International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Strawberry Disease Diagnose Mobile App Integration System

I. Ketut Agung Enriko*[1], Erika Lety Istikhomah Puspita Sari[2], Melinda[3], M. Fauzan Alfariz[4], Nanda Arisna[5], Faruq Miqdad Mudaffar[6]

Submitted: 06/11/2024 **Accepted :** 23/12/2014 **Accepted:** 05/01/2024

Abstract*:* The agriculture industry is one of the commodities driving Indonesia's economic growth. In wealthy countries, the application of technology in agriculture has already begun. One approach is to make use of the Internet of Things (IoT). IoT coupled with artificial intelligence to assist farmers in diagnosing illnesses in strawberry plants with accuracy findings meet the target goal of >90%. Using the Convolution Neural Network (CNN) whose results will be integrated as create a mobile application that can identify android-based ailments faced by strawberry plants. Where the model is fed into the flask framework, which has been connected with Firebase, and the database storage is configured to accept the data. The functionalities are then inserted into the flask code framework design connected with Firebase. Following that, the machine learning model produces predictions to detect illnesses in strawberry plants which will be displayed on the mobile application system.

Keywords: *Firebase; Flask; Strawberry; Accuracy; Application Mobile.*

1. Introduction

The agriculture industry is one of the commodities driving Indonesia's economic growth. The strawberry plant is one of the plants that has become a commercial product. Strawberry plants have a high economic value and provide several benefits [1]. Strawberry is a popular fruit due to their numerous health advantages. Strawberries have been shown to boost immunity and cognitive function, reduce joint pain, cure gout, and prevent heart attacks and high blood pressure [2]. The strawberry plant, on the other hand, is one of the horticultural products that is readily destroyed (perishable) and very subject to pest and disease assaults. The outward look and flavor of strawberries greatly influence their quality.

According to BPS data (2019), statistics on strawberry production in Indonesia from 2014 to 2017 had a large fall from 2015-2016, but strawberry output rebounded again in 2017. Strawberry production figures for 2014 were 58,884 tons, 31,801 tons in 2015, 12,091 tons in 2016, and 12,225 tons in 2017. (2017). According to this data, strawberry production decreased significantly between 2015 and 2016 but increased again in 2017. This also occurs in Indonesia's major strawberry-producing areas, including West Java, Central Java, East Java, and Bali [3].

Leaf blight, tip burn, leaf rust, red spot, and viral infections are among the illnesses that have been recognized based on their features. Diseases produced by fungi, bacteria, mycoplasma-like organisms, and viruses are common in strawberry agriculture. Pathogens commonly damage the roots, crowns, leaves, flowers, and fruit of plants. Leaf blight is one of the pathogens that attacked concurrently and caused losses in 2014 [4].

Identification of illnesses in strawberry plants is required to discover diseases early so that early prevention of disease spread may be carried out, before they cause losses, as happened in 2014. There are several tools available now to identify the kind of sickness. Direct inspection by specialists or experts derived from strawberry plants. However, due to the constraints of the strawberry plant specialists themselves, in wealthy countries, the application of technology in agriculture has already begun. One approach is to make use of the Internet of Things. The Internet of Things (IoT) is a goal-oriented interaction between sensors and gadgets linked to the internet [5]. The goal of this study is to develop an application that can identify android-based devices. This research is expected to be able to identify diseases in strawberry plants, making it easier for the general public or farmers to obtain information about the disease's symptoms or characteristics. Strawberry plant diseases.

[1] *Department of Telecomunication Engineering, Institut Teknologi Telkom Purwokerto, Purwokerto, Indonesia.(E-mail: enriko@ittelkom-pwt.ac.id)*
[2] *Department of Telecomunication Engineering, Institut Teknologi Telkom Purwokerto, Purwokerto, Indonesia. (E-mail: 2212101121@ittelkom-pwt.ac.id)*
[3] *Departement of Electrical and Computer Engineering, Faculty of Engineering, Syiah Kuala University, Banda Aceh, Indonesia. (E-mail: melinda@unsyiah.ac.id)*
[4] *Departement of Electrical and Computer Engineering, Faculty of Engineering, Syiah Kuala University, Banda Aceh, Indonesia. (E-mail: alfariz@mhs.unsyiah.ac.id)*
[5] *Departement of Electrical and Computer Engineering, Faculty of Engineering, Syiah Kuala University, Banda Aceh, Indonesia. (E-mail: nanda01@mhs.unsyiah.ac.id)*
[6] *Departement of Electrical and Computer Engineering, Faculty of Engineering, Syiah Kuala University, Banda Aceh, Indonesia. (E-mail: faruq@mhs.unsyiah.ac.id)*

