive nutritional information for each recognized fruit, empowering users to make informed dietary choices. Additionally, cultivation tips for natural farming methods are offered, promoting chemical-free produce cultivation. The backend infrastructure for the web version of the app ensures seamless data management and user experience. Through this project, we aim to promote healthy living by providing accessible and trustworthy information about fruits, ultimately empowering individuals to make healthier dietary decisions.

Table of Contents

Table of Figures

1.Introduction:

- **Problem Statement:**
  Post-COVID, there is a heightened awareness of healthy living, with fruits being a popular choice. However, concerns arise due to potential chemical use in cultivation. People seek accurate fruit nutrition data and guidance for natural cultivation, yet reliable information is scarce. Addressing this gap is crucial for meeting the demand for healthy, chemical-free produce.

- **Project Summary and Introduction:**
  FruitHub emerges as an innovative solution within this context—a mobile application designed to harness the power of machine learning for the purpose of accurately identifying a wide array of fruits and vegetables. Beyond mere recognition, this application endeavours to furnish users with comprehensive insights into the nutritional profiles of different produce items, alongside practical guidance pertaining to their cultivation. By offering a user-friendly interface replete with valuable information, FruitHub aspires to not only promote healthier eating habits but also to empower users to make well-informed decisions regarding their dietary intake.

- **Aim and Objective of the Project:**
  At its core, FruitHub seeks to achieve the following objectives:
  - Development of a robust machine learning model capable of accurately recognizing and categorizing diverse fruits and vegetables.

  - Integration of nutritional data and cultivation tips into the application interface, thereby enhancing user accessibility to pertinent information.

- Design and implementation of an intuitive user interface, ensuring seamless interaction and navigation throughout the application.
- Ensuring compatibility across a broad spectrum of mobile devices, thereby maximizing accessibility and usability for all users.

2.System Analyses:

- **Motivation:**

  The impetus behind the conception of FruitHub lies in the pressing need to foster greater awareness and understanding surrounding dietary choices and their impact on individual health. In an era characterized by burgeoning technological advancements, there exists a unique opportunity to harness machine learning algorithms to address this critical societal need effectively.

- **Brief Literature Survey:**

  - **Explanation of Different Methods with Comparison:**

    Extant literature within the realms of image recognition and machine learning offers a plethora of methodologies for fruit and vegetable identification. These approaches range from conventional image processing algorithms to more sophisticated deep learning models, such as convolutional neural networks (CNNs). A comprehensive comparison of these methods will be undertaken to ascertain the most suitable approach for the development of FruitHub.

3.Design: Analysis, Design Methodology, and Implementation Strategy:

Hardware and Software Requirements:

- **Hardware:** The application will be compatible with smartphones equipped with camera functionality, ensuring widespread accessibility.
- **Software:** Development will be conducted primarily using Kotlin and Jetpack Compose for the mobile app interface, Python libraries for the machine learning model, and Firebase for data storage and retrieval.
- **Design Methodology:** We will try to follow the Incremental Model for app development. It is beneficial because we are doing every stage of development in full detail so the changes, we encounter at the later stage of the development will be less in numbers.
- **Implementation Strategy:**
    - **Research and Data Collection:**
        - Gather comprehensive data on fruit nutrition and natural cultivation methods from reputable sources such as scientific journals, agricultural research institutes, and nutrition databases.
        - Collaborate with experts in the fields of nutrition and agriculture to ensure the accuracy and reliability of the collected information.

- **Machine Learning Model Development:**

    - Employ machine learning algorithms to develop a robust model capable of accurately recognizing and categorizing different types of fruits and vegetables.
    - Train the model using a diverse dataset of images representing various fruits and vegetables to enhance its accuracy and reliability.
    - Continuously refine and optimize the model through iterative testing and validation processes.

- **Application Development:**

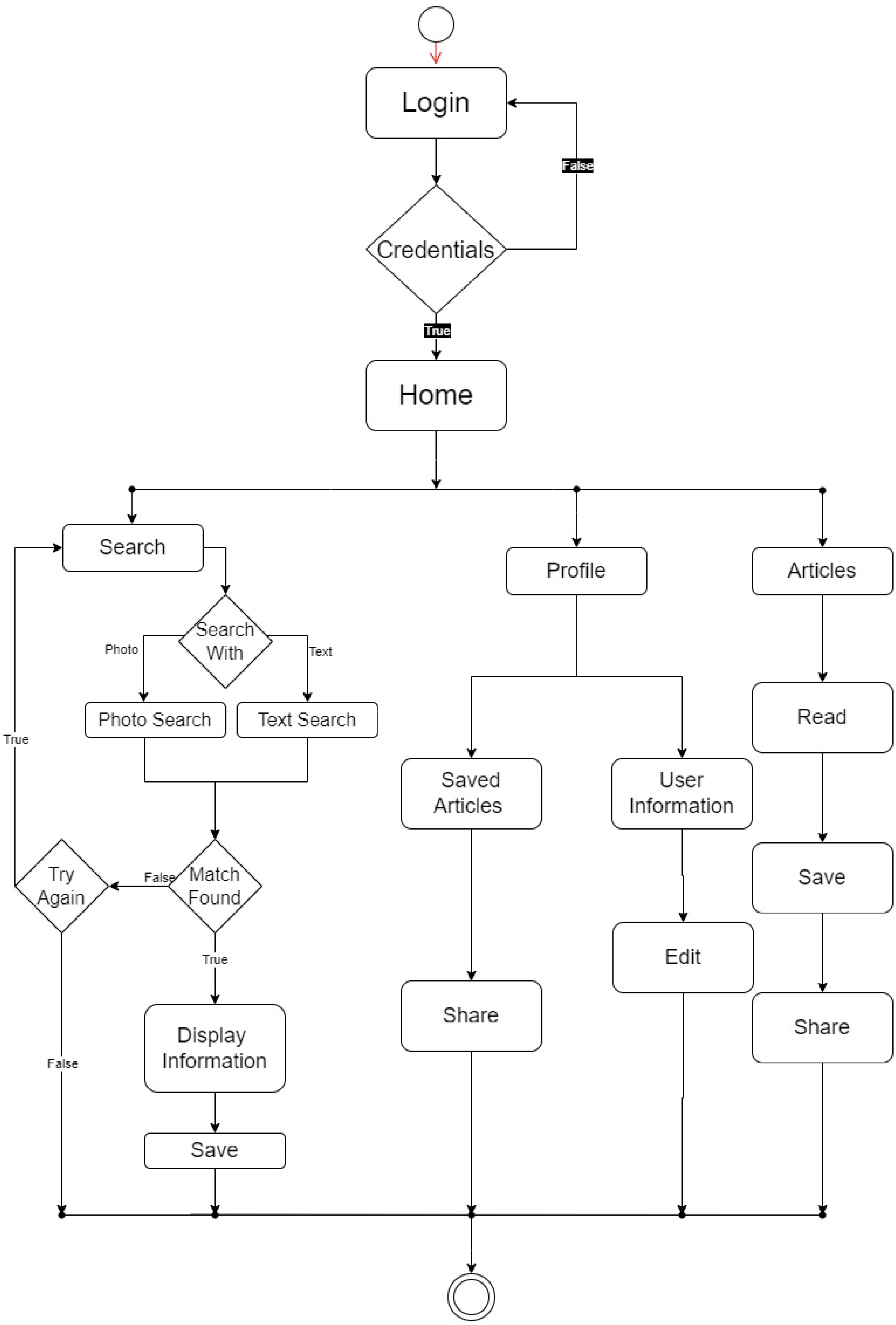
    - Utilize the Incremental Model for app development to ensure thorough attention to detail at every stage of development.
    - Begin with the development of core functionalities such as user authentication, data input, and retrieval mechanisms.
    - Incorporate the machine learning model into the application framework to enable fruit recognition capabilities.
    - Design an intuitive and user-friendly interface for seamless interaction and navigation within the application.
    - Implement features for accessing nutritional data and cultivation tips, ensuring the information is presented in a clear and understandable format.

