

# SISTEM MONITORING DAN KLASIFIKASI GREENHOUSE TANAMAN MELON BERBASIS APLIKASI MOBILE REACT NATIVE (STUDI KASUS: ROOFTOP SMART URBAN VERTICAL FARMING UNIVERSITAS TELKOM SURABAYA)

1<sup>st</sup> Rizqika Ibnu Khattab

Department of Information System  
Universitas Telkom  
Surabaya, Indonesia

[rizqikaibnu@student.telkomuniversity.ac.id](mailto:rizqikaibnu@student.telkomuniversity.ac.id)

2<sup>nd</sup> Helmy Widyantara

Department of Information  
Technology

Universitas Telkom Surabaya, Indonesia

[helmywidyantara@telkomuniversity.ac.id](mailto:helmywidyantara@telkomuniversity.ac.id)

3<sup>rd</sup> Purnama Anaking

Department of Information System  
Universitas Telkom  
Surabaya, Indonesia

[purnamaanaking@telkomuniversity.ac.id](mailto:purnamaanaking@telkomuniversity.ac.id)

**Abstrak** — Urban farming merupakan solusi dalam mengatasi ketersediaan pangan pada masa yang akan datang, mengingat tingginya peningkatan jumlah penduduk di Indonesia dan telah melampaui 278 juta jiwa. Pada tahun 2035 diprediksi 67% penduduk Indonesia akan tinggal di perkotaan, dampak dari urbanisasi tersebut akan menimbulkan masalah terutama terkait dengan ketersediaan pangan. Food and Agriculture Organization (FAO) memprediksi kebutuhan pangan mengalami peningkatan 70 %, pada tahun 2050. Urban farming memiliki kendala terkait dengan pengairan, dan nutrisi untuk tumbuh kembang tanaman. Kondisi lahan pertanian di kota yang sangat terbatas dan rata rata memiliki tingkat kesuburan yang relatif rendah, menyebabkan para petani mengandalkan pupuk kimia. Dalam menghadapi permasalahan tersebut tim smart urban vertical farming, menggagas suatu solusi dengan melakukan budidaya melon yang terintegrasi dengan Internet Of Things (IoT). Tetapi budidaya tersebut masih manual serta memakan waktu lebih banyak dan tidak efisien, hingga menyebabkan data parameter tidak terintegrasi dan tidak realtime antara alat IoT dan Cloud Database Antares, hal ini menjadi suatu permasalahan yang diangkat penulis kedalam penelitian untuk memberikan solusi. Untuk memberikan solusi penulis membangun sebuah sistem monitoring tanaman melon yang didalamnya terdapat sistem klasifikasi atau pengambil keputusan, yang diterapkan dalam aplikasi mobile untuk meningkatkan efisiensi pemantauan parameter data. Penelitian ini menggunakan sensor N, P, K, pH, kelembapan tanah, temperature tanah, dan elektrokonduktifitas untuk melakukan monitoring green house. Untuk melakukan integrasi data parameter, data dikirim melalui program PLC dan disimpan kedalam Cloud Database Antares, data kemudian dikelola menggunakan metode machine learning guna mengetahui tingkat kesuburan tanah. Dengan menggunakan API data dikirim ke aplikasi mobile. Sehingga output yang dihasilkan dalam penelitian ini berupa sistem monitoring dengan klasifikasi atau pengambil keputusan yang diimplementasikan kedalam aplikasi perangkat lunak berbasis mobile.

**Kata kunci**— Smart Urban Vertical Farming, IoT(Internet Of Things), Sensor NPK, Cloud Database Antares, Aplikasi Mobile.

## I. PENDAHULUAN

Pertanian perkotaan menjadi solusi untuk mengatasi ketersediaan pangan di masa depan, mengingat peningkatan signifikan jumlah penduduk Indonesia yang sudah melebihi 278 juta jiwa. Pada tahun 2035, diperkirakan 67% penduduk Indonesia akan tinggal di perkotaan [1]. Dampak urbanisasi ini akan menimbulkan tantangan, khususnya mengenai

ketersediaan pangan. Organisasi Pangan dan Pertanian (FAO) memperkirakan peningkatan permintaan pangan sebesar 70% pada tahun 2050 [2]. Untuk mengatasi masalah terbatasnya ketersediaan pangan, penduduk perkotaan telah mengkonsep pertanian perkotaan. Pertanian perkotaan memungkinkan individu menanam pangan secara lokal dengan cara yang fleksibel untuk mendapatkan produk segar dan terjangkau [3]. Konsep ini sangat relevan karena menghilangkan kebutuhan petani perkotaan untuk mencari lahan yang jauh untuk bercocok tanam [4].

Namun, pertanian perkotaan menghadapi tantangan karena terbatasnya dan relatif tidak subur nya lahan pertanian di perkotaan, menyebabkan petani bergantung pada pupuk kimia yang merugikan kesuburan tanah[5], [6]. Untuk mengatasi masalah tersebut, makalah ini mengusulkan untuk mengintegrasikan teknologi IoT ke dalam budidaya tanaman melon.

