

TUGAS AKHIR
IMPLEMENTASI METODE CERTAINTY FACTOR UNTUK MENILAI
KELAYAKAN BUDIDAYA PADI SALIBU BERBASIS ANDROID

Diajukan Sebagai Salah Satu Syarat untuk Menyelesaikan Program Strata-1
pada Jurusan Teknik Elektro Fakultas Teknik Universitas Andalas

Oleh :

RIFKY SYAFRIZAL

No. BP. 1110952010



JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS ANDALAS
PADANG

2016

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Saya mahasiswa/dosen/tenaga kependidikan* Universitas Andalas yang bertanda tangan di bawah ini:

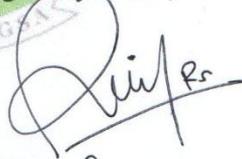
Nama lengkap : RIFKY SYAFRIZAL
No. BP/NIM/NIDN : 1110952010
Program Studi : TEKNIK ELEKTRO
Fakultas : TEKNIK
Jenis Tugas Akhir : ~~TA/Disertasi~~/Skripsi/Tesis/Disertasi/.....**

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Andalas hak atas publikasi *online* Tugas Akhir saya yang berjudul:

IMPLEMENTASI METODE CERTAINTY FACTOR UNTUK MENILAI
KELAYAKAN BUDIDAYA PADI SALIBU BERBASIS ANDROID

beserta perangkat yang ada (jika diperlukan). Universitas Andalas juga berhak untuk menyimpan, mengalihmedia/formatkan, mengelola, merawat, dan mempublikasikan karya saya tersebut di atas selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di PADANG
Pada tanggal 26 OKTOBER 2016
Yang menyatakan,


(RIFKY SYAFRIZAL)

* pilih sesuai kondisi

** termasuk laporan penelitian, laporan pengabdian masyarakat, laporan magang, dll

LEMBAR PENGESAHAN

**IMPLEMENTASI METODE CERTAINTY FACTOR UNTUK MENILAI
KELAYAKAN BUDIDAYA PADI SALIBU BERBASIS ANDROID**

Lulus Sidang Tugas Akhir Tanggal : 20 Oktober 2016

Disetujui Oleh :

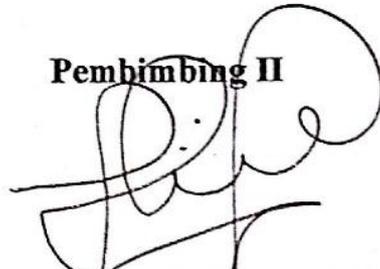
Pembimbing I



Fitrilina, MT

NIP. 19810825 200604 2 001

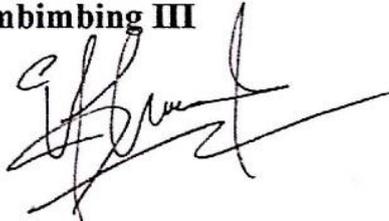
Pembimbing II



Heru Dibyo Laksono, MT

NIP. 19770107 200501 1 002

Pembimbing III



Ir. Erdiman

NIP. 19581009 199103 1 002

Mengetahui :

**Ketua Jurusan Teknik Elektro
Fakultas Teknik
Universitas Andalas**



Dr. Eng. Ariadi Hazmi

NIP. 19750314 199903 1 003

KATA PENGANTAR



Puji dan syukur penulis ucapkan kehadiran Allah SWT Yang Maha Pengasih lagi Maha Penyayang, yang telah melimpahkan rahmat dan karuniaNya sehingga penulis dapat menyelesaikan tugas akhir ini yang berjudul **“Implementasi Metode *Certainty Factor* Untuk Menilai Kelayakan Budidaya Padi Salibu Berbasis Android”**. Shalawat dan salam diucapkan kepada Nabi Muhammad SAW yang telah memberi tuntunan dan pencerahan kepada umat manusia.

Tugas akhir ini merupakan salah satu syarat untuk menyelesaikan program Strata-1 (S1) pada Jurusan Teknik Elektro Fakultas Teknik Universitas Andalas Padang.

Pelaksanaan tugas akhir serta penyelesaian laporan akhir ini tidak terlepas dari izin dan rahmat Allah SWT sehingga penulis mampu memahami materi yang diberikan. Disamping itu penulis juga mendapatkan bimbingan, motivasi, masukan serta bantuan dari berbagai pihak. Untuk itu pada kesempatan ini penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada :

1. Kedua orang tua atas kasih sayang, perhatian, doa serta dukungan dan semangat yang telah diberikan kepada penulis.
2. Ibu Fitrilina, MT dan Bapak Heru Dibyo Laksono, MT selaku pembimbing tugas akhir yang telah banyak memberikan arahan, bimbingan dan masukan dalam penyelesaian tugas akhir ini.

3. Bapak Dr. Eng, Ariadi Hazmi selaku ketua Jurusan Teknik Elektro Fakultas Teknik Universitas Andalas Padang.
4. Bapak Ir. Erdiman selaku peneliti & penemu teknologi Budidaya Padi Salibu yang telah banyak membantu dan memberi arahan dalam pembuatan aplikasi yang dirancang penulis.
5. Rekan-rekan Processor 2011 yang selalu ada untuk mengingatkan, memberi masukan serta memberikan semangat yang luar biasa sehingga penulis dapat terus mengerjakan & menyelesaikan tugas akhir ini.
6. Adik-adik Thyristor 2013 dan Elco 2014 yang menemani serta mendukung penulis dalam menyelesaikan tugas akhir ini, terutama pada Fadli Aulia Rahmat yang laptopnya selalu menjadi penunjang paling penting bagi penulis.
7. Semua pihak yang tidak dapat penulis sebutkan satu persatu yang telah membantu dalam penyelesaian tugas akhir ini.

Penulis menyadari sepenuhnya dalam penyusunan laporan tugas akhir ini masih terdapat kekurangan dan jauh dari kesempurnaan. Untuk itu penulis mengharapkan saran dan kritik yang sifatnya membangun dalam penyempurnaan laporan tugas akhir ini. Akhir kata penulis berharap semoga laporan tugas akhir ini dapat bermanfaat bagi semua pihak, baik bagi diri penulis pribadi maupun pembaca.

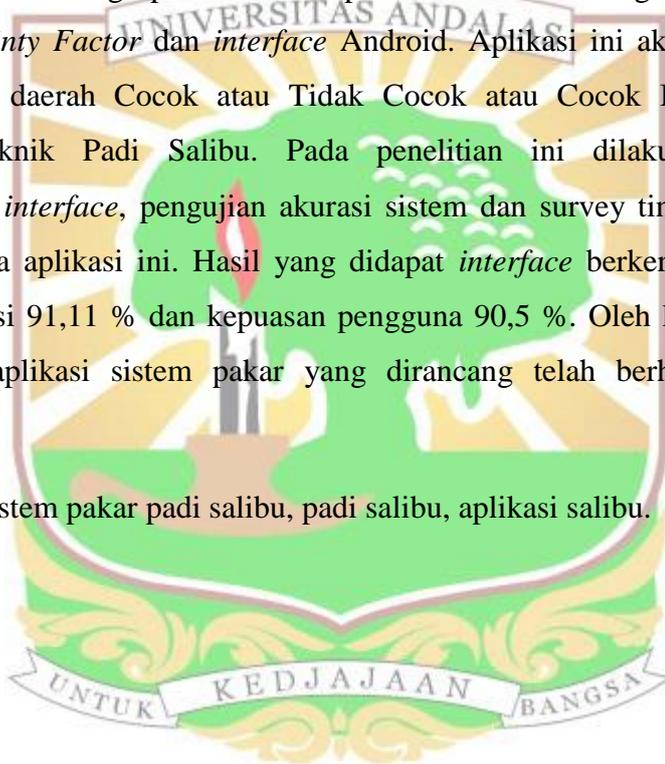
Padang, Oktober 2016

Penulis

ABSTRAK

Produktivitas padi nasional yang belum mencapai target masih menjadi permasalahan. Salah satu solusi meningkatkan produktivitas padi adalah dengan mengembangkan teknik Budidaya Padi Salibu. Teknik budidaya ini cukup menjanjikan tetapi tidak semua daerah dapat menerapkan teknik ini. Kendala lain adalah tenaga ahli yang masih kurang dan belum tersebar. Oleh karena itu pada penelitian ini dirancang aplikasi sistem pakar Padi salibu dengan menggunakan metode *Certainty Factor* dan *interface* Android. Aplikasi ini akan memutuskan kondisi suatu daerah Cocok atau Tidak Cocok atau Cocok Bersyarat untuk diterapkan teknik Padi Salibu. Pada penelitian ini dilakukan pengujian fungsionalitas *interface*, pengujian akurasi sistem dan survey tingkat kepuasan pengguna pada aplikasi ini. Hasil yang didapat *interface* berkerja dengan baik. Tingkat akurasi 91,11 % dan kepuasan pengguna 90,5 %. Oleh karena itu dapat disimpulkan aplikasi sistem pakar yang dirancang telah berhasil dan dapat digunakan.

Kata kunci : sistem pakar padi salibu, padi salibu, aplikasi salibu.

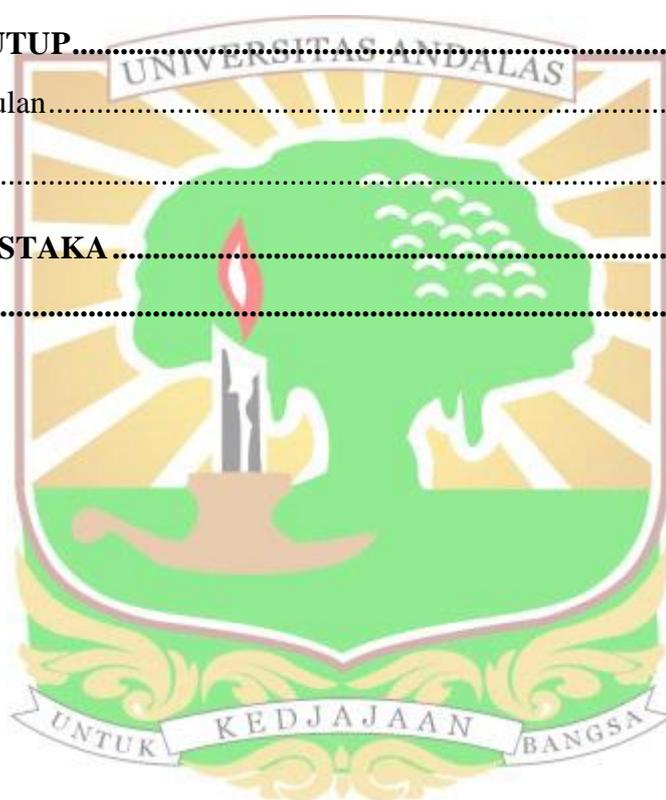


DAFTAR ISI

SCAN PERNYATAAN PUBLIKASI	Error! Bookmark not defined.
KATA PENGANTAR	iii
DAFTAR ISI	vi
DAFTAR GAMBAR	ix
DAFTAR TABEL	x
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian.....	5
1.5 Manfaat Penelitian.....	5
1.6 Sistematika Penulisan.....	5
BAB II TINJAUAN PUSTAKA	7
2.1 Sistem Pakar	7
2.2 Metode <i>Certainty Factor</i> (CF)	7
2.2.1 Model Untuk Menghitung Tingkat CF dari Sebuah Rule	8
2.2.2 Aturan Kombinasi Metode <i>Certainty Factor</i>	10
2.2.3 Langkah Perhitungan <i>Certainty Factor</i>	11
2.3 Padi Salibu.....	12
2.3.1 Prasyarat Budidaya Padi Salibu	14
2.3.2 Agroekosistem Lahan Padi Salibu	14
2.4 Android.....	18
2.4.1 Versi Android	19
2.4.2 Kelebihan & Kekurangan Android.....	19

2.4.3 Fitur-fitur pada Android	20
2.4.4 Android Studio.....	21
2.4.4.1 JDK (Java Development Kit).....	23
2.4.4.2 Android SDK (Software Development Kit).....	23
2.4.4.3 Android NDK (Native Development Kit).....	23
2.4.4.4 AVD (Android Virtual Device)	23
2.5 Skala Likert	24
BAB III PERANCANGAN DAN PEMBUATAN APLIKASI.....	28
3.1 Jenis Penelitian	28
3.2 Akuisi Data dan Sampel Penelitian	28
3.3 Perancangan Sistem.....	28
3.3.1 Pengumpulan Parameter	29
3.3.2 Pengumpulan Data Pakar.....	30
3.3.3 Perancangan Sistem Pakar Metode <i>Certainty Factor</i>	32
3.3.3.1 Penentuan Nilai CF Pakar dan CF User.....	32
3.3.3.2 Perhitungan <i>Certainty Factor</i>	33
3.3.3.3 Penetapan Rule-Rule Keputusan.....	34
3.3.3.4 Pengujian Data	34
3.4 Contoh Kasus Perhitungan <i>Certainty Factor</i>	35
3.5 Pembuatan Aplikasi <i>Interface Smartphone</i> Android.....	37
3.6 Pengujian Fungsional & Tampilan Aplikasi	38
BAB IV HASIL DAN ANALISA	40
4.1 Hasil Pengumpulan Data Pakar	40
4.2 Hasil Pengujian & Pelatihan Data	41
4.3 Hasil Pengujian Aplikasi Sistem Pakar Metode <i>Certainty Factor</i>	42