- **Testing and Quality Assurance:**

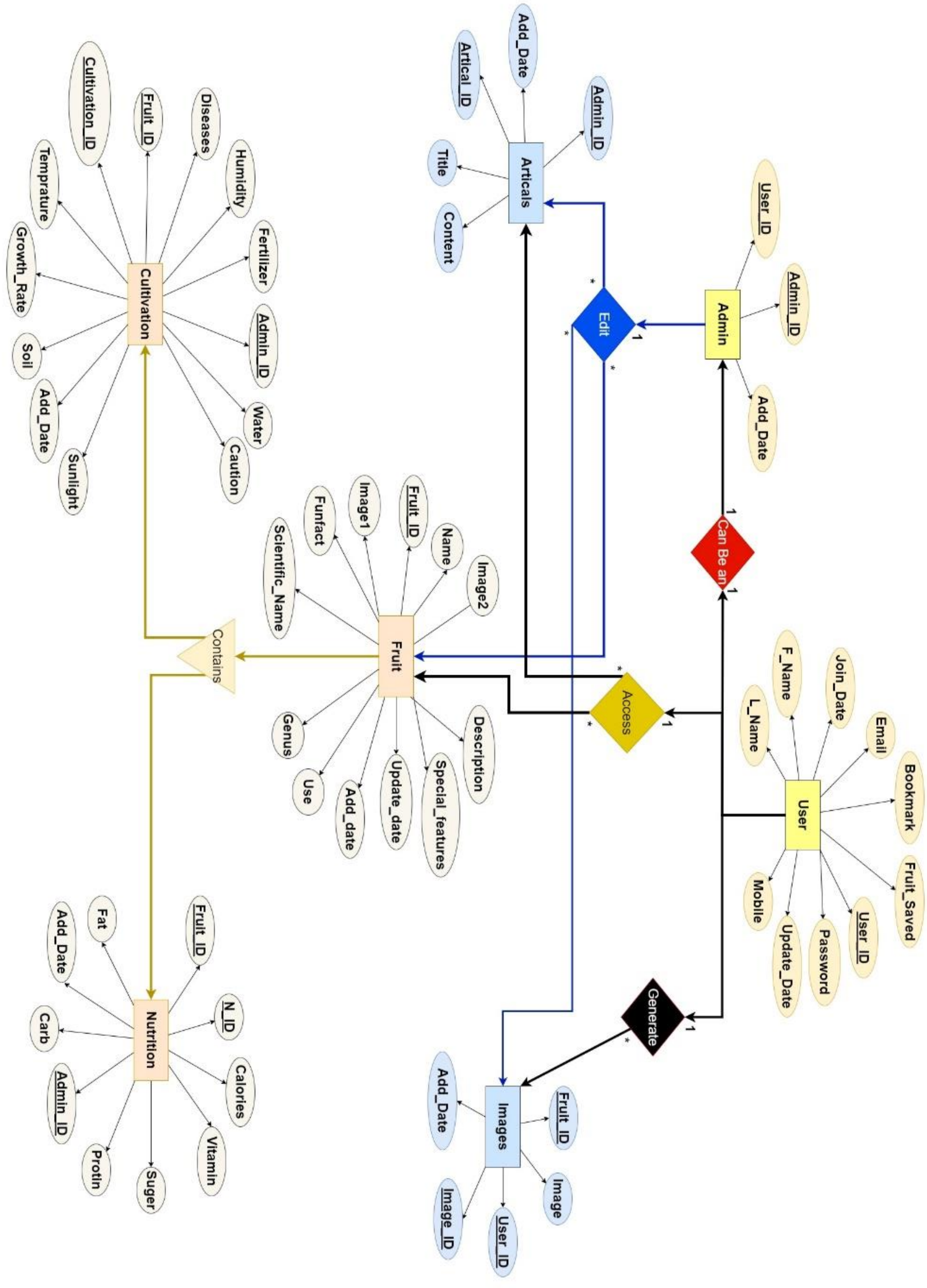
    - Conduct rigorous testing procedures to identify and rectify any bugs, glitches, or inconsistencies within the application.
    - Perform usability testing with a diverse group of users to gather feedback and insights for further improvements.

- Ensure compatibility across various mobile devices and operating systems to maximize accessibility for all users.

- **Deployment and Maintenance:**
    - Deploy the application on relevant app stores, ensuring compliance with all necessary guidelines and regulations.
    - Monitor user feedback and analytics to identify areas for further enhancement and optimization.
    - Provide regular updates and maintenance to address any issues or updates in nutritional data or cultivation practices.
    - Continuously engage with users to gather feedback and suggestions for ongoing improvement of the application.

