

BAB I

PENDAHULUAN

A. Latar Belakang

Kemajuan pesat terjadi di bidang teknologi informasi dan komunikasi, termasuk penggunaan smartphone yang semakin marak. Hal ini membuka peluang untuk memanfaatkan teknologi dalam berbagai bidang, termasuk budidaya tanaman dengan menggunakan teknik hidroponik. Hidroponik merupakan metode menanam tanaman tanpa tanah yang semakin populer karena memiliki banyak keuntungan, seperti hemat air, penggunaan lahan yang efisien, dan hasil panen yang lebih optimal (Hedro Wibowo, 2019).

Salah satu inovasi industri perkebunan yang dapat bekerja dengan baik di ruang terbatas, seperti atap rumah bertingkat, adalah metode penanaman hidroponik. Teknik ini tidak menggunakan tanah sebagai media tanam, melainkan menggunakan air. Air memiliki keuntungan karena lebih mudah digunakan di ruang terbatas, sehingga hidroponik cocok untuk lahan kecil. Selain itu, media yang digunakan seperti bak penampung air bahan-bahan bekas seperti gelas plastik, botol air mineral, dan styrofoam bekas dapat dimanfaatkan untuk membuat tempat penampungan air, pot tanaman, dan peralatan lainnya, atau benda lainnya, sehingga lebih ramah lingkungan dan ekonomis. Namun, banyak masyarakat yang masih belum familiar dengan metode ini dan membutuhkan panduan yang mudah diakses. agar mendapatkan hasil yang maksimal. Namun informasi tentang hidroponik saat ini masih tersebar luas di berbagai sumber, seperti buku, artikel online, dan video

tutorial. Hal ini membuat masyarakat kesulitan untuk menemukan informasi yang lengkap dan efisien. Selain itu, informasi yang tersedia belum tentu sesuai dengan kebutuhan dan kondisi masing-masing individu. Kurangnya panduan yang praktis dan mudah dipahami juga menjadi hambatan bagi masyarakat yang ingin mencoba hidroponik.

Aplikasi panduan hidroponik berbasis Android dapat menjadi solusi untuk mengatasi permasalahan di atas. Aplikasi ini dapat memberikan informasi yang lengkap dan efisien tentang hidroponik dengan cara yang mudah diakses dan dipahami. Aplikasi ini juga dapat membantu masyarakat untuk memilih sistem hidroponik yang sesuai dengan keadaan dan kebutuhan mereka. Selanjutnya, aplikasi yang bisa memberikan panduan praktis tentang cara budidaya tanaman hidroponik dan dilengkapi video budidaya hidroponik, serta dapat mendeteksi jenis penyakit dan hama tanaman hidroponik.

Berdasarkan masalah yang sudah dijelaskan, maka topik yang akan diangkat dengan judul “APLIKASI PANDUAN LENGKAP HIDROPONIK BERBASIS ANDROID” yang diharapkan dapat memudahkan dalam memahami cara bercocok tanam secara hidroponik agar lebih efektif dan efisien.

B. Rumusan Masalah

Berdasarkan informasi latar belakang yang diberikan, isu-isu berikut dapat diidentifikasi sebagai fokus penelitian ini:

1. Bagaimana mengembangkan aplikasi panduan hidroponik berbasis android ?
2. Bagaimana merancang aplikasi pendeteksi hama dan penyakit berbasis android?

C. Tujuan Penelitian

Berdasarkan rumusan masalah informasi yang diberikan, maka tujuan penelitian ini adalah:

1. Untuk dapat membuat aplikasi panduan hidroponik berbasis android.
2. Untuk dapat merancang aplikasi pendeteksi hama dan penyakit berbasis android.

D. Batasan Masalah

Berikut batasan masalah dalam penelitian ini, yaitu:

1. Aplikasi ini tidak menyediakan forum diskusi.
2. Pada penelitian ini untuk pendeteksi hama dan penyakit menggunakan hama dan penyakitnya, yaitu ulat kubis (*tritip plutella xylostell*), hama thrips, hama serangga dan burung, penyakit busuk daun dan penyakit kelainan genetik .
3. Pada menu pendeteksi hama dan penyakit tanaman hidroponik menggunakan kamera untuk mendeteksi, tidak mengambil dari galeri foto.

E. Manfaat Penelitian

Berikut manfaat dalam penelitian ini, yaitu:

1. Bagi Penulis

Peneliti dapat mengembangkan aplikasi yang bermanfaat bagi masyarakat luas dan meningkatkan ilmu pengetahuan tentang hidroponik dan teknologi Android.

2. Bagi Masyarakat

Informasi lengkap dan terpercaya mengenai hidroponik tersedia untuk masyarakat umum.

BAB II

TINJAUAN PUSTAKA

A. Kajian Hasil Penelitian Terdahulu

Penelitian ini mengacu pada beberapa penelitian terdahulu, beberapa di antaranya berkaitan dengan penelitian ini, antara lain:

1. Muh. Basri, Muhammad Zainal, Dedy. (2021). “Aplikasi Panduan Budidaya Tanaman Bawang Merah Teknik Hidroponik Berbasis Web *Responsive*”. Universitas Muhammadiyah Parepare. Tujuan penelitian ini dilakukan untuk merancang, mengimplementasikan yang membantu dan memudahkan masyarakat atau petani yang ingin bertanam bawang merah berbasis aplikasi web.
2. Asep Deddy Supriatna¹ , Dini Destiani Siti Fatimah² , Remyco Geovani³. (2022). “Perancangan Aplikasi Edukasi Cara Bercocok Tanam dengan Menggunakan Teknik Hidroponik Berbasis Web”. Institut Teknologi Bandung. Tujuan dari perancangan aplikasi ini adalah untuk Aplikasi ini diharapkan mampu mengatasi keterbatasan jumlah dan ragam sumber referensi tradisional. Semua informasi yang tersedia dalam aplikasi ini didasarkan pada proses nyata dan didukung oleh video tutorial, sehingga memungkinkan masyarakat untuk bercocok tanam sendiri di rumah.
3. Indra Fahrudin Rukmana. (2018). “Aplikasi Media Pembelajaran Penanaman Sayuran Hidroponik”. Institut Teknologi Nasional Malang. Tujuan penelitian ini memudahkan Masyarakat mengetahui jenis sayuran yang mudah ditanam secara hidroponik dan mengetahui metode-metode penanaman sayuran hidroponik.

B. Kajian Teori

1. Hidroponik

Hidroponik berasal dari kata *Yunani hydro* yang berarti air dan *ponos* yang berarti kekuatan. Teknik ini kadang-kadang disebut sebagai budidaya tanaman yang tidak menggunakan tanah. Metode pertanian hidroponik lebih menggunakan air dalam proses penanaman tanpa melibatkan tanah. Selama kebutuhan nutrisi (unsur hara) terpenuhi, tanaman tetap dapat tumbuh optimal meskipun tidak menggunakan media tanah. Selain nutrisi, faktor lain yang harus diperhatikan meliputi oksigen, air, dan paparan sinar matahari.

Keunggulan sistem hidroponik mencakup aspek ramah lingkungan, penghematan air hingga hanya 1/20 dari kebutuhan air pada metode konvensional, efisiensi waktu dan tenaga, *fleksibilitas* lokasi penanaman, tidak memerlukan lahan luas, mempercepat pertumbuhan tanaman, menjaga kualitas hasil panen, serta dapat diterapkan sepanjang tahun tanpa terpengaruh musim. Beberapa contoh media tanam yang sering digunakan dalam hidroponik meliputi arang sekam, rockwool, spons atau busa, serbuk sabut kelapa, zeolit, pecahan genteng atau batu bata.

Hidroponik memiliki berbagai jenis teknik penanaman berbeda, dari dasar hingga rumit. Setiap jenis teknik hidroponik memiliki cara kerja yang efektif untuk menunjang pertumbuhan tanaman. Pendeteksi hama dan penyakit pada penelitian ini dengan hama dan penyakitnya, yaitu ulat kubis (*tritif plutella xylostell*), hama thrips, hama serangga dan burung, penyakit busuk daun dan penyakit kelainan genetik. Berikut penjelasannya:

a. Hama dan Penyakit Tanaman

1). Hama Ulat Kubis (*Tritip Plutella Xylostell*)



Gambar 2.1 Ulat Kubis (*Tritip Plutella Xylostell*)

Ulat Kubis (*Tritip Plutella Xylostell*) merupakan salah satu hama yang sering menyerang tanaman, termasuk dalam sistem hidroponik. Ulat ini dapat memakan daun, batang, dan bunga tanaman, sehingga menyebabkan kerusakan yang signifikan pada tanaman.

Gejala serangan ulat Kubis:

- a) Daun yang diserang ulat grayak akan berlubang dan robek.
- b) Ulat Kubis dapat memakan daun, batang, dan bunga tanaman.
- c) Serangan ulat Kubis dapat menyebabkan daun menguning, layu, dan rontok.
- d) Dalam kasus yang parah, serangan ulat Kubis dapat menyebabkan tanaman mati.

2). Hama Thrips



Gambar 2.2 Hama Thrips

Thrips merupakan serangga berukuran sangat kecil dengan tubuh ramping dan sayap tipis menyerupai jarum. Hama ini tidak sepenuhnya terbang, melainkan berpindah dengan cara melompat antar tanaman, sering kali dibantu oleh hembusan angin atau peran manusia sebagai perantaranya. Gejala serangan hama thrips pada daun selada:

- a) Terjadi karena thrips menghisap cairan sel dari permukaan daun.
- b) Sel-sel yang rusak kehilangan klorofil, sehingga muncul bercak kekuningan atau keperakan pada daun.
- c) Biasanya terlihat tidak beraturan.
- d) Sering ditemukan di bagian bawah daun, tempat thrips lebih aktif.
- e) Daun yang terserang parah bisa menjadi coklat, kering, atau melintir.
- f) Terkadang, bercak kuning ini disertai dengan bercak hitam kecil, yang merupakan kotoran thrips.

3). Hama Serangga dan Burung



Gambar 2.3 Hama Serangga dan Burung

Hama serangga dan burung dapat menyebabkan kerusakan pada tanaman selada hidroponik. Serangga seperti kumbang daun dan belalang sering merusak daun dengan memakannya, meninggalkan lubang atau robekan pada daun selada. Sementara itu, burung seperti pipit atau burung kecil lainnya dapat

mematuk daun dan mengacak-acak tanaman, menyebabkan daun robek dan pertumbuhan terganggu.

Gejala serangan hama serangga dan burung pada daun selada:

- a) Hama seperti kumbang daun dan Belalang mengunyah daun, meninggalkan lubang atau robekan pada permukaan daun.
- b) Hama seperti kutu daun menghisap cairan dari jaringan daun, menyebabkan daun menguning, keriting, atau melengkung.
- c) Burung seperti pipit atau burung kecil lainnya mematuk daun, meninggalkan bekas robek atau rusak pada daun.
- d) Burung sering mematuk tunas muda, menyebabkan kerusakan pada bagian tanaman yang sedang berkembang.

4). Penyakit Busuk Daun



Gambar 2.4 Penyakit Busuk Daun

Penyakit busuk daun merupakan salah satu penyakit yang sering menyerang tanaman hidroponik khususnya pada jenis tanaman selada. Penyakit ini biasanya disebabkan oleh infeksi jamur *Phytophthora infestans* atau *Pythium spp.*, yang berkembang dalam kondisi lingkungan yang lembap dan basah.

Gejala serangan hama serangga dan burung pada daun selada:

- a) Bercak kecil berwarna cokelat atau hitam muncul pada daun, biasanya dimulai dari tepi atau ujung daun. Daun yang terinfeksi karat putih akan berubah menjadi kuning di bagian atas daun, terutama di sekitar bercak putih.
- b) Bercak tersebut akan melebar dan menyebar dengan cepat, terutama dalam kondisi lembap.
- c) Daun di sekitar area yang terkena mulai menguning karena jaringan tanaman mati.
- d) Daun menjadi layu, lemas, dan akhirnya membusuk. Pada beberapa kasus, daun yang terkena akan terlihat seperti basah.
- e) Pada bagian bawah daun yang terinfeksi, sering kali muncul lapisan putih atau abu-abu berupa spora jamur.

5). Penyakit Kelainan Genetik



Gambar 2.5 Penyakit Kelainan Genetik

Penyakit kelainan genetik pada daun selada dapat terjadi akibat mutasi genetik yang memengaruhi pertumbuhan dan perkembangan. Kelainan genetik pada daun selada tidak dapat diobati atau disembuhkan. Untuk menghindari

masalah ini, penting untuk menggunakan benih yang berkualitas dan memastikan pemilihan varietas yang sesuai dengan kondisi lingkungan pertumbuhan. Jika kelainan genetik menyebabkan kerugian, bisa dilakukan pemilihan tanaman yang tidak terpengaruh atau penerapan teknik pembiakan selektif tanaman.

Gejala serangan penyakit kelainan genetik pada daun selada:

- a) Tanaman selada yang mengalami kelainan genetik bisa menghasilkan daun dengan bentuk yang tidak sempurna, seperti daun yang terpotong, kerutan yang berlebihan, atau daun yang terlalu kecil.
- b) Mutasi genetik dapat menyebabkan daun selada memiliki warna yang tidak biasa, seperti daun yang sangat pucat, berwarna kuning atau merah pada jenis selada yang seharusnya hijau. Hal ini terjadi karena ketidakmampuan tanaman untuk memproduksi klorofil secara normal.
- c) Beberapa kelainan genetik dapat menyebabkan daun tumbuh tidak teratur atau bertumpuk dengan cara yang aneh, seperti daun yang terlalu rapat atau daun yang tumbuh berulang kali pada satu titik.
- d) Beberapa kelainan genetik dapat menurunkan daya tahan tanaman terhadap penyakit atau hama. Tanaman selada dengan kelainan genetik mungkin lebih rentan terhadap infeksi jamur, bakteri, atau serangan hama dibandingkan dengan tanaman yang sehat secara genetik.

2. Aplikasi

Menurut Fajri Rinaldi Chan,dkk,(2022), Program komputer atau perangkat lunak disebut aplikasi. Dirancang untuk mempermudah dan mempercepat penyelesaian berbagai tugas atau masalah. Aplikasi dapat digunakan di perangkat seperti komputer, laptop, atau ponsel pintar, dengan beragam fungsi, mulai dari mengirim pesan, menonton video, berbelanja, hingga bermain game.

3. Android



Gambar 2.6 Logo Android
(Sumber : google.com)

Android menawarkan metode menyeluruh untuk membuat aplikasi. Artinya, satu program Android dapat digunakan di berbagai perangkat berbasis Android, termasuk tablet, ponsel cerdas, jam tangan pintar, dan gadget lainnya. *AOSP (Android Open Source Project)*, yang bertugas mengembangkan sistem operasi di bawah arahan *Google*, bertanggung jawab langsung atas kemajuan teknologi Android yang sangat pesat. Sistem operasi ini awalnya dibuat oleh *Android Inc.*, sebuah bisnis di *Silicon Valley*. Salah satu manfaat kolaborasi *Open Handset Alliance (OHA)* adalah penyediaan perangkat lunak lengkap yang mencakup sistem operasi utama, *middleware*, dan aplikasi.

4. Android Studio

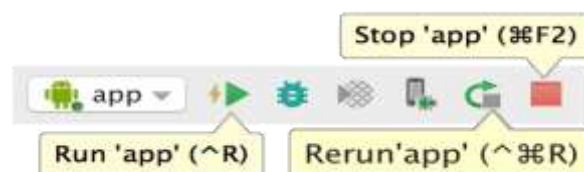


Gambar 2.7 Logo Android
(Sumber : google.com)

Google meluncurkan Android Studio, lingkungan pengembangan komprehensif baru untuk OS Android. Program ini dibuat sebagai pengganti *Eclipse*, IDE yang paling banyak digunakan untuk mengembangkan aplikasi Android. Struktur proyek dan sebagian besar file di direktori SDK ditampilkan saat proyek baru diluncurkan di Android Studio. Proses pengembangan kini lebih *fleksibel* dengan transisi Android Studio ke sistem manajemen berbasis *Gradle* (Nurhidayati et al., 2021).

Pemilihan Android Studio didasarkan pada berbagai fitur yang mendukung kemudahan penggunaan, terutama bagi programmer pemula yang ingin mempelajari pengembangan aplikasi Android lebih lanjut. Meskipun aplikasi ini cenderung membutuhkan banyak RAM pada perangkat PC, Android Studio tetap menawarkan sejumlah keunggulan lain yang membuatnya menarik untuk digunakan.

a. Instant RUN



Gambar 2.8 Instant Run
(Sumber : google.com)

Fitur *Instant Run* memungkinkan program berjalan lebih cepat dengan menghindari kebutuhan untuk mengompilasi ketika perubahan kode dilakukan, APK dapat dibuat ulang atau diterapkan kembali, sehingga mempercepat pengembangan.

b. *Intelligent Code Editor*



Gambar 2.9 *Intelligent Code Editor*
(Sumber : google.com)

Android Studio dilengkapi dengan *Intelligent Code Editor* yang mempermudah analisis kode serta memberikan rekomendasi melalui fitur *auto-complete*. Saat kita mengetik, editor ini secara otomatis menyarankan kelas yang tersedia, sesuai dengan diinstal dan memungkinkan kita untuk menambahkan kode yang sesuai dengan menggunakan kunci TAB. Berkat fitur-fitur ini, proses pengembangan aplikasi menjadi lebih cepat, sehingga meningkatkan produktivitas para pengembang.

c. Sistem Versi Yang Fleksibel

Otomatisasi versi, manajemen ketergantungan, dan alat konfigurasi versi yang dapat disesuaikan semuanya ditawarkan oleh Android Studio sesuai kebutuhan. Pengguna dapat mengatur proyek untuk menyertakan pustaka lokal maupun yang *dihosting* secara online, serta menentukan varian versi dengan kode

yang berbeda. Selain itu, pengguna juga dapat mengatur dan menginstal *library* untuk mempermudah proses pengembangan aplikasi Android. Fitur ini mencerminkan *fleksibilitas* yang ditawarkan oleh Android Studio.

d. Dioptimalkan Untuk Semua Perangkat Android

Android Studio menyediakan platform untuk mengembangkan aplikasi yang *kompatibel* dengan berbagai *gadget* Android, termasuk tablet, Android Wear, dan Android TV, dan Android Auto. Dengan struktur yang terorganisir, Anda dapat membagi proyek menjadi unit fungsional yang dapat dirancang, diuji, dan diperbaiki sesuai kebutuhan.

e. Didesain untuk Tim

Android Studio terintegrasi dengan berbagai sistem kontrol versi terkenal, *Git* dan *Subversion*, misalnya. Kami juga dapat menggunakan layanan *GitHub* langsung dari Android Studio untuk memfasilitasi kolaborasi. Tim dan pembaca dapat berkolaborasi dalam proyek terkait dengan cara ini. Serta melihat perubahan *visual real-time* yang dibuat pada aplikasi Anda dengan Android Studio. Anda juga dapat memeriksa perbedaan yang ada ketika aplikasi dijalankan pada berbagai perangkat Android, termasuk dengan berbagai konfigurasi dan resolusinya secara bersamaan.

5. Java

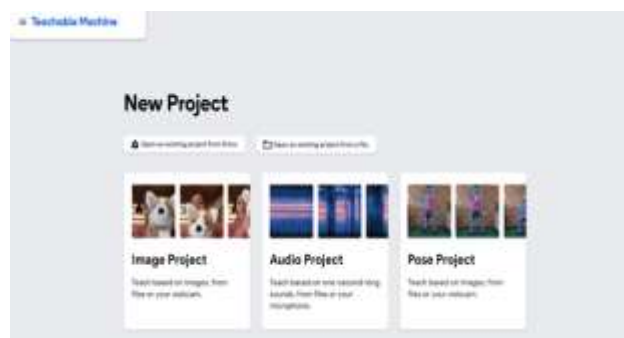


Gambar 2.10 Logo Java
(Sumber : google.com)

Seperti bahasa pemrograman tradisional lainnya, *Java* dapat digunakan untuk membuat berbagai program, termasuk desktop, online, dan aplikasi lainnya. *Java* adalah bahasa pemrograman berorientasi objek (OOP) yang berjalan pada sejumlah sistem operasi berbeda. *Java* diciptakan untuk dijalankan di berbagai sistem operasi *open source*, tidak dibatasi hanya pada satu sistem operasi saja (Nurhidayati dkk., 2021).

Java Virtual Machine dapat digunakan untuk menjalankan aplikasi berbasis *Java* ini, yang telah dikompilasi menjadi *p-code*. Sifat *Java* yang luas dan independensi dari sistem tertentu memungkinkan program ini berjalan di berbagai platform sistem operasi.

6. *Teachable Machine*



Gambar 2.11 Tampilan *Teachable Machine*
(Sumber : google.com)

Aplikasi berbasis web bernama *Teachable Machine* memungkinkan pembuatan model pembelajaran AI dasar. Menambahkan klasifikasi berdasarkan klasifikasi biner adalah cara *Teachable Machine* mengklasifikasikan data. Data pengujian dan pelatihan dapat diekstraksi dari file penyimpanan perangkat atau diunggah menggunakan kamera (Yuyun Hana Natbais dkk, 2023).

Mesin *Teachable Google* menganalisis data menggunakan metode pembelajaran tanpa memerlukan pemrograman *eksplisit*. (Chalifa Chazar dkk, 2022). Fitur *Teachable Machine* mencakup pengolahan gambar, suara, dan gerakan. Tujuan utamanya adalah untuk mempermudah pelajar, guru, desainer, dan profesional lainnya dalam memahami kecerdasan buatan dengan membuat model klasifikasi mereka sendiri.

Proses pelatihan data memerlukan banyak sumber daya dalam bentuk kumpulan data penyakit tanaman. Setiap objek penyakit tanaman harus memiliki beragam citra untuk meningkatkan akurasi klasifikasi. Semakin banyak data objek digunakan, semakin lama waktu yang dibutuhkan untuk menyelesaikan pelatihan dan menghasilkan model klasifikasi yang optimal.

Berikut adalah tahapan-tahapan penggunaan *Teachable Machine*:

- a. Pilih jenis proyek:
 - 1) *Image Project*: Untuk mengklasifikasikan, mendeteksi, atau segmentasi gambar.
 - 2) *Audio Project*: Untuk mengklasifikasikan atau mengenali suara.
 - 3) *Pose Project*: Untuk mengenali pose tubuh manusia.
- b. *Upload* data:
 - 1) *Upload* gambar, audio, atau video yang ingin digunakan.
 - 2) Dapat mengupload data dari komputer Anda, *Google Drive*, atau *URL web*.
- c. Label data:
 - 1) Beri label pada data dengan kategori yang ingin diklasifikasikan, deteksi, atau kenali.

- 2) Dapat menambahkan label baru, mengedit label yang ada, dan menghapus label yang tidak diperlukan.

d. *Train model:*

- 1) Klik tombol "*Train Model*" untuk melatih model.
- 2) *Teachable Machine* akan memproses data dan membuat model *machine learning*.

e. *Test model:*

- 1) Gunakan alat "*Test Model*" untuk mencoba model dengan data baru.
- 2) Dapat mengupload data baru, atau menggunakan *webcam* atau *mikrofon* untuk menguji model secara *real-time*.

f. *Export model:*

- 1) Setelah *performa* model sudah ada, dapat mengekspornya ke berbagai *format*, seperti *TensorFlow.js*, *ONNX*, atau *CoreML*.
- 2) Model yang diekspor dapat digunakan dalam aplikasi *web*, *mobile*, atau *desktop*.

7. *Flowchart*

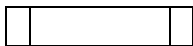
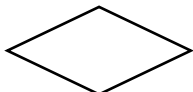
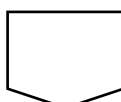
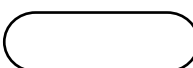
Langkah-langkah program dan urutan prosedur direpresentasikan secara grafis dalam diagram alur. Hal ini sering kali mempengaruhi penyelesaian permasalahan yang memerlukan penyelidikan dan penilaian yang lebih menyeluruh. Diagram alur dapat digunakan untuk menggambarkan proses, tugas manual, atau keduanya. Menurut Ilham Budiman dkk. (2021), diagram alur terdiri dari sejumlah simbol yang bekerja sama untuk membuat suatu kerangka kerja.

Fungsi utama *flowchart* adalah untuk mendokumentasikan, merencanakan, serta mengkomunikasikan proses yang kompleks dalam bentuk diagram agar lebih mudah dipahami oleh orang yang tidak memiliki latar belakang teknis. Selain itu, *flowchart* juga memiliki beberapa fungsi lainnya, seperti:

- a. Memberikan gambaran visual mengenai proses saat proyek dimulai.
- b. Membantu mengelola alur kerja agar lebih mudah dipahami dan terkontrol.
- c. Mendokumentasikan alur kerja sehingga tidak perlu menjelaskan ulang kepada semua pihak.
- d. Menjadi sarana komunikasi untuk semua pihak dalam memahami alur kerja.

Symbol yang digunakan sebagai berikut :

Tabel 2. 1 Symbol dan Fungsi *Flowchart*

Simbol	Fungsi
	Memulai subprogram
	Pernyataan, perbandingan, dan pemilihan data yang menawarkan pilihan untuk tindakan selanjutnya
	Menghubungkan bagian-bagian diagram alur yang berada pada halaman terpisah
	Awal/akhir program
	Arah program
	Prosedur inisialisasi dan penetapan harga awal

Lanjutan Tabel 2.1

Simbol	Fungsi
	Prosedur pengolahan dan perhitungan data
	Proses masukan/keluaran data

8. *Blackbox testing*

Tujuan pengujian perangkat lunak berbasis spesifikasi fungsional adalah untuk menentukan apakah *input*, *output*, dan fungsi perangkat lunak memenuhi persyaratan yang diperlukan tanpa memeriksa desain atau kode program. Karena pendekatan pengujian *Blackbox* hanya memerlukan batas data yang diharapkan batas bawah dan atas ini merupakan prosedur yang mudah. Jumlah *entri* data yang diperiksa, persyaratan *entri* yang harus dipenuhi, dan skenario batas atas dan bawah yang relevan semuanya dapat digunakan untuk memperkirakan jumlah data pengujian. Teknik ini memungkinkan kita untuk menentukan apakah *input* yang tidak terduga masih diterima oleh fungsionalitas perangkat lunak, yang dapat menyebabkan data yang disimpan menjadi tidak *valid* (Wahyu Nur Cholifah, dkk, 2018).