4.3.1 Hasil Pengujian Fungsionalitas Aplikasi (<i>Black Box Test</i>)	43
4.3.2 Hasil Pengujian Akurasi Sistem (<i>White Box Test</i>)	44
4.3.3 Hasil Pengolahan Keseluruhan Data Padi Salibu	51
4.4 Hasil Pengujian Fungsionalitas & Tampilan Aplikasi	54
4.4.1 Hasil Kepuasan Fungsionalitas.....	54
4.4.2 Hasil Kepuasan Tampilan.....	58
4.5 Hasil Penggunaan Pada Sistem Android	62
BAB V PENUTUP.....	63
5.1 Kesimpulan.....	63
5.2 Saran.....	63
DAFTAR PUSTAKA	65
LAMPIRAN.....	67



DAFTAR GAMBAR

Gambar 2.1 Perbandingan Teknik Salibu (kiri) dan Teknik Raton (kanan) ^[7]	13
Gambar 2.2 Penggenangan Lahan Sawah ^[7]	15
Gambar 2.3 Pemotongan Ulang Tunggul Sisa Panen ^[7]	16
Gambar 2.4 Hasil Padi Salibu ^[7]	18
Gambar 2.5 Logo Android ^[8]	18
Gambar 2.6 Tampilan Android Studio	22
Gambar 2.7 Tampilan AVD ^[18]	24
Gambar 3.1 Diagram Alir Perancangan Sistem	29
Gambar 3.2 Kuesioner Pembobotan Parameter	31
Gambar 3.3 Tampilan Aplikasi Android	38
Gambar 3.4 Tampilan Kuesioner	39
Gambar 4.1 Tampilan Hasil Program : (a) Hasil yang sama dengan pakar, dan (b) Hasil yang berbeda dengan pakar.	50
Gambar 4.2 Tampilan Aplikasi Keseluruhan	61
Gambar 4.3 Hasil Penggunaan Aplikasi : (a) Memori yang digunakan, (b) Ram yang digunakan, (c) Ukuran file	62



DAFTAR TABEL

Tabel 2.1 Nilai Interpretasi CF.....	10
Tabel 2.2 Aturan-aturan McAllister.....	11
Tabel 2.3 Penentuan Skor Jawaban.....	25
Tabel 3.1 Parameter Padi Salibu.....	30
Tabel 4.1 Data Parameter Pakar.....	40
Tabel 4.2 Pengujian fungsional berupa <i>Input – Output</i> Aplikasi.....	44
Tabel 4.3 Kondisi Lahan Beberapa Wilayah di Indonesia.....	45
Tabel 4.4 Perbandingan Hasil Keputusan.....	47
Tabel 4.5 Data Wilayah Oleh Pakar.....	51
Tabel 4.6 Hasil Perbandingan Data.....	52
Tabel 4.7 Hasil Kuesioner Fungsionalitas Tani.....	54
Tabel 4.8 Hasil Kuesioner Fungsionalitas Ahli.....	56
Tabel 4.9 Hasil Kuesioner Tampilan Tani.....	59
Tabel 4.10 Hasil Kuesioner Tampilan Ahli.....	60



BAB I

PENDAHULUAN

1.1 Latar Belakang

Padi (*Oryza Sativa* L.) merupakan komoditas tanaman paling penting di Indonesia. Hal ini dikarenakan padi menghasilkan beras yang merupakan makanan pokok bangsa Indonesia sehingga tanaman tersebut menjadi salah satu bidang pertanian yang digalakkan hampir di setiap wilayah Indonesia. Rata-rata peningkatan produksi padi nasional beberapa tahun terakhir masih rendah, yaitu sekitar 6,37 persen per tahun. Berdasarkan data Badan Pusat Statistik, produksi padi nasional tahun 2015 diperkirakan meningkat hingga 3,97 % dan diprediksi mencapai 75.36 juta ton gabah kering giling (GKG), atau setara dengan beras sebanyak 42,36 juta ton berdasarkan perhitungan KTNA (Kelompok Tani Nelayan Andalan). Berdasarkan angka tetap tahun 2015, produktivitas padi nasional sebesar 5,33 t/ha GKG^[1]. Padahal dengan laju pertumbuhan penduduk yang mencapai 1.49% dan laju konsumsi beras nasional 1.34% per tahun, rata-rata produktivitas padi nasional seharusnya minimal 6.0 t/ha.^[3]

Upaya meningkatkan produktivitas padi terus dilakukan, salah satunya adalah dengan memanfaatkan Padi Salibu. Padi Salibu merupakan tanaman padi yang tumbuh lagi setelah batang sisa panen ditebas/dipangkas, tunas akan muncul dari buku yang ada didalam tanah. Budidaya Padi Salibu cukup menjanjikan, terlihat dari hasil yang telah didapatkan oleh petani di Kabupaten Agam tahun 2011, yaitu sekitar 20% lebih tinggi dibanding dari panen pertama. Komponen hasil Padi Salibu dengan varietas lokal di Matur, Kabupaten Agam adalah 7,2 ton

per hektar dengan tinggi tanaman 102 cm, jumlah anakan 22 batang, panjang malai 24 cm, jumlah bulir per malai 120 buah dan bulir hampa hanya 17%.^[2]

Beberapa keuntungan yang dapat diperoleh dari penerapan Padi Salibu antara lain : biaya produksi lebih rendah karena tidak perlu pengolahan tanah dan penanaman ulang, pupuk yang dibutuhkan lebih sedikit yaitu setengah dari dosis yang diberikan pada tanaman utama, umur panen lebih pendek, dan hasil yang diperoleh dapat memberikan tambahan produksi dan meningkatkan produktivitas.^[4] Oleh karena itu penerapan Padi Salibu memberi keuntungan lebih cepat, mudah dan murah serta dapat meningkatkan produktivitas padi per unit area dan per unit waktu. Sehingga penerapan budidaya padi dengan sistem Padi Salibu diharapkan dapat memberi andil dalam meningkatkan produktivitas padi nasional.^[2]

Dibalik keuntungan yang begitu besar pada penerapan Padi Salibu, ternyata masih terdapat beberapa kendala pada penerapan teknik budidaya padi ini. Adapun diantaranya yaitu kurangnya pengetahuan kelompok tani tentang bagaimana teknik budidaya Padi salibu, serta tenaga ahli yang masih sedikit dan belum terdapat di seluruh daerah di Indonesia. Penerapan teknik padi Salibu memiliki banyak parameter yang harus diperhatikan seperti kondisi tanah, kondisi pengairan, iklim daerah dan parameter-parameter lain. Hal ini membutuhkan analisa dari tenaga pakar untuk memutuskan apakah suatu daerah cocok atau tidak cocok atau cocok bersyarat untuk menggunakan teknik ini. Oleh karena itu, dirancanglah suatu sistem pakar yang dapat mengatasi kendala tersebut. Sistem pakar ini nantinya akan memberikan informasi pada suatu kelompok tani mengenai teknik padi salibu dan kelayakan suatu daerah dalam menerapkan

teknik tersebut. Informasi ini dapat menjadi rujukan bagi tim penyuluh atau kelompok tani untuk melakukan observasi di daerah tersebut tanpa harus bertemu langsung dengan para pakar atau ahli dalam budidaya Padi Salibu.

Sistem pakar ini akan menggunakan metode *Certainty Factor*, hal ini dikarenakan dalam prosesnya metode ini hanya dapat mengolah 2 data saja, sehingga keakuratan data dapat terjaga. Selain itu metode ini tidak begitu sulit untuk dipahami sehingga dirasa cocok digunakan pada sistem pakar ini.

Terdapat tulisan yang membahas mengenai Padi Salibu & Sistem Pakar, yaitu oleh Erdiman (Balai Pengkajian Teknologi Pertanian Sumatera Barat) dengan judul “*Teknologi Salibu Meningkatkan Produktivitas Lahan (3-6 Ton/ Ha/ Tahun) dan Pendapatan Petani (Rp.15-25 Juta/ Tahun)*”. Penelitian yang dilakukan oleh Erdiman ini yaitu mengenai penelitian Padi Salibu, hasil serta keuntungan yang didapatkan dengan menerapkan budidaya Padi Salibu.^[2]

Pada tulisan kedua, oleh Gede Mahardika yang berjudul “*Pengembangan Sistem Pakar Diagnosa Hama dan Penyakit Tanaman Padi Menggunakan Metode Certainty Factor Berbantuan Metode Kuantifikasi Pertanyaan*”. Penelitian yang dilakukan Gede Mahardika ini membahas mengenai pengembangan suatu sistem pakar untuk memudahkan mendiagnosa hama dan penyakit pada tanaman padi
[10].

Berdasarkan penjelasan diatas maka dirancanglah suatu sistem pakar menggunakan metode *Certainty Factor* untuk menilai kelayakan lahan suatu daerah untuk diterapkan budidaya padi salibu yang berbasis Android. Hal yang membedakan penelitian ini dengan penelitian yang telah ada sebelumnya yaitu penggunaan Android sebagai *interfacenya*. Kelebihan android ini yaitu

memudahkan mobilitas dari penggunanya, sehingga para pengguna akan dapat lebih mudah dan nyaman dalam mengakses aplikasi tersebut. Berdasarkan pemikiran-pemikiran tersebut, maka proposal penelitian ini diberi judul **“Implementasi Metode Certainty Factor Untuk Menilai Kelayakan Budidaya Padi Salibu Berbasis Android”**.

1.2 Rumusan Masalah

Bagaimana merancang suatu sistem pakar dengan menggunakan metoda *Certainty Factor* untuk menilai kelayakan suatu lahan untuk diterapkan budidaya Padi Salibu berbasis Android, yang dapat memberikan keakuratan hasil dan kemudahan bagi pengguna (tim peneliti, PPL dan kelompok tani) sehingga dapat menjadi solusi alternatif dalam mengatasi kurangnya tenaga pakar padi Salibu.

1.3 Batasan Masalah

Dalam pembuatan tugas akhir ini, masalah dibatasi oleh :

1. Merancang sistem pakar untuk menilai kelayakan budidaya Padi Salibu untuk suatu lahan.
2. Metode sistem pakar yang digunakan adalah metode *Certainty Factor*.
3. Menentukan nilai *Certainty Factor* dengan cara wawancara pakar.
4. Menilai kelayakan Budidaya Padi Salibu berbasis Android.
5. Kelayakan yang dinilai adalah “Cocok”, “Tidak Cocok” dan “Cocok bersyarat”.
6. Versi minimal android yang dapat digunakan adalah versi 4.1 (Jelly Bean).
7. Menggunakan *software* Android Studio 1.5.

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk :

1. Merancang suatu sistem yang dapat menilai kelayakan suatu daerah untuk dilakukan Budidaya Padi Salibu dan diimplementasikan di Android.
2. Menganalisa kinerja sistem yang telah dirancang.

1.5 Manfaat Penelitian

Manfaat dari melakukan penelitian ini adalah :

1. Mempermudah kelompok tani dan peneliti untuk menentukan apakah suatu daerah cocok atau tidak cocok atau cocok bersyarat dalam menerapkan teknik budidaya salibu.
2. Menjadi solusi alternatif dalam mengatasi masalah kurangnya tenaga pakar Padi Salibu.
3. Memberikan aplikasi sistem pakar kelayakan lahan untuk diterapkannya Budidaya Padi Salibu dengan interface yang baik, sehingga mudah dan nyaman digunakan.
4. Dapat mengembangkan aplikasi sistem pakar certainty Factor dan menjadi penunjang materi suatu mata kuliah seperti Sistem Cerdas.

1.6 Sistematika Penulisan

Sistematika penulisan Tugas Akhir ini adalah sebagai berikut :

1. BAB I : Pendahuluan

Bab ini berisi latar belakang, tujuan, manfaat, batasan masalah, dan sistematika penulisan.

2. **BAB II : Tinjauan Pustaka**

Bab ini membahas tentang teori dan bahasan lainnya yang berhubungan dengan tugas akhir ini sebagai dasar teori pendukung yang digunakan dalam perencanaan sistem nantinya.

3. **BAB III : Metodologi Penelitian**

Berisi metoda serta cara-cara yang penulis pergunakan dalam menulis tugas akhir ini.

4. **BAB IV : Perancangan Sistem**

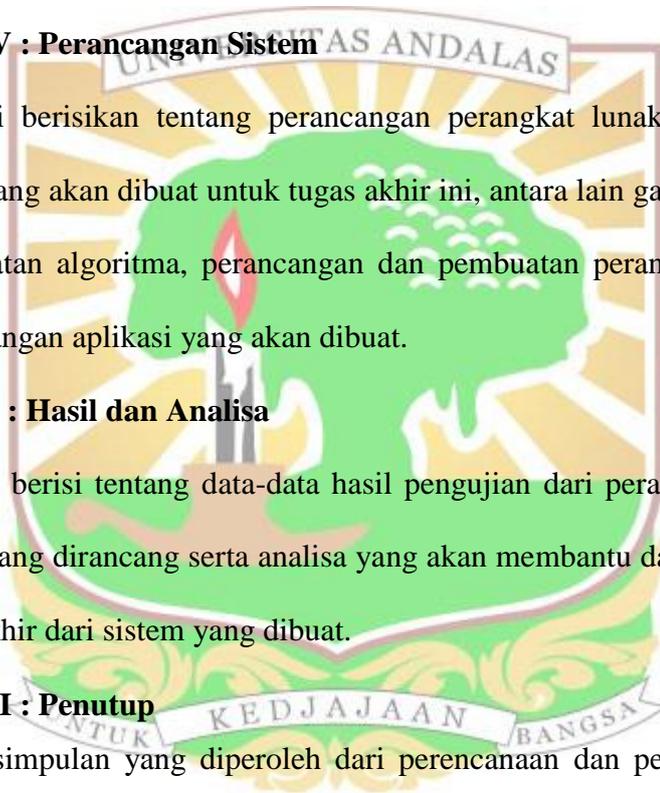
Bab ini berisikan tentang perancangan perangkat lunak dan perangkat keras yang akan dibuat untuk tugas akhir ini, antara lain garis besar sistem, pembuatan algoritma, perancangan dan pembuatan perangkat keras, dan perancangan aplikasi yang akan dibuat.