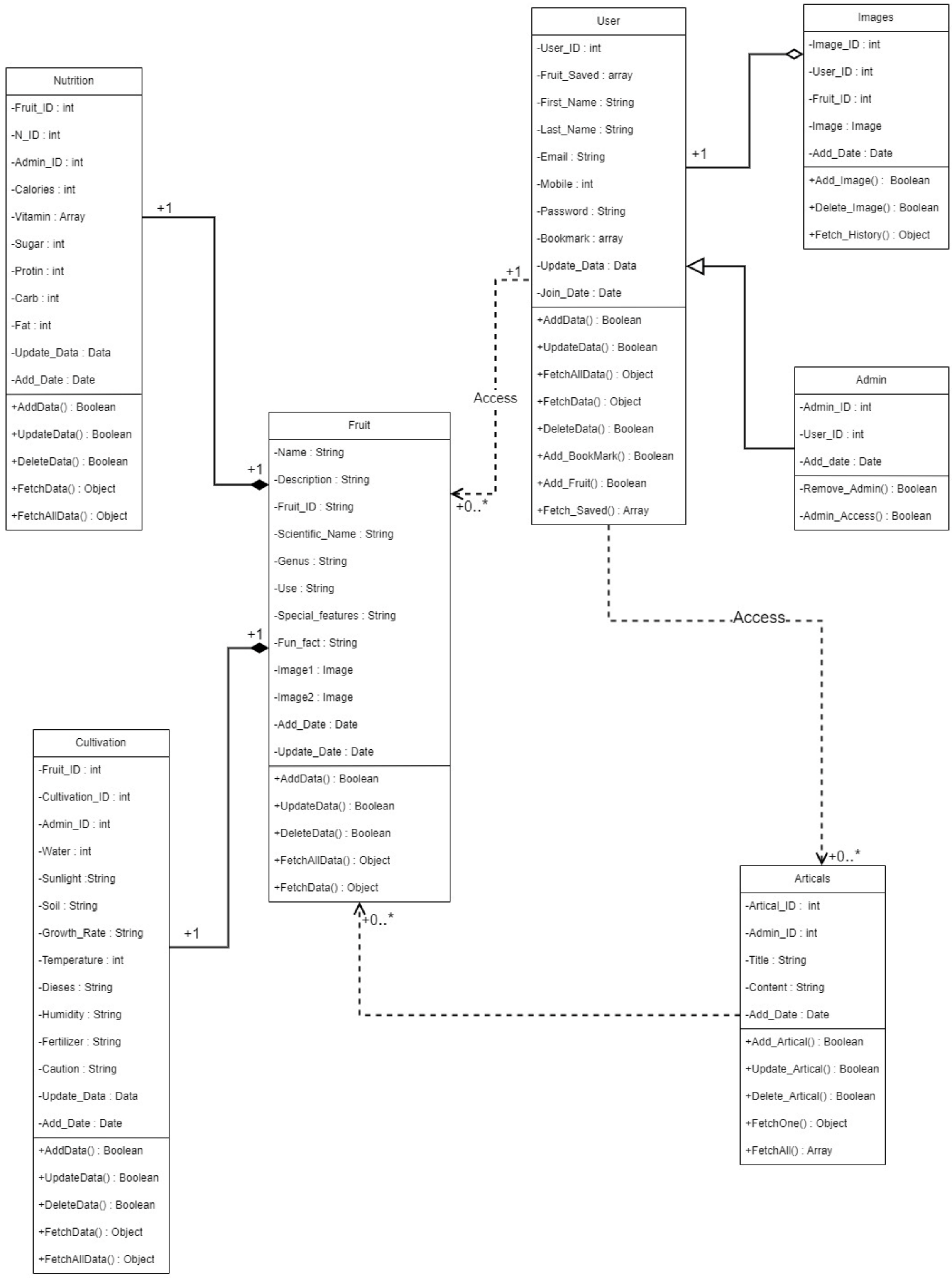


1.Fig-3(a). State Transition Diagram

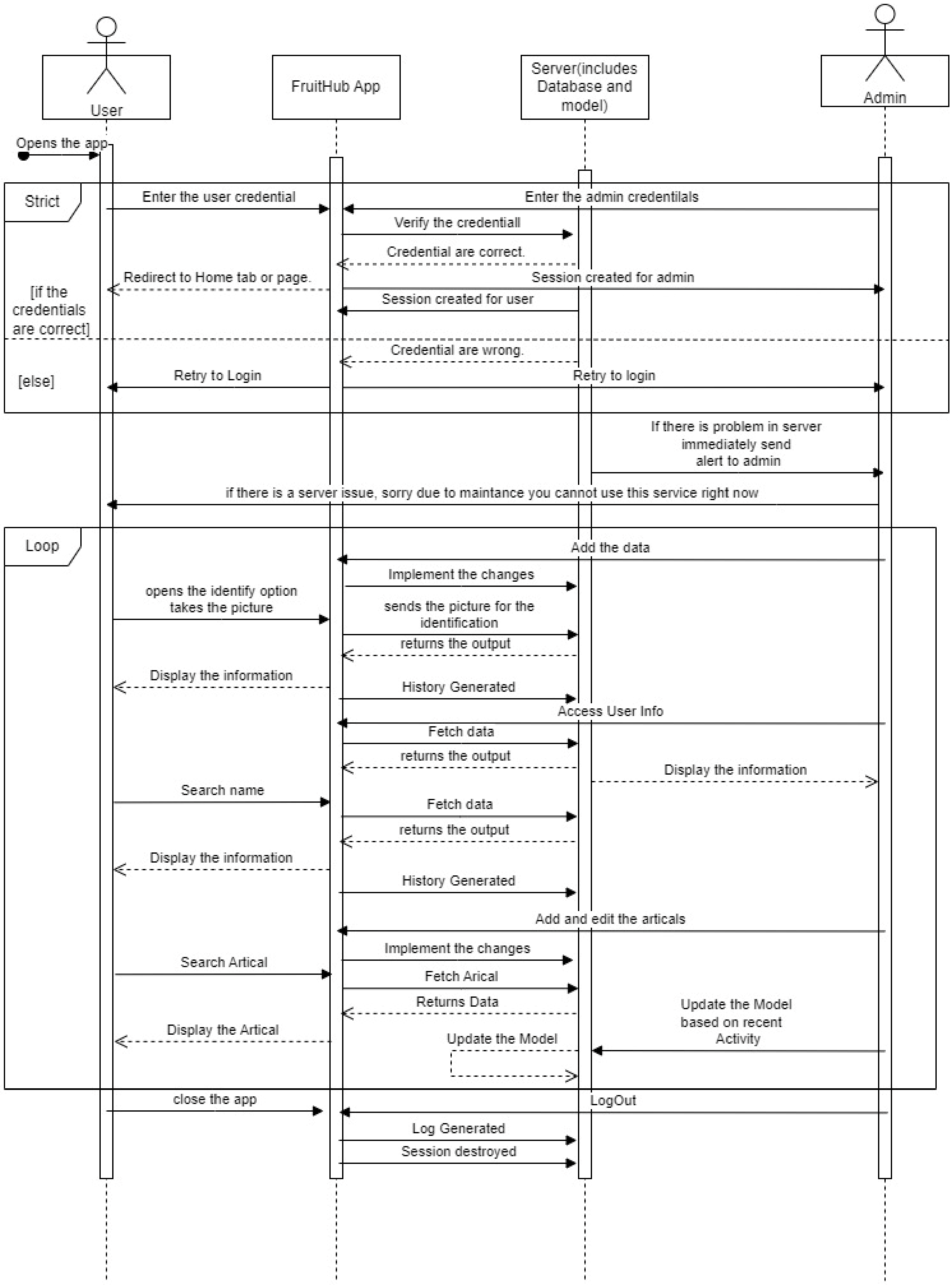


2.Fig-3(b). Entity-Relationship Diagram

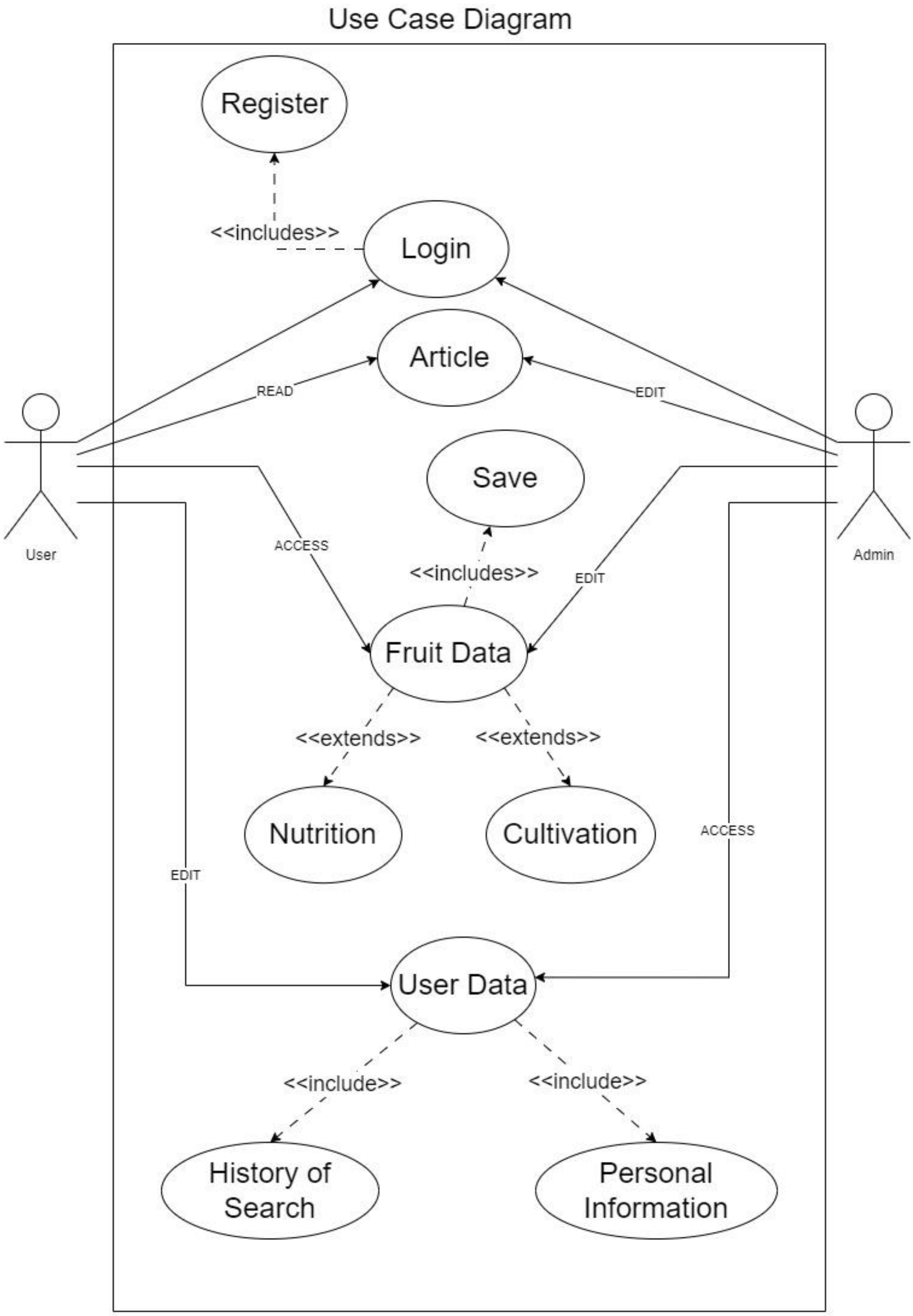
Class Diagram



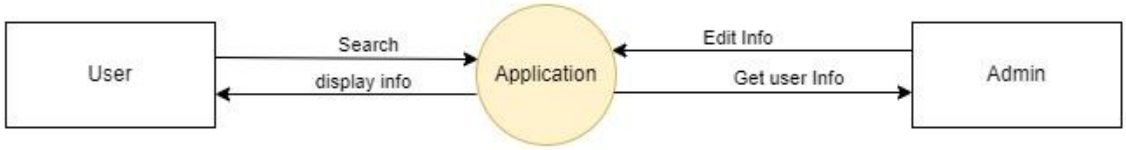
3.Fig-3(c). Class and Object Diagram



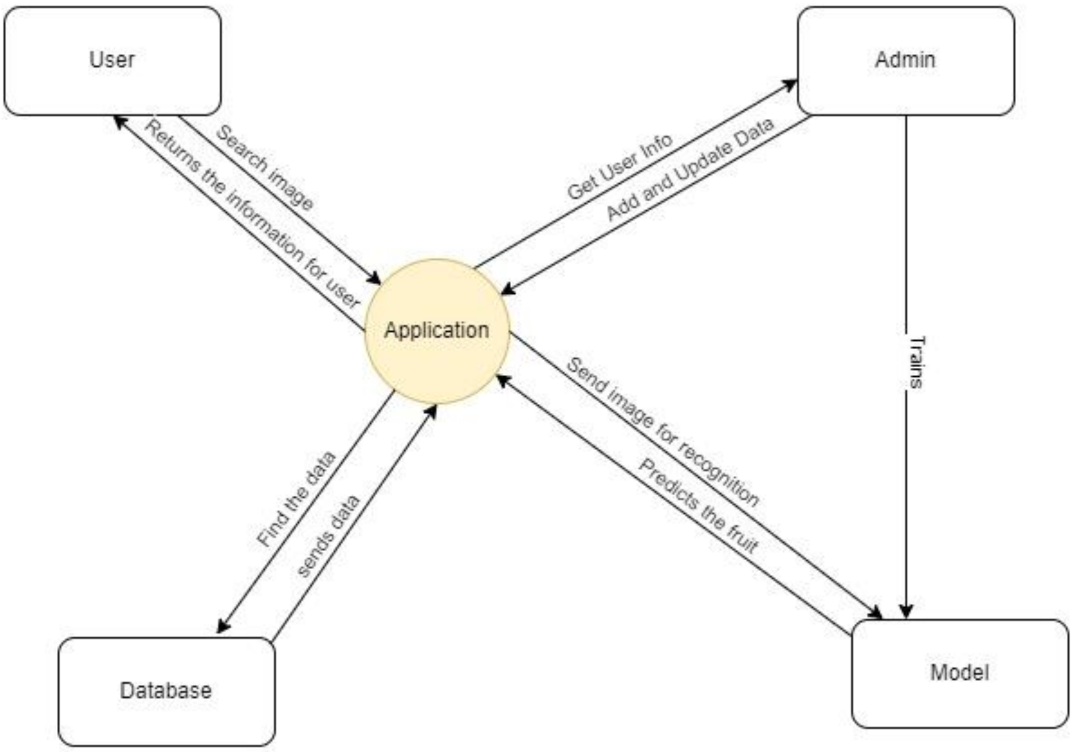
4.Fig-3(d). Sequence Diagram



5.Fig-3(e). Use-case Diagram



6.Fig-3(f). DFD level-0 Diagram.



7.Fig-3(7). DFD level-1 Diagram.

4.Implementation:

- **System Flow:**
    - Users will capture an image of a fruit or vegetable via the application interface.
    - The captured will be processed by the underlying machine learning model, facilitating accurate recognition of produce.
    - Relevant information, encompassing nutritional data and cultivation tips, will be retrieved from the database.
    - The retrieved information will then be presented to the user through the application interface in a clear and user-friendly manner.