Berikut ini adalah contoh pengujian *blackbox* pada Aplikasi Pengingat Dan Pendataan Kenaikan Golongan Gaji Berbasis Web (Firman,dkk,2021), yaitu sebagai berikut:

Tabel 2.2 Contoh Pengujian *Blackbox*

No	Fungsionalitas	Perintah dan Masukan	Hasil	Status
1	Menu Login	Admin Memasukkan username dan password dengan benar	Sistem mengarahkan admin ke halaman dashboard	Sukses
		Admin Memasukkan username dan password dengan salah	Sistem menampilkan notifikasi dan tampilan form login	Sukses
2	Menu Pegawai	Admin mengklik menu karyawan serta menambah, mengubah atau menghapus data karyawan	Sistem akan menampilkan data pegawai sesuai perintah dan masukan	Sukses
3	Menu Penggajian	Admin mengklik menu penggajian serta menambah, mengubah atau menghapus data gaji	Sistem akan menampilkan data penggajian sesuai perintah dan masukan	Sukses
4	Menu Laporan	Admin mengklik menu laporan	Sistem akan menampilkan data laporan sesuai ada ada	Sukses

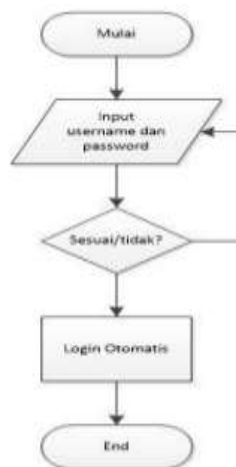
Setiap menu yang ada saat ini dilakukan pengujian aplikasi dengan teknik *black box*. Dalam tes ini, mengklik halaman login, memasukkan nama pengguna dan kata sandi yang cocok dengan database, lalu mencoba memasukkan yang salah. Hasil pengujian menunjukkan bahwa setiap fitur dan fungsi program berfungsi dengan baik dan sah.

9. *Whitebox testing*

Proses pengujian organisasi internal perangkat lunak, desain, dan kode program terkait dikenal sebagai pengujian "kotak putih". Teknik ini dapat mendeteksi kelemahan implementasi perangkat lunak. Penguji kotak putih harus menguasai kode sumber perangkat lunak yang mereka uji.

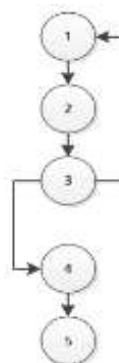
Berikut ini adalah contoh pengujian *white box* pada Aplikasi Peningat Dan Pendataan Kenaikan Golongan Gaji Berbasis Web (Firman,dkk,2021), yaitu sebagai berikut:

a. Proses Login



Gambar 2.12 Proses Login

Hasil dari prosedur login pengujian ini menunjukkan bahwa fungsi aplikasi beroperasi sesuai dengan logika saat ini. Sistem akan langsung menerima dan membawa *administrator ke dashboard* jika pengguna memasukkan login dan password yang benar. Namun, sistem akan mengingatkan pengguna dan meminta mereka mengubah data jika tidak akurat.



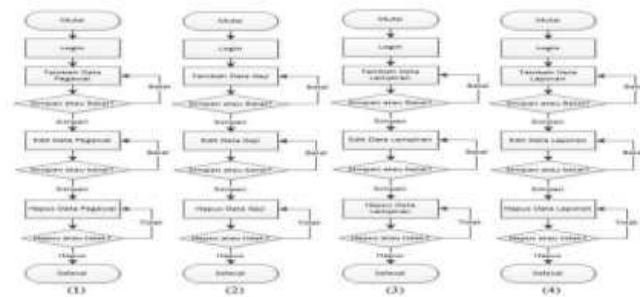
Gambar 2.13 Flowchart Node Login

Lima (N) node dan lima (E) edge membentuk diagram alur siklomatik dari node login. Untuk Path maka

$$V(G) = E - N + 2 = 5 - 5 + 2 = 2$$

Jadi, ada dua path.

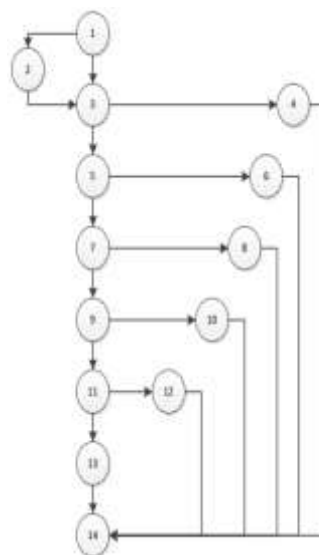
b. Proses Pengolahan Data



Gambar 2.14 Proses Pengelolaan Data Pegawai, Gaji, Lampiran, dan Laporan

Logika pengelolaan data pegawai, pengelolaan data gaji, pengelolaan data lampiran, dan pengelolaan data laporan diuji, dan ternyata semuanya berjalan lancar sesuai dengan alur yang ada.

c. Perhitungan Kompleksits Cyclomatic



Gambar 2.15 Flowchart Note

Gunakan rumus berikut untuk menentukan *kompleksitas siklomatik*:

$$(G) = E = N + 2 = 19 - 14 + 2 = 7$$

Detail: E adalah jumlah tautan atau busur.

N adalah jumlah node.

Untuk jalur *independent* dari *flowchart* mempunyai 7 path:

Path 1: 1-2-3-5-7-9--11-13-14

Path 5: 1-3-5-7-9-10-14

Path 2: 1-3-4-14

Path 6: 1-3-5-7-9--11-12-14

Path 3: 1-3-5-6-14

Path 7: 1-3-5-7-9--11-13-14

Path 4: 1-3-5-7-8-14

Tabel 2.3 Rekapitulasi Hasil *Whitebox Testing*

Flowchart	Cyclomatic Complexity	Region (R)	Independent Path
1. Login	2	2	2
2. Data Pegawai	7	7	7
3. Gaji	7	7	7
4. Lampiran	7	7	7
5. Laporan	7	7	7
Jumlah	30	30	30

Hasil pengujian pada tabel di atas menunjukkan bahwa alur logika sistem aplikasi pengingat kenaikan kelas dan gaji berbasis web sudah benar dan dapat diterima karena angka Cyclomatic Complexity = 30, Region = 30, dan Independent Path = 30 semuanya merupakan sama.

10. UML (*Unified Modelling Language*)

Perancangan sistem berorientasi objek dapat dilakukan dengan bantuan UML (*Unified Modeling Language*). Secara filosofis, kehadiran UML terinspirasi oleh konsep pemodelan berorientasi objek (OO), yang menganggap sistem sebagai entitas yang terdiri dari objek-objek, yang kemudian digambarkan atau dicatat

menggunakan simbol-simbol yang jelas dan spesifik. Proses ini bersifat standar dan independen.

Tujuan utama diagram UML adalah untuk memfasilitasi komunikasi antara tim pengembang proyek, menyelidiki kemungkinan desain dan memverifikasi kode program atau arsitektur perangkat lunak. UML terbagi menjadi tiga kategori utama: diagram struktur, diagram perilaku, dan diagram interaksi, dimana masing-masing kategori mencakup diagram yang menggambarkan arsitektur sistem dan saling terhubung satu sama lain. Adapun daftar simbol UML yaitu :

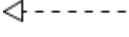
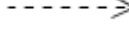
Tabel 2.4 *Symbol Use Case Diagram*

No	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menjelaskan rentang peran yang dapat dimainkan pengguna dalam interaksi <i>Use Case</i> .
2		<i>Dependency</i>	Perubahan pada suatu unsur yang berdiri sendiri akan berdampak pada unsur yang bergantung pada komponen yang tidak berdiri sendiri.
3		<i>Generalization</i>	Hubungan dimana perilaku dan struktur data objek induk (<i>Ancestor</i>) diubah oleh objek anak (<i>Descendent</i>).
4		<i>Include</i>	Sebutkan sumbernya. Nyatakan kasus penggunaan dengan jelas.
5		<i>Extend</i>	Menunjukkan bahwa perilaku <i>use case</i> sumber pada saat tertentu diperluas oleh <i>use case</i> target.
6		<i>Association</i>	Yang menghubungkan dua benda menjadi satu.
7		<i>System</i>	Mengidentifikasi paket yang memberikan pandangan terbatas pada sistem.
8		<i>Use Case</i>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor

Lanjutan tabel 2.4

No	GAMBAR	NAMA	KETERANGAN
9		<i>Collaboration</i>	Kemampuan sistem untuk memfasilitasi pertukaran pesan antar aktor atau unit
10		<i>Note</i>	Komponen fisik yang mewakili sumber daya komputasi dan hadir saat program dijalankan.

Tabel 2.5 Symbol Class Diagram

No.	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Suatu hubungan dimana perilaku dan struktur data objek di atas objek induk (leluhur) dimiliki bersama oleh objek turunannya.
2		<i>Nary Association</i>	Mencoba menjauhi asosiasi dengan lebih dari dua item.
3		<i>Class</i>	Sekelompok objek-objek berbagi atribut yang mempunyai sifat dan fungsi serupa.
4		<i>Collaboration</i>	Sekelompok item yang memiliki karakteristik dan kemampuan serupa.
5		<i>Realization</i>	Tindakan yang sebenarnya dilakukan suatu objek.
6		<i>Dependency</i>	Suatu hubungan di mana elemen-elemen yang bergantung pada elemen-elemen yang tidak independen akan terpengaruh ketika elemen independen berubah
7		<i>Association</i>	Apa yang menghubungkan satu hal dengan hal lainnya

Tabel 2.6 Symbol Sequence Diagram

No	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Antarmuka yang berkomunikasi satu sama lain disebut objek entitas.
2		<i>Message</i>	Spesifikasi komunikasi antar item yang menyimpan data tentang peristiwa yang terjadi
3		<i>Message</i>	Spesifikasi komunikasi yang mencakup rincian aktivitas yang terjadi antar objek

Tabel 2.7 *Symbol State Chart Diagram*

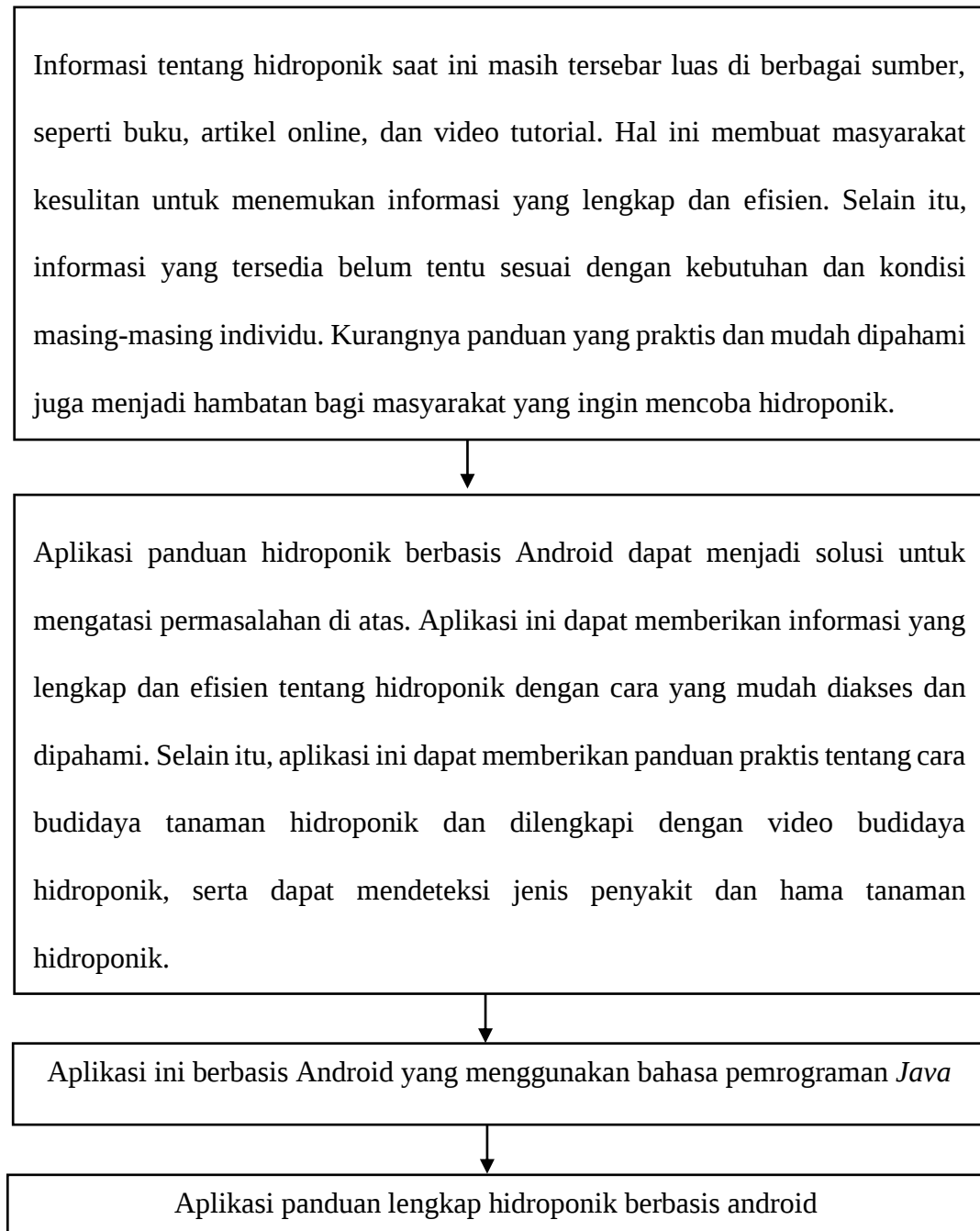
No	GAMBAR	NAMA	KETERANGAN
1		<i>State</i>	Atribut dan nilai tautan yang dimiliki suatu objek pada saat tertentu.
2		<i>Initial Pseudo State</i>	Bagaimana sesuatu diciptakan atau dimulai
3		<i>Final State</i>	Bagaimana sesuatu diciptakan dan dimusnahkan
4		<i>Transition</i>	Suatu kejadian yang mengubah satu atau lebih nilai atribut suatu objek untuk memulai keadaan objek
5		<i>Association</i>	Apa hubungan antara dua objek satu dengan yang satunya
6		<i>Node</i>	Komponen fisik yang berfungsi sebagai sumber daya komputasi dan hadir ketika suatu program dijalankan

Tabel 2.8 *Symbol Activity Diagram*

No.	GAMBAR	NAMA	KETERANGAN
1		<i>Actifity</i>	Mendemonstrasikan bagaimana setiap kelas antarmuka berkomunikasi dengan yang lain.
2		<i>Action</i>	Kondisi sistem yang menunjukkan bagaimana suatu aktivitas dilakukan
3		<i>Initial Node</i>	Bagaimana sesuatu diciptakan atau dimulai.
4		<i>Actifity Final Node</i>	Bagaimana sesuatu terbentuk dan dimusnahkan
5		<i>Fork Node</i>	Satu aliran yang pada akhirnya terpecah menjadi beberapa aliran

C. Kerangka Pikir

Untuk Memahami alur peneltian diatas, diuraikan ke dalam kerangka berpikir yang akan disajikan dalam bentuk diagram ini:



BAB III

METODE PENELITIAN

A. Jenis Penelitian

Jenis penelitian ini adalah penelitian kualitatif, dengan metode penelitian ini digunakan untuk menganalisis kebutuhan pengguna, merancang dan mengembangkan aplikasi dan pengalaman pengguna terkait dengan penggunaan aplikasi panduan lengkap hidroponik berbasis android.

B. Waktu Penelitian

Waktu penelitian ini berlangsung selama $\pm$ 3 bulan.

Tabel 3. 1 Jadwal Pelaksanaan Penelitian

No.	Uraian Kegiatan	Bulan 2024		
		Mei	Juni	Juli
1	Perencanaan			
2	Analisis dan Desain			
3	Implementasi			
4	Pengujian			

C. Alat dan Bahan Penelitian

Berikut alat dan bahan yang digunakan selama penelitian:

1. Alat

a. Spesifikasi Laptop :

- 1). Lenovo Intel CORE 3
- 2). *Processor* Gen Intel(R) Core(TM) i3-1115G4
- 3). RAM 8,00 GB
- 4). HDD 256 GB

5). Windows 11

- b. Android Studio
- c. Bahasa Pemrograman *Java*
- d. *Tools Teachable machine*

2. Bahan

- a. Jurnal
- b. Buku Panduan hidroponik
- c. Gambar Hama dan penyakit hidroponik
- d. Video budidaya hidroponik

D. Metode Pengumpulan Data

1. Secara tidak langsung

Pemanfaatan pendekatan tidak langsung ini memerlukan pengumpulan data dan pengetahuan yang relevan, misalnya dengan membaca buku referensi hidroponik atau artikel relevan dari media online.

2. Secara Langsung

Metode langsung melibatkan pengumpulan data atau informasi yang relevan melalui penyebaran kuesioner.

E. Tahapan Penelitian

Berikut tahapan dalam melakukan penelitian ini, sebagai berikut:

1. Perencanaan:

- a. Mengumpulkan data-data maupun informasi yang terkait dengan panduan hidroponik,

- b. Menganalisis aplikasi hidroponik yang sudah ada untuk mengetahui fitur-fitur yang dibutuhkan.

2. Analisis dan Desain:

- a. Merancang *flowchart* untuk menggambarkan alur kerja aplikasi.
- b. Merancang *wireframe* dan *mockup* untuk menggambarkan tampilan aplikasi antarmuka pengguna (UI).

3. Implementasi

Pada titik ini, hasil program yang telah dijelaskan dan dirancang dipraktikkan, mempersiapkan sistem untuk pengujian tambahan.

4. Pengujian Sistem

- a. Menguji aplikasi untuk memastikan bahwa aplikasi berfungsi dengan baik.
- b. Melakukan pengujian *usability* untuk memastikan aplikasi mudah digunakan oleh pengguna.

F. Metode Pengujian

Ada dua metode pengujian ini, yaitu *blackbox testing* dan *whitebox testing*:

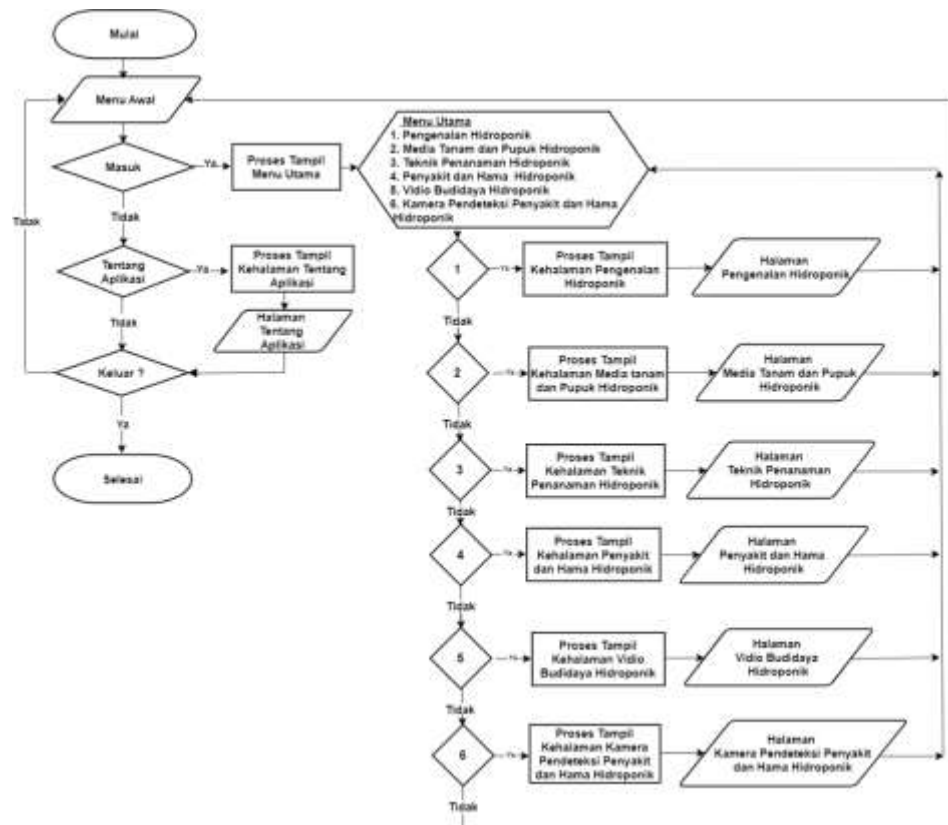
1. *Blackbox testing*

Blackbox testing digunakan untuk menguji program yang sedang berjalan apakah aplikasi panduan hidroponik yang dibuat sudah berhasil atau tidak.

2. *Whitebox testing*

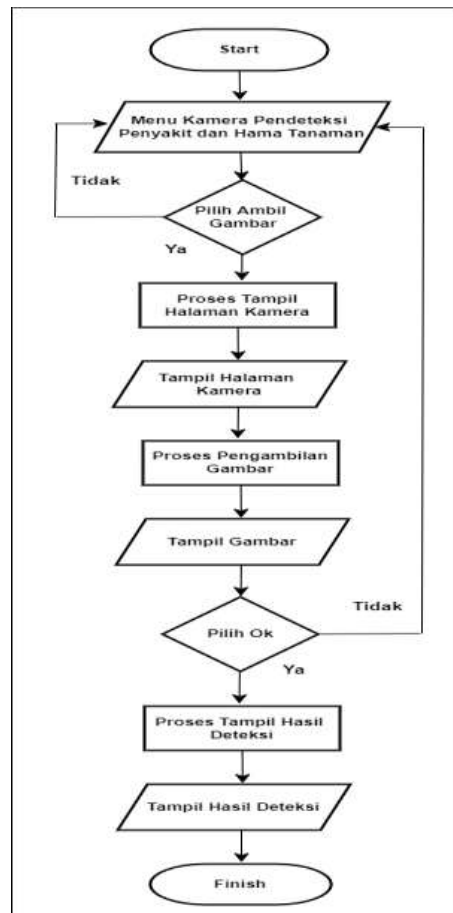
Whitebox testing digunakan untuk mengetahui struktur dan alur pada aplikasi panduan hidroponik apakah sudah sesuai dengan ketentuan kemudian menguji aplikasi tersebut untuk memperoleh hasil.

G. Flowchart



Gambar 3.1 Flowchart Aplikasi Hidroponik

Pada gambar 3.1, dimulai dari menu pertama, dilanjutkan dengan entri dan tampilan aplikasi. Jika memilih ya, Anda akan dibawa ke halaman menu utama dengan enam pilihan menu: pengenalan hidroponik, video budidaya hidroponik, kamera pendeteksi penyakit dan hama hidroponik, teknik hidroponik, penyakit dan hama hidroponik, serta media tanam dan pupuk. Anda kemudian akan dibawa kembali ke menu awal jika Anda tidak memilih salah satunya. Memilih menu aplikasi kemudian akan membawa Anda ke layar menu aplikasi; pilih ya untuk menutup aplikasi.



Gambar 3.2 Flowchart Menu Pendeteksi Hama dan Penyakit

Pada gambar 3.2 menjelaskan mengenai alur pendeteksi jenis-jenis penyakit dan hama hidroponik. Sistem ini akan dijalankan, pertama kali *user* melakukan pengambilan gambar hama atau penyakit pada tanaman bayam dengan mengambil satu gambar menggunakan kamera yang akan diklasifikasi oleh sistem, kemudian input gambar yang dipilih sistem akan merespon gambar yang diupload *user*, kemudian proses untuk mengenali dan training gambar yang diambil sesuai gambar *testing* untuk jenis penyakitnya. Setelah itu akan tampil hasil proses deteksinya berupa jenis penyakit atau hama tanaman dan cara pengendaliannya.

H. Desain Sistem

1. Desain sistem yang berjalan



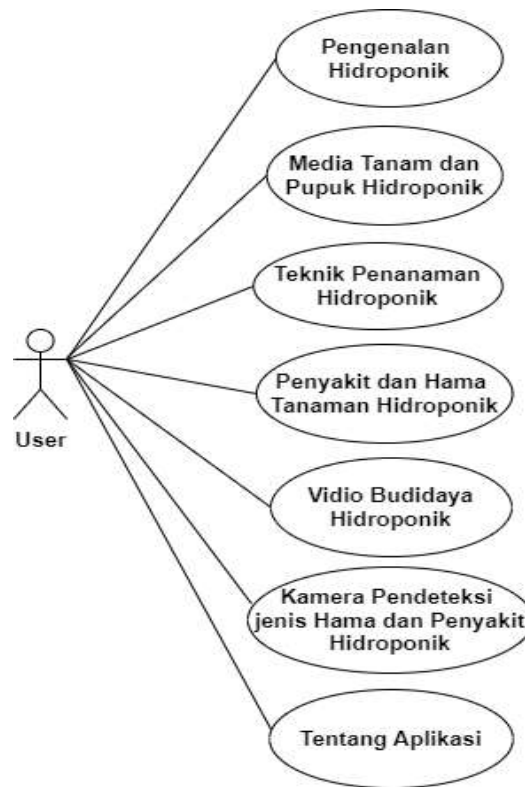
Gambar 3. 3 Desain Sistem Yang Berjalan

Pada gambar 3.3 sistem yang berjalan pada saat ini, *user* mengikuti beberapa cara untuk mengetahui tata cara budidaya hidroponik yaitu sosialisasi budidaya hidroponik, berbagi pengalaman terhadap petani lain, mencari informasi diinternet, dan membaca buku panduan hidroponik.

Tabel 3.2 Penjelasan Desain Sistem Yang Berjalan

Nama <i>use case</i>	Deskripsi
Berbagi pengalaman ke petani lain	<i>Use case</i> ini menjelaskan bahwa user berbagi pengalaman ke petani lain mengenai budidaya hidroponik
Mencari informasi diinternet	<i>Use case</i> ini menjelaskan bahwa user mencari informasi diinternet
Buku Panduan	<i>Use case</i> ini menjelaskan bahwa user menggunakan buku panduan budidaya hidroponik untuk mencari informasi

2. Desain sistem yang diusulkan



Gambar 3. 4 Desain Sistem Yang Diusulkan

Pada gambar 3.4 *user* masuk ke aplikasi tanpa login terlebih dahulu, kemudian akan muncul tampilan seperti, pengenalan hidroponik, Media tanam dan pupuk hidroponik, teknik penanaman hidroponik, penyakit dan hama tanaman hidroponik, vidio budidaya hidroponik, kamera pendeteksi jenis hama dan penyakit hidroponik, dan tentang aplikasi.