Mengacu pada penelitian sebelumnya mengenai model berbasis IoT untuk memantau pertumbuhan tanaman di rumah kaca, penggunaan IoT di rumah kaca secara signifikan membantu dalam mengelola pendinginan, pemanasan, kelembapan tanah, dan masa depan lainnya [7]. Dengan mengembangkan sistem pemantauan tanaman melon dan mengintegrasikannya ke dalam aplikasi seluler, efisiensi dalam pemantauan tanaman melon dapat ditingkatkan sehingga memberikan solusi terhadap tantangan tersebut. Metodologi penelitian meliputi studi lapangan, tinjauan literatur, analisis kebutuhan pengguna, perancangan sistem, implementasi dan pengujian black-box [8].

Pengembangan aplikasi monitoring melon plany menggunakan pemrograman ekstrim, sebuah metodologi yang telah diulas pada studi metode agile sebelumnya, yang biasa digunakan dalam pengembangan aplikasi mobile [9]. Pemrograman ekstrim (XP) melibatkan beberapa tahapan berulang: Perencanaan, Desain, Pengodean, Implementasi, dan Pengujian. Hasil pengembangan aplikasi divalidasi melalui pengujian black-box.

Pengujian black-box merupakan metode pengujian perangkat lunak dimana pengujian dilakukan tanpa mengetahui struktur internal atau logika program yang diuji [10]. Tujuan utamanya adalah untuk memverifikasi apakah program yang diimplementasikan berfungsi sesuai dengan persyaratan yang ditentukan dan untuk mengidentifikasi potensi kesalahan atau bug. Metode pengujian ini berfokus

pada proses input-output dari sudut pandang pengguna eksternal, dengan mengikuti spesifikasi yang telah ditentukan secara ketat [11].

## II. KAJIAN TEORI

### A. Urban Farming

Menurut [3], urban farming adalah praktik bercocok tanam dan berternak hewan di kota, seperti di atap atau ruang dalam, menggunakan metode seperti hidroponik, akuaponik, dan aeroponik. Ini bertujuan mengatasi tantangan krisis pangan, keterbatasan lahan, dan ketersediaan air di perkotaan, serta meningkatkan ketahanan pangan dan sistem pangan berkelanjutan. Penelitian ini menerapkan urban farming di rooftop Universitas Telkom Surabaya, memanfaatkan lahan minimal dengan teknik hidroponik dan pertanian vertikal.

### B. Sensor NPK

Sensor NPK adalah alat yang digunakan untuk mendeteksi kandungan *nitrogen, phosphorus, potassium, soil conductivity, soil measurement, soil temperature* dan pH. Dalam penelitian ini, peneliti menggunakan sensor NPK untuk melakukan pengambilan data dari setiap parameter yang dihasilkan, data tersebut dapat dibaca oleh sensor karena telah diprogram menggunakan program PLC dan dikirim menuju PLC serta ditampilkan melalui sebuah HMI. Yang kemudian, data tersebut dikirim ke platform antares melalui *IoT gateway*.

### C. Antares

Mengutip dari website resmi antares. Antares merupakan sebuah *horizontal IoT platform*, yang digunakan sebagai tempat penyimpanan data – data yang tekoneksi dengan *transmit* dari perangkat yang terhubung ke *platform*, dengan tujuan menghasilkan suatu solusi untuk mengkombinasikan antara perangkat, konektivitas dan *platform* menjadi satu kesatuan. Dalam penelitian ini peneliti menggunakan *antares* sebagai *cloud database* atau penyimpanan awan untuk menampung data dari parameter tanaman melon yang dikirimkan melalui perangkat sensor NPK melalui *IoT gateway*. Pada penelitian [12] antares digunakan sebagai koneksi jaringan pada setiap titik yang digunakan untuk mengetahui maps lingkungan perumahan, dan juga mengirimkan data melalui platform antares sehingga data dapat ditampilkan pada sistem.

### D. React Native

*React native* merupakan sebuah *framework* berbasis *javascript* yang digunakan untuk mengembangkan aplikasi berbasis *mobile* secara *cross* yakni sistem operasi *android* dan *ios*. Dalam jurnal [13] *React Native* adalah sebuah *framework* berbasis *Javascript* yang digunakan untuk mengembangkan aplikasi *mobile* di dua sistem operasi secara bersamaan, yaitu *Android* dan *iOS*. Dengan menggunakan *react native* peneliti dapat mengembangkan aplikasi *mobile* secara *cross platform*, biaya yang dikeluarkan lebih terjangkau dan waktu yang fleksibel. Dalam penelitian ini, peneliti menggunakan *react native* sebagai pengembangan *user interface* dan *user experience* atau *frontend*, dimana tampilan aplikasi *mobile* dilihat dan

di rasakan langsung oleh pengguna melalui device *smartphone*.