    - Fig 1 on page 7 is diagrammatical representation on system flow.

- **Program/Module Specification:**
    - **Mobile App:** Development will be undertaken utilizing Kotlin and Jetpack Compose, encompassing modules for image capture, machine learning integration, and content display
    - **Machine Learning Model:** Implementation will be executed using Python, leveraging libraries such as TensorFlow or PyTorch. The model will be trained on a comprehensive dataset of fruit and vegetable images to ensure robust recognition capabilities.
    - **Database:** Storage and retrieval of nutritional information and cultivation tips will be facilitated through Firebase, ensuring seamless access to pertinent data.

- **Database Segmentation**

id	name
1	Apple Braeburn
2	Apple Crimson Snow
3	Apricot
4	Avocado
5	Banana
6	Beetroot
7	Blueberry
8	Cactus Fruit
9	Cantaloupe
10	Carambula
11	Cauliflower
12	Cherry
13	Cherry rainer
14	Cherry wax black
15	Cherry wax red
16	Cherry wax yellow
17	Chestnut
18	Clementine
19	Cocos
20	Corn
21	Cucumber ripe
22	Dates
23	Eggplant
24	Fig
25	Ginger root
26	Granadilla
27	Grape blue
28	Grape pink
29	GrapeWhite
30	Guava

10. Class Names

30	Guava
31	Hazelnut
32	Huckleberry
33	Kaki
34	Kiwi
35	Kohlrabi
36	Kumquats
37	Lemon
38	Lemon Meyer
39	Limes
40	Lychee
41	Mandarine
42	Mango
43	Mango Red
44	Mangostan
45	Maracuja
46	Melon Pieel Sapo
47	Mulberry
48	Necteraine
49	Nut Forest
50	Nut Pecan
51	Onion
52	Orange
53	Papaya
54	Passion Fruit
55	Peach
56	Pear
57	Pear Abate
58	Pear Forelle
59	Pear Kaiser
60	Pear Monster

8.Class Names

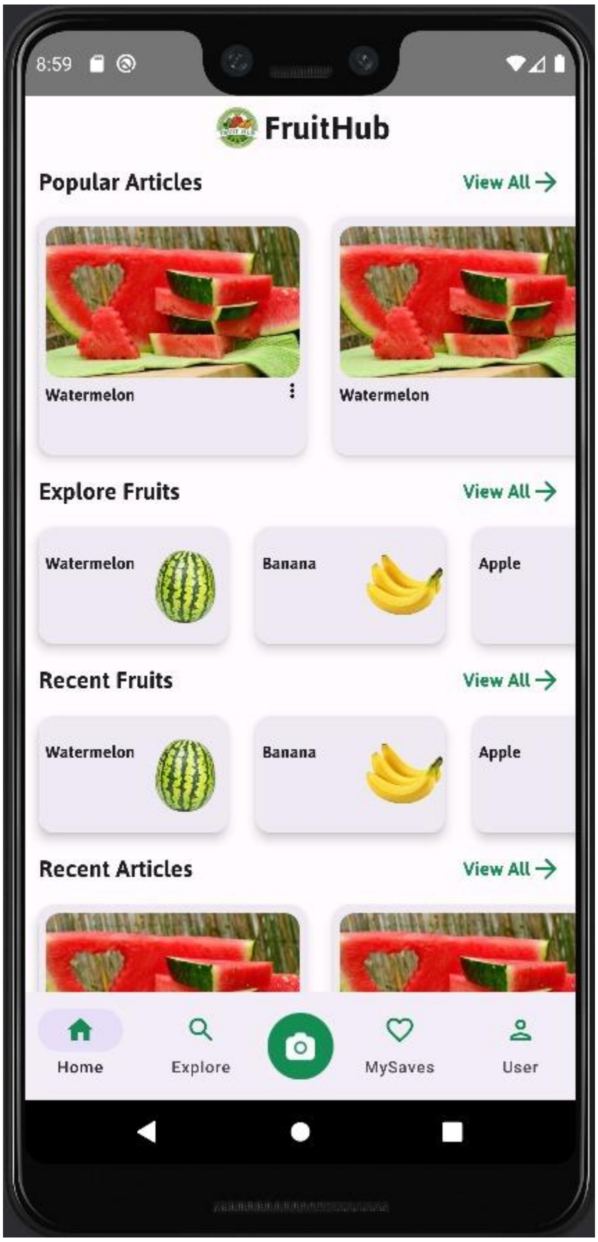
61	Pear Red
62	Pear Stone
63	Pear Williams
64	Pepino
65	Pepper Green
66	Pepper Orange
67	Pepper Red
68	Pepper Yellow
69	Physalis
70	Physalis with Husk
71	Pineapple
72	Pitahaya Red
73	Plum
74	Pomegranate
75	Pomelo Sweetie
76	Potato Red
77	Potato Sweet
78	Potato White
79	Quince
80	Rambutan
81	Raspberry
82	Redcurrant
83	Salak
84	Strawberry
85	Strawberry Wedge
86	Tamarillo
87	Tangelo
88	Tomato
89	Walnut
90	Watermelon