Tabel 3.3 Penjelasan Desain Sistem Yang Diusulkan

Nama <i>use case</i>	Deskripsi
Pengenalan hidroponik	<i>Use case</i> ini menjelaskan tentang pengenalan hidroponik yang berisi tentang definisi hidroponik dan keunggulan dan kelemahan hidroponik

Lanjutan Tabel 3.3

Nama <i>use case</i>	Deskripsi
Media tanam dan pupuk hidroponik	<i>Use case</i> ini menjelaskan tentang media tanam hidroponik, jenis-jenis pupuk hidroponik, dan tips pelarutan nutrisi hidroponik
Teknik penanaman hidroponik	<i>Use case</i> ini menjelaskan tentang jenis-jenis teknik penanaman hidroponik dan tata cara penanaman secara hidroponik
Penyakit dan Hama tanaman hidroponik	<i>Use case</i> ini menjelaskan tentang jenis-jenis penyakit dan hama tanaman hidroponik, contoh kasus penyakit dan hama, dan cara pengendalian hama dan penyakit hidroponik
Vidio Budidaya Hidroponik	<i>Use case</i> ini menjelaskan tentang vidio yang berisi pengenalan hidroponik, tata cara penanaman sampai panen, dan tata cara merakit media tanam hidroponik
Kamera pendeteksi penyakit dan hama hidroponik	<i>Use case</i> ini menjelaskan tentang user dapat mendeteksi jenis penyakit dan hama tanaman hidroponik dengan menggunakan kamera atau diambil dari galeri foto

BAB IV

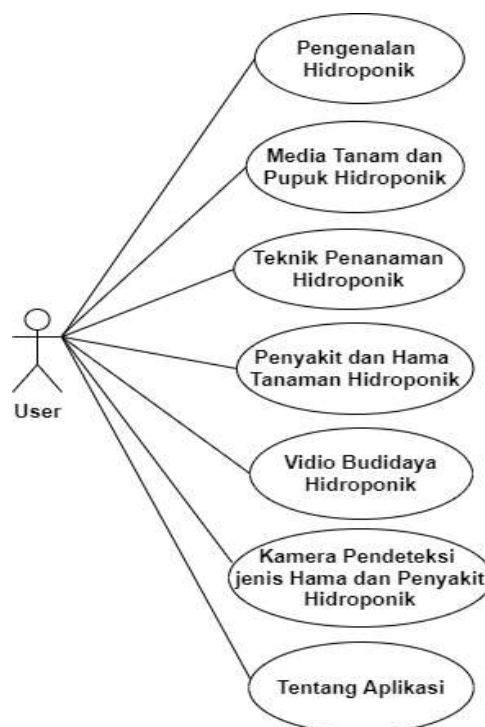
HASIL DAN PEMBAHASAN

A. Analisi Aliran Data Diagram UML

Dengan tujuan memberikan informasi tahapan dan alur aplikasi, *Use Case Diagram*, *Activity Diagram*, dan *Sequence Diagram* digunakan untuk mengorientasikan objek sistem melalui analisis aliran data.

1. *Use Case Diagram*

Diagram ini membantu menggambarkan fungsionalitas utama dari suatu sistem berdasarkan perspektif pengguna .



Gambar 4.1 *Use Case diagram*

Pada gambar 4.1 *user* masuk ke aplikasi tanpa login terlebih dahulu, kemudian akan muncul tampilan seperti, pengenalan hidroponik, Media tanam dan pupuk hidroponik, teknik penanaman hidroponik, penyakit dan hama tanaman hidroponik, vidio budidaya hidroponik, kamera pendeteksi jenis hama dan penyakit hidroponik, dan tentang aplikasi.

Tabel 4.1 Penjelasan Desain Sistem Yang Diusulkan

Nama use case	Deskripsi
Pengenalan hidroponik	Menjelaskan tentang pengenalan hidroponik yang berisi tentang definisi hidroponik dan keunggulan dan kelemahan hidroponik
Media tanam dan pupuk hidroponik	Menjelaskan tentang media tanam hidroponik, jenis-jenis pupuk hidroponik, dan tips pelarutan nutrisi hidroponik
Teknik penanaman hidroponik	Menjelaskan tentang jenis-jenis teknik penanaman hidroponik dan tata cara penanaman secara hidroponik
Penyakit dan Hama tanaman hidroponik	Menjelaskan tentang jenis-jenis penyakit dan hama tanaman hidroponik, contoh kasus penyakit dan hama, dan cara pengendalian hama dan penyakit hidroponik
Vidio Budidaya Hidroponik	Menjelaskan tentang vidio yang berisi pengenalan hidroponik, tata cara penanaman sampai panen, dan tata cara merakit media tanam hidroponik

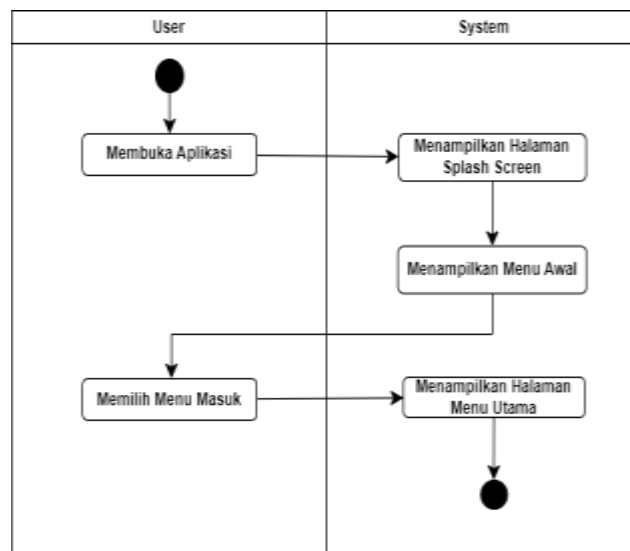
Lanjutan Tabel 4.1

Nama <i>use case</i>	Deskripsi
Kamera pendeteksi penyakit dan hama hidroponik	Menjelaskan tentang user dapat mendeteksi jenis penyakit dan hama tanaman hidroponik dengan menggunakan kamera atau diambil dari galeri foto

2. Activity Diagram

Activity diagram digunakan untuk merepresentasikan alur kerja atau aliran kontrol dari suatu proses bisnis maupun *use case*. Diagram ini menggambarkan berbagai aktivitas yang berlangsung dalam sistem beserta urutan pelaksanaannya.

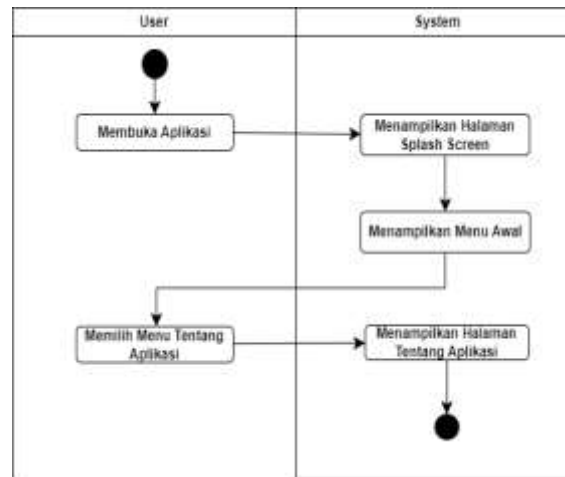
a. Activity diagram tampilan menu utama



Gambar 4. 2 Activity Diagram Tampilan Menu Utama

Pada Gambar 4.2 menjelaskan tentang tampilan menu utama, dimana user membuka aplikasi maka system akan menampilkan halaman splash screen dan ketampilan menu awal, setelah itu memilih menu masuk kemudian system akan menampilkan menu utama.

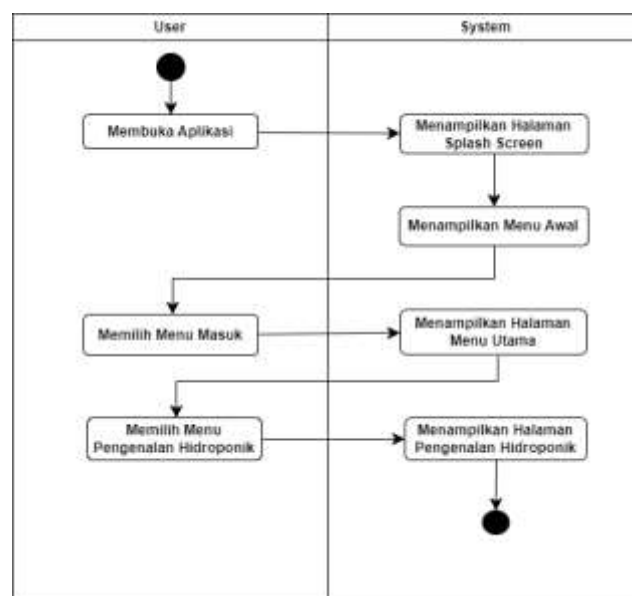
b. *Activity diagram* Tampilan Tentang Aplikasi



Gambar 4. 3 *Activity Diagram* Tampilan Tentang Aplikasi

Pada Gambar 4.3 menjelaskan bagaimana menu aplikasi ditampilkan. Saat pengguna menjalankan aplikasi, sistem akan menampilkan halaman splash screen dan menu awal aplikasi. Setelah mereka memilih menu tentang aplikasi, sistem akan menampilkan halaman tentang aplikasi.

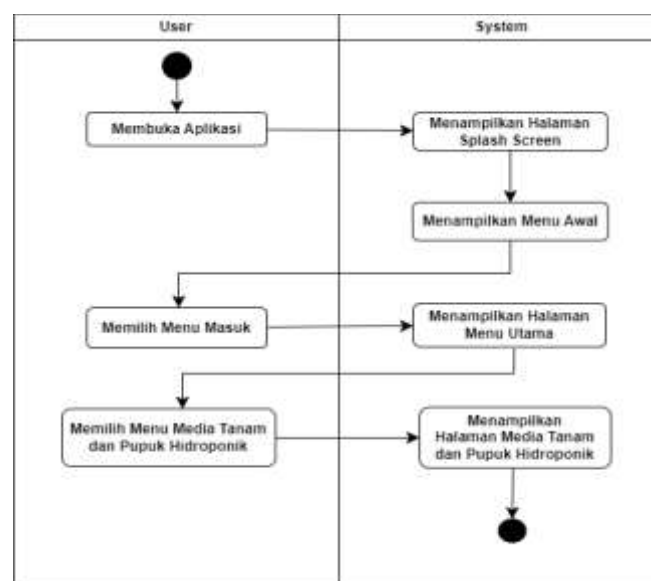
c. *Activity diagram* Tampilan Pengenalan Hidroponik



Gambar 4. 4 *Activity Diagram* Tampilan Pengenalan Hidroponik

Pada Gambar 4.4 menjelaskan tentang tampilan pengenalan hidroponik, dimana user membuka aplikasi maka system akan menampilkan halaman splash screen dan ketampilan menu awal, setelah itu memilih menu masuk kemudian system akan menampilkan menu utama. Pada menu utama user memilih menu pengenalan hidroponik, kemudian system akan menampilkan halaman pengenalan hidroponik.

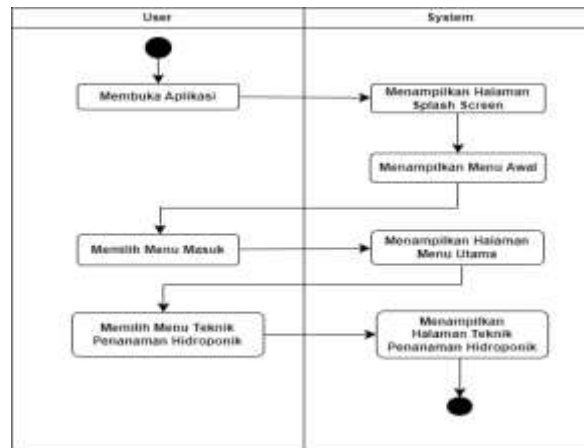
d. *Activity diagram* Tampilan Media Tanam dan Pupuk Hidroponik



Gambar 4. 5 *Activity Diagram* Tampilan Media Tanam dan Pupuk Hidroponik

Pada gambar 4.5 diatas menjelaskan tentang tampilan media tanam dan pupuk hidroponik, dimana user membuka aplikasi maka system akan menampilkan halaman splash screen dan ketampilan menu awal, setelah itu memilih menu masuk, kemudian system akan menampilkan menu utama. Pada menu utama user memilih menu media tanam dan pupuk hidroponik, kemudian system akan menampilkan halaman media tanam dan pupuk hidroponik.

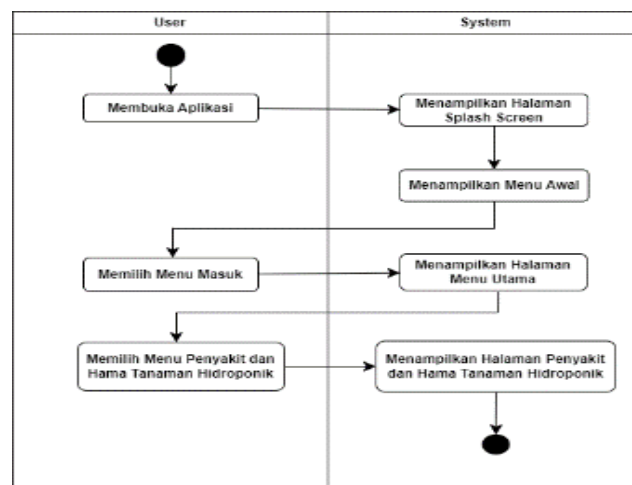
e. *Activity diagram* Tampilan Teknik Penanaman Hidroponik



Gambar 4. 6 *Activity Diagram* Tampilan Teknik Penanaman Hidroponik

Pada Gambar 4.6 menjelaskan tentang tampilan teknik penanaman hidroponik, dimana user membuka aplikasi maka system akan menampilkan halaman *splash screen* dan ketampilan menu awal, setelah itu memilih menu masuk, kemudian system akan menampilkan menu utama. Pada menu utama user memilih menu teknik penanaman hidroponik, kemudian system akan menampilkan halaman teknik penanaman hidroponik.

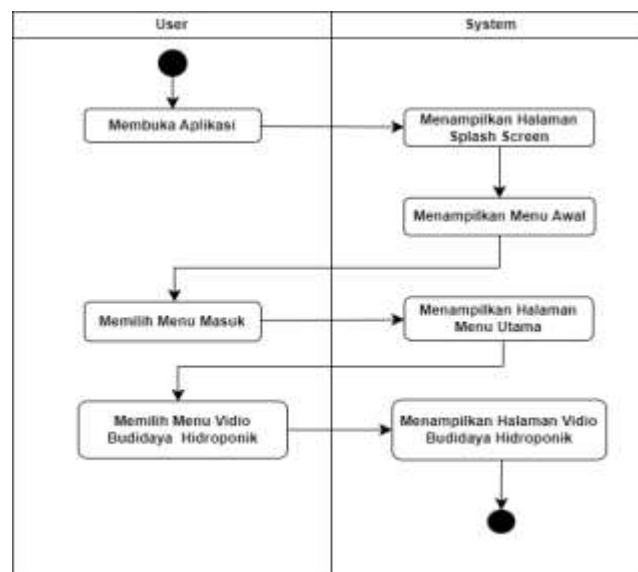
g. *Activity diagram* Tampilan Penyakit dan Hama Hidroponik



Gambar 4. 7 *Activity Diagram* Tampilan Penyakit dan Hama Hidroponik

Pada Gambar 4.7 menjelaskan tentang tampilan penyakit dan hama tanaman hidroponik, dimana user membuka aplikasi maka system akan menampilkan halaman splash screen dan ketampilan menu awal, setelah itu memilih menu masuk, kemudian system akan menampilkan menu utama. Pada menu utama user memilih menu penyakit dan hama tanaman hidroponik, kemudian system akan menampilkan halaman penyakit dan hama tanaman hidroponik.

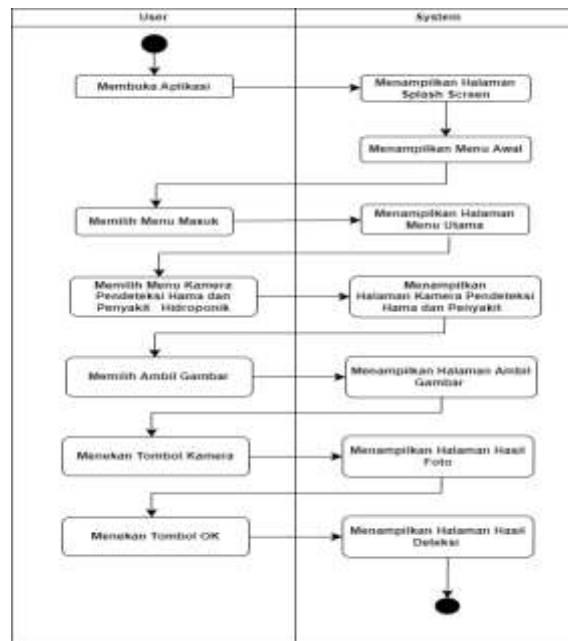
h. Activity Diagram Tampilan Vidio Budidaya Hidroponik



Gambar 4. 8 Activity Diagram Tampilan Vidio Budidaya Hidroponik

Pada Gambar 4.8 menjelaskan tentang tampilan video budidaya hidroponik, dimana user membuka aplikasi maka system akan menampilkan halaman splash screen dan ketampilan menu awal, setelah itu memilih menu masuk, kemudian system akan menampilkan menu utama. Pada menu utama user memilih menu video budidaya hidroponik, kemudian system akan menampilkan halaman video budidaya hidroponik.

i. *Activity diagram* Tampilan Kamera Pendeteksi Penyakit dan Hama Penyakit Hidroponik



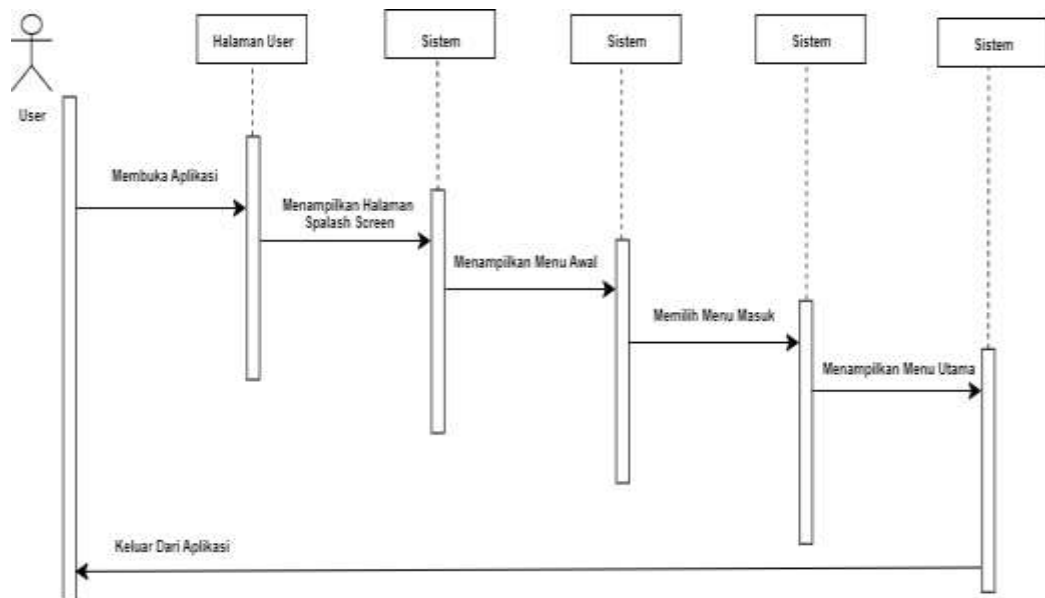
Gambar 4. 9 *Activity Diagram* Tampilan Kamera Pendeteksi Penyakit dan Hama Penyakit Hidroponik

Pada Gambar 4.9 menjelaskan tentang tampilan kamera pendeteksi penyakit dan hama hidroponik, dimana user membuka aplikasi maka system akan menampilkan halaman *splashscreen* dan ketampilan menu awal, setelah itu memilih menu masuk, kemudian sistem akan menampilkan menu utama. Pada menu utama user memilih menu kamera pendeteksi penyakit dan hama hidroponik, kemudian system akan menampilkan halaman kamera pendeteksi penyakit dan hama hidroponik. Setelah itu, user memilih tombol ambil gambar, maka system akan menampilkan halaman ambil gambar dan pada halaman tersebut user memilih tombol kamera untuk mengambil gambar dan system akan menampilkan hasil gambar tersebut lalu user memilih tombol ok, system akan menampilkan hasil deteksi sesuai dengan gambar yang di foto tersebut.

3. Sequence Diagram

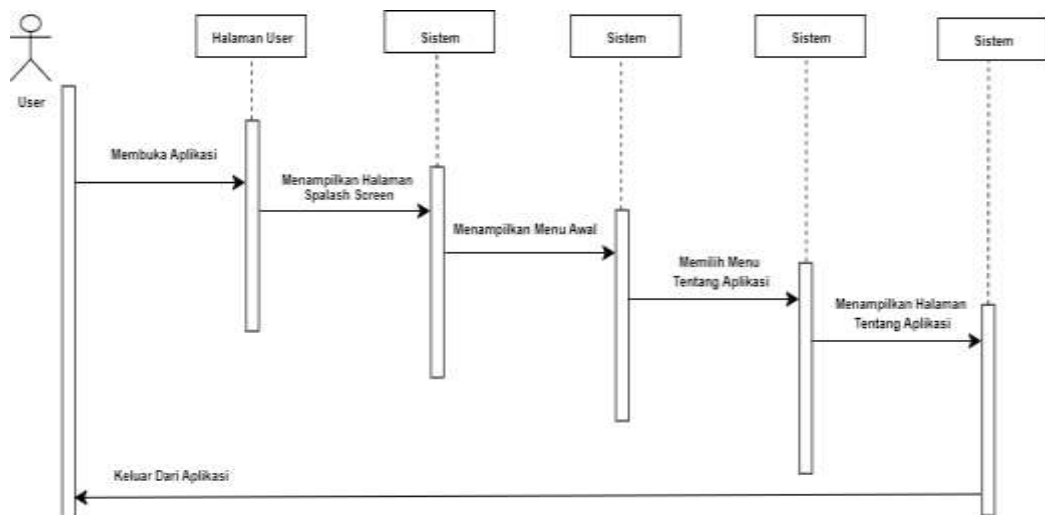
Diagram interaksi yang menunjukkan bagaimana suatu tindakan dilakukan, pesan apa yang disampaikan, dan kapan tindakan tersebut dilakukan disebut diagram urutan.

a. Sequence Diagram Tampilan Menu Utama



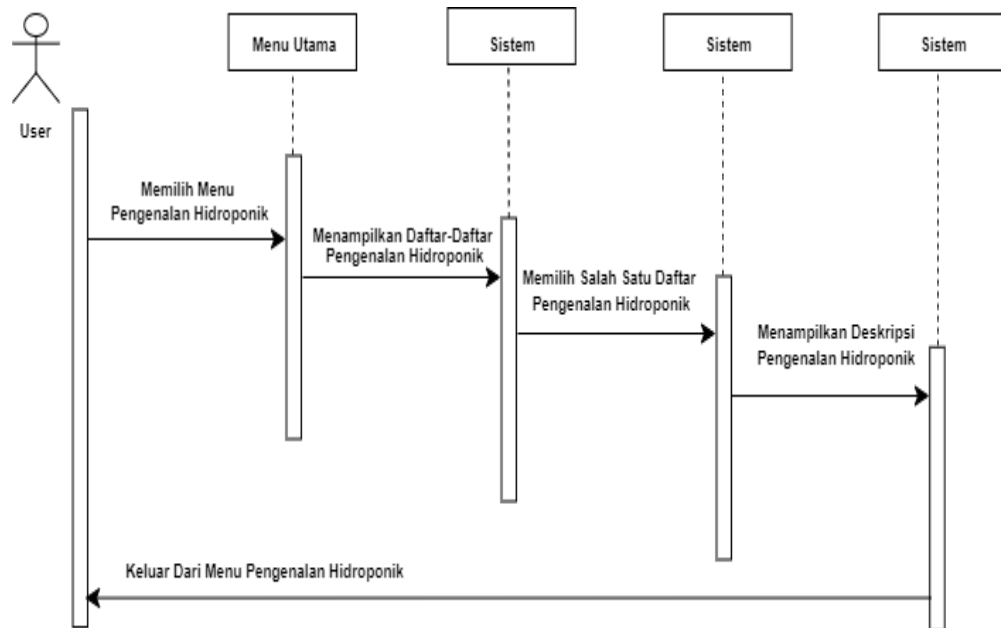
Gambar 4. 10 Sequence Diagram Tampilan Menu Utama

b. Sequence Diagram Tampilan Tentang Aplikasi



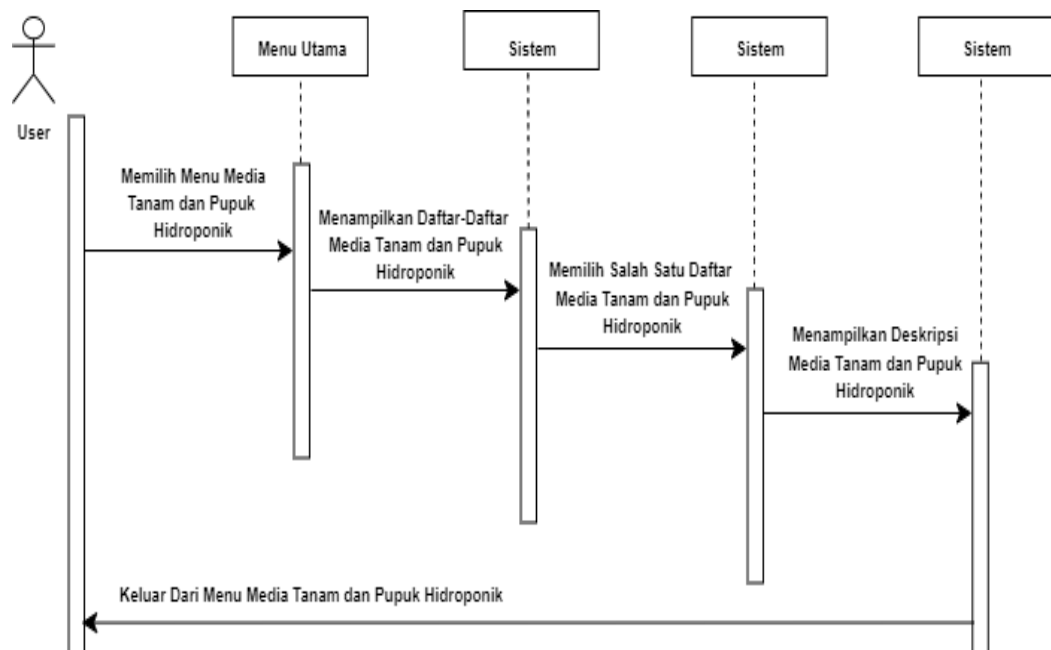
Gambar 4. 11 Sequence Diagram Tampilan Tentang Aplikasi