## III. METODE

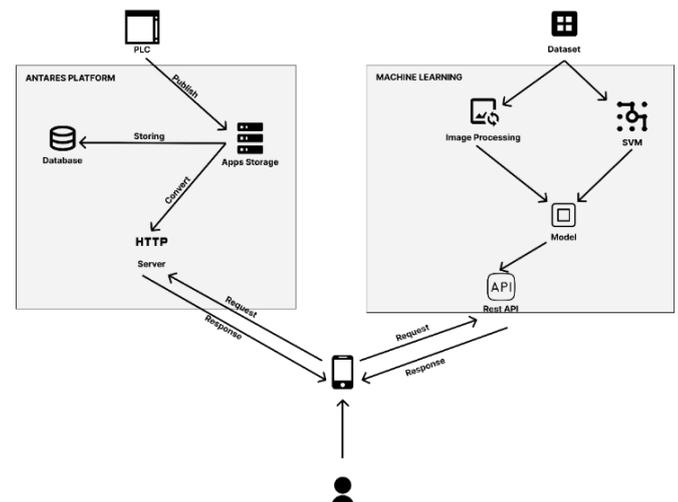
### 3.1 Analisa Kebutuhan

Dalam penelitian ini peneliti mengidentifikasi dan mengamati objek untuk memahami hakikat objek yang diteliti. Hal ini dilakukan untuk mengetahui alat apa saja yang akan disiapkan untuk melakukan penelitian. Mulai dari menyiapkan alat, menginstal perangkat lunak pendukung dan membuat lingkungan atau ruang lingkup.

### 3.2 Pembangunan Aplikasi

Pada tahap ini peneliti merancang dan membangun konsep yang dapat menghasilkan suatu keluaran berupa aplikasi. Berikut tahapan yang dilakukan peneliti untuk mengembangkan aplikasi:

#### a) Desain Sistem



**Gambar 3.** Error! No text of specified style in document..1 Desain Sistem Sistem Monitoring Melon

1. Permintaan Pengguna: Pengguna mengirimkan permintaan melalui aplikasi seluler ke server API untuk data klasifikasi dan deteksi, atau ke server HTTP untuk data pemantauan tanaman melon.
2. Penanganan server: Server HTTP pada platform Antares menerima permintaan dan memuat informasi dari database, yang menyimpan data dari server aplikasi atau server penyimpanan, yang diperoleh melalui proses berlangganan antara PLC dan platform Antares.
3. Respons Server: Server HTTP merespons dengan mengirimkan data yang diminta ke aplikasi seluler melalui protokol HTTP,
4. Visualisasi Data: Aplikasi seluler memvisualisasikan data yang diterima untuk pengguna.
5. Permintaan Klasifikasi/Deteksi: Jika pengguna meminta data klasifikasi atau deteksi, aplikasi mengirimkan permintaan ke server melalui REST API.
6. Pemrosesan Data: Server memproses permintaan dengan mengirimkan data ke server pembelajaran mesin, yang melakukan dua jenis pelatihan: mendukung mesin vektor untuk klasifikasi dan pemrosesan gambar untuk deteksi.
7. Konversi Model: Setelah pelatihan, data diubah menjadi

format model untuk diproses oleh REST API.

8. Respons API: REST API, setelah menerima permintaan dari pengguna dan model data dari server, merespons aplikasi seluler dalam format JSON.

#### b) Instalasi Sensor NPK

Dalam tahap ini, peneliti melakukan pemasangan sensor npk pada tanaman melon dengan cara ditaman pada media tanah sebanyak 4 buah sensor dan tanaman. Tanaman melon yang telah tertanam alat sensor telah dibungkus pada sebuah *polybag*, untuk menampung media tanah, tanaman, selang pengairan dan sensor npk.

#### c) Koneksi Sensor NPK ke PLC

Setelah melakukan pemasangan sensor npk dan pembuatan aliran listrik atau kabel, tahap berikutnya adalah menghubungkan sensor npk dengan plc. Dengan cara menyambungkan kabel melalui sebuah terminal, yang kemudian kabel tersebut terhubung dengan plc. Jika telah terhubung data sensor dapat ditampilkan oleh plc melalui sebuah hmi yang telah tersambung dengan plc. Yang dimana data sensor yang tampil dapat digunakan untuk melakukan monitoring dengan plc dan hmi.

#### d) Koneksi PLC ke Antares

Data yang telah terhubung dengan plc, data tersebut dikirim ke *cloud platform database antares* melalui sebuah *iot gateway*. Data yang telah terkirim dan terhubung dapat dilihat dalam situs <https://console.antares.id/>, dimana situs tersebut memuat informasi tentang data parameter tanaman melon yang telah dikirim oleh plc melalui *iot gateway*.

#### e) Pengumpulan Data

Dengan terjadinya penyimpanan data melalui *antares*, peneliti dapat melakukan pemantauan dari lokasi manapun serta dapat melakukan pengumpulan data. Data yang diambil akan berlangsung selama kurang lebih 3 bulan setiap pagi dan sore setelah dilakukannya penyiraman pada tanaman melon. Data parameter yang diambil meliputi data pH, *nitrogen*, *phosphorus*, *potassium*, *soil humidity*, *soil conductivity* dan *soil temperature*.

#### f) Pembuatan Dataset

Data yang telah dikumpulkan akan diubah menjadi sebuah dataset model untuk dilakukan *machine learning*. Untuk membuat dataset, data akan diklasifikasikan dengan label sesuai dengan tingkat kesuburan tanah. Label klasifikasi terdiri dari sangat subur, subur, kurang subur, buruk, yang dimana nilai data yang ditentukan untuk melakukan klasifikasi berasal dari wawancara peneliti bersama dengan pakar tanaman melon di Puspa Lebo, Sidoarjo.