9.Class Names

- **Timeline Chart:**

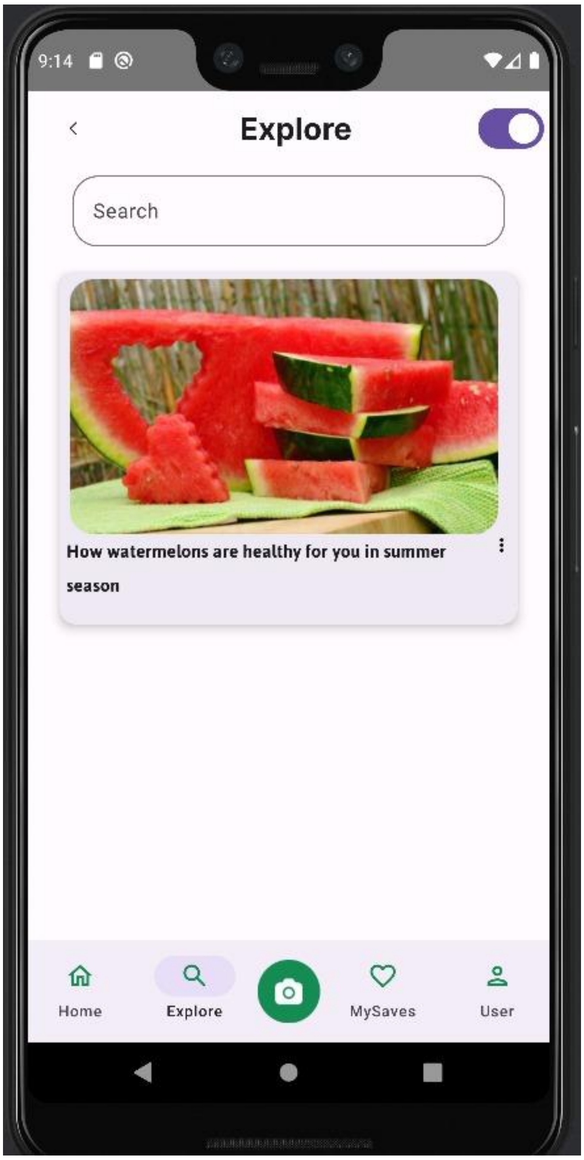
ID	Task	Start	Finish	Duration	Year-2024													
					Month													
					Jan				Feb				Mar				Apr	
					1 w	2 w	3 w	4 w	1 w	2 w	3 w	4 w	1 w	2 w	3 w	4 w	1w	2w
1	Research	1/1/24	17/2/24	7 weeks														
2	Designing and gathering data	21/1/24	3/3/24	7 weeks														
3	App Development	10/2/24	31/3/24	6 weeks														
4	ML model deployment	3/3/24	31/3/24	4 weeks														
5	Integration and Testing	31/3/24	3/4/24	4 Days														
6	Deployment	4/4/24	8/4/24	4 Days														