c. *Sequence Diagram* Tampilan Pengenalan Hidroponik



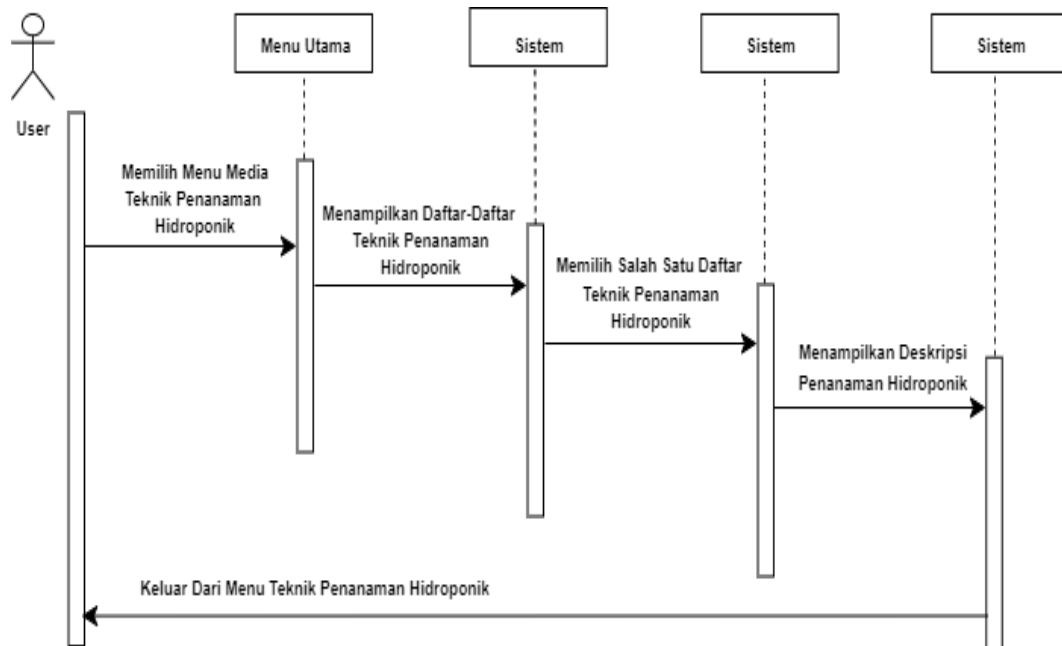
Gambar 4. 12 Sequence Diagram Tampilan Pengenalan Hidroponik

d. *Sequence Diagram* Tampilan Media Tanam Dan Pupuk Hidroponik



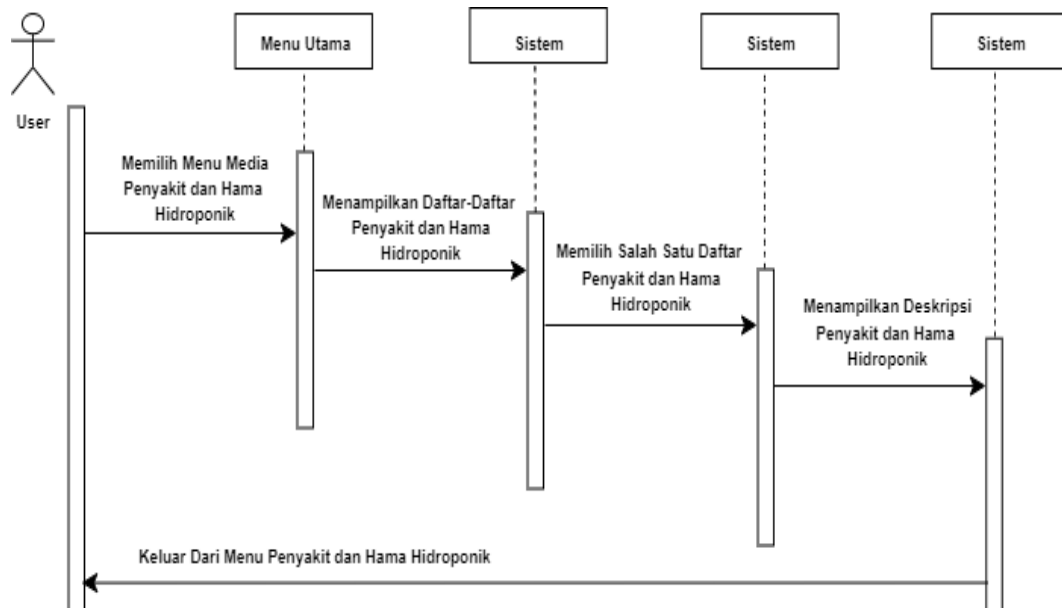
Gambar 4. 13 Sequence Diagram Tampilan Media Tanam Dan Pupuk Hidroponik

e. *Sequence Diagram* Tampilan Teknik Penanaman Hidroponik



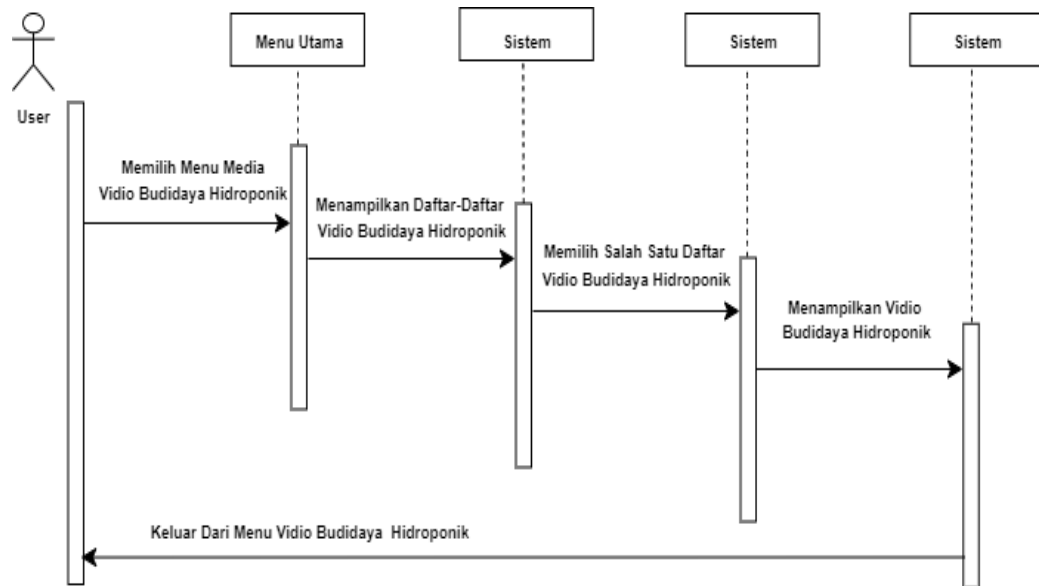
Gambar 4. 14 *Sequence Diagram* Tampilan Teknik Penanaman Hidroponik

f. *Sequence Diagram* Tampilan Penyakit Dan Hama Tanaman Hidroponik



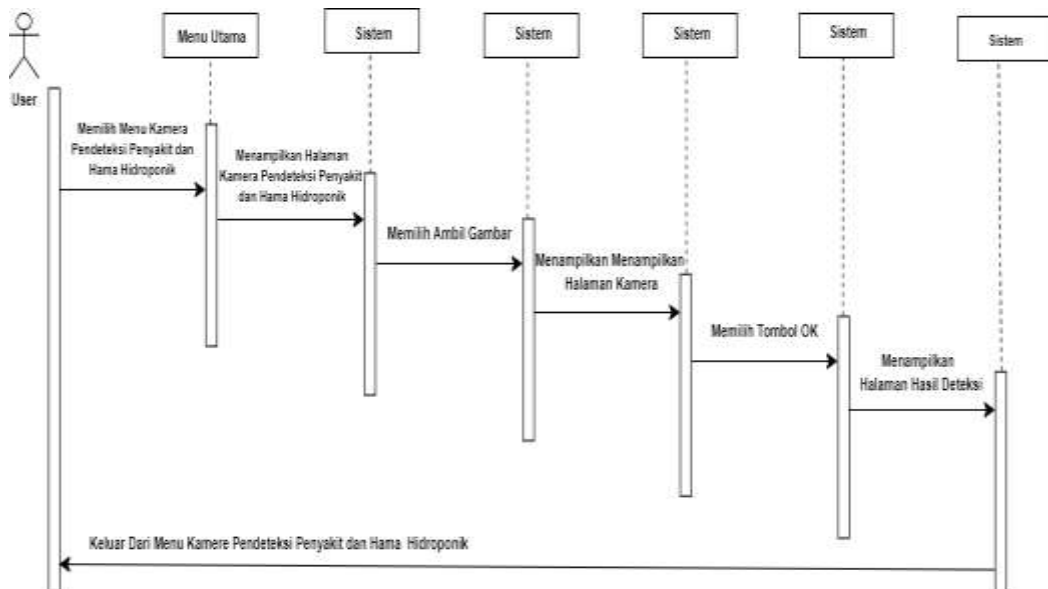
Gambar 4. 15 *Sequence Diagram* Tampilan Penyakit Dan Hama Tanaman Hidroponik

g. *Sequence Diagram* Tampilan Video Budidaya Hidroponik



Gambar 4. 16 *Sequence Diagram* Tampilan Video Budidaya Hidroponik

h. *Sequence Diagram* Tampilan Kamera Pendeteksi Jenis Hidroponik Penyakit Dan Hama Hidroponik



Gambar 4. 17 *Sequence Diagram* Tampilan Kamera Pendeteksi Jenis Hidroponik Penyakit Dan Hama Hidroponik

B. Detail Sistem

1. Tampilan Menu Awal

Pada gambar 4.18 merupakan tampilan awal pada menu aplikasi yang terdapat fitur masuk dan tentang aplikasi.



Gambar 4.18 Tampilan Awal Aplikasi

```
<de.hdodenhof.circleimageview.CircleImageView
    android:id="@+id/profil"
    android:layout_width="228dp"
    android:layout_height="228dp"
    android:layout_marginEnd="188dp"
    android:layout_marginTop="58dp"
    android:layout_centerHorizontal="true"
    android:src="@drawable/logo"
    tools:ignore="MissingConstraints"
    tools:layout_editor_absoluteY="49dp" />
```

Gambar 4.19 Script Layout Logo Aplikasi

```
<Button
    android:id="@+id/masuk"
    android:layout_width="188dp"
    android:layout_height="88dp"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="348dp"
    android:backgroundTint="@color/green_688"
    android:fontFamily="sans-serif-black"
    android:text="Masuk"
    android:textColor="@color/black"
    android:textSize="12dp" />
```

Gambar 4.20 Script Layout Tombol Masuk Menu Utama

```
masuk.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent Bukasatu =new Intent(getApplicationContext(), Menu_utama.class);
        startActivity(Bukasatu);
    }
});
```

Gambar 4.21 Script Java Tombol Masuk

```
<Button
    android:id="@+id/tentang_aplikasi"
    android:layout_width="180dp"
    android:layout_height="80dp"
    android:text="Tentang Aplikasi"
    android:backgroundTint="@color/green_600"
    android:layout_marginTop="430dp"
    android:fontFamily="sans-serif-black"
    android:textSize="12dp"
    android:textColor="@color/black"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"/>
```

Gambar 4.22 Script Layout Tombol Masuk Menu Tentang Aplikasi

```
tentang_aplikasi.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) { Bukadua(); }
});

// usage
public void Bukadua () {
    Intent Bukadua =new Intent(getApplicationContext(), Tentang_aplikasi.class);
    startActivity(Bukadua);
}
```

Gambar 4.23 Script Java Tombol Tentang Aplikasi

2. Tampilan Menu Utama

Pada gambar 4.24 merupakan tampilan menu utama aplikasi yang memiliki 6 fitur yaitu, pengenalan hidroponik, media tanam dan pupuk hidroponik, teknik penanaman hidroponik, penyakit dan hama tanaman hidroponik, vidio budidaya hidroponik, dan kamera pendeteksi jenis hama dan penyakit tanaman hidroponik.



Gambar 4.24 Tampilan Menu Utama Aplikasi

```
<ImageButton
    android:id="@+id/btnBack"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:background="@color/green_600"
    android:layout_marginTop="2dp"
    android:src="@drawable/baseline_arrow_back_ios_new_24"
    android:contentDescription="Back" />
```

Gambar 4.25 Script Layout Tombol Kembali

```
btnBack.setOnClickListener(v -> finish());
```

Gambar 4.26 Script Java Tombol Kembali

```
<androidx.core.widget.NestedScrollView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="60dp"
    tools:ignore="MissingConstraints">
    <GridLayout
        android:id="@+id/grid_layout"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginHorizontal="0dp"
        android:alignmentMode="alignMargins"
        android:columnCount="2"
        android:columnOrderPreserved="false"
        android:rowCount="3"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent">
        <androidx.cardview.widget.CardView
            android:id="@+id/btn1"
            android:layout_width="150dp"
            android:layout_height="100dp"
            android:layout_margin="20dp"
            app:cardCornerRadius="8dp">
```

Gambar 4.27 Script Layout Tombol-Tombol Menu Utama

```

    CardView cardView = findViewById(R.id.btn1);
    cardView.setCardBackgroundColor(getResources().getColor(R.color.my_card_color));

    CardView cardView1 = findViewById(R.id.btn2);
    cardView1.setCardBackgroundColor(getResources().getColor(R.color.my_card_color));

    CardView cardView2 = findViewById(R.id.btn3);
    cardView2.setCardBackgroundColor(getResources().getColor(R.color.my_card_color));

    CardView cardView3 = findViewById(R.id.btn4);
    cardView3.setCardBackgroundColor(getResources().getColor(R.color.my_card_color));

    CardView cardView4 = findViewById(R.id.btn5);
    cardView4.setCardBackgroundColor(getResources().getColor(R.color.my_card_color));

    CardView cardView5 = findViewById(R.id.btn6);
    cardView5.setCardBackgroundColor(getResources().getColor(R.color.my_card_color));

```

Gambar 4.28 Script Java Tombol-Tombol Menu Utama

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:orientation="vertical">

    <ImageView
        android:layout_width="128dp"
        android:layout_height="100dp"
        android:layout_marginBottom="20dp"
        android:src="@drawable/hidroponik" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:fontFamily="sans-serif-condensed"
        android:text="Pengertian
        Hidroponik"
        android:textColor="@color/black"
        android:textSize="15dp" />
</LinearLayout>

```

Gambar 4.29 Script Layout Gambar dan Teks

3. Tampilan Menu Tentang Aplikasi

Pada gambar 4.30 merupakan tampilan halaman tentang aplikasi yang berkaitan dengan penggunaan aplikasi hidroponik.



Gambar 4.30 Tampilan Halaman Tentang Aplikasi

```
<ImageButton
    android:id="@+id/btnBack"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:background="@color/green_600"
    android:layout_marginTop="2dp"
    android:src="@drawable/baseline_arrow_back_ios_new_24"
    android:contentDescription="Back"/>
```

Gambar 4.31 Script Layout Tombol Kembali

```
btnBack.setOnClickListener(v -> finish());
```

Gambar 4.32 Script Java Tombol Kembali

```
<ImageButton
    android:id="@+id/btnBack"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:layout_marginTop="2dp"
    android:background="@color/green_600"
    android:layout_marginLeft="10dp"
    android:src="@drawable/baseline_arrow_back_ios_new_24"
    android:contentDescription="Back"/>
<ImageButton
    android:id="@+id/btnSpeak"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:background="@color/green_600"
    android:layout_marginTop="2dp"
    android:layout_marginLeft="265dp"
    android:src="@drawable/baseline_volume_up_24"
    android:contentDescription="Speak"/>
```

Gambar 4.33 Script Layout Tombol Suara dan Berhenti

```

textToSpeech = new TextToSpeech( context: this, new TextToSpeech.OnInitListener() {
    @Override
    public void onInit(int status) {
        if (status == TextToSpeech.SUCCESS) {
            int result = textToSpeech.setLanguage(Locale.getDefault());
            if (result == TextToSpeech.LANG_MISSING_DATA ||
                result == TextToSpeech.LANG_NOT_SUPPORTED) {
            }
        } else {
        }
    }
});

buttonSpeak.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) { speakText(); }
});

buttonStop.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) { stopSpeaking(); }
});
}

private void speakText() {
    String text = textView.getText().toString();
    String text1 = textView1.getText().toString();
    String text2 = textView2.getText().toString();
    String text3 = textView3.getText().toString();
    String text4 = textView4.getText().toString();
    String text5 = textView5.getText().toString();
    String text6 = textView6.getText().toString();
    textToSpeech.speak( text + " " + text1 + " " + text2 + " " + text3 + " " + text4 + " " +
        text5 + " " + text6, TextToSpeech.QUEUE_FLUSH, params: null, utteranceId: null);
}

```

Gambar 4.34 Script Java Tombol Suara dan Berhenti

```

<TextView
    android:id="@+id/textView1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:fontFamily="sans-serif-condensed"
    android:paddingBottom="8dp"
    android:text="Fitur Utama:"
    android:textColor="@color/white"
    android:textSize="18dp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/textView2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textColor="@color/white"
    android:paddingBottom="18dp"
    android:justificationMode="inter_word"
    android:fontFamily="sans-serif-condensed"
    android:textSize="16dp"
    android:text="@string/fitur"/>

```

Gambar 4.35 Script Layout Tampilan Teks

4. Tampilan Menu Pengenalan Hidroponik

Pada gambar 4.36 merupakan tampilan halaman pengenalan hidroponik yang terbagi-bagi menjadi definisi hidroponik dan keunggulan dan kelemahan hidroponik.



Gambar 4.36 Tampilan Halaman Pengenalan Hidroponik

```
<ImageButton
    android:id="@+id/btnBack"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:layout_marginTop="2dp"
    android:background="@color/green_600"
    android:layout_marginLeft="10dp"
    android:src="@drawable/baseline_arrow_back_ios_new_24"
    android:contentDescription="Back"/>
```

Gambar 4.37 Script Layout Tombol Kembali

```
btnBack.setOnClickListener(v -> finish());
```

Gambar 4.38 Script Java Tombol Kembali

```

<Button
    android:id="@+id/btn1"
    android:layout_width="220dp"
    android:layout_height="80dp"
    android:text="Definisi Hidroponik"
    android:textSize="12dp"
    android:fontFamily="sans-serif-black"
    android:textColor="@color/black"
    android:backgroundTint="@color/green_600"
    android:layout_marginTop="250dp"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"/>
<Button
    android:id="@+id/btn2"
    android:layout_width="220dp"
    android:layout_height="80dp"
    android:text="Keunggulan dan Kelemahan Hidroponik "
    android:backgroundTint="@color/green_600"
    android:layout_marginTop="350dp"
    android:fontFamily="sans-serif-black"
    android:textSize="12dp"
    android:textColor="@color/black"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"/>

```

Gambar 4.39 Script Layout Tombol Button Definisi Hidroponik dan Keunggulan Kelemahan Hidroponik

```

btn1 = (Button) findViewById(R.id.btn1);
btn2 = (Button) findViewById(R.id.btn2);

btn1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent Bukasatu =new Intent(getApplicationContext(), Definisi_Hidroponik.class);
        startActivity(Bukasatu);
    }
});
btn2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) { Bukadua(); }
});

1 usage
public void Bukadua (){
    Intent Bukadua =new Intent(getApplicationContext(), Keunggulan_Hidroponik.class);
    startActivity(Bukadua);
}

```

Gambar 4.40 Script Java Rombol Button Definisi Hidroponik dan Keunggulan Kelemahan Hidroponik

5. Tampilan Menu Deskripsi Pengenalan Hidroponik

Pada gambar 4.41 merupakan menu pengenalan hidroponik yang berisi definisi hidroponik dan keunggulan dan kelemahan hidroponik.



Gambar 4.41 Tampilan Halaman Deskripsi Pengenalan Hidroponik

```
<ImageButton
    android:id="@+id/btnBack"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:background="@color/green_600"
    android:layout_marginTop="2dp"
    android:src="@drawable/baseline_arrow_back_ios_new_24"
    android:contentDescription="Back"/>
```

Gambar 4.42 Script Layout Tombol Kembali

```
btnBack.setOnClickListener(v -> finish());
```

Gambar 4.43 Script Java Tombol Kembali

```
<ImageView
    android:id="@+id/soundIcon1"
    android:layout_width="50dp"
    android:layout_height="38dp"
    android:layout_gravity="center"
    android:layout_marginTop="10dp"
    android:layout_marginLeft="260dp"
    android:src="@drawable/baseline_volume_up_24"
    android:contentDescription="Speak"/>
<ImageButton
    android:id="@+id/stopButton"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:layout_marginTop="2dp"
    android:background="@color/green_600"
    android:layout_marginLeft="380dp"
    android:src="@drawable/baseline_stop_circle_24"
    android:contentDescription="Stop"/>
```

Gambar 4.44 Script Layout Tombol Suara dan Berhenti

```

textToSpeech = new TextToSpeech(context, this, status -> {
    if (status != TextToSpeech.ERROR) {
        textToSpeech.setLanguage(Locale.getDefault());
        textToSpeech.setOnUtteranceProgressListener(new UtteranceProgressListener() {
            @Override
            public void onStart(String utteranceId) { isSpeaking = true; }

            @Override
            public void onDone(String utteranceId) { isSpeaking = false; }

            @Override
            public void onError(String utteranceId) {
            }
        });
    }
});

ImageView soundIcon1 = findViewById(R.id.soundIcon1);
soundIcon1.setOnClickListener(v -> {
    String text = ((TextView) findViewById(R.id.textView1)).getText().toString();
    String text1 = ((TextView) findViewById(R.id.textView2)).getText().toString();
    String text2 = ((TextView) findViewById(R.id.textView3)).getText().toString();
    speakText(text + " " + text1 + " " + text2);
});

ImageButton stopButton = findViewById(R.id.stopButton);
stopButton.setOnClickListener(v -> {
    if (textToSpeech != null) {
        textToSpeech.stop();
        isSpeaking = false;
    }
}

```

Gambar 4.45 Script Java Tombol Suara dan Berhenti

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="16dp">
    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:layout_marginBottom="16dp"
        android:layout_marginTop="16dp">

        <!-- Placeholder Image -->
        <ImageView
            android:id="@+id/videoThumbnail"
            android:layout_width="match_parent"
            android:layout_height="200dp"
            android:layout_marginBottom="16dp"
            android:layout_marginTop="16dp"
            android:scaleType="centerCrop"
            android:src="@drawable/awal2" />
    </FrameLayout>
</LinearLayout>

```

```

<ImageView
    android:id="@+id/playButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="30dp"
    android:src="@drawable/baseline_play_circle_24"
    android:contentDescription="Play Video"
    android:padding="8dp"/>

<VideoView
    android:id="@+id/videoView2"
    android:layout_width="wrap_content"
    android:layout_height="200dp"
    android:layout_marginTop="16dp"
    android:layout_marginBottom="16dp"
    android:visibility="gone" />
</FrameLayout>

```

Gambar 4.46 Script Layout Tampilan Vidio

```

VideoView videoView = findViewById(R.id.videoView2);
ImageView placeholder = findViewById(R.id.videoThumbnail);
ImageView playButton = findViewById(R.id.playButton);
MediaController mediaController = new MediaController(context, this);
mediaController.setAnchorView(videoView);
videoView.setMediaController(mediaController);

Uri videoUri = Uri.parse("android.resource://" + getPackageName() + "/" +
    R.raw.definisi_hidroponik);
videoView.setVideoURI(videoUri);
playButton.setOnClickListener(v -> {
    placeholder.setVisibility(View.GONE);
    playButton.setVisibility(View.GONE);
    videoView.setVisibility(View.VISIBLE);
    videoView.start();
});
}

```

Gambar 4.47 Script Java Tampilan Vidio

```

<TextView
    android:id="@+id/textView1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:fontFamily="sans-serif-condensed"
    android:paddingBottom="8dp"
    android:justificationMode="inter_word"
    android:layout_marginTop="16dp"
    android:text="@string/definisi"
    android:textColor="@color/white"
    android:textSize="16dp"/>

<TextView
    android:id="@+id/textView2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:fontFamily="sans-serif-condensed"
    android:text="Singkatnya:"
    android:textStyle="bold"
    android:textColor="@color/white"
    android:textSize="18dp"
    android:paddingBottom="8dp"/>

```

```

<TextView
    android:id="@+id/textView3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textColor="@color/white"
    android:fontFamily="sans-serif-condensed"
    android:textSize="16dp"
    android:justificationMode="inter_word"
    android:text="@string/kesimpulan"
    android:paddingBottom="2dp"/>

```

Gambar 4.48 Script Layout Tampilan Teks

6. Tampilan Menu Media Tanam dan Pupuk Hidroponik

Pada gambar 4.49 merupakan tampilan halaman media tanam dan pupuk hidroponik yang terbagi-bagi menjadi 3 bagian yaitu, jenis-jenis media tanam hidroponik, jenis-jenis pupuk hidroponik, dan tips pembuatan nutrisi hidroponik.



Gambar 4.49 Tampilan Halaman Media Tanam dan Pupuk Hidroponik

```

<ImageButton
    android:id="@+id/kembali_M"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:background="@color/green_600"
    android:layout_marginTop="6dp"
    android:layout_marginHorizontal="6dp"
    android:src="@drawable/baseline_arrow_back_ios_new_24" />

```

Gambar 4.50 Script Layout Tombol Kembali

```

kembali_M.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent Bukasatu =new Intent(getApplicationContext(), Menu_utama.class);
        startActivity(Bukasatu);
    }
});

```

Gambar 4.51 Script Java Tombol Kembali

```

<Button
    android:id="@+id/btnM1"
    android:layout_width="200dp"
    android:layout_height="100dp"
    android:text="Jenis-jenis Media Tanam Hidroponik"
    android:textSize="12dp"
    android:fontFamily="sans-serif-black"
    android:textColor="@color/black"
    android:backgroundTint="@color/green_600"
    android:layout_marginTop="200dp"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"/>
<Button
    android:id="@+id/btnM2"
    android:layout_width="200dp"
    android:layout_height="100dp"
    android:text="Jenis-jenis Pupuk Hidroponik "
    android:backgroundTint="@color/green_600"
    android:layout_marginTop="320dp"
    android:textSize="12dp"
    android:fontFamily="sans-serif-black"
    android:textColor="@color/black"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"/>
<Button
    android:id="@+id/btnM3"
    android:layout_width="200dp"
    android:layout_height="100dp"
    android:text="Tips Pembuatan Nutrisi Hidroponik "
    android:backgroundTint="@color/green_600"
    android:layout_marginTop="440dp"
    android:textSize="12dp"
    android:fontFamily="sans-serif-black"
    android:textColor="@color/black"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"/>

```

Gambar 4.52 Script Layout Tombol Button Jenis-Jenis Pupuk dan Nutrisi Hidroponik

```

    btnM1.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent Bukasatu = new Intent(getApplicationContext(), Media_Hidroponik.class);
            startActivity(Bukasatu);
        }
    });

    btnM2.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent Bukasatu = new Intent(getApplicationContext(), Pupuk_Hidroponik.class);
            startActivity(Bukasatu);
        }
    });

    btnM3.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) { Bukadua(); }
    });

    public void Bukadua () {
        Intent Bukadua = new Intent(getApplicationContext(), Tips_Pupuk.class);
        startActivity(Bukadua);
    }
}

```

Gambar 4.53 Script Java Tombol Button Jenis-Jenis Pupuk dan Nutrisi Hidroponik

7. Tampilan Menu Deskripsi Media Tanam dan Pupuk Hidroponik

Pada gambar 4.54 merupakan menu media tanam dan pupuk hidroponik yang salah satu menunya , yaitu jenis-jenis media tanah hidroponik.