#### g) Melatih dan Mengolah Data

Dataset dalam format .csv digunakan untuk membuat sistem klasifikasi menggunakan algoritma support vector machine (SVM). Proses ini melibatkan persiapan dataset, pre-processing dengan pelabelan dan standarisasi, pembagian data menjadi data pelatihan dan pengujian, serta evaluasi skor akurasi model. Model SVM akan memplot data untuk menemukan hyperplane yang memisahkan data

ke dalam empat kategori: sangat subur, subur, kurang subur, dan buruk. Setelah model dilatih, data baru dapat diklasifikasikan sesuai tingkat kesuburan tanah.

#### h) Pembuatan Sistem Backend

Peneliti membangun sistem server side atau backend menggunakan Firebase untuk menangani autentikasi (auth), pembuatan, pembacaan, pembaruan, dan penghapusan data (CRUD). Sistem auth mengenali pengguna melalui login, memungkinkan pengaturan ulang dan pembaruan kata sandi. Sistem CRUD mengelola data non-user seperti ketersediaan alat dan pupuk, memungkinkan pengguna untuk membuat, mengubah, menampilkan, dan menghapus data produk.

#### i) Membangun API

Dalam tahap sebelumnya, yakni penghubungan data plc ke antares, pelatihan dan pengolahan data secara *machine learning*, serta pembuatan sistem *backend*. Langkah tersebut menghasilkan sebuah proses akhir, yaitu pembuatan *endpoint api*, dimana semua *endpoint* dikumpulkan menjadi satu *directory* bernama *collection*. Untuk melakukan tahap ini, peneliti menggunakan *software tools postman*. Dalam *postman collection*, menampung data atribut pH, *nitrogen*, *phosphorus*, *potassium*, *soil humidity*, *soil conductivity*, *soil temprature*, *classification data* dan *object detection*.

#### j) Desain Aplikasi Mobile

Setelah membuat *backend*, peneliti membuat *prototype* desain aplikasi *mobile* yang dimana peneliti menggunakan *tools figma* untuk membuat tampilan *interface* pengguna. Dengan membuat desain di *figma*, peneliti dapat melakukan aksi *prototype* desain yang bertujuan untuk melakukan simulasi desain guna mendapatkan pengalaman pengguna lebih lanjut terhadap tampilan *interface* yang dibuat.

#### k) Evaluasi UI/UX

Peneliti melakukan pengujian desain menggunakan User Experience Questionnaire (UEQ) untuk mengukur pengalaman pengguna (UX) aplikasi. UEQ mencakup enam skala penilaian: daya tarik, kejelasan, efisiensi, ketepatan, stimulasi, dan kebaruan.

#### l) Integrasi

Pada tahap ini, peneliti melakukan integrasi antara alat, sistem dan aplikasi. Dimana diumulai dari melakukan integrasi data sensor dan data *classification* ke *react native* melalui *endpoint api*.

#### m) Pembuatan Aplikasi

Setelah melakukan proses diatas, peneliti akan membangun aplikasi *mobile* dengan bantuan *framework react native* untuk menampilkan data – data yang telah dikelola kedalam sebuah *mobile apps* berbasis *platform android* dan *ios*.

### 3.3 Pengujian Aplikasi

Dalam tahap ini, peneliti melakukan pengujian aplikasi yang telah dibuat. Dimana peneliti mengecek keseluruhan fitur dalam aplikasi untuk mencari *bug* dan *error* pada sistem. Selain pencarian *bug* dan *error*, peneliti juga melakukan tes *performance* pada aplikasi yang telah dibuat.

### 3.4 Deployment

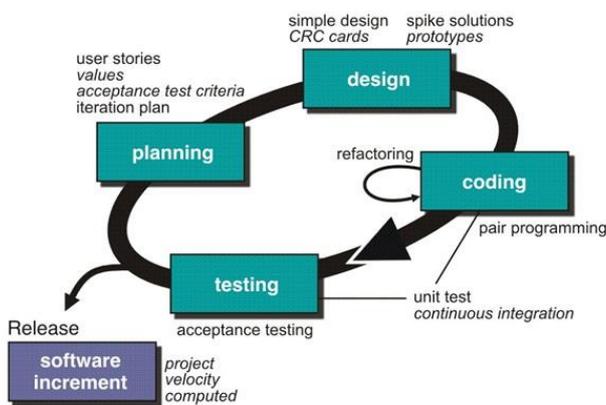
Aplikasi mobile yang telah dibuat dilakukan deploy melalui *easy build* untuk menghasilkan file berbentuk *.apk*. Yang dimana file *.apk* tersebut dapat di jalankan atau di *install* ke *device smartphone android*.

### 3.5 Penerapan Aplikasi

Aplikasi yang telah di-*build* dan diuji akan di *install* ke masing – masing device anggota *r/c* yang bertugas untuk melakukan budidaya tanaman melon.