2. Research Methodology

2.1. Literature Review

2.1.1. Strawberry

Is a delicious herb that was discovered in Chile, America. Fragaria Chiloensis L. is a strawberry species that has spread over North America, Europe, and Asia. Other species, such as F. Vesca L., are more prevalent in the wild; nevertheless, strawberry is being introduced to Indonesia for the first time. The strawberry is a cross between Fragaria virginiana L. var Duchesne from North America and Fragaria Chiloensis L. var Duchesne from Chile. Required for the production of contemporary strawberry hybrids [5].

2.1.2. Flask

Flask is a Python web framework that is categorized as a micro-framework. Even though Flask is referred to as a micro-framework, it does not lack functionality. Flask strives to keep the core of the application as basic as possible while yet making it easy to add to. Flask is a high-performance application and web interface framework. Developers may use Flask with Python to construct well-structured websites and alter the functionality of any website more quickly [6]. As a result, as compared to other frameworks, Flask's flexibility and scalability are relatively high. Flask has the following features:

1. Integrated development server.
2. Rapid debuggers.
3. Support for unit testing is built in.
4. Google application engine is supported.
5. Request dispatching through REST.
6. Templates for Jinja2.
7. Secure cookies are supported.
8. Unicode is used.
9. After WSGI 1.0.

Aside from that, Flask is well-documented, and there are numerous forums on the internet where you can discuss F flask-related issues [19].

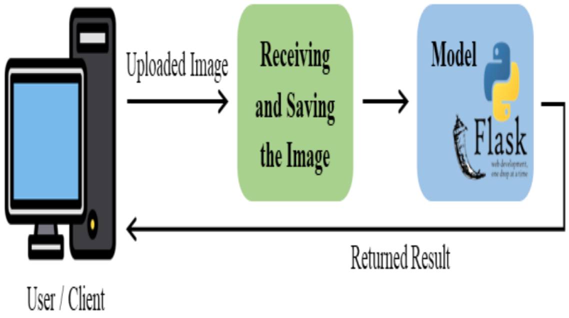


Fig 1: Framework Flask [10]

The first step is to use an HTTP Post request to upload the picture to the server. The server receives and stores images. The picture should then be checked for any possible security issues. The saved image is then processed by the model. The results of the model are returned to the user in text form [6].

2.1.3. Firebase

Firebase is a provider of real-time database and backend services. A tool that allows developers to design APIs (Application Programming Interfaces) that can sync with different clients and are saved in the Firebase cloud. Firebase offers a wide library that enables integration with Android, iOS, JavasScript, Java, Objective-C, and Node.JS [7].

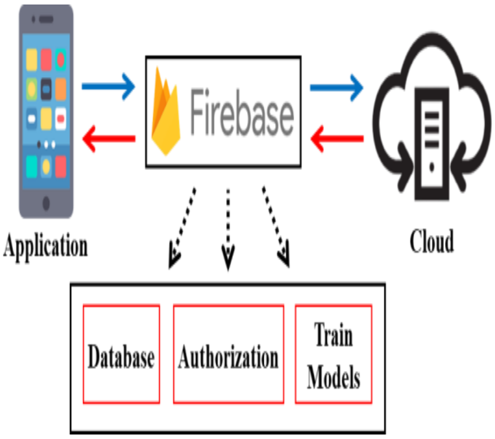


Fig 2: firebase [11]

Firebase employs a Real-Time database, in which the database, training model, and server processes determine whether or not the client is authorized to use resources or access files hosted in the cloud. Data is saved as JSON and is constantly synced for each related client. The application was then integrated [7].

A NoSQL database is Firebase real-time database. The database structure is flat to decrease bandwidth and does not comply with the normalization required by conventional databases [13]. Storage on the Firebase This enables quick and safe file transfers for Firebase apps independent of network quality. It is driven by Google Cloud Storage, a low-cost object storage service. It may be used by developers to store photographs, music, video, and other user-generated information [14].