5. **BAB V : Hasil dan Analisa**

Bab ini berisi tentang data-data hasil pengujian dari perangkat keras dan lunak yang dirancang serta analisa yang akan membantu dalam memahami hasil akhir dari sistem yang dibuat.

6. **BAB VI : Penutup**

Berisi simpulan yang diperoleh dari perencanaan dan pembuatan sistem pada tugas akhir. Selain itu, bab ini juga berisikan saran-saran yang dapat digunakan untuk penyempurnaan tugas akhir ini kedepannya.



BAB II

TINJAUAN PUSTAKA

2.1 Sistem Pakar

Sistem pakar adalah salah satu cabang dari *Artificial Intelligence* (AI). Prinsip Sistem pakar adalah sebuah perangkat lunak komputer yang memiliki basis pengetahuan untuk domain tertentu dan menggunakan penalaran inferensi menyerupai seorang pakar dalam memecahkan masalah^[9]. Seorang pakar adalah orang yang memiliki keahlian dibidang tertentu, yaitu pakar yang mempunyai *knowledge* atau kemampuan khusus yang orang lain tidak mengetahui atau mampu dalam bidang yang dimilikinya.^[10]

Bagian dari sistem pakar terdiri dari 2 komponen utama, yaitu *Knowledge-Base* yang berisi *knowledge* dan mesin inferensi yang menggambarkan kesimpulan. Kesimpulan tersebut merupakan respon dari sistem pakar atas permintaan pengguna.^[10]

2.2 Metode *Certainty Factor* (CF)

Certainty Factor diperkenalkan oleh Shortliffe Buchanan dalam pembuatan sistem pakar MYCIN pada tahun 1975 untuk mengakomodasi ketidakpastian pemikiran (*inexact reasoning*) seorang pakar^[12]. Teori ini berkembang bersamaan dengan pembuatan sistem pakar MYCIN. Team pengembang MYCIN mencatat bahwa dokter sering kali menganalisa informasi yang ada dengan ungkapan seperti misalnya : “mungkin, kemungkinan besar, hampir pasti”. Untuk mengakomodasi hal ini, tim MYCIN menggunakan *Certainty Factor* (CF) guna menggambarkan tingkat keyakinan pakar terhadap

permasalahan yang sedang dihadapi. Secara umum, rule direpresentasikan dalam bentuk sebagai berikut ^[12]:

IF E1 [AND / OR] E2 [AND / OR] ... En
THEN H (CF = CFi)

dimana:

E1 ... En : fakta – fakta (*evidence*) yang ada.

H : hipotesa atau konklusi yang dihasilkan.

CF : tingkat keyakinan (*Cerainty Factor*) terjadinya hipotesa H akibat adanya fakta– fakta E1 s/dEn .

2.2.1 Model Untuk Menghitung Tingkat CF dari Sebuah Rule

Sampai saat ini ada dua model yang sering digunakan untuk menghitung tingkat keyakinan (CF) dari sebuah rule adalah sebagai berikut^[5]:

1. Menggunakan metode *Belief Disbelief* yang diusulkan oleh E. H. Shortliffe dan B. G. Buchanan, yaitu^[6]:

$$CF(H,E) = MB(H, E) - MD(H, E) \dots\dots\dots (1)$$

$$MB(H | E) = \begin{cases} 1 & ; \text{if } P(H) = 1 \\ \frac{[P(H | E)] - P(H)}{1 - P(H)} & ; \text{otherwise} \end{cases} \dots\dots\dots (2)$$

$$MD(H | E) = \begin{cases} 1 & ; \text{if } P(H) = 0 \\ \frac{P(H) - [P(H|E)]}{P(H)} & ; \text{otherwise} \end{cases} \dots\dots\dots(3)$$

dimana:

CF(H, E) = CF dari hipotesis H yang dipengaruhi oleh gejala (*evidence*) E. Besarnya CF antara -1 sampai dengan 1.

Nilai -1 menunjukkan ketidakpercayaan mutlak sedangkan nilai 1 menunjukkan kepercayaan mutlak.

$MB(H, E)$ = ukuran kenaikan kepercayaan (*measure of increased belief*) terhadap hipotesis H yang dipengaruhi oleh gejala E.

$MD(H, E)$ = ukuran kenaikan ketidakpercayaan (*measure of increased disbelief*) terhadap hipotesis H yang dipengaruhi oleh

gejala E.

$P(H)$ = probabilitas kebenaran hipotesa H.

$P(H|E)$ = probabilitas bahwa H benar karena fakta E.

$P(H)$ dan $P(H|E)$ merepresentasikan keyakinan dan ketidakyakinan pakar.

E (Evidence) = Fakta / Gejala pada pasien.

H (Hipotesa) = Jawaban sementara dari suatu masalah yang masih bersifat praduga.

Selanjutnya untuk menentukan nilai gabungan *Certainty Factor* maka digunakanlah aturan-aturan gabungan metode *Certainty Factor*.

2. Menggali informasi dari hasil wawancara dengan pakar. Nilai CF didapat dari *Qualitatif Term* yang diberikan pakar sehingga menjadi nilai CF tertentu. Nilai *Qualitatif Term* dan nilai *Certainty Factornya* dapat dilihat pada Tabel 2.1.

Tabel 2.1 Nilai Interpretasi CF^[6]

<i>Qualitatif Term</i>	<i>Certainty Factor</i>
<i>Definitely not</i>	- 1.0
<i>Almost certainly not</i>	- 0.8
<i>Probably not</i>	- 0.6
<i>Maybe not</i>	- 0.4
<i>Unknown</i>	- 0.2 to 0.2
<i>Maybe</i>	0.4
<i>Probably</i>	0.6
<i>Almost certainly</i>	0.8
<i>Definitely</i>	1.0

Nilai *Certainty Factor* (CF) yang didapatkan dari hasil wawancara dengan pakar, selanjutnya akan digunakan sebagai nilai CF(H).

2.2.2 Aturan Kombinasi Metode *Certainty Factor*

Defenisi *Certainty Factor* menurut David McAllister adalah suatu metode untuk membuktikan apakah suatu fakta itu pasti ataukah tidak pasti yang berbentuk *metric* yang biasanya digunakan dalam sistem pakar. Metode ini sangat cocok untuk sistem pakar yang mendiagnosis sesuatu yang belum pasti^[13]. Aturan-aturan kombinasi *Certainty Factor* yang berlaku menurut McAllister dapat di lihat pada Tabel 2.2.

Tabel 2.2 Aturan-aturan McAllister

No	Kondisi Aturan	Rumus
1	Apabila kedua nilai positif (+ +)	$CF_c (CF_1,CF_2) = CF_1 + CF_2*(1-CF_1)$
2	Apabila kedua nilai negatif (- -)	$CF_c (CF_1,CF_2) = CF_1 + CF_2*(1+CF_1)$
3	Apabila salah satu nilai negatif (+ -)	$CF_c = \frac{CF_1 + CF_2}{1 - \min(CF_1 , CF_2)}$

2.2.3 Langkah Perhitungan *Certainty Factor*

Adapun langkah-langkah untuk melakukan perhitungan pada metode *Certainty Factor* yaitu^[13] :

- **Langkah Pertama**

Tentukan nilai CF Pakar {CF(H)} dan CF Pengguna {CF(E)} yang didapat dari hasil wawancara pakar dengan jawaban berdasarkan pada Tabel 2.1.

- **Langkah Kedua**

Selanjutnya akan dilakukan perhitungan nilai CF-nya dengan mengalikan antara nilai CF pakar {CF(H)} dan nilai CF pengguna {CF(E)}. Perhitungannya :

$$CF(H,E)_n = CF(H)_n * CF(E)_n$$

.

.

$$CF(H,E)_{n+1} = CF(H)_{n+1} * CF(E)_{n+1}$$

- **Langkah Ketiga**

Selanjutnya kombinasikan setiap 2 nilai CF sesuai dengan aturan-aturan yang ada. Apabila terdapat lebih dari 2 nilai CF, maka dilakukan pengurangan pengkombinasian nilai CF tersebut.

- **Langkah Keempat**

Hasil dari nilai CF kombinasi yang didapatkan akan dikalikan dengan 100%. Sehingga jawaban yang didapatkan oleh pengguna akan dalam bentuk persentase.

2.3 Padi Salibu

Padi (*Oryza Sativa* L.) merupakan komoditas tanaman paling penting di Indonesia. Rata-rata peningkatan produksi padi nasional beberapa tahun terakhir masih rendah, yaitu sekitar 6,37 persen per tahun. Berdasarkan data Badan Pusat Statistik, produksi padi nasional tahun 2015 diperkirakan meningkat hingga 3,97 persen dan diprediksi mencapai 75.36 juta ton gabah kering giling (GKG), atau setara dengan beras sebanyak 42,36 juta ton berdasarkan perhitungan KTNA (Kelompok Tani Nelayan Andalan). Berdasarkan angka tetap tahun 2015, produktivitas padi nasional sebesar 5,33 t/ha GKG^[1]. Padahal dengan laju pertumbuhan penduduk yang mencapai 1.49% dan laju konsumsi beras nasional 1.34% per tahun, rata-rata produktivitas padi nasional seharusnya minimal 6.0 t/ha.^[3]

Upaya meningkatkan produktivitas padi terus dilakukan, salah satunya adalah dengan memanfaatkan Padi Salibu. Padi Salibu merupakan tanaman padi yang tumbuh lagi setelah batang sisa panen ditebas/dipangkas, tunas akan muncul dari buku yang ada didalam tanah. Tunas ini akan mengeluarkan akar baru

sehingga suplai hara tidak lagi tergantung pada batang lama. Tunas ini bisa membelah atau bertunas lagi seperti padi tanaman pindah biasa, inilah yang membuat pertumbuhan dan produksinya sama atau lebih tinggi dibanding tanaman pertama (ibunya).^[2]

Padi salibu berbeda dengan padi ratun (Gambar 2.1). Padi Ratun adalah padi yang tumbuh dari batang sisa panen tanpa dilakukan pemangkasan batang, tunas akan muncul pada buku paling atas, suplai hara tetap dari batang lama.^[2]



Gambar 2.1 Perbandingan Teknik Salibu (kiri) dan Teknik Ratun (kanan)^[7]

Pertumbuhan tunas setelah dipotong sangat dipengaruhi oleh ketersediaan air tanah, dan pada saat panen sebaiknya kondisi air tanah dalam keadaan kapasitas lapang. Untuk mengimbangi kebutuhan unsur hara pada masa pertumbuhan anakan padi salibu perlu pemupukan yang cukup, terutama hara nitrogen.^[2]

Unsur nitrogen merupakan komponen utama dalam sintesis protein, sehingga sangat dibutuhkan pada fase vegetatif tanaman, khususnya dalam proses pembelahan sel. Tanaman yang cukup mendapatkan nitrogen memperlihatkan

daun yang hijau tua dan lebar, fotosintesis berjalan dengan baik, unsur nitrogen adalah faktor penting untuk produktivitas tanaman.^[2]

2.3.1 Prasyarat Budidaya Padi Salibu

Persyaratan utama yang harus dipenuhi pada budidaya padi salibu antara lain^[7]:

- a) Bukan daerah endemik Organisme Pengganggu Tanaman (OPT) khususnya penyakit tungro, busuk batang, hawar daun bakteri, keong mas, dan lain-lain.
- b) Ketersediaan air mudah dikondisikan dan cukup.
- c) Tidak terjadi genangan dan kekeringan yang lama.
- d) Kondisi lahan dengan drainase baik.
- e) Kondisi air tanah pada saat dua minggu sebelum dan setelah panen sebaiknya pada kondisi kapasitas lapang (lembab). Dalam hal ini tunas padi salibu lebih baik tumbuhnya jika kondisi tanah lembab dibandingkan kondisi tergenang.

Di wilayah dengan sistem tanam serempak, pengembangan padi salibu disarankan pada suatu hamparan dengan luas minimal 25 ha untuk mengurangi serangan OPT.^[7]

2.3.2 Agroekosistem Lahan Padi Salibu

Secara umum, budidaya padi salibu dapat dilakukan pada berbagai agroekosistem dan ketinggian tempat (dari rendah sampai 1.100 m dpl) seperti di lahan irigasi desa atau sederhana yang sistem pengairannya diusahakan secara mandiri oleh kelompok tani, di lahan tadah hujan dan pasang surut.^[7]

Teknik budidaya Padi Salibu pada berbagai agroekosistem tersebut sebagai berikut^[7]:

1. Lahan Irigasi Desa

- Kondisi lahan subur dengan sistem pengairan yang mudah diatur atau dikendalikan secara swadaya oleh kelompok tani.