## IV. HASIL DAN PEMBAHASAN

### 4.1 Hasil Penelitian



Gambar 4.1 Agile Extreme Programming

Sumber 1 <https://machlizadevi.blog.binusian.org/>

Dalam metodologi Agile, khususnya Extreme Programming, metode ini menghasilkan iterasi untuk mencapai kemajuan dalam mencapai tujuan. Artinya setiap iterasi menghasilkan umpan balik yang akan digunakan untuk melanjutkan ke iterasi berikutnya sesuai kondisi yang diperlukan. Dalam pengembangan aplikasi mobile monitoring melon ini dilakukan beberapa iterasi dengan tujuan yang berbeda-beda, yaitu:

- Iterasi pertama dengan tujuan mengembangkan fitur inti yaitu monitoring dan klasifikasi.
- Iterasi kedua dengan tujuan deteksi penyakit.
- Iterasi ketiga dengan tujuan sistem notifikasi manajemen.

### 4.2 Planning

Berdasarkan analisis kebutuhan yang dilakukan pada tahap persyaratan, persyaratan pemantauan dan klasifikasi sistem dapat diringkas dalam format cerita pengguna seperti yang ditunjukkan pada tabel di bawah ini:

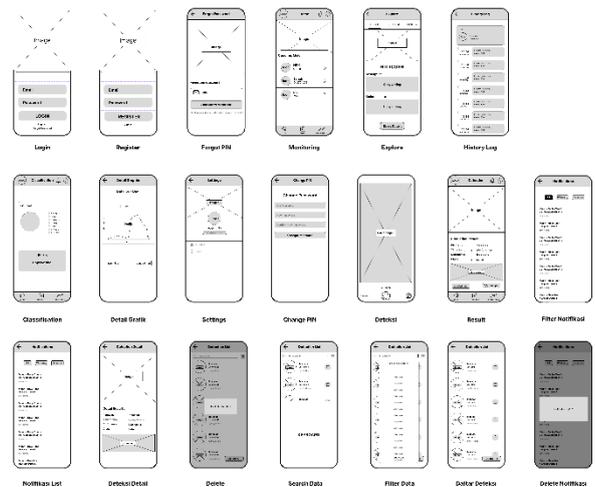
Table 4.1 User Stories Sistem Monitoring Melon

Feature	User Stories	Tasks
Authentication	User dapat melakukan	Konfigurasi autentikasi

	Registrasi, login dan reset PIN.	firebase dan membuat tampilan frontend.
Monitoring	User dapat memantau perkembangan tanaman melon dan riwayat kondisi tanaman melon.	Menghubungkan endpoint API dengan sistem monitoring dan membuat tampilan frontend.
Classification	User mengetahui kondisi kesuburan tanaman melon.	Menghubungkan endpoint API dan membuat tampilan frontend.
Settings	User dapat melihat profil, mengubah pin dan melakukan logout.	Membuat fungsi ubah pin, logout dan membuat tampilan frontend.
Deteksi Penyakit	Identifikasi kesehatan tanaman melon melalui kamera	Membuat API deteksi, kamera akses dan membuat tampilan hasil deteksi.
List Deteksi	User dapat mengolah data deteksi penyakit tanaman melon.	Konfigurasi firebase realtime database, firestore dan firebase storage.
Background Service	Mengirim notifikasi	Membuat fungsi service dan

Notification	rutin pagi dan sore.	menjalankan di server.
List Notifikasi	User dapat mengolah data notifikasi.	Membuat penyimpanan dan pengambilan data notifikasi dan membuat tampilan frontend.
Filter Notifikasi	User dapat memilih jenis data notifikasi.	Membuat fungsi dan tombol filter.
Delete Notifikasi	User dapat menghapus data notifikasi.	Membuat fungsi hapus notifikasi.
Count Notifikasi	User mengetahui jumlah data notifikasi.	Membuat fungsi total data notifikasi.

## B. Wireframe



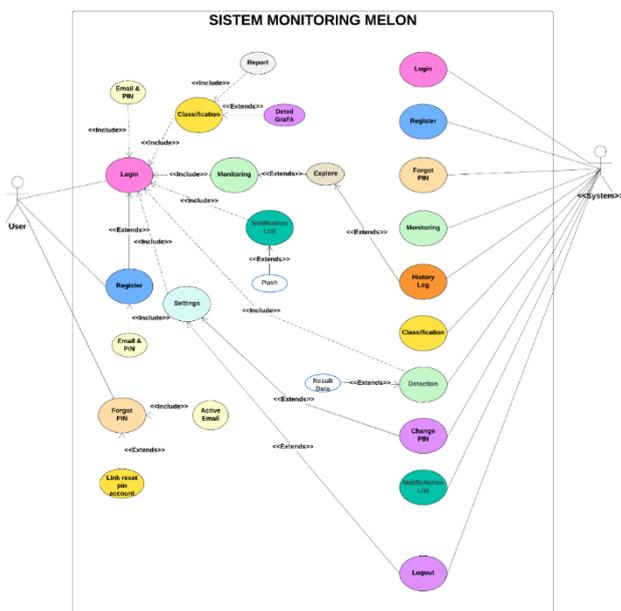
Gambar 4. **Error! No text of specified style in document.**3 Wireframe Sistem Monitoring Melon

Dalam wireframe, proses autentikasi dimulai dengan pendaftaran akun menggunakan email dan pin, diikuti dengan login. Setelah autentikasi, pengguna dapat memantau nutrisi dan elemen fisik tanaman melon di halaman utama, eksplorasi, dan riwayat, serta melihat klasifikasi kondisi kesehatan melalui grafik. Pengguna juga dapat mengeksport laporan data. Pada sistem deteksi menggambarkan deteksi penyakit tanaman melon melalui kamera, dengan hasil deteksi ditampilkan dan data dikelola di layar daftar deteksi. Dalam sistem notifikasi menunjukkan pengelolaan notifikasi, termasuk melihat, memfilter, dan menghapus data notifikasi.