2.1.4. Flutter

Flutter is a software development kit (SDK) from Google that allows you to create mobile apps for Android or iOS using the Dart programming language. Flutter allows you to create apps for Android and iOS using the same code and programming language that Google created in 2011. Low-level (native) Android applications had previously required to be built in Java or Kotlin, whilst iOS apps had to be written in Objective-C or Swift [8].

The architecture of the Flutter framework. Flutter architecture is made up of four parts: the Flutter Engine, Foundation Libraries, Widgets, and widget Design Specific [8]. Flutter combines UI and logic into a single codebase.

Flutter's programming is implemented using widgets. In flutter, widgets can be either visual components or placeholders for other widgets. As a result, flutter has a hierarchical code. Flutter differs from other multiplatform solutions in that it does not employ connectors, as do other multiplatform approaches [15].

2.1.5. Figma

Figma is a browser-based interface design program. Figma has all of the tools needed for a project's design phase, such as vector drawing tools capable of producing fully illustrated designs and prototyping capabilities. Figma is incredibly adaptable and simple to learn. Figma allows you to create, collaborate, prototype, and vector graphics editors and tools that can be used for prototyping in real-time, and the key benefit is that it is available on the internet as well as an application. Can create anything, including user interfaces for web pages and vector graphics [9].

2.1.6. User Interface (UI)

User interface (UI) is the design of an interface for tools and software such as computers, online applications, and other electronic devices that focuses on improving the user experience. A "user interface" is a system of interdependence or thorough integration of hardware and software that delivers a computer experience. The user interface (UI) refers to how the system and users interact to run the system, enter data, and utilize its contents via instructions or processes. Computer systems, mobile devices, and games, as well as application programs and content consumption, all have user interfaces [10].

The user interface (UI) refers to all interactive system components (software and hardware) that offer information and control to the user for the interactive system to accomplish certain tasks. Windows, icons, menus, and pointers are examples of UI components. In addition to the visual aspects, the user interface design must be able to explain its function. According to Nielsen's research, the ultimate aim of user design is usability [17].

2.1.7. User Experience

The process of boosting user pleasure (application users, website visitors) in increasing application usability and the interaction offered between the user and the product is known as user experience. UX is a design that interacts with diverse cross-functionality to find a sweet spot between user wants, business goals, and technical improvements, then create a magical experience via meaningful, helpful, and engaging product design [10].

Because the experience of a device will differ from that of other devices, UX design is also expanding. Accessing online from a desktop computer is not the same as accessing the same web from a smartphone. Similarly, with the emergence of social media, the digital material that is exhibited is becoming more diversified [18].

2.2. Design System

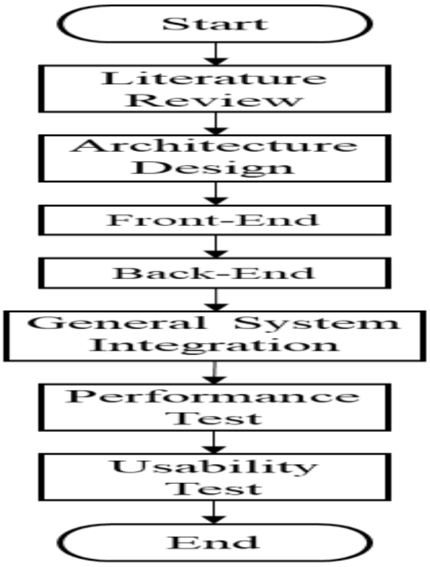


Fig 3: Design System

The first stage is to perform a literature study in which references from relevant journals are examined in order to determine the best approach from some of these article observations on the topic of model integration in machine learning. The Flask framework is then linked to Firebase. Following that, the architectural design focuses on the Front-End, or the scope of the user interface or application display, and the Back-End, or the system behind the screen when the program is operating. Then it will be properly integrated and performance tests on mobile apps will be performed to examine the speed of each responsive menu and so on in application components, and the last level will look at system performance capabilities in mobile applications.