Gambar 4.54 Tampilan Halaman Deskripsi Media Tanam dan Pupuk Hidroponik

```
<ImageButton
    android:id="@+id/btnBack"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:background="@color/green_600"
    android:layout_marginTop="2dp"
    android:src="@drawable/baseline_arrow_back_ios_new_24"
    android:contentDescription="Back"/>
```

Gambar 4.55 Script Layout Tombol Kembali

```
btnBack.setOnClickListener(v -> finish());
```

Gambar 4.56 Script Java Tombol Kembali

```
<ImageView
    android:id="@+id/soundIcon2"
    android:layout_width="100dp"
    android:layout_height="30dp"
    android:layout_gravity="center"
    android:src="@drawable/baseline_volume_up_24"
    android:contentDescription="Speak"/>

<ImageButton
    android:id="@+id/stopButton"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:background="@color/green_600"
    android:layout_marginTop="2dp"
    android:layout_marginLeft="300dp"
    android:src="@drawable/baseline_stop_circle_24"
    android:contentDescription="Stop"/>
```

Gambar 4.57 Script Layout Tombol Suara Dan Berhenti

```
textToSpeech = new TextToSpeech(context, this, status -> {
    if (status != TextToSpeech.ERROR) {
        textToSpeech.setLanguage(Locale.getDefault());
        textToSpeech.setOnUtteranceProgressListener(new UtteranceProgressListener() {
            @Override
            public void onStart(String utteranceId) { isSpeaking = true; }

            @Override
            public void onDone(String utteranceId) { isSpeaking = false; }

            @Override
            public void onError(String utteranceId) {
            }
        });
    }
});

ImageView soundIcon2 = findViewById(R.id.soundIcon2);
soundIcon2.setOnClickListener(v -> {
    String text1 = ((TextView) findViewById(R.id.textView1)).getText().toString();
    String text2 = ((TextView) findViewById(R.id.textView2)).getText().toString();
    speakText(text1 + " " + text2);
});

ImageButton stopButton = findViewById(R.id.stopButton);
stopButton.setOnClickListener(v -> {
    if (textToSpeech != null) {
        textToSpeech.stop();
        isSpeaking = false;
    }
});
```

Gambar 4.58 Script Java Tombol Suara Dan Berhenti

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="16dp">
    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:layout_marginBottom="16dp"
        android:layout_marginTop="16dp">
        <ImageView
            android:id="@+id/videoThumbnail"
            android:layout_width="match_parent"
            android:layout_height="200dp"
            android:layout_marginBottom="16dp"
            android:layout_marginTop="16dp"
            android:scaleType="centerCrop"
            android:src="@drawable/awal" />
        <ImageView
            android:id="@+id/playButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginTop="30dp"
            android:src="@drawable/baseline_play_circle_24"
            android:contentDescription="Play Video"
            android:padding="8dp"/>

        <VideoView
            android:id="@+id/videoView"
            android:layout_width="wrap_content"
            android:layout_height="200dp"
            android:layout_marginTop="16dp"
            android:layout_marginBottom="16dp"
            android:visibility="gone" />
    </FrameLayout>
</LinearLayout>

```

Gambar 4.59 Script Layout Tampilan Vidio

```

VideoView videoView = findViewById(R.id.videoView);
ImageView placeholder = findViewById(R.id.videoThumbnail);
ImageView playButton = findViewById(R.id.playButton);
MediaController mediaController = new MediaController( context, this);
mediaController.setAnchorView(videoView);
videoView.setMediaController(mediaController);

Uri videoUri = Uri.parse( UriString "android.resource://" + getPackageName() + "/" +
    R.raw.jenis_media);
videoView.setVideoURI(videoUri);
playButton.setOnClickListener(v -> {
    placeholder.setVisibility(View.GONE);
    playButton.setVisibility(View.GONE);
    videoView.setVisibility(View.VISIBLE);
    videoView.start();
});
}

```

Gambar 4.60 Script Layout Tampilan Vidio

```
<ImageView
    android:layout_width="match_parent"
    android:layout_height="150dp"
    android:src="@drawable/rockwool"/>
```

Gambar 4.61 Script Layout Tampilan Gambar

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:gravity="center_vertical">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:fontFamily="sans-serif-condensed"
        android:paddingBottom="8dp"
        android:gravity="center"
        android:text="Jenis-jenis Media Tanam Hidroponik"
        android:textColor="@color/white"
        android:textSize="18dp"
        android:textStyle="bold"/>
    <TextView
        android:id="@+id/textView1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:fontFamily="sans-serif-condensed"
        android:paddingBottom="8dp"
        android:text="1. Rockwool"
        android:textColor="@color/white"
        android:textSize="18dp"
        android:textStyle="bold" />
    <TextView
        android:id="@+id/textView2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textColor="@color/white"
        android:paddingBottom="18dp"
        android:layout_marginTop="18dp"
        android:justificationMode="inter_word"
        android:fontFamily="sans-serif-condensed"
        android:textSize="16dp"
        android:text="@string/rockwool"/>
```

Gambar 4.62 Script Layout Tampilan Teks

8. Tampilan Menu Teknik Penanaman Hidroponik

Pada gambar dibawah 4.63 merupakan tampilan halaman teknik penanaman hidroponik yang terbagi-bagi menjadi 2 bagian yaitu, Jenis-jenis teknik penanaman dan tata cara penanaman sampai panen hidroponik.



Gambar 4.63 Tampilan Halaman Teknik Penanaman Hidroponik

```
<ImageButton
    android:id="@+id/kembali_M"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:background="@color/green_600"
    android:layout_marginTop="6dp"
    android:layout_marginHorizontal="6dp"
    android:src="@drawable/baseline_arrow_back_ios_new_24" />
```

Gambar 4.64 Script Layout Tombol Kembali

```
kembali_M.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent Bukasatu = new Intent(getApplicationContext(), Menu_utama.class);
        startActivity(Bukasatu);
    }
});
```

Gambar 4.65 Script Java Tombol Kembali

```
<Button
    android:id="@+id/btnT1"
    android:layout_width="250dp"
    android:layout_height="90dp"
    android:text="Jenis-jenis Teknik Penanaman Hidroponik"
    android:textSize="12dp"
    android:fontFamily="sans-serif-black"
    android:textColor="@color/black"
    android:backgroundTint="@color/green_600"
    android:layout_marginTop="250dp"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"/>
<Button
    android:id="@+id/btnT2"
    android:layout_width="250dp"
    android:layout_height="90dp"
    android:text="Tata cara Penanaman Secara Hidroponik"
    android:backgroundTint="@color/green_600"
    android:layout_marginTop="350dp"
    android:textSize="12dp"
    android:fontFamily="sans-serif-black"
    android:textColor="@color/black"
```

Gambar 4.66 Script Layout Tombol Button Teknik Penanaman dan Tata Cara Penanaman Hidroponik

```

    btnT1 = (Button) findViewById(R.id.btnT1);
    btnT2 = (Button) findViewById(R.id.btnT2);
    btnT1.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent Bukasatu = new Intent(getApplicationContext(), Teknik_Hidroponik.class);
            startActivity(Bukasatu);
        }
    });
    btnT2.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) { Bukadua(); }
    });
}
1 usage:
public void Bukadua () {
    Intent Bukadua = new Intent(getApplicationContext(), Penanaman_Hidroponik.class);
    startActivity(Bukadua);
}
}

```

Gambar 4.67 Script Java Tombol Button Teknik Penanaman dan Tata Cara Penanaman Hidroponik

9. Tampilan Menu Deskripsi Teknik Penanaman Hidroponik

Pada gambar 4.68 merupakan teknik penanaman hidroponik yang salah satu menunya , yaitu jenis-jenis teknik hidroponik.



Gambar 4.68 Tampilan Halaman Deskripsi Teknik Penanaman Hidroponik

```

<ImageButton
    android:id="@+id/btnBack"
    android:layout_width="56dp"
    android:layout_height="56dp"
    android:background="@color/green_688"
    android:layout_marginTop="2dp"
    android:src="@drawable/baseline_arrow_back_ios_new_24"
    android:contentDescription="Back"/>

```

Gambar 4.69 Script Layout Tombol Kembali

```
btnBack.setOnClickListener(v -> finish());
```

Gambar 4.70 Script Java Tombol Kembali

```
<ImageView
    android:id="@+id/soundIcon2"
    android:layout_width="100dp"
    android:layout_height="30dp"
    android:layout_gravity="center"
    android:src="@drawable/baseline_volume_up_24"
    android:contentDescription="Speak"/>

<ImageButton
    android:id="@+id/stopButton"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:background="@color/green_600"
    android:layout_marginTop="2dp"
    android:layout_marginLeft="300dp"
    android:src="@drawable/baseline_stop_circle_24"
    android:contentDescription="Stop"/>
```

Gambar 4.71 Script Layout Tombol Suara dan Berhenti

```
textToSpeech = new TextToSpeech( context: this, status -> {
    if (status != TextToSpeech.ERROR) {
        textToSpeech.setLanguage(Locale.getDefault());
        textToSpeech.setOnUtteranceProgressListener(new UtteranceProgressListener() {
            @Override
            public void onStart(String utteranceId) { isSpeaking = true; }

            @Override
            public void onDone(String utteranceId) { isSpeaking = false; }

            @Override
            public void onError(String utteranceId) {
            }
        });
    }
});

ImageView soundIcon2 = findViewById(R.id.soundIcon2);
soundIcon2.setOnClickListener(v -> {
    String text1 = ((TextView) findViewById(R.id.textView1)).getText().toString();
    String text2 = ((TextView) findViewById(R.id.textView2)).getText().toString();
    speakText(text1 + " " + text2);
});

ImageButton stopButton = findViewById(R.id.stopButton);
stopButton.setOnClickListener(v -> {
    if (textToSpeech != null) {
        textToSpeech.stop();
        isSpeaking = false;
    }
});
```

Gambar 4.72 Script Java Tombol Suara dan Berhenti

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="16dp">
    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:layout_marginBottom="16dp"
        android:layout_marginTop="16dp">
        <ImageView
            android:id="@+id/videoThumbnail"
            android:layout_width="match_parent"
            android:layout_height="200dp"
            android:layout_marginBottom="16dp"
            android:layout_marginTop="16dp"
            android:scaleType="centerCrop"
            android:src="@drawable/awal" />

        <ImageView
            android:id="@+id/playButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginTop="30dp"
            android:src="@drawable/baseline_play_circle_24"
            android:contentDescription="Play Video"
            android:padding="8dp"/>

        <VideoView
            android:id="@+id/videoView"
            android:layout_width="wrap_content"
            android:layout_height="200dp"
            android:layout_marginTop="16dp"
            android:layout_marginBottom="16dp"
            android:visibility="gone" />
    </FrameLayout>
</LinearLayout>

```

Gambar 4.73 Script Layout Tampilan Vidio

```

VideoView videoView = findViewById(R.id.videoView);
ImageView placeholder = findViewById(R.id.videoThumbnail);
ImageView playButton = findViewById(R.id.playButton);
MediaController mediaController = new MediaController(context, this);
mediaController.setAnchorView(videoView);
videoView.setMediaController(mediaController);

Uri videoUri = Uri.parse("android.resource://" + getPackageName() + "/" +
    R.raw.contoh_penanaman);
videoView.setVideoURI(videoUri);
playButton.setOnClickListener(v -> {
    placeholder.setVisibility(View.GONE);
    playButton.setVisibility(View.GONE);
    videoView.setVisibility(View.VISIBLE);
    videoView.start();
});
}

```

Gambar 4.74 Script Java Tampilan Vidio

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:gravity="center_vertical">
    <TextView
        android:id="@+id/textView4"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:fontFamily="sans-serif-condensed"
        android:paddingBottom="8dp"
        android:text="2. Seledri"
        android:textColor="@color/white"
        android:textSize="18dp"
        android:textStyle="bold" />
    <TextView
        android:id="@+id/textView5"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textColor="@color/white"
        android:layout_marginTop="15dp"
        android:justificationMode="inter_word"
        android:fontFamily="sans-serif-condensed"
        android:textSize="16dp"
        android:text="@string/seledri"/>
    <ImageView
        android:layout_width="310dp"
        android:layout_height="180dp"
        android:layout_gravity="center"
        android:src="@drawable/selada" />

```

Gambar 4. 75 Script Layout Tampilan Teks dan Gambar

10. Tampilan Menu Penyakit dan Hama Hidroponik

Pada gambar 4.75 merupakan tampilan halaman penyakit dan hama hidroponik yang terdiri dari jenis-jenis hama dan penyakit tanaman hidroponik, dan upaya penaggulangannya.



Gambar 4.76 Tampilan Halaman Penyakit dan Hama Hidroponik

```

<ImageButton
    android:id="@+id/kembali_P"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:layout_marginTop="6dp"
    android:background="@color/green_600"
    android:layout_marginHorizontal="6dp"
    android:src="@drawable/baseline_arrow_back_ios_new_24" />

```

Gambar 4. 77 Script Layout Tombol Kembali

```

kembali_P.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent Bukasatu =new Intent(getApplicationContext(), Menu_utama.class);
        startActivity(Bukasatu);
    }
});

```

Gambar 4. 78 Script Java Tombol Kembali

```

<Button
    android:id="@+id/btnP1"
    android:layout_width="250dp"
    android:layout_height="100dp"
    android:text="Jenis-jenis Penyakit dan Hama Hidroponik"
    android:textSize="12dp"
    android:fontFamily="sans-serif-black"
    android:textColor="@color/black"
    android:backgroundTint="@color/green_600"
    android:layout_marginTop="250dp"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"/>
<Button
    android:id="@+id/btnP2"
    android:layout_width="250dp"
    android:layout_height="100dp"
    android:text="Upaya Pengendalian dan Pencegahan Penyakit dan Hama"
    android:backgroundTint="@color/green_600"
    android:layout_marginTop="350dp"
    android:textSize="12dp"
    android:fontFamily="sans-serif-black"

```

Gambar 4.79 Script Layout Tampilan Button Hama Penyakit dan Pengendalian Hama Penyakit Hidroponik

```

    btnP1 = (Button) findViewById(R.id.btnP1);
    btnP2 = (Button) findViewById(R.id.btnP2);
    btnP1.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent Bukasatu =new Intent(getApplicationContext(), HamaPenyakit_Hidroponik.class);
            startActivity(Bukasatu);
        }
    });
    btnP2.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) { Bukadua(); }
    });
}

//image
public void Bukadua (){
    Intent Bukadua =new Intent(getApplicationContext(), Pengendalian_PenyakitHama.class);
    startActivity(Bukadua);
}
}

```

Gambar 4.80 Script Layout Tampilan Button Hama Penyakit dan Pengendalian Hama Penyakit Hidroponik

11. Tampilan Menu Deskripsi Penyakit dan Hama Hidroponik

Pada gambar 4.81 merupakan penyakit dan hama hidroponik yang salah satu menunya , yaitu jenis-jenis penyakit dan hama hidroponik.



Gambar 4.81 Tampilan Halaman Deskripsi Penyakit dan Hama Hidroponik



Gambar 4.82 Script Layout Tombol Kembali



Gambar 4.83 Script Java Tombol Kembali



Gambar 4.84 Script Layout Tombol Suara dan Berhenti

```

textToSpeech = new TextToSpeech(this, status -> {
    if (status != TextToSpeech.ERROR) {
        textToSpeech.setLanguage(Locale.getDefault());
        textToSpeech.setOnUtteranceProgressListener(new UtteranceProgressListener() {
            @Override
            public void onStart(String utteranceId) { isSpeaking = true; }

            @Override
            public void onDone(String utteranceId) { isSpeaking = false; }

            @Override
            public void onError(String utteranceId) {
            }
        });
    }
});

ImageView soundIcon2 = findViewById(R.id.soundIcon2);
soundIcon2.setOnClickListener(v -> {
    String text1 = ((TextView) findViewById(R.id.textView1)).getText().toString();
    String text2 = ((TextView) findViewById(R.id.textView2)).getText().toString();
    speakText(text1 + " " + text2);
});

ImageButton stopButton = findViewById(R.id.stopButton);
stopButton.setOnClickListener(v -> {
    if (textToSpeech != null) {
        textToSpeech.stop();
        isSpeaking = false;
    }
});

```

Gambar 4.85 Script Java Tombol Suara dan Berhenti

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="16dp">
    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="288dp"
        android:layout_marginBottom="16dp"
        android:layout_marginTop="16dp">
        <ImageView
            android:id="@+id/videoThumbnail"
            android:layout_width="match_parent"
            android:layout_height="288dp"
            android:layout_marginBottom="16dp"
            android:layout_marginTop="16dp"
            android:scaleType="centerCrop"
            android:src="@drawable/awal6" />
        <ImageView
            android:id="@+id/playButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginTop="45dp"
            android:src="@drawable/baseline_play_circle_24"
            android:contentDescription="Play Video"
            android:padding="8dp" />

        <VideoView
            android:id="@+id/videoView5"
            android:layout_width="wrap_content"
            android:layout_height="200dp"
            android:layout_marginTop="16dp"
            android:layout_marginBottom="16dp"
            android:visibility="gone" />
    </FrameLayout>

```

Gambar 4.86 Script Layout Tampilan Vidio

```

        VideoView videoView = findViewById(R.id.videoView5);
        ImageView placeholder = findViewById(R.id.videoThumbnail);
        ImageView playButton = findViewById(R.id.playButton);
        MediaController mediaController = new MediaController(context, this);
        mediaController.setAnchorView(videoView);
        videoView.setMediaController(mediaController);

        Uri videoUri = Uri.parse("android.resource://" + getPackageName() + "/" +
            R.raw.hama_penyakit_hidroponik);
        videoView.setVideoURI(videoUri);
        playButton.setOnClickListener(v -> {
            placeholder.setVisibility(View.GONE);
            playButton.setVisibility(View.GONE);
            videoView.setVisibility(View.VISIBLE);
            videoView.start();
        });
    }
}

```

Gambar 4.87 Script Java Tampilan Vidio

```

<TextView
    android:id="@+id/textView4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:fontFamily="sans-serif-condensed"
    android:paddingBottom="8dp"
    android:layout_marginTop="18dp"
    android:text="2. Thrips"
    android:textColor="@color/white"
    android:textSize="18dp"
    android:textStyle="bold" />
<TextView
    android:id="@+id/textView5"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textColor="@color/white"
    android:justificationMode="inter_word"
    android:paddingBottom="18dp"
    android:fontFamily="sans-serif-condensed"
    android:textSize="16dp"
    android:text="@string/thrips" />
<ImageView
    android:layout_width="300dp"
    android:layout_height="150dp"
    android:layout_marginHorizontal="20dp"
    android:src="@drawable/tungau" />

```

Gambar 4.88 Script Layout Tampilan Teks dan Gambar

12. Tampilan Menu Vidio Budidaya Hidroponik

Pada gambar 4.89 merupakan tampilan halaman vidio budidaya hidroponik yang terdiri dari vidio tata cara merakit media tanam hidroponik dan vidio tata cara penanaman dan panen hidroponik.



Gambar 4.89 Tampilan Halaman Vidio Budidaya Hidroponik

```
<ImageButton
    android:id="@+id/kembali_V"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:layout_marginTop="0dp"
    android:background="@color/green_688"
    android:layout_marginHorizontal="0dp"
    android:src="@drawable/baseline_arrow_back_ios_new_24" />
```

Gambar 4.90 Script Layout Tombol Kembali

```
kembali_V.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent Bukasatu =new Intent(getApplicationContext(), Menu_utama.class);
        startActivity(Bukasatu);
    }
});
```

Gambar 4.91 Script Java Tombol Kembali

```
<Button
    android:id="@+id/btnV1"
    android:layout_width="300dp"
    android:layout_height="100dp"
    android:text="Vidio Tutorial Merakit Teknik-Teknik Hidroponik"
    android:textSize="12dp"
    android:fontFamily="sans-serif-black"
    android:textColor="@color/black"
    android:backgroundTint="@color/green_688"
    android:layout_marginTop="250dp"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"/>

<Button
    android:id="@+id/btnV2"
    android:layout_width="300dp"
    android:layout_height="100dp"
    android:text="Vidio Tips Penanaman Hidroponik dari barang bekas"
    android:backgroundTint="@color/green_688"
    android:layout_marginTop="350dp"
    android:textSize="12dp"
    android:fontFamily="sans-serif-black"
    android:textColor="@color/black"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"/>
```

Gambar 4.21 Script Layout Tombol Button Vidio

```

        btnV1 = (Button) findViewById(R.id.btnV1);
        btnV2 = (Button) findViewById(R.id.btnV2);

        btnV1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent Bukasatu = new Intent(getApplicationContext(), VidioMerakit_Hidroponik.class);
                startActivity(Bukasatu);
            }
        });

        btnV2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) { Bukadua(); }
        });
    }

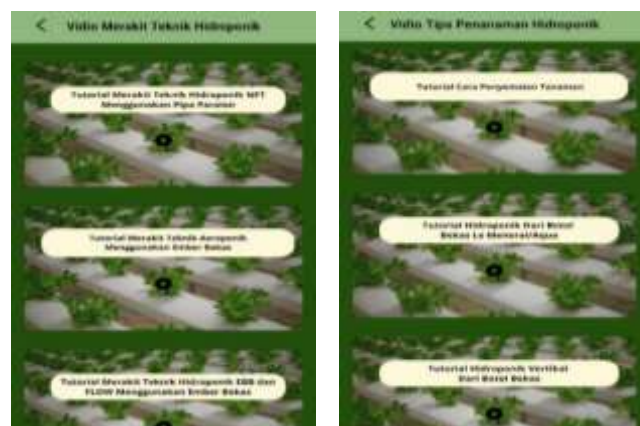
    1 usage
    public void Bukadua () {
        Intent Bukadua = new Intent(getApplicationContext(), VidioPenanaman_Hidroponik.class);
        startActivity(Bukadua);
    }
}

```

Gambar 4.93 Script Java Tombol Button Vidio

13. Tampilan Menu Deskripsi Vidio Budidaya Hidroponik

Pada gambar 4.93 merupakan vidio budidaya hidroponik yang salah satu menunya, yaitu vidio tutorial merakit teknik-teknik hidroponik.



Gambar 4.94 Tampilan Halaman Deskripsi Vidio Budidaya Hidroponik

```

<ImageButton
    android:id="@+id/btnBack"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:background="@color/green_600"
    android:layout_marginTop="2dp"
    android:src="@drawable/baseline_arrow_back_ios_new_24"
    android:contentDescription="Back"/>

```

Gambar 4.95 Script Layout Tombol Kembali

```
btnBack.setOnClickListener(v -> finish());
```

Gambar 4.96 Script Layout Tombol Kembali

```
<ScrollView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_below="@id/toolbar">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:padding="16dp">
        <FrameLayout
            android:layout_width="match_parent"
            android:layout_height="200dp"
            android:layout_marginBottom="16dp"
            android:layout_marginTop="16dp">
            <ImageView
                android:id="@+id/placeholder1"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:scaleType="centerCrop"
                android:src="@drawable/awal18"
                android:contentDescription="Video Placeholder 1" />
            <ImageView
                android:id="@+id/playButton1"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_gravity="center"
                android:layout_marginTop="25dp"
                android:src="@drawable/baseline_play_circle_24"
                android:contentDescription="Play Video 1"
                android:padding="8dp"/>
            <VideoView
                android:id="@+id/videoView1"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:visibility="gone" />
        </FrameLayout>
```

Gambar 4.97 Script Layout Tampilan Vidio

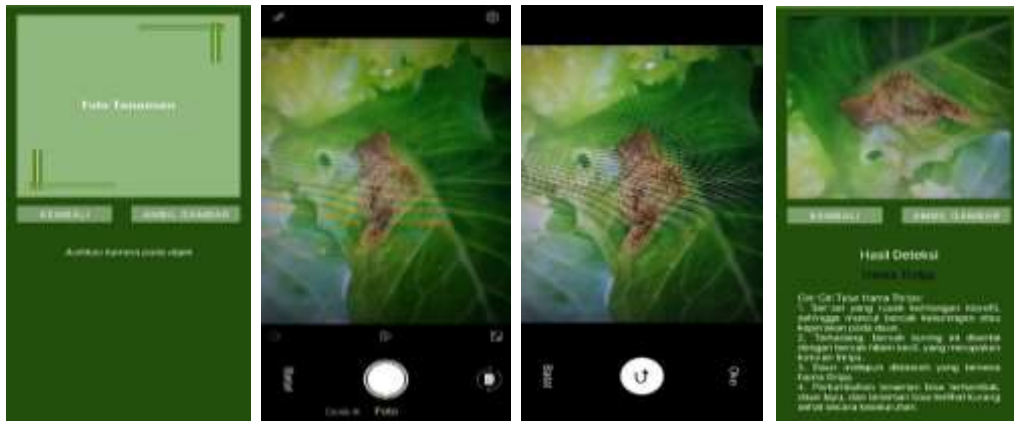
```
initializeVideo(R.id.placeholder1, R.id.playButton1, R.id.videoView1, R.raw.merakit_teknik_nft);
initializeVideo(R.id.placeholder2, R.id.playButton2, R.id.videoView2, R.raw.merakit_teknik_aerponik);
initializeVideo(R.id.placeholder3, R.id.playButton3, R.id.videoView3, R.raw.merakit_teknik_abbflow);
initializeVideo(R.id.placeholder4, R.id.playButton4, R.id.videoView4, R.raw.merakit_teknik_ranitopung);
initializeVideo(R.id.placeholder5, R.id.playButton5, R.id.videoView5, R.raw.merakit_teknik_dripsistem);
initializeVideo(R.id.placeholder6, R.id.playButton6, R.id.videoView6, R.raw.gelon_bekas);
}

private void initializeVideo(int placeholderId, int playButtonId, int videoViewId, int videoResId) {
    ImageView placeholder = findViewById(placeholderId);
    ImageView playButton = findViewById(playButtonId);
    VideoView videoView = findViewById(videoViewId);
    MediaController mediaController = new MediaController(this);
    mediaController.setAnchorView(videoView);
    videoView.setMediaController(mediaController);
    Uri videoUri = Uri.parse("android.resource://" + getPackageName() + "/" + videoResId);
    videoView.setVideoURI(videoUri);
    playButton.setOnClickListener(v -> {
        placeholder.setVisibility(View.GONE);
        playButton.setVisibility(View.GONE);
        videoView.setVisibility(View.VISIBLE);
        videoView.start();
    });
}
```

Gambar 4.98 Script Java Tampilan Vidio

14. Tampilan Menu Kamera Pendeteksi Penyakit dan Hama Hidroponik

Pada gambar 4.99 merupakan tampilan halaman kamera pendeteksi jenis penyakit dan hama hidroponik yang dapat menggunakan kamera, jika hasil gambar sesuai maka akan tampil hasil deteksi.



Gambar 4.99 Tampilan Halaman Kamera Pendeteksi Penyakit dan Hama Hidroponik