## 4.3 Design

### A. Use Case Diagram

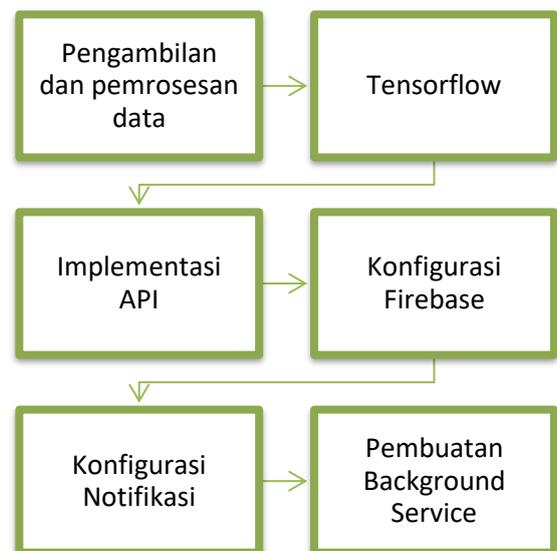
Sistem yang dikembangkan dapat dijelaskan melalui use case diagram. Diagram use case merupakan salah satu jenis representasi UML yang bertujuan untuk menyajikan fungsi-fungsi sistem yang dikembangkan sebagai dasar pertimbangan. Diagram ini digambarkan dengan merepresentasikan pengguna sebagai aktor dan sistem sebagai use case [14]. Dalam use case sistem yang dibuat, terdapat pengguna dan aktor sistem. Berikut gambaran keseluruhan sistem yang dibangun pada sistem monitoring dan klasifikasi.



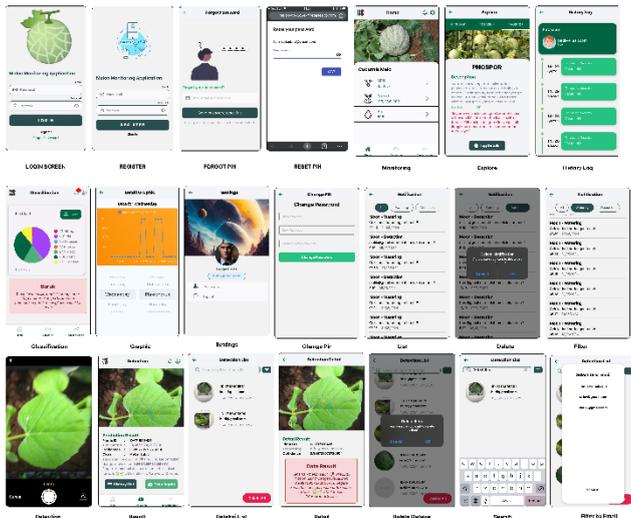
Gambar 4. **Error! No text of specified style in document.**2 Use Case Sistem Monitoring Melon

## 4.4 Coding

Berikut merupakan alur dari proses *coding*, dimulai dari pengolahan data hingga pembuatan *service*.



## 4.5 Implementation



Gambar 4.4 Implementasi Sistem Monitoring Melon

Dalam proses implementasi *wireframe* yang telah dibuat pada gambar 4.3 diimplementasikan pada bentuk tampilan pada gambar 4.4, dengan terdiri dari beberapa sistem dan tampilan seperti. Sistem autentikasi memungkinkan pengguna mendaftar, masuk, dan mengatur ulang PIN melalui email. Layar pemantauan menyediakan informasi real-time tentang nutrisi dan parameter fisik tanaman melon, dengan detail tambahan di layar eksplorasi dan riwayat. Sistem klasifikasi menampilkan data nutrisi melalui diagram lingkaran dan grafik garis, serta memungkinkan ekspor data dalam bentuk PDF. Sistem deteksi memungkinkan penghapusan, pencarian, dan pemfilteran data, serta deteksi penyakit tanaman melalui kamera. Sistem manajemen notifikasi menyediakan tampilan, pemfilteran, dan penghapusan notifikasi, sementara layar pengaturan memungkinkan pengguna mengganti PIN dan logout.