a. Front-End

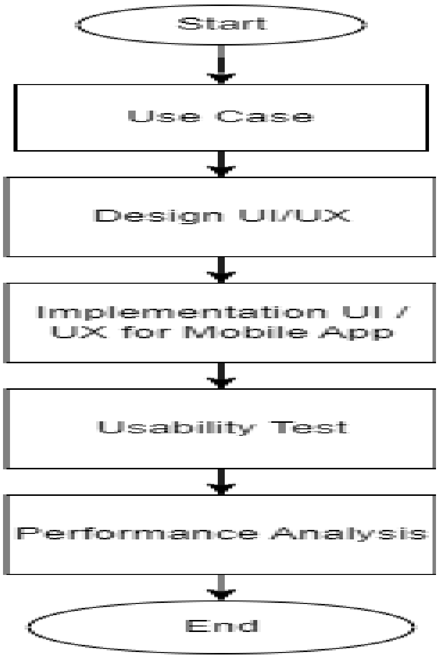


Fig 4: Front-End

It starts with a use case in the Front-End system that defines the primary guidelines involved and what can be done on the system. Then comes the User Interface (UI) design, which seeks to produce an appealing look through animation, font, colors, videos, and buttons, and the User Experience (UX), which focuses on user convenience through usability, navigation, design structure, features, and interaction design. Then, on the Mobile Application, integrate the UI/UX findings and develop the essential application functionalities. Following that, a test of the mobile application's capabilities with the Front-End that has been incorporated into the mobile application is performed, followed by a performance study of the mobile application

b. Firebase

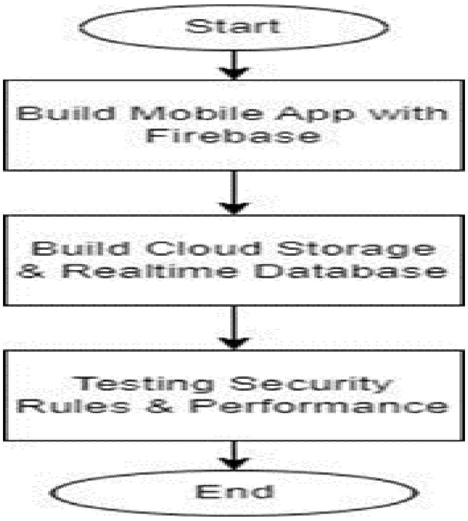

Start
Build Mobile App with Firebase
Build Cloud Storage & Realtime Database
Testing Security Rules & Performance
End

Fig 5: Firebase

The Back-End system begins with the development of Android Apps that are connected with Firebase. Then, on Firebase, establish Cloud Storage as a data repository that has been entered into Firebase storage with the acquired Real-Time Database. Next, do a performance test on the mobile application's back-end and verify for security in accordance with the application's criteria.

c. Flask

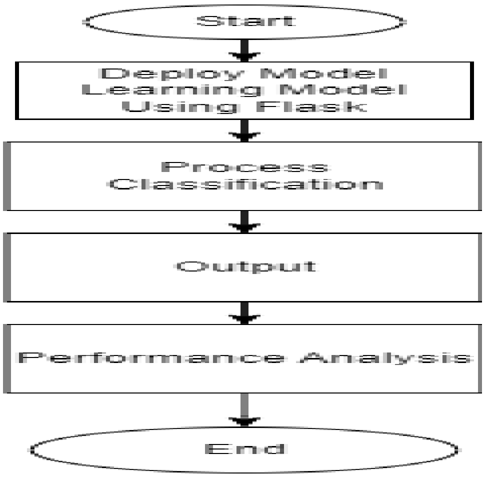


Fig 6: Framework Flask

The Back-End system then starts by delivering machine learning model results using the flask framework. Then, do the categorization procedure for strawberry leaf plants and report the findings. Then, do a performance analysis to see if the model that was deployed on the flask successfully performs the categorization procedure.