```
<Button
    android:id="@+id/backButton"
    android:layout_width="160dp"
    android:layout_height="40dp"
    android:layout_below="@id/imageView"
    android:layout_alignParentStart="true"
    android:layout_marginTop="8dp"
    android:text="Kembali" />
<Button
    android:id="@+id/picture"
    android:layout_width="wrap_content"
    android:layout_height="40dp"
    android:layout_below="@id/imageView"
    android:layout_alignParentEnd="true"
    android:layout_marginTop="8dp"
    android:text="Ambil Gambar" />
```

Gambar 4.100 Script Layout Tombol Kembali dan Tombol Ambil Gambar

```
// Aksi saat tombol kembali diklik
backButton.setOnClickListener(v -> finish());

// Aksi saat tombol ambil gambar diklik
pictureButton.setOnClickListener(v -> {
    if (ContextCompat.checkSelfPermission(this,
        Manifest.permission.CAMERA) == PackageManager.PERMISSION_GRANTED) {
        Intent cameraIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
        startActivityForResult(cameraIntent, requestCode: 1);
    } else {
        ActivityCompat.requestPermissions(this, new
            String[]{Manifest.permission.CAMERA}, requestCode: 100);
    }
});
```

Gambar 4.101 Script Java Tombol Kembali dan Tombol Ambil Gambar

```

<TextView
    android:id="@+id/demoTxt"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Arahkan kamera pada objek"
    android:textStyle="italic"
    android:textColor="@android:color/white"
    android:textSize="14dp"
    android:layout_gravity="center_horizontal" />

```

Gambar 4.102 Script Layout Tampilan Teks

```

<ScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/picture"
    android:layout_marginTop="16dp">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:padding="16dp">
        <!-- TextView untuk menampilkan hasil deteksi -->
        <TextView
            android:id="@+id/classified"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Hasil Deteksi"
            android:layout_gravity="center_horizontal"
            android:textSize="18sp"
            android:visibility="gone"
            android:textStyle="bold"
            android:textColor="@color/white"
            android:fontFamily="sans-serif-medium" />
        <TextView
            android:id="@+id/result"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Hasil deteksi akan ditampilkan di sini"
            android:textSize="17sp"
            android:gravity="center"
            android:visibility="gone"
            android:layout_marginTop="10dp"
            android:textColor="@color/black"
            android:textStyle="bold"
            android:layout_gravity="center_horizontal" />

```

Gambar 4.103 Script Layout Tampilan Deteksi

```

protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    try {
        if (requestCode == 1 && resultCode == RESULT_OK && data != null) {
            Bitmap image = (Bitmap) data.getExtras().get("data");
            if (image != null) {
                int dimension = Math.min(image.getWidth(), image.getHeight());
                image = ThumbnailUtils.extractThumbnail(image, dimension, dimension);
                imageView.setImageBitmap(image);
                demoTxt.setVisibility(View.GONE);
                classified.setVisibility(View.VISIBLE);
                result.setVisibility(View.VISIBLE);
                image = Bitmap.createScaledBitmap(image, imageSize, imageSize, true);
                classifyImage(image);
            } else {
                Toast.makeText(context, this, "Gagal mengambil gambar", Toast.LENGTH_SHORT).show();
            }
        }
    } catch (Exception e) {
        Log.e("tag: onActivityResult", "msg: Terjadi kesalahan: " + e.getMessage());
        Toast.makeText(context, this, "Terjadi kesalahan saat memproses gambar", Toast.LENGTH_SHORT).show();
    }
}

private void classifyImage(Bitmap image) {
    try {
        // Buat model
        ModelHamapenyakit model = ModelHamapenyakit.newInstance(getApplicationContext());

        // Skalakan gambar ke ukuran model
        image = Bitmap.createScaledBitmap(image, imageSize, imageSize, true);

        // Konversi gambar ke TensorBuffer dengan format numerik
        TensorBuffer inputFeature0 = TensorBuffer.createFixedSize(new int[]{1, imageSize, imageSize, 3},
            DataType.FLOAT32);
        ByteBuffer byteBuffer = ByteBuffer.allocateDirect(capacity: 4 * imageSize * imageSize * 3);
        byteBuffer.order(ByteOrder.nativeOrder());

        int[] intValues = new int[imageSize * imageSize];
        image.getPixels(intValues, 0, imageSize, 0, 0, imageSize, imageSize);

        // Normalisasi pixel dari 0-255 ke 0-1 dengan 3 komponen warna RGB
        for (int i = 0; i < intValues.length; i++) {
            int pixel = intValues[i];
            byteBuffer.putFloat(value: ((pixel >> 16) & 0xFF) / 255.0f); // Merah
            byteBuffer.putFloat(value: ((pixel >> 8) & 0xFF) / 255.0f); // Hijau
            byteBuffer.putFloat(value: (pixel & 0xFF) / 255.0f); // Biru
        }

        inputFeature0.loadBuffer(byteBuffer);
        // Proses gambar melalui model
        ModelHamapenyakit.Outputs outputs = model.process(inputFeature0);
        TensorBuffer outputFeature0 = outputs.getOutputFeature0AsTensorBuffer();
        // tingkat kepercayaan akurasi tertinggi
        float[] confidence = outputFeature0.getFloatArray();
        int maxPos = 0;
        float maxConfidence = 0;
        for (int i = 0; i < confidence.length; i++) {
            if (confidence[i] > maxConfidence) {
                maxConfidence = confidence[i];
                maxPos = i;
            }
        }
    }
}

```

```

float confidenceThreshold = 0.7f;
String[] classes = {
    "Hama Ulat Kubis (Tritip Plutella Xylostell)",
    "Penyakit Busuk Daun",
    "Penyakit Kelainan Genetik",
    "Hama Thrips",
    "Hama Serangga dan Burung",
    "Gambar Tidak Dikenali"
};

// Tentukan hasil klasifikasi
String currentClass = classes[maxPos];
if (maxConfidence < confidenceThreshold) {
    currentClass = classes[classes.length - 1]; // "Gambar Tidak Dikenali"
}

result.setText(classes[maxPos]);
result.setPaintFlags(result.getPaintFlags() | Paint.UNDERLINE_TEXT_FLAG);
// Tampilkan teks pencegahan jika gambar dikenali
displayPreventionText(classes[maxPos]);
model.close();
} catch (IOException e) {
    e.printStackTrace();
    Toast.makeText(context, this, "Gagal memproses gambar", Toast.LENGTH_SHORT).show();
}
}

```

Gambar 4.104 Script Java Proses Tampilan Deteksi Penyakit dan Hama

```

<TextView
    android:id="@+id/prevention"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Informasi pencegahan"
    android:layout_gravity="center_horizontal"
    android:visibility="gone"
    android:justificationMode="inter_word"
    android:fontFamily="sans-serif-medium"
    android:textColor="@color/white"
    android:layout_marginTop="18dp" />
<TextView
    android:id="@+id/unknown_image_message"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Arahkan kembali gambar yang benar dan
    fokuskan kamera"
    android:textSize="14sp"
    android:textStyle="italic"
    android:textColor="@android:color/white"
    android:layout_gravity="center_horizontal"
    android:gravity="center"
    android:layout_marginTop="140dp"
    android:visibility="gone"
    android:layout_centerHorizontal="true" />

```

Gambar 4.105 Script Layout Tampilan Ciri-Ciri Dan Cara Pengendalian Hama Penyakit Hidroponik

```

private void displayPreventionText(String className) {
    String preventionText = getPreventionText(className);
    if (className.equals("Gambar Tidak Dikenali")) {
        prevention.setVisibility(View.GONE);
        click_here.setVisibility(View.GONE);
        unknownImageMessage.setVisibility(View.VISIBLE); // Tampilkan pesan untuk gambar tidak dikenali
    } else {
        prevention.setText(preventionText);
        prevention.setVisibility(View.VISIBLE);
        click_here.setVisibility(View.VISIBLE);
        unknownImageMessage.setVisibility(View.GONE);
    }
}

private String getPreventionText(String className) {
    switch (className) {
        case "Hama Ulat Kubis (Tritip Plutella Xylostelli)":
            return "Ciri-Ciri Hama Ulat Kubis (Tritip Plutella Xylostelli):\n\n" +
                "1. Ulat akan memakan jaringan daun dan menyebabkan lubang-lubang kecil yang dapat mem-  

                "2. Daun yang terinfeksi menjadi kering, terutama di bagian bercak, sehingga tampak se-  

                "3. Terdapat jejak kotoran kecil yang terletak di sekitar lubang-lubang yang dibuat ol-  

                "Cara Pencegahan dan Pengendalian Hama Ulat Kubis (Tritip Plutella Xylostelli):\n\n" +
                "1. Gunakan jaring pelindung atau netting untuk mencegah kupu-kupu dewasa masuk dan be-  

                "2. Lakukan pemantauan secara berkala untuk melihat gejala-gejala serangan hama, seper-  

                "3. Bersihkan secara rutin area sekitar tanaman hidroponik dari daun-daun yang jatuh a-  

        case "Penyakit Busuk Daun":
            return "Ciri-Ciri Penyakit Busuk Daun:\n\n" +
                "1. Bercak awalnya kecil, berwarna coklat atau kuning pucat. Bercak biasanya memiliki t-  

                "2. Daun yang terinfeksi menjadi kering, terutama di bagian bercak, sehingga tampak se-  

                "3. Bagian tepi daun sering kali tampak seperti terbakar dan kering jika infeksi telah  

                "Cara Pencegahan dan Pengendalian Penyakit Busuk Daun:\n\n" +
                "1. Buang daun yang terinfeksi untuk mencegah penyebaran.\n" +
                "2. Gunakan fungisida organik atau kimia yang sesuai.\n" +
                "3. Terapkan biofungisida seperti Trichoderma spp. untuk pengendalian alami.";
    }
}

```

Gambar 4.106 Script Java Tampilan Ciri-Ciri Dan Cara Pengendalian Hama Penyakit Hidroponik

```

<TextView
    android:id="@+id/click_here"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Klik di sini untuk mencari lebih lanjut"
    android:textColor="@android:color/holo_blue_light"
    android:textStyle="italic"
    android:layout_gravity="center_horizontal"
    android:visibility="gone"
    android:layout_marginTop="16dp" />

```

Gambar 4.107 Script Layout Tombol Kepencarian Google