## 4.6 Testing

### a) Test Case

Table 4.2 Test Case Sistem Monitoring Melon

<b>Id Case</b>	<b>Test</b>	<b>Expected output</b>	<b>Observed output</b>	<b>Pass / Fail</b>
TC1	Invalid login, please fill the fields	Unable to log in, an alert appears		Pass
TC2	Invalid email	Invalid email format		Pass
TC3	Invalid email	Invalid email format		Pass
TC4	Invalid email	Invalid email format		Pass
TC5	Invalid email	Invalid email format		Pass

TC6	Invalid email	Invalid email format		Pass
TC7	Email is empty	Please enter email and pin		Pass
TC8	Empty pin	Please enter email and pin		Pass
TC9	User is not registered	Invalid login: check account credentials		Pass
TC10	Wrong Pin	Invalid login: check account credentials		Pass
TC11	Login successful	Login successful		Pass
TC12	Incorrect email or pin	Invalid login: check account credentials		Pass
TC13	Please fill all fields	Cannot register		Pass
TC14	Invalid email	Invalid email format		Pass
TC15	Invalid email	Invalid email format		Pass
TC16	Invalid email	Invalid email format		Pass
TC17	Invalid email	Invalid email format		Pass
TC18	Invalid email	Invalid email format		Pass
TC19	Email is empty	Please enter email and pin		Pass

TC20	Empty pin	Please enter email and pin	Pass	TC40	Password updated successfully	Password updated successfully	Pass
TC21	Successful registration	Registration successful	Pass	TC41	Displays information in the form of identifying healthy melon plants	Provides identification of healthy melons	Pass
TC22	Invalid email	Invalid email format	Pass	TC42	Displays information in the form of identifying sick melon plants	Provides identification of diseased melons	Pass
TC23	Successful registration	Registration successful	Pass	TC43	Does not display disease identification information	Displays plant disease identification information	Fail
TC24	Pin Too Short	Pin Too Short	Pass	TC44	Does not display disease identification information	Displays plant disease identification information	Fail
TC25	Please fill all fields	Please fill all fields alert	Pass	TC45	Does not display disease identification information	Displays plant disease identification information	Fail
TC26	Invalid email	Invalid email format	Pass	TC46	Does not display disease identification information	Displays plant disease identification information	Fail
TC27	Invalid email	Invalid email format	Pass	TC47	Displays data based on the searched plant ID	Data found	Pass
TC28	Invalid email	Invalid email format	Pass	TC48	Data not found	Does not display data	Pass
TC29	Invalid email	Invalid email format	Pass	TC49	Data is missing	Does not display data	Pass
TC30	Invalid email	Invalid email format	Pass	TC50	Data has been successfully filtered based on email	Displays data according to email filter	Pass
TC31	Email not active	Send password reset link success	Fail	TC51	User email is not available	User email is not displayed	Pass
TC32	Email is not registered	Send password reset link success	Fail	TC52	Data deleted	Deleted successfully	Pass
TC33	Invalid email	Invalid email format	Pass	TC53	Data is not deleted	Cancel delete	Pass
TC34	Send password reset link success	Invalid email format	Pass				
TC35	Please fill all fields	Please fill all fields alert	Pass				
TC36	Account password is wrong	Account password is wrong	Pass				
TC37	Password at least 6 characters	Password at least 6 characters	Pass				
TC38	New password does not match	New password does not match	Pass				
TC39	New pin must be different from field current pin	New pin must be different from field current pin	Pass				

TC54	Successfully sent disease notification data	The notification data for the village category has been successfully generated	Pass	testing			
				Risk-based testing	55%	70%	80%
				Security testing	50%	65%	80%
				Subroutine testing	27%	45%	60%
				System testing	27%	42%	55%
TC55	Error when sending notification village data	Failure information appears	Pass	External Beta testing	30%	40%	50%
				Performance testing	30%	40%	45%
TC56	Displays all notification data	All categories of notification data appear	Pass	Supply-chain testing	20%	40%	47%
				Cloud testing	25%	40%	55%
TC57	Displays a list of notification data for the watering category	Displays all notification data for the watering category	Pass	Function testing	33%	40%	55%
				Unit testing (automated)	20%	40%	50%
TC58	Displays a list of disease category notification data	Displays all notification data for the village category	Pass	Unit testing (manual)	15%	38%	50%
				Regression testing	35%	35%	45%
TC59	Data deleted	Deleted successfully	Pass	Independent verification	20%	35%	47%
				Clean-room testing	20%	35%	50%
TC60	Data is not deleted	Cancel delete	Pass	Acceptance testing	15%	35%	40%
				Independent testing	15%	35%	42%
				Average	29.24%	44.12%	55.06%

#### b) Deffect Removal Effect

Pada tahap evaluasi hasil pengujian, tahap ini melakukan proses evaluasi untuk memeriksa keefektifan metode dalam pengujian sistem dan aplikasi. Evaluasi dapat dilakukan dengan menghitung efisiensi penghilangan cacat sehingga diperoleh persamaan berikut [15]

$$DRE = \frac{NUMBER\ OF\ ERRORS\ FOUND}{TOTAL\ NUMBER\ OF\ TEST\ CASES} \quad (1)$$

Dimana setiap jumlah kesalahan yang ditemukan pada proses pengujian yang mengakibatkan kegagalan akan dibagi dengan jumlah total kasus pengujian yang terjadi pada aplikasi. Sehingga menghasilkan keluaran efisiensi penghapusan cacat berupa persentase kesalahan atau bug atau error pada aplikasi.