Gambar 2.2 Penggenangan Lahan Sawah^[7]

- Jika saat panen kondisi tanah kurang basah, maka masukkan air ke lahan segera setelah dilakukan panen tanaman utama yang menyisakan tunggul tanaman setinggi 25 cm dari permukaan tanah (seperti Gambar 2.2), untuk mencapai kondisi kapasitas lapang (lembab).
- Tunggul sisa panen dibiarkan selama 7-10 hari setelah panen atau hingga keluar anakan baru. Apabila tunas yang keluar kurang 70% dari populasi, maka tidak disarankan untuk dilakukan budidaya salibu.
- Jika tunas yang tumbuh > 70% dari populasi, lakukan pemotongan ulang tunggul sisa panen secara seragam dengan alat pemotong hingga tersisa 3-5 cm dari permukaan tanah. Gunakan dekomposer untuk perombakan sisa jerami bekas potongan tunggul padi.

2. Lahan Tadah Hujan

- Sebelum tanaman utama ditanam, dilakukan pengolahan tanah secara sempurna dan penambahan bahan organik sekitar 2-5 ton/ha.
- Saat panen tanaman utama, upayakan kondisi tanah tidak terlalu kering. Jika kering, maka lakukan pemberian air segera setelah panen dengan ketinggian 2-5 cm dari tanah untuk mencapai kondisi kapasitas lapang (lembab).
- Sisa pemotongan panen tanaman utama sebaiknya diletakkan di sekitar tanaman atau sebagai penutup permukaan tanah untuk mempertahankan kelembaban tanah seperti pada Gambar 2.3.
- Tunggul sisa panen dibiarkan selama 7-10 hari setelah panen atau hingga keluar anakan baru. Apabila tunas yang keluar kurang 70% dari populasi, maka tidak disarankan untuk dilakukan budidaya salibu.



Gambar 2.3 Pemotongan Ulang Tunggul Sisa Panen^[7]

- Jika tunas yang tumbuh > 70% dari populasi, maka dilakukan pemotongan ulang tunggul sisa panen secara seragam dengan alat pemotong hingga

tersisa 3-5 cm dari permukaan tanah (seperti Gambar 2.3). Gunakan dekomposer untuk perombakan sisa jerami bekas potongan tunggul padi.

3. Lahan Pasang Surut

- Teknologi budidaya padi salibu di lahan pasang surut, harus dilakukan kajian terlebih dahulu dan sebaiknya dipilih lokasi-lokasi yang memiliki tipe luapan A ke B yang tidak tergenangi ketika air pasang.
- Sistem budidaya padi sistem ratun di lahan pasang surut selama ini banyak dilakukan pada musim tanam periode Oktober – Maret, dan diasumsikan bahwa sistem budidaya salibu juga dapat dilakukan pada periode tersebut.
- Tunggul sisa panen dibiarkan selama 7-10 hari setelah panen atau hingga keluar anakan baru. Apabila tunas yang keluar kurang 70% dari populasi, maka tidak disarankan untuk dilakukan budidaya salibu.
- Jika tunas yang tumbuh > 70% dari populasi, lakukan pemotongan ulang tunggul sisa panen secara seragam dengan alat pemotong hingga tersisa 3-5 cm dari permukaan tanah. Gunakan dekomposer untuk perombakan sisa jerami bekas potongan tunggul padi.

Apabila semua proses dilakukan dengan benar, dan lahan yang akan ditanami padi salibu memenuhi kriteria maka hasil yang diperoleh akan sama dengan padi indukannya (seperti Gambar 2.4).



Gambar 2.4 Hasil Padi Salibu^[7]

2.4 Android

Android merupakan platform *mobile* yang bersifat open source dan didukung oleh Google OS. Pada mulanya Android dikembangkan oleh Google Inc. dan kemudian diselesaikan oleh Open Handset Alliance (OHA).^[8]



Gambar 2.5 Logo Android^[8]

Android (Gambar 2.5) merupakan subset perangkat lunak untuk perangkat *mobile* yang meliputi sistem operasi *middleware* dan aplikasi inti yang dirilis oleh Google. Sebagai pelengkapanya berupa Android SDK (*Software Development Kit*) yang menyediakan Tools dan API (*Application Programming Interface*) yang diperlukan untuk mengembangkan aplikasi pada platform Android dengan menggunakan bahasa pemrograman Java.^[8]

Android dikembangkan secara bersama – sama antara Google, Intel, Motorola, Qualcomm, T-Mobile, NVIDIA serta 47 perusahaan lain yang tergabung dalam OHA (Open Handset Alliance) dengan tujuan membuat sebuah standar terbuka untuk perangkat bergerak (mobile device).^[8]

2.4.1 Versi Android

Selain pemberian kode nomor ke dalam setiap versi, Android juga diberi nama berupa nama makanan sesuai dengan huruf alphabet.^[8] Adapun namanya yaitu :^[9]

- Pada 9 Maret 2009, Google pertama kali merilis Android versi 1.1.
- Android versi 1.5 bernama *Cupcake*.
- Android versi 1.6 bernama *Donut*.
- Android versi 2.0, 2.0.1 dan 2.1 bernama *Éclair*.
- Android versi 2.2, 2.2.1-2.2.3 bernama *Froyo (Frozen Yogurt)*.
- Android versi 2.3, 2.3.1-2.3.7 bernama *Gingerbread*.
- Android versi 3.0-3.2 bernama *Honeycomb*.
- Android versi 4.0, 4.0.1-4.0.4 bernama *Ice Cream Sandwich*.
- Android versi 4.1-4.3 bernama *Jelly Bean*.
- Android versi 4.4 bernama *KitKat*.
- Android versi 5.0 bernama *Lollipop*.
- Android versi 6.0 bernama *Marshmallow*.

2.4.2 Kelebihan & Kekurangan Android

Kelebihan sistem operasi ini yaitu, sistem operasinya terbuka, sehingga dapat dikembangkan oleh siapa saja. Akses mudah ke Android Market. Multitasking, yaitu ponsel android mampu menjalankan beberapa aplikasi

sekaligus. Mudah dalam hal notifikasi, maksudnya sistem operasi ini dapat memberitahukan anda tentang adanya SMS, Email, atau bahkan artikel terbaru dari RSS Reader.^[8]

Kelebihan lainnya yaitu mendukung semua layanan Google. Fasilitas penuh USB, maksudnya yaitu dengan fitur USB pengguna mampu mengisi baterai, menggunakannya sebagai USB *mass storage*, dan juga USB *tethering*.^[8]

Selain kelebihan yang banyak, terdapat kekurangan sistem operasi dari android yaitu berupa iklan. Terkadang aplikasi yang di download secara gratis dan mudah akan terdapat iklan di dalam aplikasi tersebut. Lambatnya pembaharuan perangkat yang tersedia, serta terdapatnya malware. Baterai yang boros, karena OS ini punya banyak proses yang bekerja di balik layar hingga lebih boros baterai.^[8]

2.4.3 Fitur-fitur pada Android

Fitur – fitur yang terdapat pada perangkat smartphone berbasis Android antara lain :^[8]

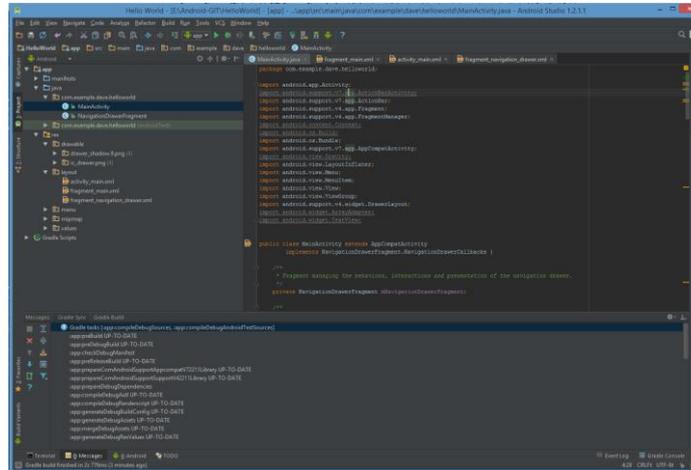
1. ***Dalvik Virtual Machine***, merupakan Java Runtime Environment yang telah dioptimasi untuk *device* atau perangkat dengan ukuran memori yang kecil. Fitur ini menjadikan aplikasi dapat dijalankan dengan baik pada perangkat berbasis Android.
2. ***Touch-screen* atau layar sentuh**. Fitur ini cukup fenomenal karena belum terdapat pada perangkat ponsel yang lama. Dengan menggunakan fitur ini maka proses navigasi menu menjadi lebih mudah karena pengguna hanya memilih dan menekan menu yang akan dijalankan tanpa perlu harus melakukan *scroll* ke atas-bawah atau samping kiri-kanan.

3. **Multipage.** Pengguna dapat menambahkan halaman baru pada layar sehingga tampilan ikon aplikasi pada layar semakin banyak. Hal ini berguna untuk semakin mempercepat akses ke berbagai aplikasi.
4. **Bersifat terbuka (Open Source),** sehingga pengguna dapat mempelajari, membuat, serta memodifikasi sesuai keinginan tanpa harus membayar.
5. **Kualitas suara dan grafik yang bagus,** karena dalam sistem Android telah tersedia dengan standar suara dan video seperti MP3, AAC, dll.
6. **SQLite.** Sebagai *database* untuk media penyimpanan aplikasi – aplikasi *smartphone*.
7. **Tersedianya berbagai macam library/services,** yang dapat langsung digunakan seperti *browser*, GPS, kamera, Bluetooth, dan Wifi.
8. **Miracast.** Sebuah bentuk protokol yang memperbolehkan perangkat baru untuk melakukan streaming audio dan video pada televisi yang mempunyai fitur Miracast. Fitur ini memiliki kemiripan dengan *Airplay* yang dimiliki oleh sejumlah perangkat keluaran Apple.
9. **Gesture Typing Keyboard.** Sebenarnya fitur ini mirip dengan *Swipe Keyboard* yang sudah lama diperkenalkan, tetapi Google menyempurnakannya dalam Android 4.2 dengan memberikan akurasi yang lebih baik dan respon yang lebih cepat.

2.4.4 Android Studio

Android Studio adalah sebuah IDE (*Integrated Development Environment*) yang bisa digunakan untuk pengembangan aplikasi Android, dan dikembangkan oleh Google. Android Studio merupakan pengembangan dari Eclipse IDE dan dibuat berdasarkan IDE Java yang populer, yaitu JetBrains' IntelliJ IDEA

software. Android Studio direncanakan akan menggantikan Eclipse ke depannya sebagai IDE resmi untuk pengembangan aplikasi Android^[14]. Berikut pada Gambar 2.6 tampilan umum pada Android Studio.



Gambar 2.6 Tampilan Android Studio

Sebagai pengembangan dari Eclipse, Android Studio mempunyai banyak fitur-fitur baru dibandingkan dengan Eclipse IDE. Berbeda dengan Eclipse yang menggunakan Apache Ant, Android Studio menggunakan Gradle sebagai *build systemnya*. Fitur-fitur dari Android Studio adalah sebagai berikut ^[14]:

- Menggunakan *Gradle-based build system* yang fleksibel.
- Bisa mem-*build multiple* APK .
- *Template support* untuk Google Services dan berbagai macam tipe perangkat.
- *Layout editor* grafis yang lebih bagus.
- *Built-in support* untuk *Google Cloud Platform*, sehingga mudah untuk integrasi dengan *Google Cloud Messaging* dan *App Engine*.
- *Import library* langsung dari Maven Repository.
- Mendukung layanan *Android Native Development Kit (NDK)*.

2.4.4.1 JDK (*Java Development Kit*)

JDK adalah sebuah perangkat peralatan yang digunakan untuk membangun perangkat lunak dengan menggunakan bahasa pemrograman Java. JDK berjalan diatas sebuah *virtual machine* yang dinamakan JVM (*Java Virtual Machine*).^[16]

2.4.4.2 Android SDK (*Software Development Kit*)

Android SDK adalah tools API (*Application Programming Interface*) yang diperlukan untuk mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java. Sebagai platform aplikasi netral, Android memberi kesempatan bagi semua orang untuk membuat aplikasi yang dibutuhkan, yang bukan merupakan aplikasi bawaan *Handphone/Smartphone*.^[18]

2.4.4.3 Android NDK (*Native Development Kit*)

Android NDK adalah peralatan pendamping Android SDK yang memungkinkan *developer* untuk mengimplementasikan kode *native* seperti bahasa C dan C++ di beberapa bagian aplikasi yang diinginkan.

2.4.4.4 AVD (*Android Virtual Device*)

Android Virtual Device merupakan emulator untuk menjalankan aplikasi android. Tampilan AVD dapat dilihat pada Gambar 2.7. Setiap AVD terdiri dari^[18]:

1. Sebuah profil perangkat keras yang dapat mengatur pilihan untuk menentukan fitur hardware emulator. Misalnya, menentukan apakah menggunakan perangkat kamera, apakah menggunakan

keyboard QWERTY fisik atau tidak, berapa banyak memori internal, dan lain-lain.