11.Fig-4(a). Timeline Chart



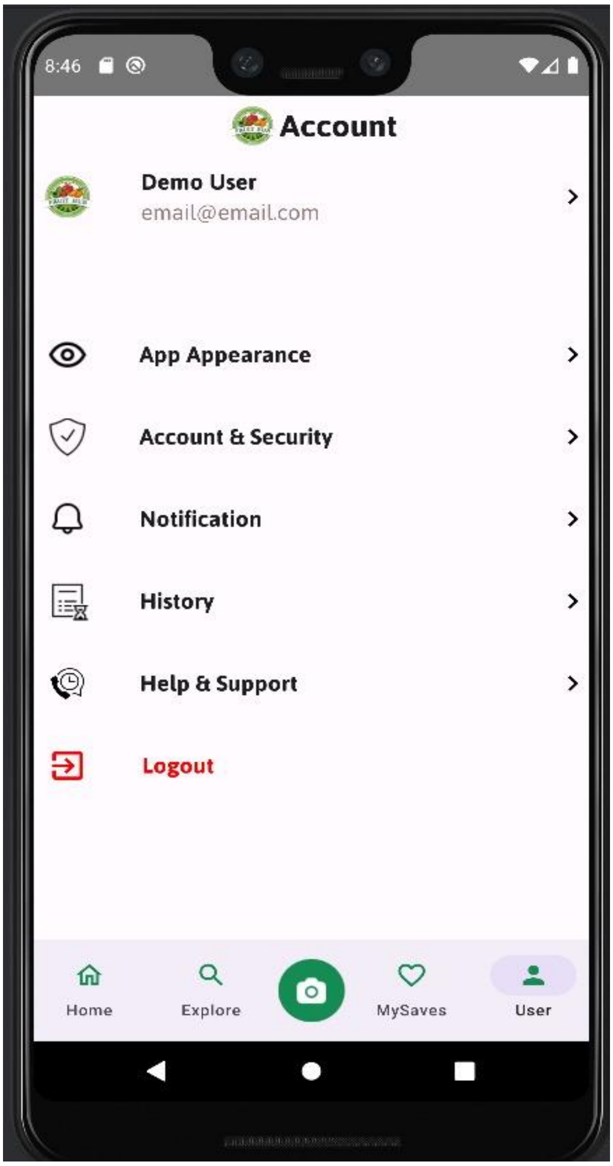
12.Fig-4(b).Popular Articles

13.Fig-4(c). Explore Fruits Page



14.Fig-4(d). Explore Page

15.Fig-4(e). Save page



17.Fig-4(g).Capture Page

16Fig-4(f).Account Page

- **Implementing CNN Model for Fruit Classification:**

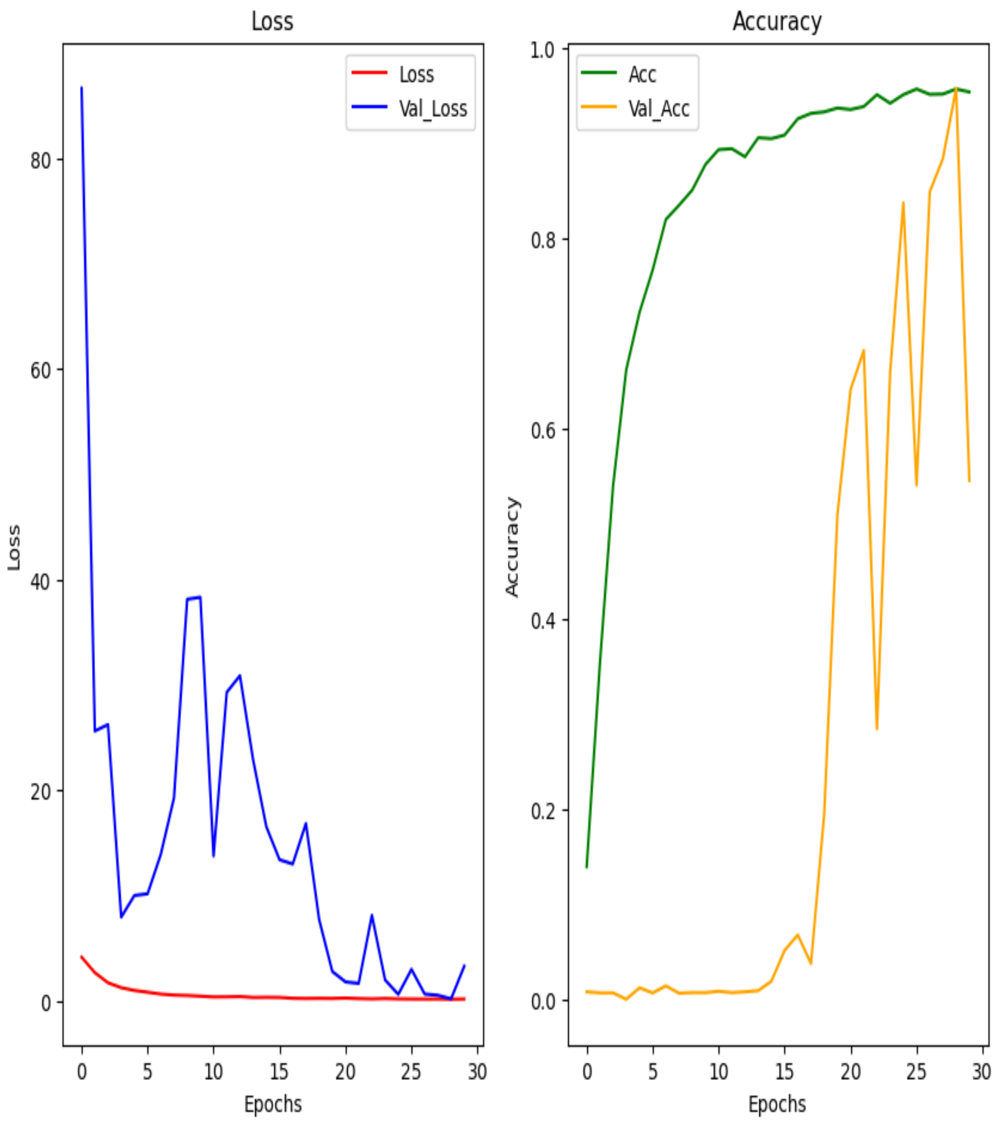
```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 100, 100, 32)      896

 batch_normalization (BatchN  (None, 100, 100, 32)     128
 ormalization)

 max_pooling2d (MaxPooling2D  (None, 50, 50, 32)       0
 )

 dropout (Dropout)           (None, 50, 50, 32)        0

 conv2d_1 (Conv2D)           (None, 50, 50, 64)        18496

 batch_normalization_1 (Batc  (None, 50, 50, 64)       256
 hNormalization)

 max_pooling2d_1 (MaxPooling  (None, 25, 25, 64)       0
 2D)

 dropout_1 (Dropout)         (None, 25, 25, 64)        0

 conv2d_2 (Conv2D)           (None, 25, 25, 128)       73856

 batch_normalization_2 (Batc  (None, 25, 25, 128)      512
 hNormalization)

 max_pooling2d_2 (MaxPooling  (None, 12, 12, 128)      0
 2D)

 dropout_2 (Dropout)         (None, 12, 12, 128)       0

 flatten (Flatten)           (None, 18432)             0

 dense (Dense)               (None, 131)               2414723

=================================================================
Total params: 2,508,867
Trainable params: 2,508,419
Non-trainable params: 448
```

18.Fig-4(h).Model Architecture

- This deep learning model is a convolutional neural network (CNN) designed for image classification tasks, specifically for recognizing different types of fruits from input images. Here's a breakdown of the model architecture and its components:

1) Input Layer: The input images are fed into the network with a shape of (100, 100, 3), indicating images of 100x100 pixels with 3 color channels (RGB).
2) Convolutional Layers: The model consists of three convolutional layers (`Conv2D`) with 32, 64, and 128 filters respectively. Each convolutional layer is followed by a rectified linear unit (ReLU) activation function, which introduces non-linearity into the model, enabling it to learn complex patterns in the input data.
3) Batch Normalization: Batch normalization layers (`Batch Normalization`) are included after each convolutional layer. Batch normalization helps stabilize and speed up the training process by normalizing the activations of each layer.
4) MaxPooling Layers: MaxPooling layers (`MaxPooling2D`) are used to down sample the feature maps produced by the convolutional layers, reducing the spatial dimensions of the data while retaining important features. This helps reduce computational complexity and prevent overfitting.
5) Dropout Layers: Dropout layers (`Dropout`) are employed after each max-pooling layer to mitigate overfitting by randomly dropping a fraction of the neurons during training.
6) Flatten Layer: The flatten layer (`Flatten`) transforms the 3D feature maps into a 1D vector, preparing the data for input into the fully connected layers.
7) Fully Connected Layer: The flattened output is then passed through a fully connected dense layer (`Dense`) with 131 units, representing the number of classes (types of fruits) in the classification task. The dense layer applies a SoftMax activation function, which converts the raw output into probability scores for each class, indicating the likelihood of the input image belonging to each fruit category.

Overall, this model architecture is designed to efficiently learn and extract meaningful features from input fruit images, enabling accurate classification of fruits into their respective categories. With over 2.5 million trainable parameters, the model can capture intricate patterns and nuances in the input data, ultimately leading to reliable fruit recognition performance.



19.Fig-4(i).Model Performance

The deep learning model's performance was assessed over 30 epochs. The loss plot reveals a significant initial decrease in training loss, stabilizing after approximately 5 epochs. Validation loss mirrored this trend but plateaued at a slightly higher value, suggesting the model's good fit to the training data with a slight overfitting to the validation set.

Accuracy metrics showed progressive improvement, with training accuracy increasing steadily. However, validation accuracy displayed fluctuations post the 15th epoch, indicating the model's potential overfitting. Despite this, the overall trend suggests the model's capability to generalize, albeit with room for improvement in reducing the gap between training and validation accuracy.