```

click_here.setOnClickListener(v -> {
    String query = result.getText().toString();
    Intent browserIntent = new Intent(Intent.ACTION_VIEW, Uri.parse
        ( UriString: "https://www.google.com/search?q=" + query));
    startActivity(browserIntent);
});
}

```

Gambar 4.108 Script Layout Tombol Kepencarian Google

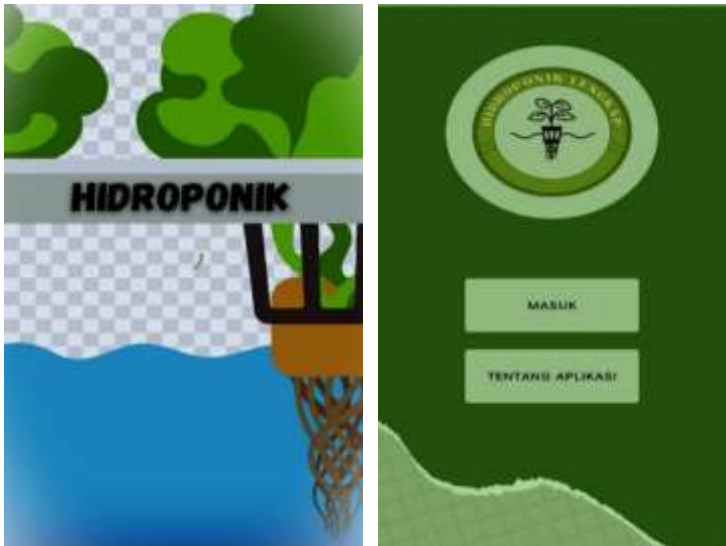
C. Pengujian Sistem

Penelitian ini menggunakan tiga metode pengujian sistem, yaitu pengujian *Black Box*, *White Box*, dan pengujian akurasi. Berikut adalah hasil pengujian yang diperoleh melalui pendekatan-pendekatan tersebut.

1. *Black Box Testing*

a. *Black Box Testing* Halaman Menu Awal

Tabel 4. 2 *Black Box Testing* Halaman Menu Awal

Tes Faktor	Hasil	Keterangan
User pertama kali menjalankan aplikasi	✓	Berhasil tampil halaman splash screen menu awal
Screenshot		
		

b. Black Box Testing Halaman Menu Utama

Tabel 4.3 Black Box Testing Halaman Menu Utama

Tes Faktor	Hasil	Keterangan
<i>User</i> menekan menu masuk	✓	Berhasil tampil halaman menu utama
Screenshot		
		

c. *Black Box Testing* Halaman Menu Tentang Aplikasi

Tabel 4. 4 *Black Box Testing* Halaman Menu Tentang Aplikasi

Tes Faktor	Hasil	Keterangan
User menekan menu tentang aplikasi	✓	Berhasil tampil halaman tentang aplikasi
Screenshot		
 Aplikasi Hidroponik adalah alat yang dirancang untuk membantu Anda dalam mengelola sistem hidroponik secara efisien. Dengan fitur panduan lengkap dan pendeteksi penyakit dan hama tanaman, aplikasi ini memungkinkan Anda untuk mengoptimalkan pertumbuhan tanaman hidroponik Anda dan bagi pemula yang ingin mencoba penanaman secara hidroponik. Fitur Utama: - Panduan Lengkap: Bersel tentang Pengenalan Hidroponik, Media tanam dan pupuk hidroponik, teknik penanaman hidroponik, penyakit dan hama tanaman hidroponik dan video budaya hidroponik. - Pendeteksi Penyakit dan Hama Tanaman: dapat mendeteksi jenis penyakit dan hama tanaman hidroponik dengan menggunakan kamera. Panduan Penggunaan: Pendeteksi Penyakit dan Hama Tanaman Hidroponik: 1. Nyalakan kamera dan tempatkan kamera di area di mana tanaman dan hama sering terlihat. 2. Pastikan area yang dipantau memiliki pencahayaan yang cukup. 3. Jika gambar telah selesai di foto, maka akan muncul nama jenis Penyakit atau Hama yang di deteksi dan cara penanggulangannya.		

d. *Black Box Testing* Halaman Menu Pengenalan Hidroponik**Tabel 4. 5** *Black Box Testing* Halaman Pengenalan Hidroponik

Tes Faktor	Hasil	Keterangan
User menekan menu pengenalan hidroponik	✓	Berhasil tampil halaman pengenalan hidroponik
Screenshot		
		

e. *Black Box Testing* Halaman Deskripsi Pengenalan Hidroponik**Tabel 4. 6** *Black Box Testing* Halaman Deskripsi Pengenalan Hidroponik

Tes Faktor	Hasil	Keterangan
User menekan menu definisi hidroponik dan keunggulan dan kelemahan hidroponik	✓	Berhasil tampil definisi hidroponik dan keunggulan dan kelemahan hidroponik
Screenshot		
 Definisi Hidroponik Dalam bahasa Inggris, hidroponik (hydroponic) merupakan berasal dari kata Yunani yaitu hydro yang berarti air serta ponos yang artinya daya. Hidroponik sendiri dikenal dengan soilless culture maupun dengan arti budidaya tanpa tanah. Jadi hidroponik ini adalah sebuah budidaya tanaman yang menggunakan air dengan tidak memakai tanah sebagai media tanamnya. Hidroponik merupakan sebuah cara budidaya menanam tanpa media tanah, dengan cara memanfaatkan air. Satu hal yang dapat ditekankan dalam sebuah hidroponik yaitu pemenuhan kebutuhan nutrisi untuk sebuah tanaman. Teknik dalam menanam hidroponik ini memerlukan air lebih sedikit apabila dibandingkan dengan menanam di tanah pada umumnya.  Keunggulan dan Kelemahan Hidroponik Keunggulan Hidroponik: 1. Tidak membutuhkan tanah. 2. Tidak membutuhkan banyak air. 3. Mudah dalam pengendalian nutrisi, sehingga pemberian nutrisi lebih efisien. 4. Relatif tidak menghasilkan polusi nutrisi ke lingkungan. 5. Memberikan hasil yang lebih banyak. 6. Mudah dalam memanen. 7. Steril dan bersih. 8. Media tanam dapat dilakukan bertahun-tahun. 9. Tanaman tumbuh lebih cepat. 10. Sangat cocok untuk lahan terbatas. 11. Mengurangi zat kimia ke tanah. 12. Kandungan gizi tanaman hidroponik lebih tinggi.		

f. *Black Box Testing* Halaman Menu Media Tanam Dan Pupuk Hidroponik**Tabel 4. 7** *Black Box Testing* Halaman Menu Media Tanam Dan Pupuk Hidroponik

Tes Faktor	Hasil	Keterangan
User menekan menu media tanam dan pupuk hidroponik	✓	Berhasil tampil halaman media tanam dan pupuk hidroponik
Screenshot		
		

g. *Black Box Testing* Halaman Deskripsi Media Tanam Dan Pupuk Hidroponik

Tabel 4. 8 *Black Box Testing* Halaman Deskripsi Media Tanam Dan Pupuk Hidroponik

Tes Faktor	Hasil	Keterangan
User menekan menu jenis-jenis media tanam hidroponik, jenis-jenis pupuk hidroponik dan tips pembuatan nutrisi hidroponik	✓	Berhasil tampil halaman jenis-jenis media tanam hidroponik, jenis-jenis pupuk hidroponik dan tips pembuatan nutrisi hidroponik
Screenshot		
  		

h. *Black Box Testing* Halaman Menu Teknik Penanaman Hidroponik**Tabel 4. 9** *Black Box Testing* Halaman Teknik penanaman hidroponik

Tes Faktor	Hasil	Keterangan
User menekan menu Teknik penanaman hidroponik	✓	Berhasil tampil halaman Teknik penanaman hidroponik
Screenshot		
		

i. *Black Box Testing* Halaman Deskripsi Teknik Penanaman Hidroponik**Tabel 4.10** *Black Box Testing* Halaman Deskripsi Teknik Penanaman Hidroponik

Tes Faktor	Hasil	Keterangan
User menekan menu jenis-jenis Teknik penanaman hidroponik dan tata cara penanaman secara hidroponik	✓	Berhasil tampil halaman jenis-jenis Teknik penanaman hidroponik dan tata cara penanaman secara hidroponik
Screenshot		
 		

j. *Black Box Testing* Halaman Menu Penyakit Dan Hama Hidroponik**Tabel 4.11** *Black Box Testing* Halaman Menu Penyakit Dan Hama Hidroponik

Tes Faktor	Hasil	Keterangan
User menekan menu penyakit dan hama hidroponik	✓	Berhasil tampil halaman penyakit dan hama hidroponik
Screenshot		
		

k. *Black Box Testing* Halaman Deskripsi Penyakit Dan Hama Hidroponik**Tabel 4.12** *Black Box Testing* Halaman Deskripsi Penyakit Dan Hama Hidroponik

Tes Faktor	Hasil	Keterangan
User menekan menu jenis-jenis penyakit dan hama hidroponik dan upaya pengendalian dan pencegahan penyakit dan hama hidroponik	✓	Berhasil tampil halaman jenis-jenis penyakit dan hama hidroponik dan upaya pengendalian dan pencegahan penyakit dan hama hidroponik
Screenshot		
 		

1. *Black Box Testing* Halaman Menu Video Budidaya Hidroponik

Tabel 4.13 *Black Box Testing* Halaman Video Budidaya Hidroponik

Tes Faktor	Hasil	Keterangan
User menekan menu video budidaya hidroponik	✓	Berhasil tampil halaman video budidaya hidroponik
Screenshot		
		

m. *Black box testing* halaman deskripsi video budidaya hidroponik

Tabel 4.14 *Black Box Testing* Halaman Deskripsi Video Budidaya Hidroponik

Tes Faktor	Hasil	Keterangan
User menekan menu video tutorial merakit teknik-teknik hidroponik dan video tips penanamn hidroponik dari barang bekas	✓	Berhasil tampil halaman video tutorial merakit teknik-teknik hidroponik dan video tips penanamn hidroponik dari barang bekas
Screenshot		
 		

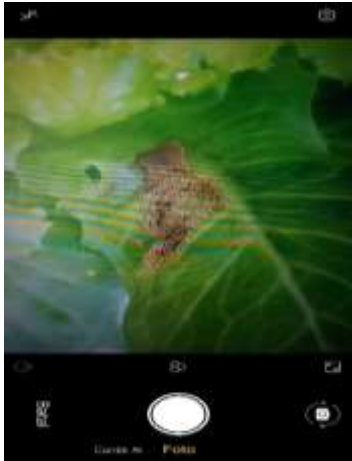
n. *Black Box Testing* Halaman Menu Kamera Pendeteksi Penyakit Dan Hama Tanaman Hidroponik

Tabel 4.15 *Black Box Testing* Halaman Deskripsi Video Budidaya Hidroponik

Tes Faktor	Hasil	Keterangan
User menekan menu kamera pendeteksi penyakit dan hama tanaman hidroponik	✓	Berhasil tampil halaman kamera pendeteksi penyakit dan hama tanaman hidroponik
Screenshot		
		

- o. *Black Box Testing* Halaman Hasil Deteksi Penyakit Dan Hama Tanaman Hidroponik

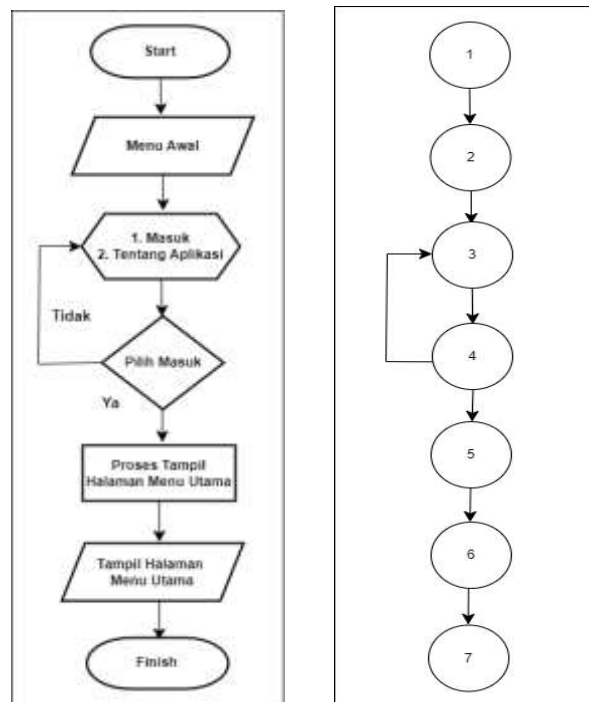
Tabel 4. 16 *Black Box Testing* Halaman Hasil Deteksi Penyakit Dan Hama Tanaman Hidroponik

Tes Faktor	Hasil	Keterangan
User menekan ambil gambar	✓	Berhasil tampil halaman kamera dan hasil deteksi penyakit dan hama tanaman hidroponik
Screenshot		
   		

2. White Box Testing

Dalam pengujian ini akan menampilkan *flowchart* dan *flowgraph* dari sistem yang telah dibuat. Berikut ini adalah hasil dari pengujian menggunakan *white box testing* secara keseluruhan.

a. Flowchart dan Flowgraph Menu Awal



Gambar 4.109 Flowchart dan Flowgraph Menu Awal

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau *Flowchart* dan *Flowgraph* di atas:

1). Menghitung kompleksitas cyclomatic $V(G)$ pada *edge* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 7$$

$$N \text{ (node)} = 7$$

$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 7 - 7 + 2 \\ &= 2 \end{aligned}$$

$$\begin{aligned} \text{Predikat (P)} &= P + 1 \\ &= 1 + 1 \\ &= 2 \end{aligned}$$

2). Berdasarkan perhitungan kompleksitas *cyclomatic* dari *flowgraph* di atas, memiliki = 2

3). Jalan independen pada *flowgraph* ini adalah:

Path 1 = 1 – 2 – 3 – 4 – 3

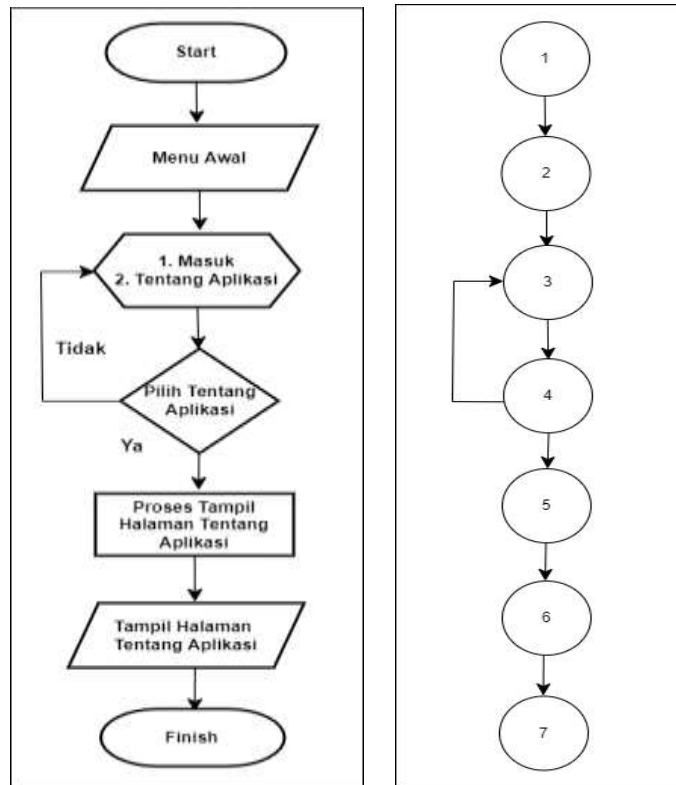
Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 7

4).Grafik matriks menu menu awal

Tabel 4. 17 Grafik matriks menu menu awal

	1	2	3	4	5	6	7	E - 1
1		1						$1 - 1 = 0$
2			1					$1 - 1 = 0$
3				1				$1 - 1 = 0$
4			1		1			$2 - 1 = 1$
5						1		$1 - 1 = 0$
6							1	$1 - 1 = 0$
7								0
	SUM (E + 1)							$1 + 1 = 2$

b. *Flowchart* dan *Flowgraph* Menu Tentang Aplikasi



Gambar 4.110 *Flowchart* dan *Flowgraph* Tentang Aplikasi

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau *Flowchart* dan *Flowgraph* di atas:

1). Menghitung kompleksitas *cyclomatic* $V(G)$ pada *edge* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 7$$

$$N \text{ (node)} = 7$$

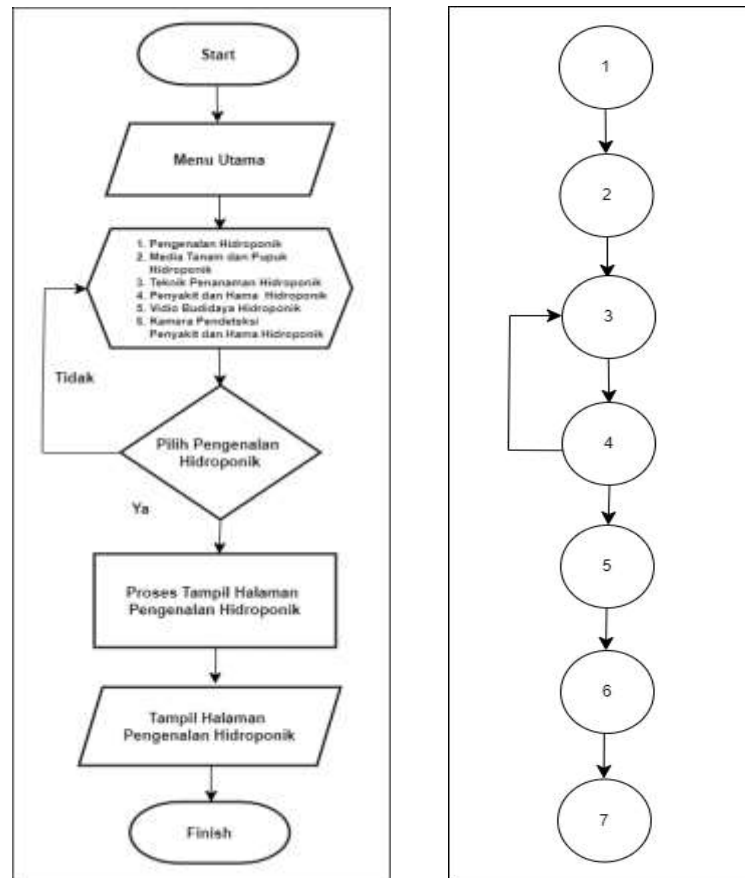
$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

$$V(G) = E - N + 2$$

$$= 7 - 7 + 2$$

c. *Flowchart* dan *Flowgraph* Menu Pengenalan Hidroponik



Gambar 4.111 Flowchart dan Flowgraph Pengenalan Hidroponik

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau

Flowchart dan *Flowgraph* di atas:

1). Menghitung kompleksitas *cyclomatic* $V(G)$ pada *edge* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 7$$

$$N \text{ (node)} = 7$$

$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 7 - 7 + 2 \\ &= 2 \end{aligned}$$

$$\begin{aligned} \text{Predikat (P)} &= P + 1 \\ &= 1 + 1 \\ &= 2 \end{aligned}$$

2). Berdasarkan perhitungan kompleksitas *cyclomatic* dari *flowgraph* di atas, memiliki = 2

3). Jalan independen pada *flowgraph* ini adalah:

Path 1 = 1 – 2 – 3 – 4 – 3

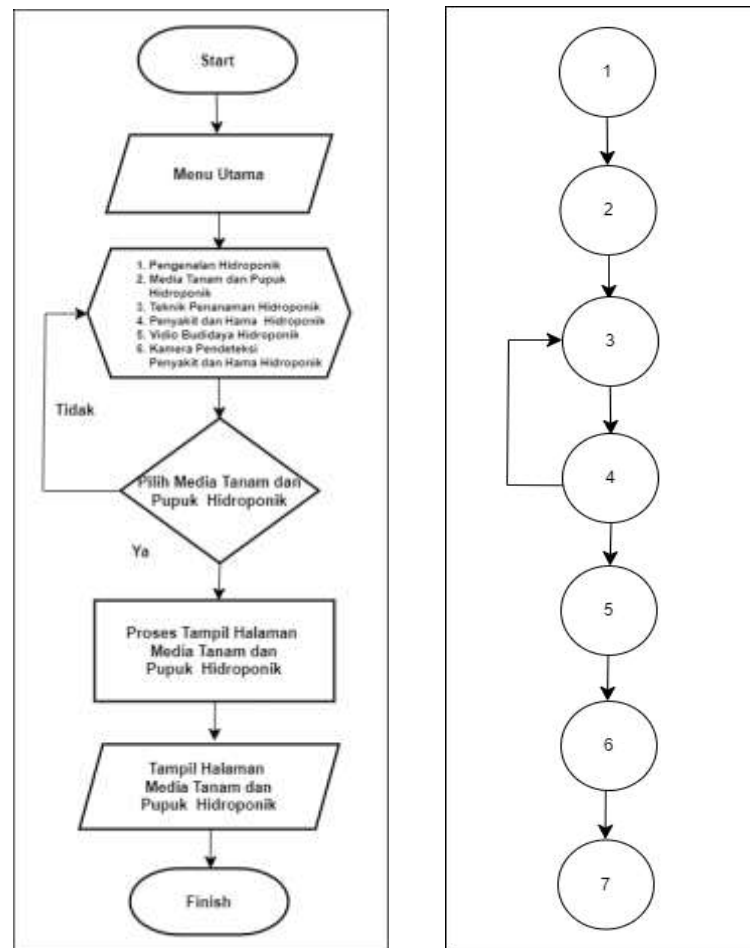
Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 7

4).Grafik matriks menu pengenalan hidroponik

Tabel 4. 19 Grafik matriks menu pengenalan hidroponik

	1	2	3	4	5	6	7	E - 1
1		1						$1 - 1 = 0$
2			1					$1 - 1 = 0$
3				1				$1 - 1 = 0$
4			1		1			$2 - 1 = 1$
5						1		$1 - 1 = 0$
6							1	$1 - 1 = 0$
7								0
	SUM (E + 1)							$1 + 1 = 2$

d. *Flowchart* dan *Flowgraph* Menu Media Tanam dan Pupuk Hidroponik



Gambar 4.112 *Flowchart* dan *Flowgraph* Menu Media Tanam dan Pupuk Hidroponik

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau

Flowchart dan *Flowgraph* di atas:

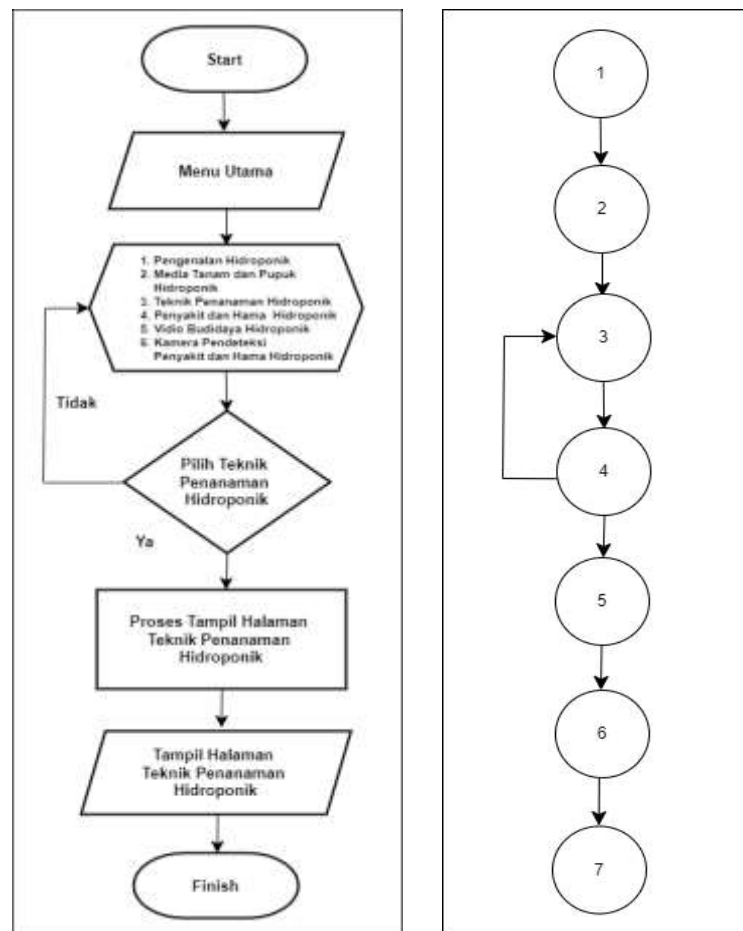
1). Menghitung kompleksitas *cyclomatic* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 7$$

$$N \text{ (node)} = 7$$

e. *Flowchart dan Flowgraph Menu Teknik Penanaman Hidroponik*



Gambar 4.113 *Flowchart dan Flowgraph Menu Teknik Penanaman Hidroponik*

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau Flowchart dan Flowgraph di atas:

1). Menghitung kompleksitas *cyclomatic* $V(G)$ pada *edge* dan *node*

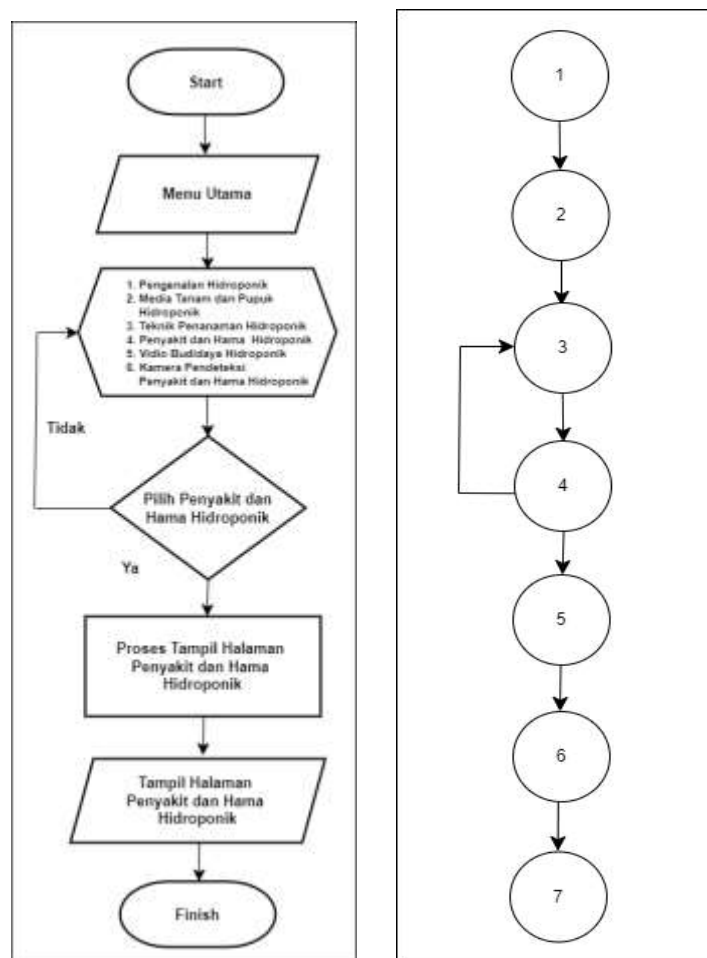
$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 7$$

$$N \text{ (node)} = 7$$

$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

f. *Flowchart* dan *Flowgraph* Menu Penyakit dan Hama Tanaman Hidroponik

Gambar 4.114 *Flowchart* dan *Flowgraph* Menu Penyakit dan Hama Tanaman Hidroponik

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau

Flowchart dan *Flowgraph* di atas:

1). Menghitung kompleksitas *cyclomatic* $V(G)$ pada *edge* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 7$$

$$N \text{ (node)} = 7$$

$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 7 - 7 + 2 \\ &= 2 \end{aligned}$$

$$\begin{aligned}\text{Predikat (P)} &= P + 1 \\ &= 1 + 1 \\ &= 2\end{aligned}$$

2). Berdasarkan perhitungan kompleksitas *cyclomatic* dari *flowgraph* di atas, memiliki = 2

3). Jalan independen pada *flowgraph* ini adalah:

Path 1 = 1 – 2 – 3 – 4 – 3

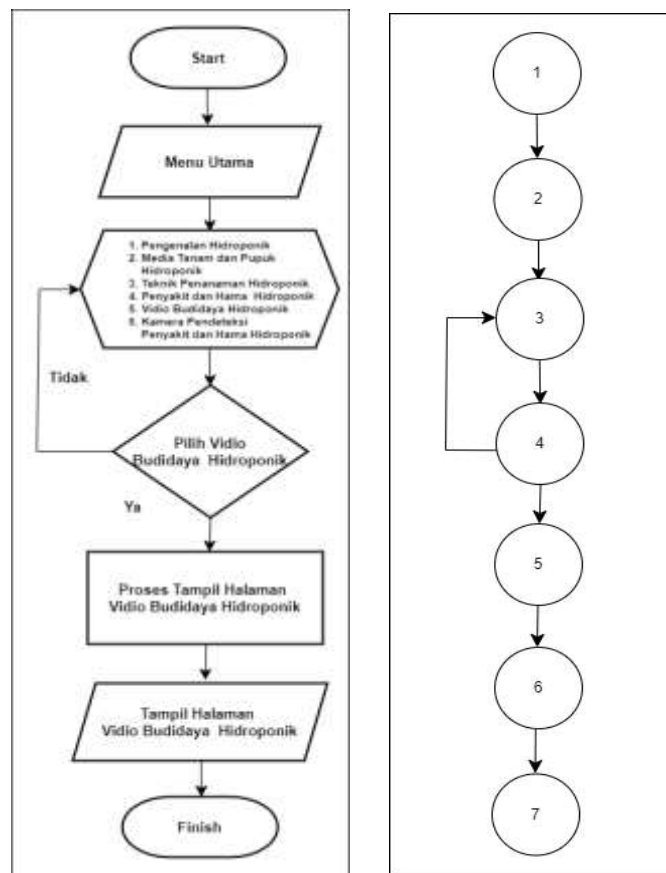
Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 7

4). Grafik matriks menu penyakit dan hama tanaman hidroponik

Tabel 4. 22 Grafik matriks menu penyakit dan hama tanaman hidroponik

	1	2	3	4	5	6	7	E - 1
1		1						1 - 1 = 0
2			1					1 - 1 = 0
3				1				1 - 1 = 0
4			1		1			2 - 1 = 1
5						1		1 - 1 = 0
6							1	1 - 1 = 0
7								0
	SUM (E + 1)							1 + 1 = 2

g. *Flowchart dan Flowgraph Menu Vidio Budidaya Hidroponik*



Gambar 4.115 *Flowchart dan Flowgraph Menu Vidio Budidaya Hidroponik*

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau

Flowchart dan Flowgraph di atas:

1). Menghitung kompleksitas *cyclomatic* $V(G)$ pada *edge* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 7$$

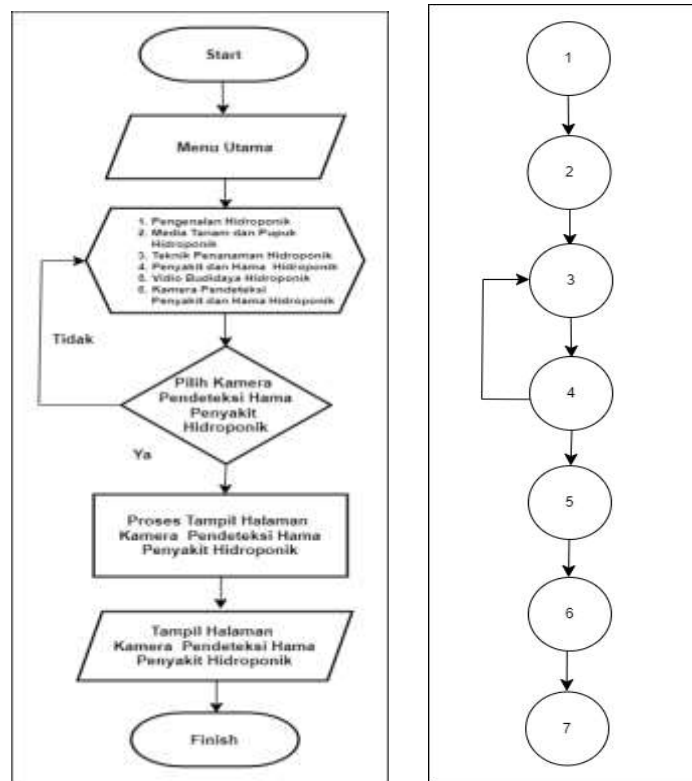
$$N \text{ (node)} = 7$$

$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

$$V(G) = E - N + 2$$

h. *Flowchart* dan *Flowgraph* Menu Kamera Pendeteksi Penyakit dan Hama Tanaman Hidroponik



Gambar 4.116 *Flowchart* dan *Flowgraph* Menu Kamera Pendeteksi Penyakit dan Hama Tanaman Hidroponik

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau

Flowchart dan *Flowgraph* di atas:

1). Menghitung kompleksitas *cyclomatic* $V(G)$ pada *edge* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

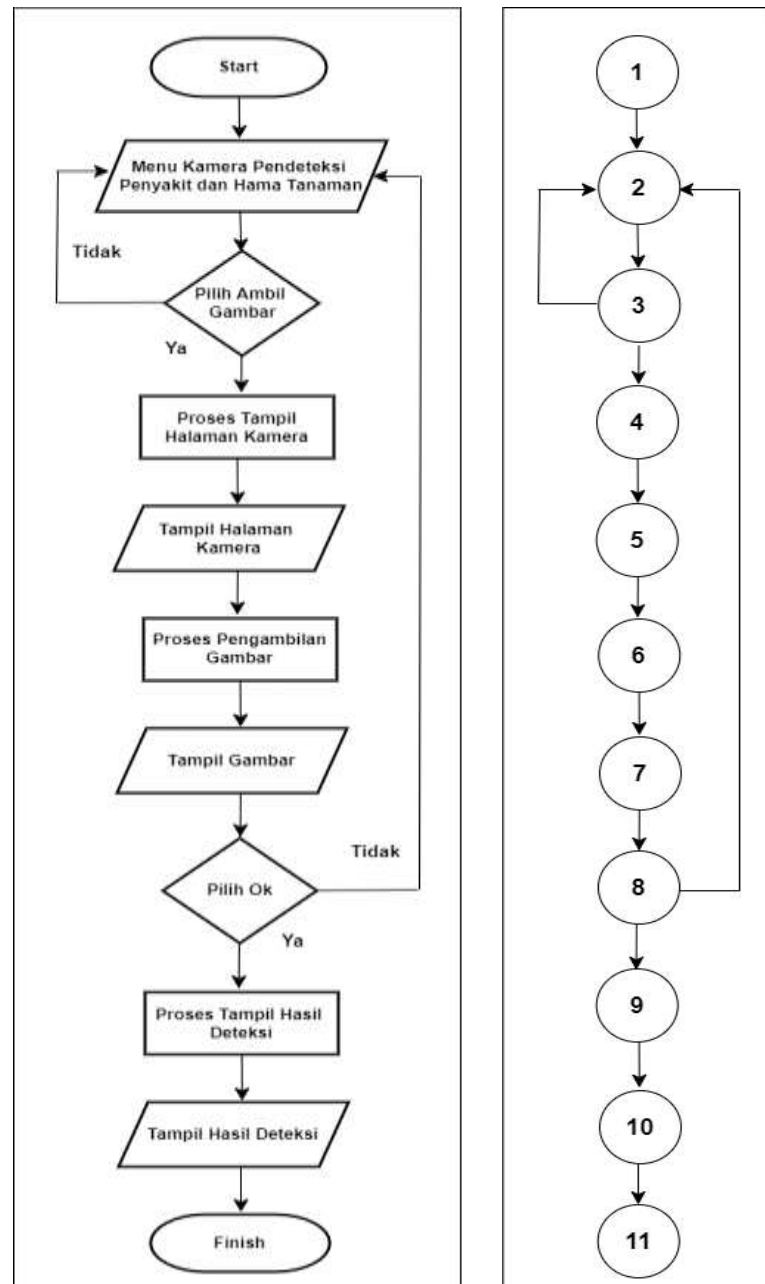
$$E \text{ (edge)} = 7$$

$$N \text{ (node)} = 7$$

$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

i. *Flowchart dan Flowgraph* Proses Deteksi Penyakit dan Hama Tanaman Hidroponik



Gambar 4.117 *Flowchart dan Flowgraph* Proses Deteksi Penyakit dan Hama Tanaman Hidroponik

Perhitungannya dapat diselesaikan sebagai berikut setelah meninjau *Flowchart* dan *Flowgraph* di atas:

- 1). Menghitung kompleksitas *cyclomatic* $V(G)$ pada *edge* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 12$$

$$N \text{ (node)} = 11$$

$$P \text{ (Predikat node)} = 2$$

Penyelesaian :

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 12 - 11 + 2 \\ &= 3 \end{aligned}$$

$$\begin{aligned} \text{Predikat (P)} &= P + 1 \\ &= 2 + 1 \\ &= 3 \end{aligned}$$

- 2). Berdasarkan perhitungan kompleksitas *cyclomatic* dari *flowgraph* di atas, memiliki = 3

- 3). Jalan independen pada *flowgraph* ini adalah:

$$\text{Path 1} = 1 - 2 - 3 - 2$$

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 2$$

$$\text{Path 3} = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11$$

4). Grafik matriks proses deteksi penyakit dan hama tanaman hidroponik

Tabel 4. 25 Grafik matriks proses deteksi penyakit dan hama tanaman hidroponik

	1	2	3	4	5	6	7	8	9	10	11	E - 1
1		1										$1 - 1 = 0$
2			1									$1 - 1 = 0$
3		1		1								$2 - 1 = 1$
4					1							$1 - 1 = 0$
5						1						$1 - 1 = 0$
6							1					$1 - 1 = 0$
7								1				$1 - 1 = 0$
8		1							1			$2 - 1 = 1$
9										1		$1 - 1 = 0$
10											1	$1 - 1 = 0$
11												
	SUM (E + 1)											$2 + 1 = 3$

3. Pengujian Akurasi Pendeteksi Hama Dan Penyakit

Pengujian ini dilakukan untuk mengukur dan mengevaluasi sejauh mana sistem pendeteksi dalam aplikasi dapat mengenali dan mengidentifikasi hama dan penyakit pada tanaman secara akurat dari berbagai pengambilan sudut foto, yaitu depan, samping kanan, atas, dan samping kiri.



Depan



Samping Kanan



Gambar 4.118 Pengambilan foto dari berbagai sudut tanaman

Berikut rata-rata persentase akurasi pada tanaman selada:

- a. Daun selada yang terkena hama ulat kubis (*tritip plutella xylostell*)

Tabel 4.26 Akurasi daun selada yang terkena hama ulat kubis (*tritip plutella xylostell*)

No	Sudut Foto (°)	Jumlah Foto	Deteksi Benar	Deteksi Salah	Akurasi (%)
1	0° (Depan)	5	4	1	80%
2	90° (Samping Kanan)	5	5	0	100%
3	180° (Atas)	5	5	0	100%
4	90° (Samping Kiri)	5	5	0	100%

$$\text{Rata-rata akurasi} = \frac{80+100+100+100}{4} = 95\%$$

- b. Daun selada yang terkena hama thrips

Tabel 4.27 Akurasi daun selada yang terkena hama thrips

No	Sudut Foto (°)	Jumlah Foto	Deteksi Benar	Deteksi Salah	Akurasi (%)
1	0° (Depan)	5	4	1	80%
2	90° (Samping Kanan)	5	4	1	80%

Lanjutan Tabel 4.27

No	Sudut Foto (°)	Jumlah Foto	Deteksi Benar	Deteksi Salah	Akurasi (%)
3	180° (Atas)	5	5	0	100%
4	90° (Samping Kiri)	5	3	2	60%

$$\text{Rata-rata akurasi} = \frac{80+80+100+60}{4} = 80\%$$

c. Daun selada yang terkena hama serangga dan burung

Tabel 4.28 Akurasi daun selada yang terkena hama serangga dan burung

No	Sudut Foto (°)	Jumlah Foto	Deteksi Benar	Deteksi Salah	Akurasi (%)
1	0° (Depan)	5	5	0	100%
2	90° (Samping Kanan)	5	5	0	100%
3	180° (Atas)	5	2	3	40%
4	90° (Samping Kiri)	5	5	0	100%

$$\text{Rata-rata akurasi} = \frac{100+100+40+100}{4} = 85\%$$

d. Penyakit busuk daun pada daun selada

Tabel 4.29 Akurasi penyakit busuk daun pada daun selada

No	Sudut Foto (°)	Jumlah Foto	Deteksi Benar	Deteksi Salah	Akurasi (%)
1	0° (Depan)	5	3	2	60%
2	90° (Samping Kanan)	5	4	1	100%
3	180° (Atas)	5	5	0	100%
4	90° (Samping Kiri)	5	5	0	100%

$$\text{Rata-rata akurasi} = \frac{60+100+100+100}{4} = 90\%$$

e. Penyakit kelainan genetik pada daun selada

Tabel 4.30 Akurasi penyakit kelainan genetik pada daun selada

No	Sudut Foto (°)	Jumlah Foto	Deteksi Benar	Deteksi Salah	Akurasi (%)
1	0° (Depan)	5	5	0	100%
2	90° (Samping Kanan)	5	4	1	80%
3	180° (Atas)	5	5	0	100%
4	90° (Samping Kiri)	5	5	0	100%

$$\text{Rata-rata akurasi} = \frac{100+100+40+100}{5} = 85\%$$

$$\text{Rata-rata Keseluruhan} = \frac{95+80+85+90+85}{5} = 87\%$$

Berdasarkan hasil pengujian, sistem pendeteksi dalam aplikasi menunjukkan tingkat akurasi yang bervariasi tergantung pada jenis hama dan penyakit yang diidentifikasi. Daun selada yang terkena hama ulat kubis (*Plutella xylostella*) memiliki rata-rata akurasi sebesar 95%, menunjukkan performa deteksi yang sangat baik di semua sudut foto. Untuk daun selada yang terkena hama thrips, akurasi rata-ratanya sebesar 80%, yang merupakan nilai terendah dibandingkan kategori lainnya, terutama karena rendahnya akurasi pada sudut tertentu. Deteksi hama serangga dan burung memiliki rata-rata akurasi 85%, dengan kelemahan utama pada sudut pengambilan foto dari atas. Penyakit busuk daun pada daun selada