2. Sebuah pemetaan versi Android, maksudnya kita menentukan versi dari platform Android akan berjalan pada emulator.
3. Pilihan lainnya. Misalnya menentukan *skin* yang kita ingin gunakan pada emulator, yang memungkinkan untuk menentukan dimensi layar, tampilan, dan sebagainya. Kita juga dapat menentukan SD Card virtual untuk digunakan dengan di emulator.



Gambar 2.7 Tampilan AVD^[18]

2.5 Skala Likert

Skala Likert adalah metode pengukuran yang digunakan untuk mengukur sikap, pendapat dan persepsi seseorang atau kelompok orang tentang fenomena sosial.^[19] Prosedur penskalaan dengan metode Likert dapat dilakukan dengan langkah berikut :

1. Penentuan Skor Jawaban

Skor jawaban merupakan nilai jawaban yang akan diberikan oleh responden. Hal pertama yang dilakukan adalah menentukan skor dari tiap jawaban yang akan diberikan. Contohnya, sikap yang akan kita pakai yaitu "setuju". Selanjutnya kita menentukan banyaknya jawaban pada tiap pertanyaan yang akan kita berikan. Misalnya 5 skala, berarti sangat tidak setuju, kurang setuju, setuju, cukup setuju dan sangat setuju.^[19]

Jika pertanyaan yang diberikan bersifat susah untuk diberikan jawabannya, otomatis responden cenderung statik. Oleh karena itu kita dapat memberikan pilihan jawaban yang banyak, misal 7 atau 9 jawaban dari tiap pertanyaan. Hal ini bertujuan agar responden dapat memberikan penilaian sesuai dengan kriteria mereka berdasarkan pilihan yang ada. Lihat Tabel 2.3 dibawah ini.^[20]

Tabel 2.3 Penentuan Skor Jawaban

Skala Jawaban	Skor / Bobot
Sangat Setuju	5
Cukup Setuju	4
Setuju	3
Kurang Setuju	2
Sangat Tidak Setuju	1

2. Skor Ideal

Skor ideal merupakan skor yang digunakan untuk menghitung skor untuk menentukan rating scale dan jumlah seluruh jawaban. Untuk menghitung jumlah skor ideal (Ideal) dari seluruh item, digunakan rumus berikut^{[19][20]} :

$$\text{Skor Ideal} = \text{Nilai Skala maks} \times \text{Jumlah responden} \dots\dots\dots (4)$$

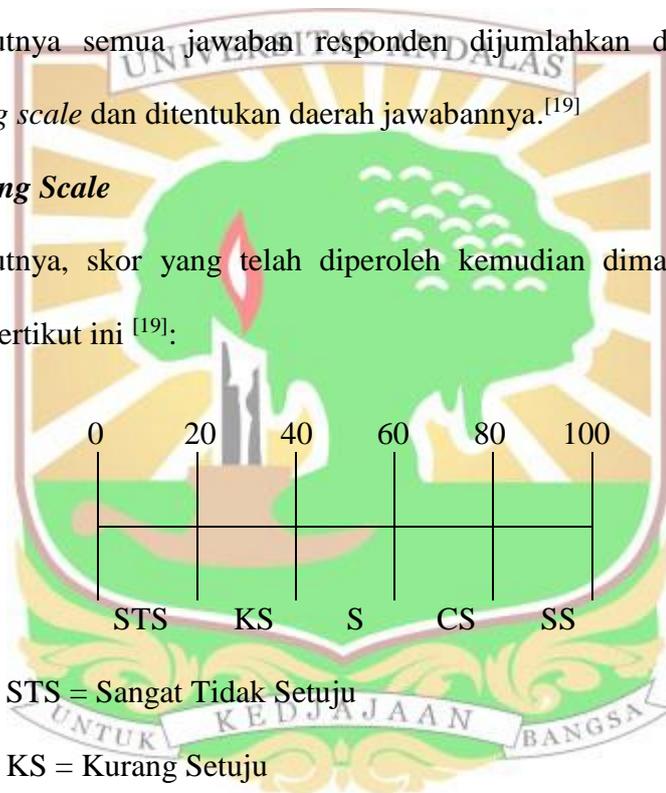
Seandainya skor tertinggi adalah 5 (sesuai Tabel 2.3) dan jumlah responden 20, maka dapat dirumuskan menjadi:

Rumus	Skala Jawaban
$5 \times 20 = 100$	Sangat Setuju
$4 \times 20 = 80$	Cukup Setuju
$3 \times 20 = 60$	Setuju
$2 \times 20 = 40$	Kurang Setuju
$1 \times 20 = 20$	Sangat Tidak Setuju

Selanjutnya semua jawaban responden dijumlahkan dan dimasukkan kedalam *rating scale* dan ditentukan daerah jawabannya.^[19]

3. Rating Scale

Selanjutnya, skor yang telah diperoleh kemudian dimasukkan kedalam *rating scale* bertikut ini ^[19]:



Keterangan : STS = Sangat Tidak Setuju

KS = Kurang Setuju

S = Setuju

CS = Cukup Setuju

SS = Sangat Setuju

Rating scale berfungsi untuk mengetahui hasil data angket (kuesioner) dan wawancara secara umum dan keseluruhan yang didapat dari penilaian angket (kuesioner) dan wawancara^[19].

4. Persentase Jawaban

Selanjutnya untuk mengetahui jumlah persentase jawaban dari responden, digunakan rumus berikut :

$$p = \frac{f}{n} \times 100\% \quad \dots\dots\dots (5)$$

Keterangan : p = Persentase jawaban

f = Frekuensi dari setiap jawaban di kuesioner

n = Jumlah skor ideal



0=20

BAB III

PERANCANGAN DAN PEMBUATAN APLIKASI

Bab ini membahas tentang perancangan yang digunakan dalam penyelesaian tugas akhir.

3.1 Jenis Penelitian

Berdasarkan tujuan dan manfaat dari tugas akhir ini maka penelitian ini adalah penelitian terapan. Penelitian terapan merupakan pemecahan terhadap suatu masalah untuk tujuan tertentu dan merupakan pengembangan dari penelitian yang telah ada dengan berpedoman pada data sekunder (data dari hasil penelitian) yang relevan.

3.2 Akuisi Data dan Sampel Penelitian

3.2.1 Akuisi Data

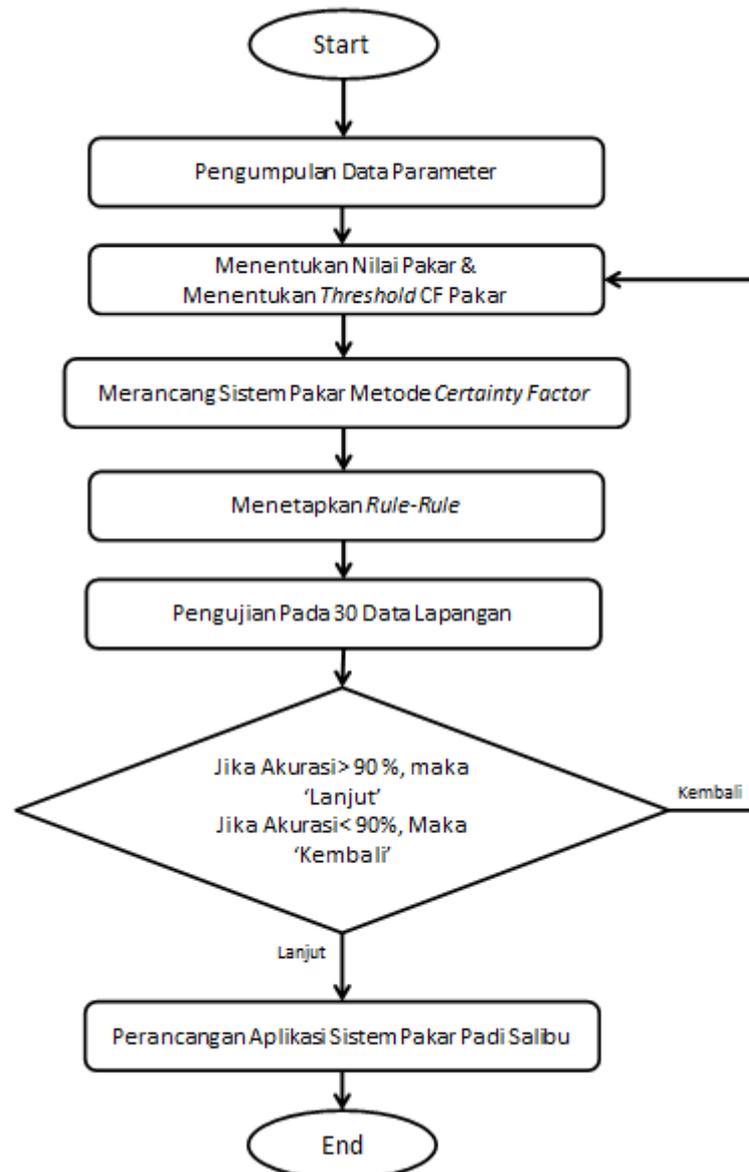
Data berupa kumpulan parameter-parameter yang didapatkan dari referensi dan pakar yang ahli di bidangnya. Parameter-parameter ini akan digunakan untuk menentukan kelayakan dari teknik yang akan diterapkan di lapangan.

3.2.2 Sampel Penelitian

Sampel penelitian didapatkan dari pengujian lapangan setelah teknik ini diterapkan. Sampel penelitian ini nantinya akan dibandingkan dengan sampel yang telah ada oleh pembanding berupa sawah yang akan diterapkan teknik padi salibu.

3.3 Perancangan Sistem

Perancangan Sistem secara keseluruhan dapat dilihat Gambar 3.1 berikut.



Gambar 3. 1 Diagram Alir Perancangan Sistem

3.3.1 Pengumpulan Parameter

Pengumpulan parameter yang dibutuhkan dilakukan dengan cara mencari informasi dari referensi yang ada. Selain itu parameter yang dibutuhkan juga didapatkan dari wawancara dengan pakar Padi Salibu yang ada yaitu Ir. Erdiman (Peneliti & Penemu Teknologi Salibu, BPTP

Sumbar). Berikut pada Tabel 3.1 data berupa parameter-parameter yang dibutuhkan.

Tabel 3.1 Parameter Padi Salibu

No	Parameter	Pilihan Parameter	Keterangan
1	Jenis Irigasi	• Irigasi Teknis	Bendungan permanen, debit air terukur
		• Setengah Teknis	Bendungan sederhana, debit air tak terukur
		• Irigasi Pompa	Sumber air dari waduk atau air tanah
		• Tadah Hujan	Sumber air dari air hujan
2	Kondisi Drainase Tanah	• Lancar	Kehilangan air sangat cepat
		• Sedikit Terhambat	Kehilangan air agak lambat
		• Terhambat	Kehilangan air sangat lambat
3	Kondisi Hamparan Lahan	• Miring	Kemiringan lebih dari 5%
		• Datar	Kemiringan kurang dari 5%
		• Cekung (Lembah)	Dataran diantara 2 lembah
4	Kadar Organik Tanah	• Tinggi	Warna tanah lebih gelap
		• Sedang	Warna tanah agak terang
		• Rendah (Kurang)	Warna tanah lebih terang
5	Tekstur Tanah	• Sedang (Lempung)	Tanah yang lempung
		• Halus (Liat)	Seperti tanah liat
		• Kasar (Pasir)	Hampir seperti pasir

3.3.2 Pengumpulan Data Pakar

Pengumpulan data ini dilakukan dengan cara wawancara dan pemberian kuesioner kepada pakar, sehingga data yang diperlukan akan sesuai dengan yang diharapkan.

Adapun data pakar yang diperlukan yaitu berupa bobot/nilai pada masing-masing parameter. Oleh karena itu sebelum melakukan wawancara

dengan pakar untuk mendapatkan bobot masing-masing parameter, terlebih dahulu penulis melakukan wawancara untuk menanyakan parameter-parameter yang dibutuhkan pada Padi Salibu seperti yang terdapat pada subbab 3.3.1.

Setelah didapatkan data parameter yang parameter tepat, selanjutnya penulis memberikan kuesioner kepada pakar sembari melakukan wawancara. Kuesioner yang diberikan dapat dilihat pada Gambar 3.1.

KUESIONER
BUDIDAYA PADI SALIBU

Responden (pakar) yang terhormat,
Saya Rizky Syafizal Mahasiswa Teknik Elektro Universitas Andalas sedang menyusun Tugas Akhir bermaksud melakukan penelitian dengan judul "IMPLEMENTASI METODE CERTAINTY FACTOR UNTUK MENILAI KELAYAKAN BUDIDAYA PADI SALIBU BERBASIS ANDROID". Oleh sebab itu, perkenankanlah saya untuk meminta sedikit waktu Anda untuk membantu penelitian saya dengan mengisi kuesioner ini.

a. Petunjuk Pengisian
Pada lembar Kuesioner ini akan diberikan beberapa parameter yang berkaitan mengenai budidaya penanaman padi salibu dan diharapkan kepada responden (pakar) untuk memilih pilihan kondisi (*term*) yang disediakan dan mengisi pada kolom nilai masing-masing parameter yang ada.

b. Pilihlah Kondisi (*term*)
Dalam mengisi Kuesioner ini, diberikan beberapa pilihan kepada responden (pakar) berupa kondisi-kondisi agar lebih memudahkan dalam penilaian.