2.3. Integration System

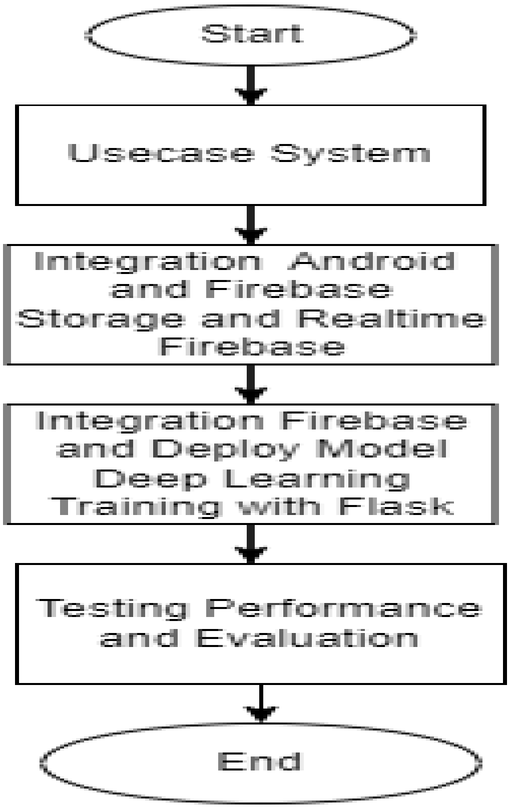

Start
Usecase System
Integration Android and Firebase Storage and Realtime Firebase
Integration Firebase and Deploy Model Deep Learning Training with Flask
Testing Performance and Evaluation
End

Fig 7: Integration system

The entire Integration System begins with an overview of a system's functioning. Then, connect with Android and build Firebase storage. Then it connects to Firebase in real time. At this point, the connection with Firebase is linked to the Flask framework, where the machine learning model's findings will be deployed. Furthermore, a performance test will be performed on the system integration that has been completed to determine the viability of the performance system on the mobile

2.4. Architecture Model View Controller

This research employs the Model View Controller, which is an architecture for developing applications that is divided into three parts: model, view, and controller. The model serves as an identification of the data structure, the View serves as a location for the program's design or User Interface (UI), and the Controller serves as the logic of an application, also known as the back end of an application in this research.

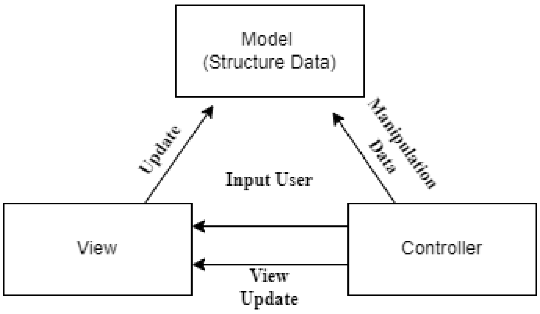


Fig 8: architecture model view controller

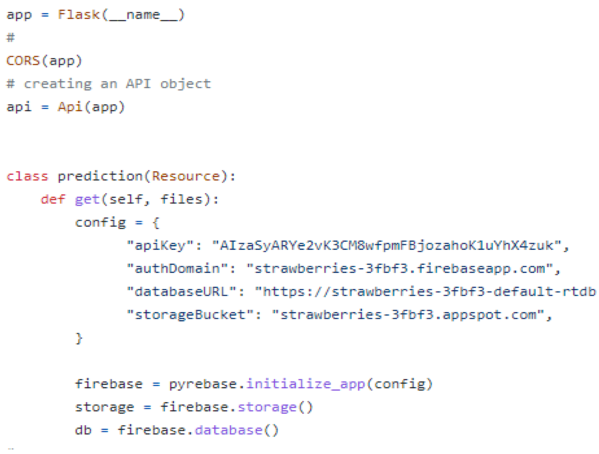
3. Result and Discussion

3.1. Result Research

3.1.1. Application Design

Application Development The Smart Strawberry Disease Classification System in Android Mobile App is the name of the design of this Android-based mobile application, which includes architectural design, back-end, front-end, and system integration phases.

3.1.2. Integration System Firebase



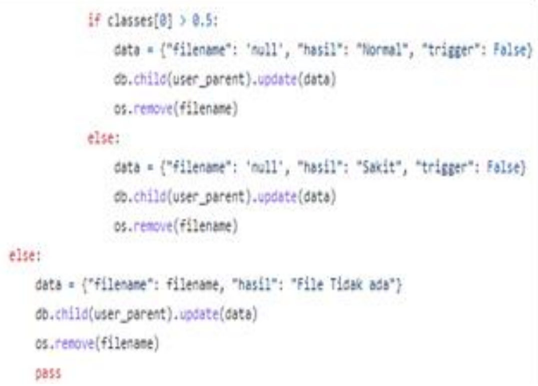
```
app = Flask(__name__)
#
CORS(app)
# creating an API object
api = Api(app)


class prediction(Resource):
    def get(self, files):
        config = {
              "apiKey": "AIzaSyARYe2vK3CM8wfpmFBjozahoK1uYhX4zuk",
              "authDomain": "strawberries-3fbf3.firebaseapp.com",
              "databaseURL": "https://strawberries-3fbf3-default-rtdb
              "storageBucket": "strawberries-3fbf3.appspot.com",
        }

        firebase = pyrebase.initialize_app(config)
        storage = firebase.storage()
        db = firebase.database()
```