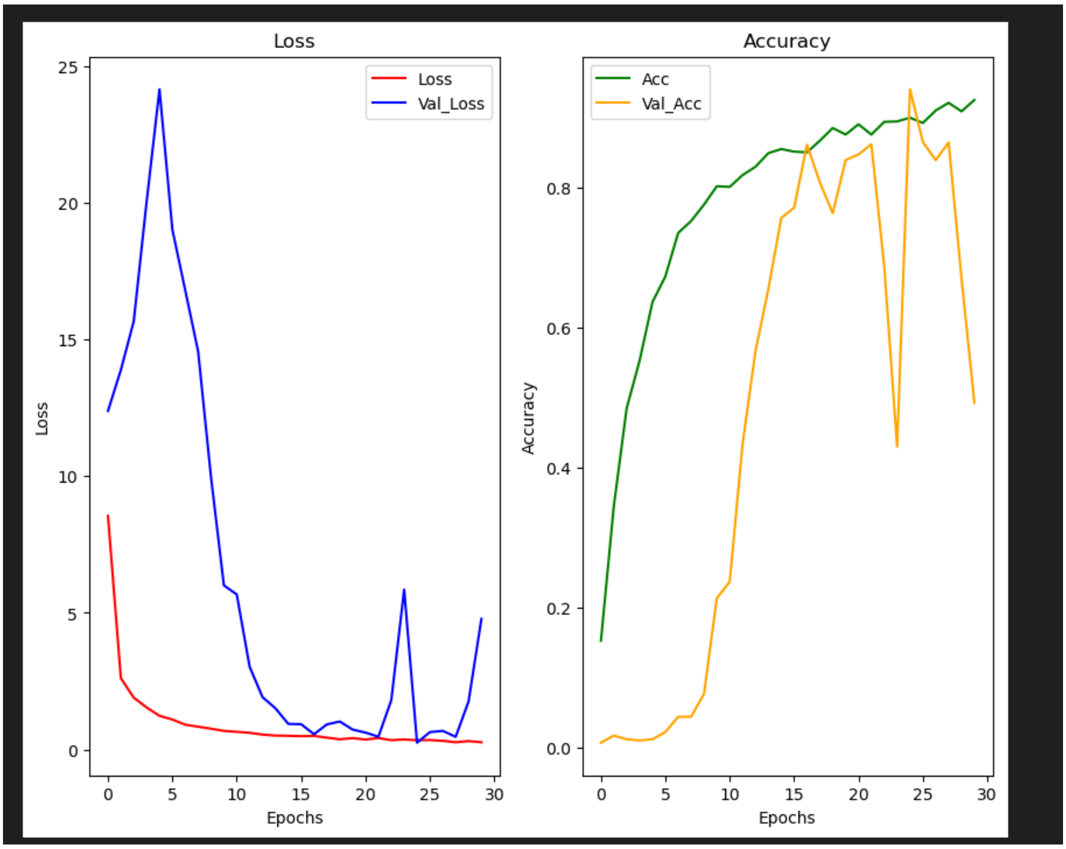
To enhance the model's generalization, strategies such as dropout, regularization, or data augmentation could be considered for future iterations.

20.Fig-4(j).Testing Model

To enhance the model's accuracy, transfer learning was employed utilizing the ResNet50 architecture, renowned for its depth and performance in image recognition tasks. The pre-trained weights of ResNet50, trained on the ImageNet dataset, served as a foundational knowledge base, enabling the model to leverage learned features from a vast array of images.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 resnet50 (Functional)       (None, 4, 4, 2048)        23587712

 global_average_pooling2d (G  (None, 2048)             0
 lobalAveragePooling2D)

 dense (Dense)               (None, 256)               524544

 dropout (Dropout)           (None, 256)               0

 dense_1 (Dense)             (None, 131)               33667

=================================================================
Total params: 24,145,923
Trainable params: 24,092,803
Non-trainable params: 53,120
```

21.Fig-4(k).Architecture of new model



22.Fig- 4(l).Accuracy for new Model

5.Conclusion & Future Work:

- ## Conclusion

FruitHub epitomizes a pioneering endeavour aimed at fostering healthier dietary habits and enhancing nutritional literacy through the utilization of cutting-edge mobile technology. By amalgamating image recognition with comprehensive informational resources, the application endeavours to empower users to make informed decisions regarding their dietary choices and cultivate a deeper understanding of the fruits and vegetables they consume.

- ## Future Work:

1. **Collaboration with Farmers**: In the future, we envision collaborating with farmers to enhance the functionality of the Fruit Recognition App. By incorporating feedback from farmers and integrating features tailored to their needs, such as real-time crop monitoring and pest detection, we can empower farmers to make data-driven decisions and improve crop yield and quality.

2. **Development of Web Module**: As part of our expansion plans, we intend to develop a web module for the Fruit Recognition App. This web-based platform will provide users with additional features and functionalities, such as accessing their account and viewing their saved fruit recognition history from any device with internet access. The web module will also serve as a valuable resource for farmers, allowing them to analyse crop data and trends more comprehensively.

3. **Collaboration with Botany Experts**: To further enhance the accuracy and reliability of our fruit recognition system, we aim to collaborate with botany experts in the field. By leveraging their expertise and domain knowledge, we can refine our machine learning algorithms and expand Page | 33 our database of fruit images and botanical information. This collaboration will ensure that our app remains at the forefront of fruit recognition technology and continues to provide users with the most accurate and up-to-date information.

6.References:

1. Knowledge Academy. (n.d.). SQL Array: A Complete Guide. Retrieved from https://www.theknowledgeacademy.com/blog/sql-array/
2. U.S. Food and Drug Administration. (n.d.). Nutrition Information for Raw Fruits, Vegetables, and Fish. Retrieved from https://www.fda.gov/food/food-labeling-nutrition/nutrition-information-raw-fruits-vegetables-and-fish
3. Moltean, D. (n.d.). Fruits Dataset. Retrieved from https://www.kaggle.com/datasets/moltean/fruits/data
4. WhatFlower. (n.d.). Flower Identification Guide. Retrieved from https://whatflower.net/
5. 7ESL. (n.d.). Fruits Vocabulary in English. Retrieved from https://7esl.com/fruits-vocabulary-english/
6. Rawat, W., & Wang, Z. (2017). Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. *Neural Computation, 29*(9), 1-98. https://doi.org/10.1162/NECO_a_00990
7. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In Advances in Neural Information Processing Systems (pp. 1097-1105).
8. Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv preprint arXiv:1409.1556.
9. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9).
10. Chollet, F. (2017). Deep Learning with Python. Manning Publications.
11. Géron, A. (2017). Hands-On Machine Learning with Scikit-Learn and TensorFlow. O'Reilly Media.
12. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

13. Brownlee, J. (2020). Deep Learning for Computer Vision. Machine Learning Mastery.