Kondisi	Nilai
Pasti Tidak Ada	1
Hampir Pasti Tidak Ada	2
Kemungkinan Besar Tidak Ada	3
Mungkin Tidak Ada	4
Tidak Tahu	5
Mungkin Ada	6
Kemungkinan Besar Ada	7
Hampir Pasti Ada	8
Pasti Ada	9

c. Pertanyaan
Pada parameter-parameter berikut, seberapa besar kepastian dari pengaruh tiap parameter pada suatu lahan sehingga cocok dan dapat diterapkannya Budidaya Padi Salibu ?

d. Kuesioner

No	Parameter	Pilihan Parameter	Nilai
1	Jenis Irigasi	• Irigasi Teknis	
		• Setengah Teknis	
		• Irigasi Pempu	
		• Tidak Ada Hujan	
2	Kondisi Drainase Tanah	• Lancar	
		• Sedikit Terhambat	
		• Terhambat	
3	Kondisi Hamparan Lahan	• Miring	
		• Datar	
4	Kadar Organik Tanah	• Cukup (Lembah)	
		• Tinggi	
5	Tekstur Tanah	• Sedang (Kurang)	
		• Sedang (Lempung)	
		• Halus (Liat)	
		• Kasar (Pasir)	

Gambar 3.2 Kuesioner Pembobotan Parameter

Bobot yang ada pada kuesioner dibuat berdasarkan tabel kondisi masing-masing nilai CF yang ada pada Tabel 2.1. Pembobotan dilakukan untuk mempermudah pakar dalam mengisi kuesioner. Berikut nilai CF untuk masing-masing bobotnya :

- Bobot 1 = -1
- Bobot 2 = -0.8
- Bobot 3 = -0.6
- Bobot 6 = 0.4
- Bobot 7 = 0.6
- Bobot 8 = 0.8

- Bobot 4 = -0.4
- Bobot 5 = -0.2 sampai 0.2
- Bobot 9 = 1

3.3.3 Perancangan Sistem Pakar Metode *Certainty Factor*

Perancangan Sistem Pakar ini bertujuan untuk membantu mempermudah tani/kelompok tani dan penyuluh untuk melakukan budidaya Padi Salibu dan membantu memudahkan seorang pakar Padi Salibu.

Metode sistem pakar yang digunakan adalah metode *Certainty Factor*. Metode ini digunakan untuk menarik kesimpulan/keputusan dalam Sistem Pakar Padi Salibu. Terdapat beberapa langkah untuk menerapkan *Certainty Factor*. Langkah-langkah tersebut adalah:

1. Penentuan Nilai-Nilai CF Pakar (CF(h)) dan CF *User* (CF(e)) dengan cara wawancara pakar.
2. Perhitungan *Certainty Factor*
3. Penetapan *Rule-Rule* Keputusan
4. Pengujian Data & Penentuan *Threshold* Keputusan

Berikut ini adalah langkah-langkah penerapan *Certainty Factor* :

3.3.3.1 Penentuan Nilai CF Pakar dan CF User

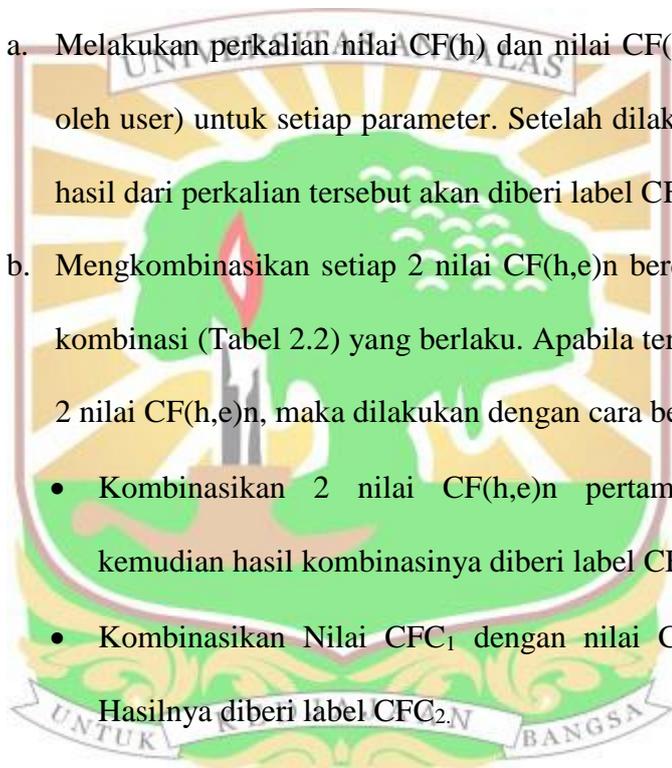
Nilai CF Pakar dan CF *User* ditentukan dengan cara wawancara kepada pakar. Wawancara dilakukan dengan bantuan kuesioner yang mana untuk kondisi (*term*) serta nilai CFnya ditentukan sesuai dengan yang dijelaskan pada subbab 3.3.2.

Nilai CF Pakar ($CF(h)$) adalah nilai yang terbaik pada masing-masing parameter yang ada. Selanjutnya nilai $CF(h)$ akan dijadikan referensi untuk perhitungan nilai *Certainty Factor*.

3.3.3.2 Perhitungan Certainty Factor

Perhitungan *Certainty Factor* dilakukan dengan cara sebagai berikut :

- a. Melakukan perkalian nilai $CF(h)$ dan nilai $CF(e)$ (yang dipilih oleh user) untuk setiap parameter. Setelah dilakukan perkalian, hasil dari perkalian tersebut akan diberi label $CF(h,e)_n$.
- b. Mengkombinasikan setiap 2 nilai $CF(h,e)_n$ berdasarkan aturan kombinasi (Tabel 2.2) yang berlaku. Apabila terdapat lebih dari 2 nilai $CF(h,e)_n$, maka dilakukan dengan cara berikut.
 - Kombinasikan 2 nilai $CF(h,e)_n$ pertama dan kedua, kemudian hasil kombinasinya diberi label CFC_1 .
 - Kombinasikan Nilai CFC_1 dengan nilai $CF(h,e)_n$ ketiga. Hasilnya diberi label CFC_2 .
 - Langkah tersebut dilanjutkan sampai nilai $CF(H,e)_n$ terakhir dikombinasikan.
- c. Hasil dari kombinasi nilai terakhir (CFC_n) adalah nilai *Certainty Factor* (CF) yang digunakan untuk menentukan keputusan.



3.3.3.3 Penetapan Rule-Rule Keputusan

Penetapan *rule* ini bertujuan untuk menilai dan memberikan hasil keputusan yang akan ditampilkan oleh aplikasi nantinya. Penetapan *rule* dilakukan bersama pakar dengan cara mencobakan aplikasi yang dirancang terhadap beberapa kasus yang diberikan pakar. Percobaan ini dilakukan berulang kali sehingga didapatkan *rule-rule* dengan *threshold* yang dirasa sesuai dengan keputusan pakar.

Setelah dilakukan penentuan *rule-rule*, maka dilakukan pengujian pada 30 data pada Lampiran B-1. Apabila akurasi yang didapatkan setelah pengolahan 30 data tersebut lebih besar dari 90 %, maka akan dilanjutkan ke perancangan aplikasi *interface* androidnya. Akan tetapi apabila akurasi yang didapat lebih kecil dari 90 %, maka akan dilakukan penentuan ulang *threshold* dari masing-masing *rule* yang ada.

3.3.3.4 Pengujian Data

Pengujian Data ini dilakukan untuk menguji tingkat akurasi dari aplikasi yang dirancang. Pengujian data dilakukan dengan cara mengujikan metode *Certainty Factor* ke 30 data yang diberikan oleh pakar (Lampiran B-1). Hasil dari pengujian tersebut akan menghasilkan 30 nilai CF yang akan di sesuaikan dengan *rule-rule* yang telah ditetapkan.

Setelah didapatkan hasil keputusan masing-masing data menggunakan metode *Certainty Factor*, selanjutnya hasil keputusan tersebut akan dibandingkan dengan hasil keputusan pakar pada 30 data yang sama.

Dari perbandingan yang dilakukan, akan didapatkan hasil keputusan yang sama dan hasil keputusan yang berbeda. Sehingga didapatkan hasil akurasi menggunakan persamaan berikut.

$$\text{akurasi} = \frac{\text{jumlah kasus dengan keputusan yang benar}}{\text{jumlah keseluruhan kasus}} \times 100 \%$$

3.4 Contoh Kasus Perhitungan *Certainty Factor*

Setelah metode *Certainty Factor* diujikan ke 30 data (di lampiran B-1) dan didapatkan nilai *threshold*-nya pada subbab 4.2, maka diberikan suatu kasus sesuai dengan kondisi lapangan yang ada. Berdasarkan data yang ada, diambil satu kasus yang akan dihitung nilai CF nya serta ditentukan hasil keputusan sesuai dengan nilai CF yang didapat.

Diambil satu kasus dari data di Kecamatan Matur, Kab. Agam. Selanjutnya untuk menentukan nilai CF serta hasil keputusannya, dilakukan langkah-langkah yang telah dijelaskan pada subbab 2.2.3 sebagai berikut :

a. Langkah 1 : Penentuan Nilai CF Pakar (CF(h)) dan CF *User* (CF(e)).

- Jenis irigasi yang digunakan adalah “Irigasi Setengah Teknis”.
CF(e)₁ = 0.4
- Kondisi drainase lahan “Lancar”.
CF(e)₂ = 0.8
- Bentuk hamparannya “Datar”.
CF(e)₃ = 0.4
- Kadar bahan organik tanah yang “Sedang”.
CF(e)₄ = 0.4
- Tekstur tanah “Kasar(berpasir)”.
CF(e)₅ = -0.6

Penentuan nilai CF(h) masing-masing parameter (berdasarkan subbab 3.3.3) sebagai berikut.

- $CF(h)_1 = CF(h)$ Irigasi = 0.8
- $CF(h)_2 = CF(h)$ Drainase = 0.8
- $CF(h)_3 = CF(h)$ Hampan Lahan = 0.8
- $CF(h)_4 = CF(h)$ Kadar Organik = 0.8
- $CF(h)_5 = CF(h)$ Tekstur Tanah = 0.6

b. Langkah 2 : Perkalian Nilai CF(h) dan CF(E).

Selanjutnya, dilakukan perkalian nilai $CF(h)_n$ dan nilai $CF(E)_n$.

- $CF(h,e)_1 = CF(h)_1 \times CF(e)_1 = 0.8 \times 0.4 = 0.32$
- $CF(h,e)_2 = CF(h)_2 \times CF(e)_2 = 0.8 \times 0.8 = 0.64$
- $CF(h,e)_3 = CF(h)_3 \times CF(e)_3 = 0.8 \times 0.4 = 0.32$
- $CF(h,e)_4 = CF(h)_4 \times CF(e)_4 = 0.8 \times 0.4 = 0.32$
- $CF(h,e)_5 = CF(h)_5 \times CF(e)_5 = 0.6 \times -0.6 = -0.36$

c. Langkah 3 : Mengkombinasikan setiap 2 nilai $CF(h,e)_n$ berdasarkan pada aturan-aturan yang ada pada Tabel 2.2.

- $CFC_1 = CF(h,e)_1 + CF(h,e)_2 \times (1 - CF(h,e)_1)$; kedua nilai positif
 $= 0.32 + 0.64 \times (1 - 0.32)$
 $= 0.7552$
- $CFC_2 = CFC_1 + CF(h,e)_3 \times (1 - CFC_1)$; kedua nilai positif
 $= 0.7552 + 0.32 \times (1 - 0.7552)$
 $= 0.8335$

- $CFC_3 = CFC_2 + CF(h,e)_4 \times (1 - CFC_2)$; kedua nilai positif

$$= 0.8335 + 0.32 \times (1 - 0.8335)$$

$$= 0.8867$$
- $CFC_4 = \frac{CFC_3 + CF(h,e)_5}{1 - \min(|CFC_3|, |CF(h,e)_5|)}$; salah satu nilai negatif

$$= \frac{0.8867 + (-0.36)}{1 - \min(|0.8867|, |-0.36|)}$$

$$CFC_4 = \textit{Certainty Factor} = 0.823$$

Didapatkanlah nilai *Certainty Factor* (CF) sebesar 0.823. Selanjutnya untuk menentukan apakah keputusan yang diperoleh ”Cocok”, ”Cocok Bersyarat” atau ”Tidak Cocok”, maka digunakanlah *threshold* yang telah didapatkan pada subbab 4.2. Sehingga untuk nilai $CF = 0.823$ didapatkanlah keputusan ”Cocok Bersyarat” dengan *threshold* $0.76 < CF \geq 0.9275$.

3.5 Pembuatan Aplikasi *Interface Smartphone Android*

Pembuatan program aplikasi android menggunakan *software* Android Studio versi 1.5. Pembuatan program aplikasi android ini bertujuan untuk mendesain tampilan dari aplikasi sistem pakar Padi Salibu. Tampilan aplikasi android yang didesain dapat dilihat pada Gambar 3.1.



Gambar 3.3 Tampilan Aplikasi Android

3.6 Pengujian Fungsional & Tampilan Aplikasi

Kuesioner aplikasi ini dibuat untuk menilai kepuasan *user* pada saat menggunakan aplikasi yang dirancang. Kepuasan *user* dinilai dari 2 aspek, yaitu kepuasan terhadap fungsionalitas aplikasi dan kepuasan terhadap tampilan aplikasi.