Fig 9: Integrated System

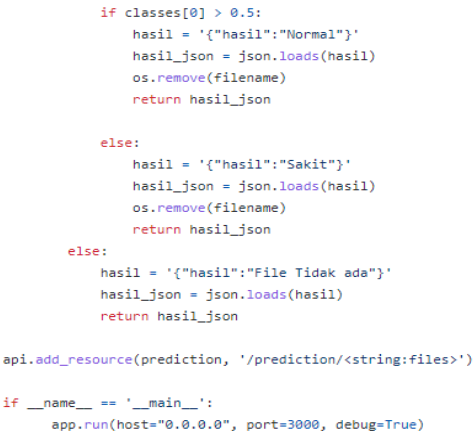
In an integrated system which connects code to the flask framework on firebase by creating an API Key dictionary, auth Domain, URL database, and Storage Bucket. Then create storage for the database and initialize it with the dictionary that was created.



```
          if classes[0] > 0.5:
              data = {"filename": 'null', "hasil": "Normal", "trigger": False}
              db.child(user_parent).update(data)
              os.remove(filename)
          else:
              data = {"filename": 'null', "hasil": "Sakit", "trigger": False}
              db.child(user_parent).update(data)
              os.remove(filename)
  else:
      data = {"filename": filename, "hasil": "File Tidak ada"}
      db.child(user_parent).update(data)
      os.remove(filename)
      pass
```

Fig 10: Store Database

Store data on the firebase server where if classes [0] > 0.5 will be inserted into the database to be presented in normal conditions and otherwise will be included in sick strawberry leaf plants. If the file is not shown, the results will indicate that the image file does not exist and the database will continue to be updated.



```
            if classes[0] > 0.5:
                hasil = '{"hasil":"Normal"}'
                hasil_json = json.loads(hasil)
                os.remove(filename)
                return hasil_json

            else:
                hasil = '{"hasil":"Sakit"}'
                hasil_json = json.loads(hasil)
                os.remove(filename)
                return hasil_json
        else:
            hasil = '{"hasil":"File Tidak ada"}'
            hasil_json = json.loads(hasil)
            return hasil_json

api.add_resource(prediction, '/prediction/<string:files>')

if __name__ == '__main__':
      app.run(host="0.0.0.0", port=3000, debug=True)
```

Fig 11: Display on Local Server

The process of displaying images on the web server in Json form is depicted above, where if classes [0] > 0.5 will be displayed in Normal conditions and otherwise it will enter a sick strawberry leaf plant. If the file is not displayed, the results assume that the image file does not exist.

Table 1: Prediction Result Data

1	Classes	Result
2	> 0.5	Normal
3	< 0.5	Sick
4	-	Missing Files

3.1.3 System Testing

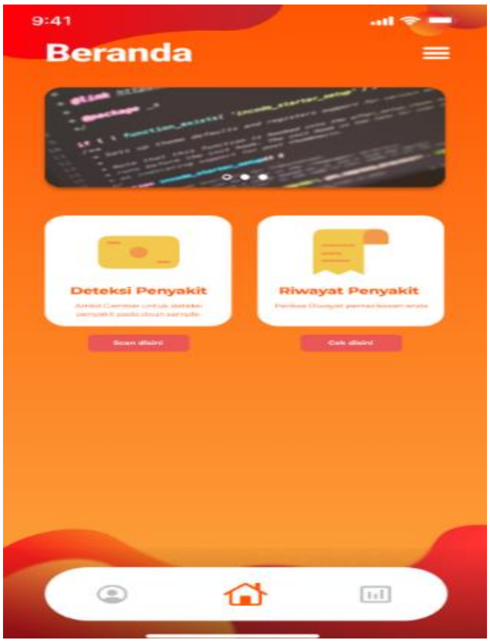


Fig 12: Application Main View

The main menu display displays disease detection and disease history, which is designed in figma and connected to a responsive display via flutter where in the application there is a menu to enter the disease detection screen display and the menu to enter the disease history screen display.