menunjukkan rata-rata akurasi sebesar 90%, yang cukup baik dengan hanya sedikit kesalahan pada sudut depan. Sedangkan untuk penyakit kelainan genetik pada daun selada, rata-rata akurasi mencapai 85%, dengan kelemahan kecil pada sudut samping kanan.

Secara keseluruhan, rata-rata akurasi sistem untuk semua kategori adalah 87%, yang menunjukkan performa deteksi yang cukup andal. Namun, diperlukan peningkatan akurasi untuk kategori dengan hasil lebih rendah, seperti hama thrips, guna memastikan deteksi yang optimal pada semua sudut pengambilan foto.

D. Proses Deteksi Kamera Pendeteksi Hama dan Penyakit Hidroponik

Proses deteksi dan klasifikasi gambar penyakit pada daun selada melibatkan serangkaian langkah untuk memproses gambar dan menentukan jenis hama atau penyakit berdasarkan *model TensorFlow Lite*. Berikut adalah penjelasan lengkap langkah-langkahnya:

1. Muat Model *TensorFlow Lite*

Model machine learning (ModelHamaPenyakit) dimuat ke dalam aplikasi.

Memungkinkan pemrosesan data gambar menggunakan model terlatih.

2. Persiapan Input Gambar

a. Mengubah ukuran gambar:

- 1) Gambar asli diubah ukurannya menjadi 224×224 piksel.
- 2) Setiap piksel mengandung nilai R (Red), G (Green), dan B (Blue).

b. Normalisasi nilai piksel:

- 1). Nilai piksel gambar dinormalisasi dari skala 0-255 menjadi skala 0-1.

Menggunakan Rumus:

$$\text{Komponen Normalisasi} = \frac{\text{Nilai Asli}}{255}$$

Simulasi piksel:

Piksel 1: R=200, G=150, B=100

$$R_{norm} = \frac{200}{255} = 0.784, \quad G_{norm} = \frac{150}{255} = 0.588, \quad B_{norm} = \frac{100}{255} = 0.392$$

Piksel 2: R=50, G=100, B=200

$$R_{norm} = \frac{50}{255} = 0.196, \quad G_{norm} = \frac{100}{255} = 0.392, \quad B_{norm} = \frac{200}{255} = 0.784$$

Piksel 3 (Putih Murni): R=255, G=255, B=255

$$R_{norm} = \frac{255}{255} = 1.0, \quad G_{norm} = \frac{255}{255} = 1.0, \quad B_{norm} = \frac{255}{255} = 1.0$$

Output Tensor: *Tensor* normalisasi dengan dimensi:

$$[1, 224, 224, 3]$$

3. Proses *Inferensi* dengan Model

- a. Gambar yang sudah diproses dikirimkan ke model untuk diproses melalui *model.process(inputFeature0)*.
- b. Model menghasilkan output berupa *array confidence level* untuk setiap kelas.

Berikut adalah tahapan detail *inferensi* dengan simulasi perhitungan:

1) Operasi Linear (*Fully Connected Layer*)

Setiap *neuron* dalam model menghitung output *Z* menggunakan rumus berikut:

$$Z = W \cdot X + b$$

Ket:

- Simulasi Bobot (W) : [0.5,0.3,-0.2]
- Bias (b) : 0.1
- Input (X) : Nilai normalisasi piksel

Simulasi Perhitungan untuk 3 Piksel:

Piksel 1: $R_{norm} = 0.784$, $G_{norm} = 0.588$, $B_{norm} = 0.392$

$$Z = (0.5 \cdot 0.784) + (0.3 \cdot 0.588) + (-0.2 \cdot 0.392) + 0.1$$

$$Z = 0.392 + 0.176 - 0.078 + 0.1 = 0.59$$

Piksel 2: $R_{norm} = 0.196$, $G_{norm} = 0.392$, $B_{norm} = 0.784$

$$Z = (0.5 \cdot 0.196) + (0.3 \cdot 0.392) + (-0.2 \cdot 0.784) + 0.1$$

$$Z = 0.098 + 0.118 - 0.157 + 0.1 = 0.159$$

Piksel 3: $R_{norm} = 1.0$, $G_{norm} = 1.0$, $B_{norm} = 1.0$

$$Z = (0.5 \cdot 1.0) + (0.3 \cdot 1.0) + (-0.2 \cdot 1.0) + 0.1$$

$$Z = 0.5 + 0.3 - 0.2 + 0.1 = 0.7$$

2) Fungsi Aktivasi (ReLU)

Nilai Z kemudian diproses melalui fungsi aktivasi ReLU yang akan menghilangkan nilai negatif menggunakan rumus:

$$f(Z) = \max(0, Z)$$

Hasil Perhitungan ReLU:

Piksel 1: $f(Z) = \max(0, 0.59) = 0.59$

Piksel 2: $f(Z) = \max(0, 0.159) = 0.159$

Piksel 3: $f(Z) = \max(0, 0.7) = 0.7$

3) Softmax pada Lapisan Akhir

Softmax mengubah nilai menjadi *probabilitas*:

$$\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$$

Keterangan:

- Z_i : Nilai logit dari *neuron* ke- i (hasil perhitungan dari lapisan sebelumnya).
- e^{z_i} : *Eksponensial* dari Z_i , yaitu nilai e dipangkatkan dengan Z_i (di mana $e = 2.718$).
- $\sum_j e^{z_j}$: Total *eksponensial* dari semua nilai logit Z_j , yaitu jumlah semua e^{z_j} untuk $j = 1, 2, 3, \dots, n$ di mana n adalah jumlah kelas.
- **Softmax(z_i)**: *Probabilitas* yang dinormalisasi untuk kelas ke- i . Nilainya selalu berada dalam rentang 0 hingga 1, dan total dari semua *probabilitas* $\sum_i \text{Softmax}(z_i) = 1$.

Perhitungan untuk $Z = [0.59, 0.159, 0.7]$:

a. *Eksponensial*:

$$e^Z = [e^{0.59}, e^{0.159}, e^{0.7}] = [1.803, 1.172, 2.013]$$

b. Total *eksponensial*:

$$\sum_j e^{z_j} = 1.803 + 1.172 + 2.013 = 4.988$$

c. Normalisasi untuk Masing-Masing Kelas:

$$\text{Softmax}(Z_1) = \frac{e^{0.59}}{\sum_j e^{z_j}} = \frac{1.803}{4.988} = 0.361$$

$$\text{Softmax}(Z_2) = \frac{e^{0.159}}{\sum_j e^{z_j}} = \frac{1.172}{4.989} = 0.234$$

$$\text{Softmax}(Z_2) = \frac{e^{0.7}}{\sum_j e^{z_j}} = \frac{2.013}{4.989} = 0.403$$

d. *Probabilitas* kelas:

[0.361, 0.235, 0.403]

4. Identifikasi *Confidence*

Probabilitas tertinggi digunakan untuk menentukan kelas:

- a. *Probabilitas* tertinggi: ***maxConfidence* = 0.403**
- b. Bandingkan dengan ambang batas (0.7):
 - 1) Jika *maxConfidence* ≥ 0.7 gambar dikenali.
 - 2) Jika *maxConfidence* < 0.7 hasil klasifikasi adalah "Gambar Tidak Dikenali".

5. Output dan Tindakan

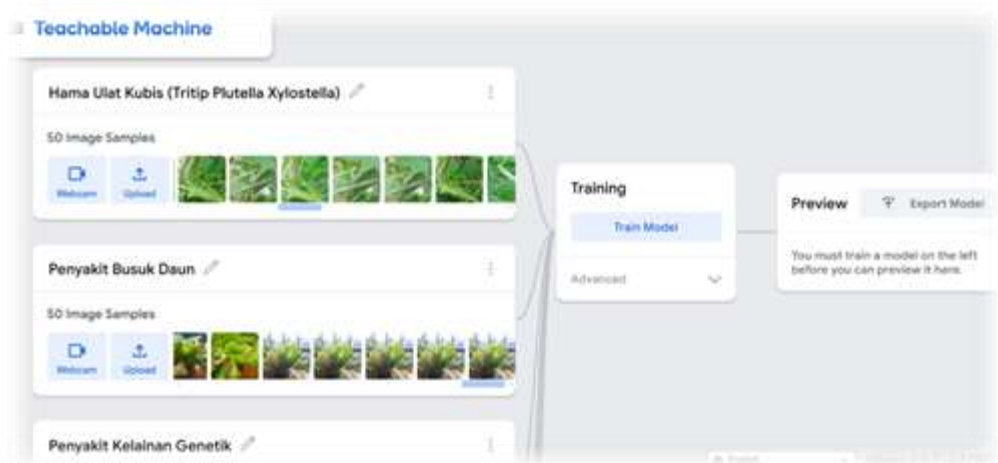
- a. Hasil Klasifikasi:

Hasil Klasifikasi="Gambar Tidak Dikenali" (Karena $0.403 < 0.7$).
- b. Tindakan Pencegahan: Tidak ditampilkan karena gambar tidak dikenali.

6. Tampilkan Hasil Klasifikasi

- a. Nama kelas ditampilkan pada UI menggunakan komponen teks (*result*).
- b. Jika gambar dikenali, teks pencegahan atau pengendalian ditampilkan berdasarkan kelas menggunakan metode *displayPreventionText*.

E. Data Pendeteksi Penyakit dan Hama Tanaman Hidroponik



Gambar 4.119 Data Gambar Penyakit dan Hama Tanaman Hidroponik

Pada gambar 4.42 menunjukkan data gambar penyakit dan hama tanaman yang berisi lima class diantaranya ,yaitu ulat kubis (*tritip plutella xylostell*), hama thrips, hama serangga dan burung, penyakit busuk daun, penyakit kelainan genetik dan gambar tidak dikenali. Berikut isi sampel data gambar penyakit dan hama tanaman hidroponik:

Tabel 4.31 Data Sampel Gambar Penyakit Dan Hama Tanaman

Data Gambar	Sampel
Hama Ulat Kubis (<i>Triptip Plutella Xylostella</i>)	50
Hama Thrips	50
Hama Seranga dan Burung	50
Penyakit Busuk Daun	50
Penyakit Kelainan Genetik	50
Gambar Tidak Dikenali	20

Pada tabel 4.31 menunjukkan data gambar kelas setiap penyakit dan hama masing-masing memiliki 50 sampel, kecuali gambar tidak dikenali hanya 20 sampel.

F. Pengujian Blackbox Dan Whitebox

Tabel 4.32 Pengujian *Blackbox* dan *Whitebox*

No	Tampilan Aplikasi	Pengujian <i>Blackbox</i>	Pengujian <i>Whitebox</i>	Keterangan
1.	Tampilan menu awal	Berhasil tampil	2	Tampilan menu awal pada pengujian <i>blackbox</i> berhasil tampil dan sesuai dengan <i>flowchart</i> dan <i>flowgraph</i> untuk pengujian <i>whitebox</i> .
2.	Tampilan menu utama	Berhasil tampil	2	Tampilan menu utama pada pengujian <i>blackbox</i> berhasil tampil dan sesuai dengan <i>flowchart</i> dan <i>flowgraph</i> untuk pengujian <i>whitebox</i> .
3.	Tampilan menu pengenalan hidroponik	Berhasil tampil	2	Tampilan menu pengenalan hidroponik pada pengujian <i>blackbox</i> berhasil tampil dan sesuai dengan <i>flowchart</i> dan <i>flowgraph</i> untuk pengujian <i>whitebox</i> .
4.	Tampilan menu media tanam dan pupuk hidroponik	Berhasil tampil	2	Tampilan menu media tanam dan pupuk hidroponik pada pengujian <i>blackbox</i> berhasil tampil dan sesuai dengan alur <i>flowchart</i> dan <i>flowgraph</i> untuk pengujian <i>whitebox</i> .
5.	Tampilan menu penanaman hidroponik	Berhasil tampil	2	Tampilan menu penanaman hidroponik pada pengujian <i>blackbox</i> berhasil tampil dan sesuai dengan <i>flowchart</i> dan <i>flowgraph</i> untuk pengujian <i>whitebox</i> .
6.	Tampilan menu penyakit dan hama tanaman hidroponik	Berhasil tampil	2	Tampilan menu penyakit dan hama tanaman hidroponik pada pengujian <i>blackbox</i> berhasil tampil dan sesuai dengan <i>flowchart</i> dan <i>flowgraph</i> untuk pengujian <i>whitebox</i> .
7.	Tampilan menu vidio budidaya hidroponik	Berhasil tampil	2	Tampilan menu vidio budidaya hidroponik pada pengujian <i>blackbox</i> berhasil tampil dan sesuai dengan <i>flowchart</i> dan <i>flowgraph</i> untuk pengujian <i>whitebox</i> .
8.	Tampilan menu kamera pendeteksi hama dan penyakit hidroponik	Berhasil tampil	2	Tampilan menu kamera pendeteksi hama dan penyakit hidroponik pada pengujian <i>blackbox</i> berhasil tampil dan

Lanjutan Tabel 4.32

No	Tampilan Aplikasi	Pengujian Blackbox	Pengujian Whitebox	Keterangan
				sesuai dengan <i>flowchart</i> dan <i>flowgraph</i> untuk pengujian <i>whitebox</i> .
9.	Proses deteksi penyakit dan hama tanaman hidroponik	Berhasil tampil	3	Proses deteksi penyakit dan hama tanaman hidroponik pada pengujian <i>blackbox</i> berhasil tampil dan sesuai dengan <i>flowchart</i> dan <i>flowgraph</i> untuk pengujian <i>whitebox</i> .
10.	Tampilan tentang aplikasi	Berhasil tampil	2	Tampilan menu tentang aplikasi pada pengujian <i>blackbox</i> berhasil tampil dan sesuai dengan <i>flowchart</i> dan <i>flowgraph</i> untuk pengujian <i>whitebox</i> .

G. Implementasi Aplikasi

Implementasi aplikasi dilakukan dengan memberikan kuesioner kepada 30 orang dengan tujuan untuk mengetahui tingkat kepuasan pengguna menggunakan aplikasi panduan hidroponik berbasis android. Berikut ini adalah hasil persentase masing-masing jawaban:

$$\text{Rumus} = \frac{\text{Jumlah Skor}}{\text{Jumlah Pertanyaan} \times \text{Jumlah responden}} \times 100\%$$

Tabel 4. 33 Rata-rata hasil jawaban responden

Pertanyaan	Jawaban			
	Sangat Mudah	Mudah	Sulit	Sangat Sulit
Apakah aplikasi mudah dioperasikan?	10	17	2	1
Apakah tampilan aplikasi memudahkan anda dalam mengetahui informasi pada aplikasi?	5	22	3	0
Apakah informasi dan fitur yang tersedia sesuai dengan yang dibutuhkan?	1	23	6	0
Apakah informasi hidroponik pada aplikasi mudah dimengerti?	4	22	4	0

Lanjutan Tabel 4.33

Pertanyaan	Jawaban			
	Sangat Mudah	Mudah	Sulit	Sangat Sulit
Apakah gambar dan teks pada aplikasi memudahkan dalam mengetahui informasi hidroponik?	10	14	6	0
Apakah audio suara pada aplikasi memudahkan dalam mengetahui informasi hidroponik?	5	21	4	0
Apakah fitur video pada aplikasi memudahkan dalam mengetahui informasi hidroponik?	5	21	4	0
Apakah fitur kamera pendeteksi Penyakit dan hama pada aplikasi berguna bagi pengguna hidroponik?	4	18	7	1
Apakah panduan langkah-langkah bercocok tanam hidroponik yang tersedia di aplikasi ini mudah diikuti?	2	24	4	0
Apakah Anda merasa aplikasi ini dapat memudahkan anda memperkaya pengetahuan Anda tentang perawatan dan solusi untuk masalah yang umum dihadapi dalam hidroponik?	3	24	3	0
Jumlah Skor	49	206	43	2
Persentase (%)	16,3%	68,7%	14,3%	0,7%

Berdasarkan hasil persentase pada Tabel 4.27, terdapat empat kategori penilaian, yaitu Sangat Baik, Baik, Cukup, dan Kurang. Persentase penilaian Sangat Baik adalah 16,3%, Baik 68,7%, Cukup 14,3%, dan Kurang 0,7%. Secara keseluruhan, aplikasi panduan hidroponik dinilai memudahkan pengguna dalam mengakses informasi dan fitur yang tersedia, dengan 85% responden memberikan respons positif (Sangat Mudah dan Mudah). Tampilan aplikasi dianggap membantu pengguna dalam memahami informasi, dan mayoritas responden merasa bahwa informasi serta fitur yang disediakan sesuai dengan kebutuhan mereka. Informasi mengenai hidroponik, termasuk gambar, teks, audio, dan video, dinilai cukup

mudah dipahami. Meskipun demikian, terdapat ruang untuk perbaikan, khususnya pada fitur kamera pendeteksi hama dan penyakit. Panduan langkah-langkah bercocok tanam dan fitur edukasi lainnya terbukti efektif dalam meningkatkan pengetahuan pengguna. Secara keseluruhan, aplikasi ini telah memenuhi ekspektasi pengguna, meskipun diperlukan optimalisasi lebih lanjut pada fitur tertentu untuk meningkatkan pengalaman penggunaan secara keseluruhan.

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan penelitian, penulis dapat mengambil kesimpulan sebagai berikut:

1. Aplikasi panduan lengkap hidroponik yang telah dirancang berhasil dibangun dengan menggunakan Android studio bahasa pemograman *Java*. Aplikasi ini dilengkapi dengan berbagai fitur yang dirancang untuk mempermudah pengguna dalam memahami dan menerapkan teknik bercocok tanam hidroponik. Fitur-fitur yang disediakan dalam aplikasi ini, seperti pengenalan hidroponik, media tanam dan pupuk hidroponik, teknik penanaman hidroponik, penyakit dan hama tanaman hidroponik, video budidaya hidroponik, dan kamera pendeteksi hama dan penyakit hidroponik.
2. Fitur kamera pendeteksi hama dan penyakit berhasil dibangun menggunakan *Web Teachable Machine* untuk melatih model yang kemudian diubah menjadi *TensorFlow Lite*. Model tersebut diintegrasikan ke dalam aplikasi Android yang dikembangkan dengan menggunakan bahasa pemrograman *Java* dan *platform* Android Studio.
3. Pada pengujian *Blackbox* dan *Whitebox* memastikan struktur internal kode dan logika berjalan sesuai dengan yang dirancang. Seluruh tombol yang ada pada aplikasi panduan lengkap hidroponik dapat berjalan seluruhnya tanpa ada eror.

4. Pada pengujian akurasi aplikasi menunjukkan performa yang baik dengan rata-rata akurasi keseluruhan sebesar 87%. Hasil pengujian menunjukkan bahwa deteksi hama ulat kubis (*Plutella xylostella*) memiliki akurasi tertinggi sebesar 95%, diikuti oleh deteksi penyakit busuk daun dengan akurasi 90%. Deteksi hama serangga dan burung serta penyakit kelainan genetik sama-sama mencapai akurasi sebesar 85%. Namun, deteksi hama thrips memiliki akurasi terendah, yaitu 80%, terutama pada sudut tertentu. Meskipun secara keseluruhan sistem menunjukkan kinerja yang baik, diperlukan peningkatan akurasi pada kategori dengan hasil lebih rendah, khususnya untuk deteksi hama thrips, agar mampu memberikan hasil yang optimal di berbagai sudut pengambilan foto.
5. Pada pengujian tingkat kepuasan pengguna aplikasi hidroponik terdapat empat kategori penilaian, yaitu Sangat Baik, Baik, Cukup, dan Kurang. Persentase penilaian Sangat Baik 16,3%, Baik 68,7%, Cukup 14,3%, dan Kurang 0,7%. Secara keseluruhan, aplikasi panduan hidroponik dinilai memudahkan pengguna dalam mengakses informasi dan fitur yang tersedia, dengan 85% responden memberikan respons positif (Sangat Mudah dan Mudah).

B. Saran

Penulis penelitian ini mengakui bahwa masih banyak permasalahan yang memerlukan pengembangan dan perbaikan dalam penelitian lebih lanjut. Oleh karena itu, penulis menawarkan rekomendasi berikut untuk pengembangan studi lebih lanjut sebagai berikut:

1. Pengembangan dapat dilakukan dengan menggunakan database untuk memperkaya informasi dan mengurangi volume kodingan serta mengurangi penyimpanan aplikasi.
2. Aplikasi ini disarankan lebih banyak gambar penyakit dan hama tanaman hidroponik untuk menu kamera pendeteksi penyakit tanaman agar lebih akurat saat dideteksi.

DAFTAR PUSTAKA

- Asep Deddy Supriatna, Dini Destiani Siti Fatimah, Rexyco Geovani. (2022). *Perancangan Aplikasi Edukasi Cara Bercocok Tanam dengan Menggunakan Teknik Hidroponik Berbasis Web*. Jurnal Algoritma. 19(1). 140-148.
- Andre Setiawan. (2019). *Buku Pintar Hidroponik Penulis*. Yogyakarta : Laksana.
- Chalifa Chazar , Muhammad Helmi Rafsanjani. (2022). *Penerapan Teachable Machine Pada Klasifikasi Machine Learning Untuk Identifikasi Bibit Tanaman*. Jurnal Seminar Nasional. 32-40.
- Chan, F. R., Dusri, H., Ramadani, M., & Efriyanti, L. (2022). Perancangan Aplikasi Pengelolaan Gudang Berbasis Android Menggunakan Android Studio. *Journal of Informatics and Advanced Computing (JIAC)*, 3(2), 103-107.
- Daniel Novianto. (2018). *Perancangan Aplikasi Android Untuk Media Pembelajaran Budidaya Tanaman Secara Hidroponik Dengan Metode Demon-Disco Learning*. Jurnal Elektronik dan Komputer. 11(1). 1-8.
- Firman,Fauziah, Ratih Titi Komalasari. (2021). *Aplikasi Pengingat Dan Pendataan Kenaikan Golongan Gaji Berbasis Web Menggunakan Metode White Box Testing dan Black Box Testing*. Jurnal Teknologi Informasi, 7(1). 50-57.
- Hendro Wibowo, (2019). *Panduan Terlengkap Hidroponik*. Diedit oleh Farel Kurniawan. Yogyakarta: Laksana.
- Umi Fadillah Umar,Yudhan Nur Akhmadi, (2016). *Jago Bertanam Hidroponik Untuk Pemula*. Penerbit oleh PT AgroMedia Pustaka. Jakarta.
- Irvansyah, F., Setiawansyah, S., & Muhaqiqin, M. (2020). *Aplikasi Pemesanan Jasa Cukur Rambut Berbasis Android*. Jurnal Ilmiah Infrastruktur Teknologi Informasi, 1(1), 26-32.
- Indra Fahrudin Rukmana. (2018). *Aplikasi Media Pembelajaran Penanaman Sayuran Hidroponik*. Jurnal Mahasiswa Teknik Informatika. 2(1). 36-45.
- Ilham Budiman, Sopyan Saori, Ramdan Nurul Anwar, Fitriani, Muhamamd Yuga Pangestu, (2021), *Analisis Pengendalian Mutu Di Bidang Industri Makanan*, Jurnal Inovasi Penelitian, 1(10), 2185-2190.

- Jerrycho Azarea Aditama, Jati Sasongko Wibowo. (2023). *Aplikasi Media Pembelajaran Menanam Sayuran Hidroponik Berbasis Android*. Jurnal Elektronika dan Komputer. 16(1). 182-189.
- Joelyan Vicky, Frisca Ayu, Bagas Julianto. (2023). *Implementasi Pendeteksi Penyakit pada Daun Alpukat Menggunakan Metode CNN*. Prosiding Seminar Nasional Teknologi Dan Sains. 2. 155-162.
- Judith Bryan L Sie, Izmy Alwiah Musdar, Syamsul Bahri. (2022). *Pengujian White Box Testing Terhadap Website Room Menggunakan Teknik Basis Path*. Jurnal Ilmu Komputer. 17(02). 45-57.
- Kadek Teguh Purwanto, Gede Sukadarmika, D.M.Wiharta. (2021). *Rancang Bangun Hidroponik Tanaman Paprika Dengan Drip System Berbasis Internet Of Things*. Jurnal SPEKTRUM. 8(4). 193-199.
- Muh. Basri, Muhammad Zainal 2, Dedy. (2021). *Aplikasi Panduan Budidaya Tanaman Bawang Merah Teknik Hidroponik Berbasis Web Responsive*. Jurnal sintaks Logika. 1(1). 1-11.
- Muhammad Khaisar Wirawan, Syalam Ali Wira Dinata, Muliady Faisal. (2022). *Buku Monitoring Pembudidayaan Tanaman Hidroponik*. Yogyakarta: Deeplish.
- Nurhidayati, Amri Muliawan Nur. (2021). *Pemanfaatan Aplikasi Android Dalam Rancang Bangun Sistem Informasi Persebaran Indekos di Wilayah Pancor Kabupaten Lombok Timur*. Jurnal Informatika dan Teknologi. 4(1). 51-62.
- Recky Arianto. (2023). *Perancangan Aplikasi Pengenalan dan Pembelajaran Pertanian Milenial Hidroponik Berbasis Android Flutter & Dart Dengan Metode RAD*. Jurnal Ilmu Komputer dan Science. 2(3). 928-935.
- Rahmadya Trias. (2023). *Cara Menggunakan Teachable Machine*. Diperoleh dari: <https://www.youtube.com/watch?v=LPJZXSX8Ng0>. (Diakses 3 Mei 2024).
- Wahyu Nur Cholifah, Yulianingsih Yulianingsih, Sri Melati Sagita. (2018). *Pengujian black box testing pada aplikasi action & strategy berbasis android dengan teknologi phonegap*. STRING (Satuan Tulisan Riset dan Inovasi Teknologi), 3(2), 206-210.
- Yuyun Hana Natbais, Ari Bangkit Sanjaya Umbu. (2023). *Aplikasi Deteksi Penyakit pada Daun Tomat Berbasis Android Menggunakan Model Terlatih Tensorflow Lite*. Teknotan. 17(2). 83-90.