Kuesioner yang dirancang disebarakan kepada 10 orang petani/kelompok petani dan 10 orang tenaga ahli yang mengerti mengenai pertanian. Selain mengerti di bidang pertanian, pemilihan responden tersebut juga bertujuan agar hasil dari pengujian ini tepat pada sasarannya. Bentuk kuesioner yang disebarakan dapat dilihat pada Gambar 3.2.



Jurusan Teknik Elektro – Fakultas Teknik
Universitas Andalas
Padang

Rifky Syafizal – 1110952010

Bapak/Ibu yang saya hormati,

Saya mahasiswa Jurusan Teknik Elektro, Fakultas Teknik – Universitas Andalas Padang. Dalam hal ini saya sedang mengadakan penelitian Tugas Akhir. Kuesioner ini berhubungan dengan pendapat anda mengenai aplikasi yang saya buat. Hasil kuisisioner ini tidak untuk dipublikasikan, melainkan untuk kepentingan penelitian.

Atas bantuan, kesediaan waktu dan kerjasamanya saya ucapkan terima kasih

Hari/Tg : Terapa ANI

KUESIONER

Penunjuk pengisian : Berilah tanda (✓) pada kolom jawaban yang Anda pilih.
Isi (.....) sesuai dengan pendapat Anda.

A. Data Responden

- Nama :
- Umur : Tahun Laki-Laki Perempuan
- Alamat :
- Pendidikan Terakhir :

<input type="checkbox"/> Tidak Sekolah	<input type="checkbox"/> SMA/ Sederajat
<input type="checkbox"/> SD/ Sederajat	<input type="checkbox"/> Diploma
<input type="checkbox"/> SMP/ Sederajat	<input type="checkbox"/> Sarjana
	<input type="checkbox"/> Lainnya :
- Pekerjaan :
- Bidang Keahlian :

B. Kuesioner Fungsional Aplikasi

- Apakah aplikasi dapat terinstall dengan baik di android anda ?
 - Ya
 - Tidak
- Bagaimana respon aplikasi saat akan di buka ?
 - Respon Cepat
 - Kurang Cepat
 - Lambat

- Selama menggunakan aplikasi, apakah fungsi-fungsi yang terdapat pada aplikasi seperti tombol, scroll dan lainnya dapat berfungsi dengan baik ?
 - Ya
 - Kurang
 - Tidak
 Jika Tidak, fungsi mana yang tidak berfungsi dengan baik :
- Apakah terdapat masalah (*error*) pada android anda selama menjalankan aplikasi ?
 - Ya
 - Sedikit
 - Tidak
- Apakah aplikasi yang digunakan berfungsi dengan baik ?
 - Baik
 - Kurang Baik
 - Tidak
 Jika Tidak, bagian mana yang tidak berfungsi dengan baik :
- Dari keseluruhan aplikasi yang digunakan, seberapa puas anda ketika menggunakan aplikasi tersebut ?
 - Sangat Puas
 - Puas
 - Kurang Puas
 - Sedikit
 - Tidak Puas
- Harapan anda mengenai aplikasi untuk kedepannya :

C. Kuesioner Tampilan Aplikasi

- Seberapa menarik tampilan pada aplikasi ?
 - Sangat Menarik
 - Menarik
 - Biasa Saja
 - Kurang Menarik
 - Tidak Menarik
- Sudah cukup membantah tampilan yang ada pada aplikasi ?
 - Sudah
 - Belum Membantu
 - Tidak Membantu

Gambar 3.4 Tampilan Kuesioner



BAB IV

HASIL DAN ANALISA

4.1 Hasil Pengumpulan Data Pakar

Pada tahap ini dibahas mengenai hasil dari pengumpulan data pakar berupa pembobotan pada masing-masing parameter yang telah ditentukan sesuai dengan yang dijelaskan pada subbab 3.3.1.

Tabel 4.1 Data Parameter Pakar

No	Parameter	Pilihan Parameter	Nilai / Bobot	Nilai CF
1	Jenis Irigasi	• Irigasi Teknis	8	0.8
		• Setengah Teknis	6	0.4
		• Irigasi Pompa	5	0.2
		• Tadah Hujan	4	-0.4
2	Kondisi Drainase Tanah	• Lancar	8	0.8
		• Sedikit Terhambat	6	0.4
		• Terhambat	3	-0.6
3	Kondisi Hampan Lahan	• Miring	8	0.8
		• Datar	6	0.4
		• Cekung (Lembah)	3	-0.6
4	Kadar Organik Tanah	• Tinggi	8	0.8
		• Sedang	6	0.4
		• Rendah (Kurang)	4	-0.4
5	Tekstur Tanah	• Sedang (Lempung)	7	0.6
		• Halus (Liat)	6	0.4
		• Kasar (Pasir)	3	-0.6

Proses pengumpulan data pakar ini dilakukan berulang-ulang dengan tujuan agar hasil dari pembobotan sesuai dengan yang diharapkan. Selain itu,

proses ini dilakukan agar bobot masing-masing parameter sesuai dengan pengaruh yang akan ditimbulkannya di lapangan.

Pemberian bobot yang dilakukan pada Tabel 4.1 semuanya berdasarkan pada nilai interpretasi CF yang ada pada Tabel 2.1, dimana nilai tertinggi pada tiap parameternya merupakan keputusan paling “Cocok” untuk diterapkan pada budidaya Padi Salibu. Sedangkan nilai terendah untuk tiap parameternya merupakan keputusan paling “Tidak Cocok” untuk diterapkan.

Berikut hasil parameter yang paling “Cocok” untuk diterapkan pada budidaya Padi Salibu :

1. Jenis Irigasi → Irigasi Teknis, CF = 0.8
2. Kondisi Drainase Tanah → Lancar, CF = 0.8
3. Kondisi Hampan Lahan → Miring, CF = 0.8
4. Kadar Organik Tanah → Tinggi, CF = 0.8
5. Tekstur Tanah → Sedang (Lempung), CF = 0.6

Nilai tertinggi dari masing-masing pakar ini nantinya akan dijadikan nilai referensi CF Pakar (CF(h)). Sedangkan seluruh nilai-nilai yang terdapat pada Tabel 4.1 selanjutnya bisa dijadikan nilai CF User (CF(e)) pada saat dipilih di aplikasi.

4.2 Hasil Pengujian & Pelatihan Data

Pada tahap ini menjelaskan hasil dari pengujian dan pelatihan yang dilakukan pada 30 data mengenai Padi Salibu yang ada di lapangan yang terdapat pada lampiran B-1.

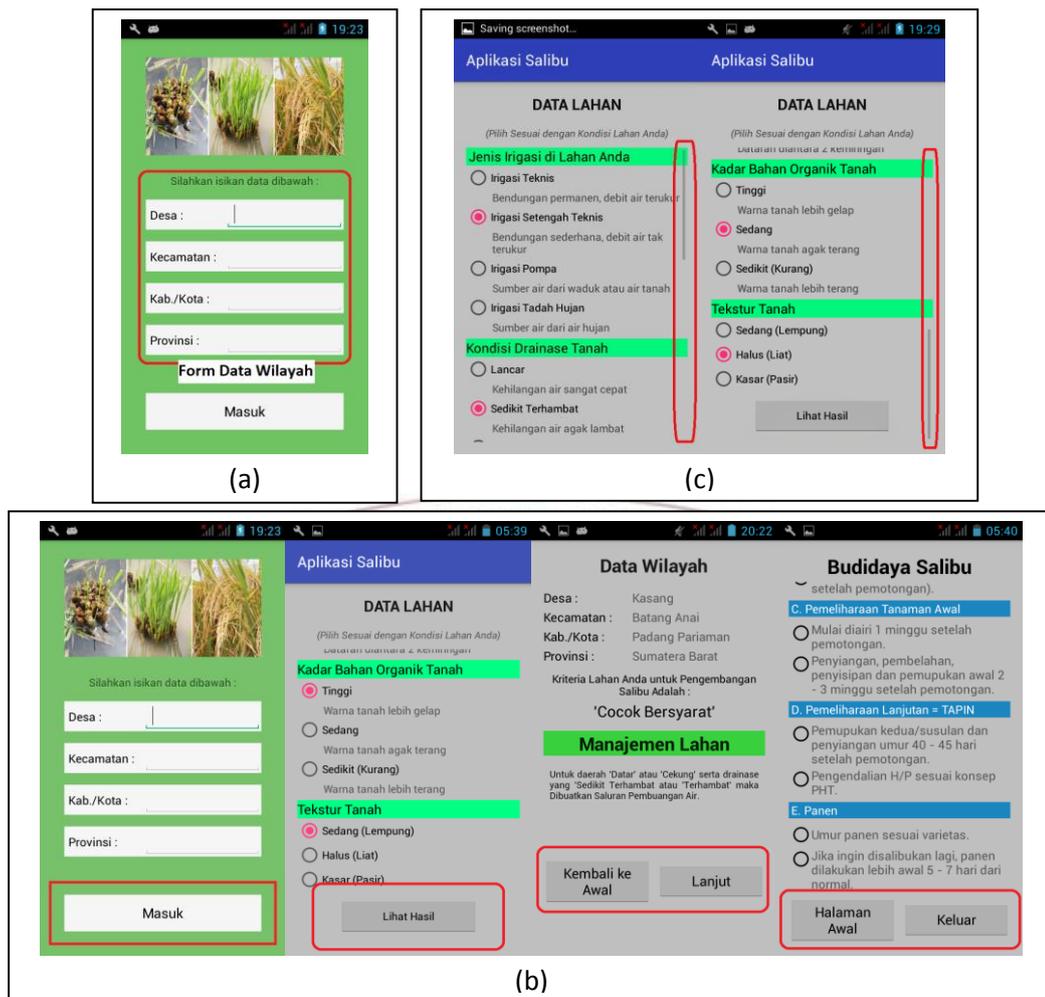
Pada subbab 3.3.3.3 telah dijelaskan mengenai *rule-rule* yang akan didapatkan. Selanjutnya dilakukan pengujian dan pelatihan data pada Lampiran B1, sehingga didapatkan *rule-rule* sebagai berikut :

- *Rule 1* = Keputusan “Cocok” jika nilai $CF > 0.9275$
- *Rule 2* = Keputusan “Cocok Bersyarat” jika nilai $0.76 < CF \leq 0.9275$
- *Rule 3* = Keputusan “Tidak Cocok” jika nilai $CF \leq 0.76$

Terdapat suatu kondisi tambahan yang diberikan ke dalam aplikasi yang dirancang. Kondisi ini berlaku apabila *user* memilih parameter “Irigasi Tadah Hujan”, maka hasil keputusan akan langsung menjadi “Tidak Cocok”. Hal ini dikarenakan menurut pakar apabila suatu lahan memiliki jenis irigasi tadah hujan, maka tidak memungkinkan lagi lahan tersebut untuk diterapkan budidaya Padi Salibu.

4.3 Hasil Pengujian Aplikasi Sistem Pakar Metode Certainty Factor

Pada tahap ini dibahas hasil pengujian aplikasi yang dirancang. Pengujian aplikasi ini terbagi menjadi tiga Tahapan. Tahapan yang pertama berupa pengujian fungsionalitas sistem, pengujian dilakukan dengan cara menguji semua fungsi-fungsi yang terdapat dalam aplikasi seperti fungsi tombol-tombol, fungsi *scroll*, serta *form* pengisian data. Tahapan yang kedua adalah menguji akurasi sistem dengan cara membandingkan hasil keputusan yang diberikan pakar dengan hasil keputusan yang didapatkan dari aplikasi yang dirancang. Tahapan yang ketiga adalah dengan menguji fungsionalitas aplikasi, yaitu dengan cara mengedarkan angket/kuesioner kepada petani/ kelompok tani dan tenaga ahli (Penyuluh/PPL) dibidang pertanian.



Gambar 4.1 Fungsional Aplikasi : a. Form Data Wilayah, b. Tombol Fungsi dan c. Fungsi Scrolling

4.3.1 Hasil Pengujian Fungsionalitas Aplikasi (*Black Box Test*)

Pada tahap ini, setiap fungsi-fungsi yang ada pada aplikasi telah diujikan. Terdapat 3 macam fungsi yang diujikan yaitu “Form Data Wilayah” untuk mengisikan data mengenai lahan, “Tombol Fungsi” berupa tombol-tombol yang dapat di *click* dan “Fungsi Scroll” untuk menggeser halaman aplikasi ke atas dan ke bawah. Adapun bentuk pengujian fungsional Aplikasi Budidaya Padi Salibu pada *smartphone Android* dapat dilihat pada Tabel 4.2 berikut.

Tabel 4.2 Pengujian fungsional berupa *Input – Output* Aplikasi

No	Komponen Pengujian	<i>Input</i>	<i>Output</i>	Status
1	<i>Form Data Wilayah</i> (Gambar 4.1a)	Data wilayah diisikan ke dalam <i>form</i> pengisian data	Data wilayah dapat diisi dengan baik dan dapat ditampilkan sesuai yang diisikan.	Berhasil
2	Tombol Fungsi (Gambar 4.2b)	Menekan tombol-tombol fungsi seperti tombol “lanjut”, tombol “kembali” yang ada pada tiap halaman dan tombol “keluar” yang ada pada halaman akhir	Tombol “lanjut” akan menampilkan halaman berikutnya, tombol “kembali” menampilkan halaman sebelumnya dan tombol “keluar” untuk keluar dari aplikasi	Berhasil
3	Fungsi <i>Scroll</i> (Gambar 4.2c)	Menggeser halaman tampilan ke atas dan ke bawah pada setiap halaman yang bersangkutan	Halaman tampilan dapat digeser ke atas dan ke bawah dengan baik	Berhasil

4.3.2 Hasil Pengujian Akurasi Sistem (*White Box Test*)

Berdasarkan kasus-kasus yang ada di lapangan, pengujian ini dilakukan dengan cara meminta kepada ahli/pakar budidaya Padi Salibu untuk memberikan 30 contoh kasus beserta hasil keputusannya. Dimana untuk setiap kasusnya memiliki kondisi parameter lahan yang berbeda-beda. Pada Tabel 4.3 dapat dilihat kondisi parameter lahan yang terdapat di beberapa wilayah di Indonesia.