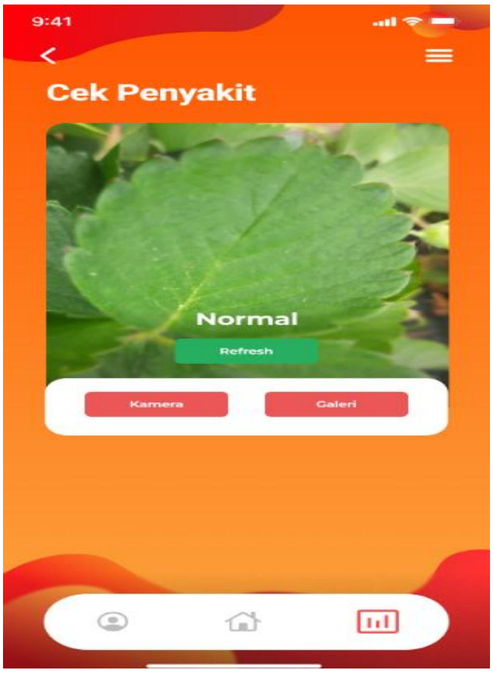
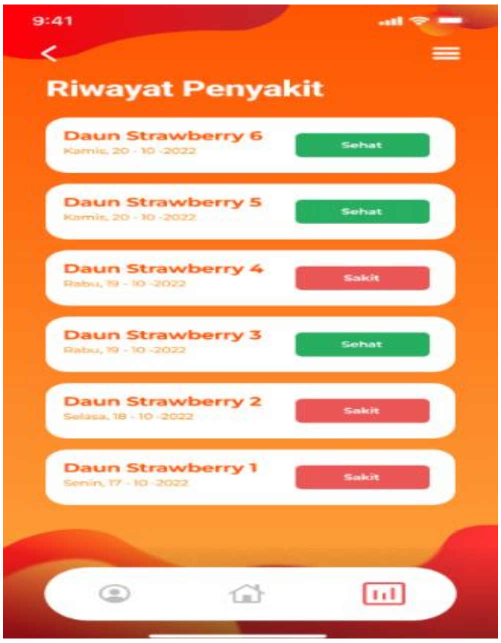


Fig 13: Display Menu Check Plant Disease

This menu display shows checking for diseases in strawberry leaf plants by inputting image data from the gallery and also using the camera directly and will be uploaded to be detected connected to the flask framework which has deployed a model from the training data results and integrated into firebase where the user interface will display the prediction results for healthy/normal and unhealthy plants.



Riwayat Penyakit

Daun Strawberry 6
Kamis, 20 - 10 -2022
Sehat

Daun Strawberry 5
Kamis, 20 - 10 -2022
Sehat

Daun Strawberry 4
Rabu, 19 - 10 -2022
Sakit

Daun Strawberry 3
Rabu, 19 - 10 -2022
Sehat

Daun Strawberry 2
Selasa, 18 - 10 -2022
Sakit

Daun Strawberry 1
Senin, 17 - 10 -2022
Sakit

Fig 14: Menu History

This menu display shows the history of diseases that have been detected, where data is stored in a database that is linked to the application in real-time by displaying the date and time when the application was detected.

Tabel 2: Test Result

No.	Jumlah Gambar	Tipburn	Leafspot	Healthy	Duration (ms/image)	Average (s)
1	6	1	3	2	68 ms	1.133 s
2	4	1	1	2	70 ms	0.175 s
3	8	4	2	2	50 ms	1.625 s
4	10	3	4	3	36 ms	1.460 s
5	3	1	1	1	73 ms	2,433 s

4. Conclusion

The conclusion drawn in this paper is the process of deploying a model that is integrated with flask and the Firebase database which can be seen in real time in the application and the advantages of this application are that it can monitor diseases in strawberry plants in real time and can detect them directly or can be taken from a direct file, and the shortcomings cannot be applied or marketed directly by making a design to be able to see strawberry leaf plant diseases as a whole without checking one by one and having accuracy more greather than 90% So that it can see the clas-sification of diseases in plants on strawberry leaves with good accuracy.

Acknowledgment

This research is fully funded by PT Telkom Indonesia, in the division of Telkom Corporate University Center or Indonesia Telecommunication and Digital Research Institute (ITDRI).

References

[1] A. Efrilla and S. A., "Klasifikasi Penyakit Pada Daun Stroberi Menggunakan K-Means Clustering dan Jaringan Syaraf Tiruan," Jurnal Keteknikan Pertanian Tropis Dan Biosistem, p. 161–170, 2020.