Berikut tingkat efek penghilangan cacat atau dre yang dikutip dalam jurnal [16]

Table 4.3 Deffect Removal Effect Range

Test Deffect Removal	Minimum	Average	Maximum
Experiment-based	60%	75%	85%

## V. KESIMPULAN

Penelitian ini berhasil mengembangkan sistem pemantauan dan klasifikasi budidaya melon di lingkungan urban farming menggunakan teknologi IoT dan aplikasi seluler React Native. Sistem ini mengotomatiskan pengumpulan data dari sensor yang mengukur N, P, K, kelembaban tanah, suhu, konduktivitas listrik, dan pH, mengatasi inefisiensi pemantauan manual. Modul klasifikasi memberikan informasi kesehatan tanaman melon secara real-time, sehingga meningkatkan efisiensi pemantauan. Framework React Native terbukti efektif untuk mengembangkan aplikasi seluler ramah pengguna yang kompatibel dengan berbagai perangkat Android. Secara keseluruhan, studi ini menunjukkan bahwa mengintegrasikan IoT dan teknologi seluler menawarkan solusi efisien untuk tantangan pertanian perkotaan dan pertanian modern.

## REFERENSI

- [1] D. I. W. Sarosa, "Panduan Praktis Implementasi Agenda Baru Perkotaan Untuk Kota Berkelanjutan Di Indonesia," *Pupur*, vol. 1, p. 126, 2017.
- [2] B. Parvez, R. A. Haidri, and J. K. Verma, "IoT in Agriculture," 2020.
- [3] S. Oh and C. Lu, "Vertical farming - smart urban agriculture for enhancing resilience and sustainability in food security," *Journal of Horticultural Science and Biotechnology*, vol. 98, no. 2. Taylor and Francis Ltd., pp. 133–140, 2023.
- [4] T. Qureshi, M. Saeed, K. Ahsan, A. A. Malik, E. S. Muhammad, and N. Touheed, "Smart Agriculture for Sustainable Food Security Using Internet of Things (IoT)," *Wireless Communications and Mobile Computing*, vol. 2022. Hindawi Limited, 2022.
- [5] S. V. S. Ramakrishnam Raju, B. Dappuri, P. Ravi Kiran Varma, M. Yachamaneni, D. M. G. Verghese, and M. K. Mishra, "Design and Implementation of Smart Hydroponics Farming Using IoT-Based AI Controller with Mobile Application System," *J. Nanomater.*, vol. 2022, 2022.
- [6] Y. Wu, Z. Yang, and Y. Liu, "Internet-of-Things-Based Multiple-Sensor Monitoring System for Soil Information Diagnosis Using a Smartphone," *Micromachines*, vol. 14, no. 7, Jul. 2023.
- [7] A. Q. Mohabuth and D. Nem, "An IoT-Based Model for Monitoring Plant Growth in Greenhouses," *J. Inf. Syst. Informatics*, vol. 5, no. 2, pp. 536–549, May 2023.
- [8] H. J. Christanto *et al.*, "Design and Development of the Mobile-Based Hydroponic Planting Machine Application MyHydro," *J. Inf. Syst. Informatics*, vol. 5, no. 4, pp. 1321–1339, Dec. 2023.
- [9] R. Setiawan and I. Effendy, "Implementation of Agile Methods on Development of Savings and Loan Cooperative Information Systems," *J. Inf. Syst. Informatics*, vol. 4, no. 3, 2022.
- [10] M. Nurudin, W. Jayanti, R. D. Saputro, M. P. Saputra, and D. Yulianti, "Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Teknik Boundary Value Analysis," vol. 4, no. 4, pp. 2622–4615, 2019.
- [11] E. S. J. Atmadji, I. R. Sanjaya, and H. A. Putranto, "Pemanfaatan boundary value analysis dan equivalence partitioning pada automated testing aplikasi berbasis website," *Angkasa J. Ilm. Bid. Teknol.*, vol. 15, no. 1, p. 97, May 2023.
- [12] A. Gunawan, I. Nawangsih, and S. B. Rahardjo, "MIND (Multimedia Artificial Intelligent Networking Database Penerapan Sistem Elektronik Keamanan Lingkungan Berbasis Internet of Things Menggunakan Modul LoRa Garuda," *J. MIND J. / ISSN*, vol. 8, no. 1, pp. 92–106, 2023.
- [13] I. F. Abiyyu and H. A. Tawakal, "Pengembangan Aplikasi Pendeteksi Kematangan Buah Melon: Studi Kasus Aplikasi Melonku," *J. Inform. Terpadu*, vol. 7, no. 1, pp. 27–32, 2021.
- [14] H. Muhammad and Subandi, "Perancangan Aplikasi SIGAB pada Polresta Banjarmasin Berbasis Sistem Informasi Geografis Mobile Android," *Pros. Semin. Nas. Sist. Inf. dan Teknol.*, no. 2, pp. 97–101, 2023.
- [15] I Dewa Made Widia, Sovia Rosalin, Salnan Ratih Asriningtias, and Elta Sonalita, "1. Black Box Testing Menggunakan Boundary..."