[2] P. Harviana, H. Fitriyah and E. Setiawan, "Sistem Penghitung Stroberi Matang di Kebun berdasarkan Hue dan Saturation menggunakan Algoritme Watershed berbasis Raspberry Pi," Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, vol. 4, pp. 463-471, 2020.

[3] B. F. S. Pramedistian, "EFEKTIVITAS BEBERAPA MACAM PESTISIDA ALAMI SEBAGAI PENGENDALI

HAMA DAN PENYAKIT PADA TANAMAN STROBERI (Fragaria sp.)," Jrnal Universitas Muhammadiyah malang, 2019.

[4] A. V. Efrilla, S. B. Sulistyo, K. Wijaya, P. H. Kuncoro and A. Sudarmaji, "Klasifikasi Penyakit Pada Daun Stroberi Menggunakan K-Means Clustering dan Jaringan Syaraf Tiruan," Jurnal Keteknikan Pertanian Tropis dan Biosistem, vol. 8, no. 2, pp. 161-170, 2020.

[5] M. Faisal, "Rancang Bangun Sistem Monitoring Tanaman Stroberi Menggunakan Pengolahan Citra Digital Berbasis Internet Of Things," Jurnal Universitas Komputer Indonesia, 2021.

[6] H. F. R. S. Oka Mahendra, "Comparison of Features for Strawberry Grading Classification with Novel Dataset," in International Conference on Computer, Control, Informatics and its Applications (IC3INA), Tangerang, Indonesia, 2018.

[7] N. Chatterjee, S. Chakraborty, A. Decosta and A. Nath, "Real-time Communication Application Based on Android Using Google Firebase," International Journal of Advance Research in Computer Science and Management Studies, vol. 6, no. 4, pp. 74-78, 2018.

[8] A. Singh, R. Akash and G. Rajan V, Flower Classifier Web App Using Ml & Flask Web Framework, India: 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), 2022.

[9] P. Karuppusamy, I. Perikos, F. Shi and T. N. Nguyen, InSustainable Communication Networks and Application, India: Conference Chair, ICSCN, 2020.

[10] Rully Pramudita, Rita Wahyuni A., Ari Nurul Alfian, and Nadya S., "Penggunaan Aplikasi Figma Dalam Membangun UI/UX Yang Interaktif," Buana Pengabdian, vol. 3, no.1, pp. 149-154, 2021.

[11] D. Dharmayanti A.M Bachtiar & A.P Wibawa "Analysis of User Interface and User Experience on Comrades Application," in IOP Conference Series: Materials Science and Engineering, Jawa Barat, Indonesia, 2018.

[12] M. R. Mufid, A. Basofi, M. U. H. Al Rasyid, I. F. Rochmansyah, and A. rokhim, "Design an MVC Model using Python for Flask Framework Development," International Electronics Symposium (IES), Surabaya, Indonesia, pp. 214–219, 2019

[13] Firebase, 'Structure Your Database', https://firebase.google.com/docs/database/web/structure-data#best_practices_for_data_structur

[14] Chunnu Khawas and Pritam S., "Application of Firebase in Android App Development-A Study," International Journal of Computer Applications , vol. 179, pp. 49-53, 2018.

[15] S. Santoso, D. J. Surjawan, and E. D. Handoyo, "Pengembangan Sistem Informasi Tukar Barang Untuk Pemanfaatan Barang Tidak Terpakai dengan Flutter Framework," J. Tek. Inform. dan Sist. Inf., vol. 6, no. 3, pp. 589–598, 2020.

[16] Benny Suhendra, Tb.Dedy Fuady, and Yoga Herdian, Rancang Bangun Sistem Monitoring dan Controlling Suhu Ideal Tanaman Stroberi Berbasis Internet of Things (IoT)," Universitas Banten Jaya., vol. 5, no. 1, pp. 48-60, 2021.

[17] Caesaron, "Evaluasi Heuristic Desain Antar Muka (Interface) Portal Mahasiswa (Studi Kasus Portal Mahasiswa Universitas X)," J. Metris, vol. 16, pp. 9–14, 2019.

[18] M. Aisyiyah, "Analisis User Interface Dalam Aplikasi Mobile," Library Ipusnas, vol. 4, no. 0271. 2019.

[19] R. Irsyad, "Penggunaan Python Web Framework Flask Untuk Pemula," ResearchGate, p. 4, 2018.