Tabel 4.3 Kondisi Lahan Beberapa Wilayah di Indonesia

No	Nama Daerah	Parameter					Keputusan Pakar
		Jenis Irigasi	Kondisi Drainase	Kondisi Hamparan	Kadar Organik	Tekstur Tanah	
1	Sei. Tarab, Batusangkar	Irigasi Teknis	Lancar	Miring	Tinggi	Sedang (Lempung)	Cocok
2	Nagari Tabek, Tanah Datar	Irigasi ½ Teknis	Lancar	Miring	Tinggi	Sedang (Lempung)	Cocok
3	Rambatan, Tanah Datar	Irigasi Tadah Hujan	Lancar	Miring	Tinggi	Sedang (Lempung)	Tidak Cocok
4	Lima Kaum, Tanah Datar	Irigasi Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok
5	Batipuh, Tanah Datar	Irigasi ½ Teknis	Lancar	Miring	Tinggi	Sedang (Lempung)	Cocok
6	X Koto, Tanah Datar	Irigasi ½ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok
7	Singkarak, Solok	Irigasi Pompa	Sedikit Terhambat	Datar	Tinggi	Halus (Liat)	Cocok Bersyarat
8	Sumani, Solok	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Halus (Liat)	Cocok Bersyarat
9	Selayo, Solok	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Sedang (Lempung)	Cocok Bersyarat
10	Talang, Solok	Irigasi Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok
11	Cupak, Solok	Irigasi ½ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok
12	Sungai Lasi, Solok	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedang	Kasar (Pasir)	Cocok Bersyarat
13	Parabek, Agam	Irigasi ½ Teknis	Lancar	Miring	Tinggi	Sedang (Lempung)	Cocok
14	Matur, Agam	Irigasi ½ Teknis	Lancar	Datar	Sedang	Kasar (Pasir)	Cocok Bersyarat
15	Lubuk Basung, Agam	Irigasi ½ Teknis	Lancar	Datar	Sedang	Sedang (Lempung)	Cocok
16	Biaro, Agam	Irigasi ½ Teknis	Lancar	Datar	Sedang	Kasar (Pasir)	Cocok
17	Kuranji, Padang	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedikit (Kurang)	Halus (Liat)	Tidak Cocok
18	Koto Tangah, Padang	Irigasi Teknis	Lancar	Datar	Sedang	Sedang (Lempung)	Cocok

19	Pauh, Padang	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Sedang (Lempung)	Cocok
20	Lubuk Alung, Padang Pariaman	Irigasi Teknis	Sedikit Terhambat	Cekung (Lembah)	Sedang	Kasar (Pasir)	Tidak Cocok
21	Buayan, Padang Pariaman	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Halus (Liat)	Cocok Bersyarat
22	Kasang, Padang Pariaman	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedang	Halus (Liat)	Cocok Bersyarat
23	Sitiung, Dharmasraya	Irigasi Teknis	Sedikit Terhambat	Datar	Sedikit (Kurang)	Sedang (Lempung)	Cocok Bersyarat
24	Sijunjung	Irigasi Tadah Hujan	Sedikit Terhambat	Datar	Sedang	Halus (Liat)	Tidak Cocok
25	Karawang	Irigasi Teknis	Terhambat	Datar	Sedang	Halus (Liat)	Tidak Cocok
26	Sukamandi	Irigasi Teknis	Terhambat	Datar	Sedikit (Kurang)	Halus (Liat)	Tidak Cocok
27	Purwakarta	Irigasi Teknis	Lancar	Miring	Tinggi	Halus (Liat)	Cocok
28	Brebes	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Sedang (Lempung)	Cocok
29	Lombok	Irigasi ½ Teknis	Sedikit Terhambat	Miring	Tinggi	Sedang (Lempung)	Cocok
30	Muara Enim	Irigasi Pompa	Sedikit Terhambat	Datar	Tinggi	Sedang (Lempung)	Cocok Bersyarat

Setelah didapatkan hasil dari 30 kasus yang berbeda berupa data wilayah yang diberikan pakar, selanjutnya hasil tersebut diujikan kembali menggunakan Aplikasi Sistem Pakar Padi Salibu yang telah dirancang. Hasil yang didapatkan dari aplikasi selanjutnya akan dibandingkan dengan hasil yang ada di lapangan (Tabel 4.3). Hasil perbandingan antara keputusan pakar dan keputusan aplikasi dapat dilihat pada tabel 4.4.

Tabel 4.4 Perbandingan Hasil Keputusan

No	Nama Daerah	CF	Keputusan Pakar	Keputusan Aplikasi	Keterangan
1	Sei. Tarab, Batusangkar	0.9892	Cocok	Cocok	Sama
2	Nagari Tabek, Tanah Datar	0.9796	Cocok	Cocok	Sama
3	Rambatan, Tanah Datar	0.956	Tidak Cocok	Tidak Cocok	Sama
4	Lima Kaum, Tanah Datar	0.9892	Cocok	Cocok	Sama
5	Batipuh, Tanah Datar	0.9796	Cocok	Cocok	Sama
6	X Koto, Tanah Datar	0.9616	Cocok	Cocok	Sama
7	Singkarak, Solok	0.8937	Cocok Bersyarat	Cocok Bersyarat	Sama
8	Sumani, Solok	0.9139	Cocok Bersyarat	Cocok Bersyarat	Sama
9	Selayo, Solok	0.9276	Cocok Bersyarat	Cocok	Berbeda
10	Talang, Solok	0.9796	Cocok	Cocok	Sama
11	Cupak, Solok	0.9616	Cocok	Cocok	Sama
12	Sungai Lasi, Solok	0.6659	Cocok Bersyarat	Cocok Bersyarat	Sama
13	Parabek, Agam	0.9796	Cocok	Cocok	Sama
14	Matur, Agam	0.8231	Cocok Bersyarat	Cocok Bersyarat	Sama
15	Lubuk Basung, Agam	0.9276	Cocok	Cocok	Sama
16	Biaro, Agam	0.8231	Cocok	Cocok Bersyarat	Berbeda
17	Kuranji, Padang	0.6485	Tidak Cocok	Tidak Cocok	Sama

18	Koto Tangah, Padang	0.9616	Cocok	Cocok	Sama
19	Pauh, Padang	0.9276	Cocok	Cocok	Sama
20	Lubuk Alung, Padang Pariaman	0.4998	Tidak Cocok	Tidak Cocok	Sama
21	Buayan, Padang Pariaman	0.9139	Cocok Bersyarat	Cocok Bersyarat	Sama
22	Kasang, Padang Pariaman	0.8375	Cocok Bersyarat	Cocok Bersyarat	Sama
23	Sitiung, Dharmasraya	0.8433	Cocok Bersyarat	Cocok Bersyarat	Sama
24	Sijunjung	0	Tidak Cocok	Tidak Cocok	Sama
25	Karawang	0.7567	Tidak Cocok	Tidak Cocok	Sama
26	Sukamandi	0.4738	Tidak Cocok	Tidak Cocok	Sama
27	Purwakarta	0.9872	Cocok	Cocok	Sama
28	Brebes	0.9276	Cocok	Cocok	Sama
29	Lombok	0.9616	Cocok	Cocok	Sama
30	Muara Enim	0.9105	Cocok Bersyarat	Cocok Bersyarat	Sama

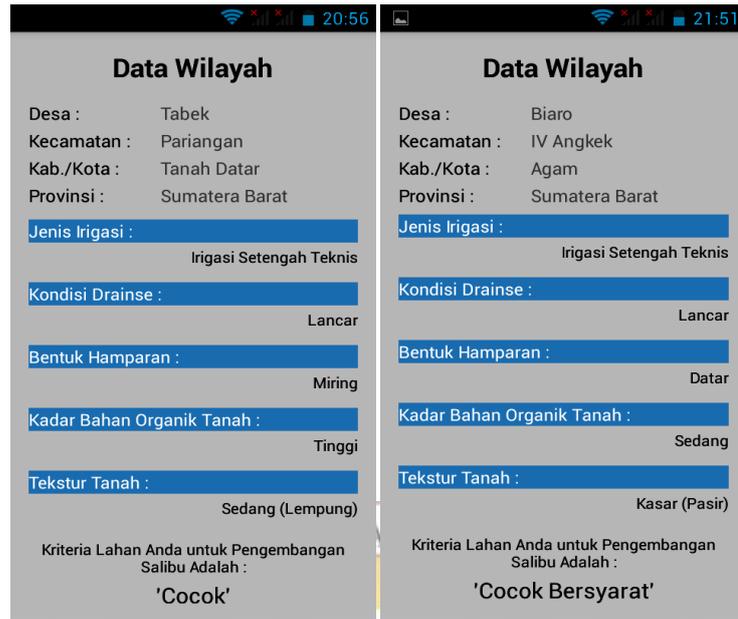
Pada Tabel 4.4 dapat dilihat terdapat 2 hasil keputusan yang berbeda (berwarna merah) antara keputusan pakar dan keputusan aplikasi. Hasil keputusan yang berbeda terjadi pada Nagari Selayo, Kab. Solok dan Nagari Biaro, Kab. Agam.

Pada Nagari Selayo, Kab. Solok, terdapat pertimbangan khusus dari pakar yang menyebabkan keputusan dari pakar dan aplikasi menjadi berbeda. Pada aplikasi, hasil dari nilai CF-nya yaitu 0.92755. Apabila dilihat dari nilai *threshold*-nya, maka nilai $CF = 0.92755$ masuk pada rentang keputusan “Cocok”

($CF > 0.9275$). Namun, pakar memiliki penilaian lain yaitu pada parameter kondisi drainase dan hamparannya. Menurut pakar, lahan tersebut akan sulit untuk diatur ketinggian air genangannya karena memiliki drainase yang sedikit terhambat, ditambah lagi karena lahannya memiliki hamparan yang datar. Sehingga, walaupun dilakukan perbaikan pada bagian drainasinya, air akan tetap tergenang karena faktor lahan yang datar.

Pada Nagari Biaro, Kab. Agam, pakar menentukan bahwa kondisi lahannya adalah “Cocok” untuk dilakukan budidaya salibu. Hal ini dikarenakan tekstur tanahnya yang kasar (berupa pasir), sehingga proses kehilangan airnya akan cepat dan air tidak akan menggenang lama pada lahan walaupun kondisi hamparannya datar. Namun, aplikasi memiliki keputusan lain yaitu “Cocok”, hal ini dikarenakan nilai CF -nya 0.8231. Sehingga apabila dilihat pada nilai *threshold* yang ada, nilai $CF = 0.8231$ masuk pada rentang keputusan “Cocok Bersyarat” ($0.76 < CF \leq 0.9275$).

Secara keseluruhan, pada Tabel 4.4 terdapat 28 kasus hasil keputusan yang sesuai/sama dari total 30 kasus. Hasil tersebut akan dibandingkan persentasinya dengan cara membandingkan antara jumlah kasus dengan keputusan yang benar dan jumlah keseluruhan kasus. Pada Gambar 4.1 terlihat contoh hasil keputusan menggunakan Aplikasi Sistem Pakar Padi Salibu.



(a) (b)
Gambar 4.1 Tampilan Hasil Program : (a) Hasil yang sama dengan pakar, dan (b) Hasil yang berbeda dengan pakar.

Persentasi akurasi didapatkan dengan cara melakukan perhitungan yang ada pada persamaan (1) berikut.

$$\% \text{ Akurasi} = \frac{\text{Jumlah kasus yang benar}}{\text{Jumlah Kasus Keseluruhan}} \times 100 \% = \frac{28}{30} \times 100 \% = 93,33\% \dots (1)$$

Setelah diujikan, didapatkan hasil akurasi sebesar 93,33 %. Dari hasil tersebut, dapat disimpulkan bahwa pengujian akurasi ini dinyatakan berhasil namun belum sempurna karena tingkat akurasinya belum mencapai 100%.

Hasil yang tidak sempurna ini disebabkan karena metode *Certainty Factor* yang digunakan masih belum bisa menyamakan hasil yang ada antara pakar dengan sistem yang dirancang. Hal ini dikarenakan pakar dapat melakukan pertimbangan-pertimbangan lebih pada suatu lahan untuk menentukan keputusan akhir, sedangkan sistem yang dirancang menentukan keputusan hanya mengandalkan *rule-rule* yang telah ditetapkan saja.

4.3.3 Hasil Pengolahan Keseluruhan Data Padi Salibu

Pada tahap ini menjelaskan mengenai hasil dari pengolahan data-data Padi Salibu yang baru setelah dilakukannya pengujian akurasi aplikasi. Data-data yang diolah merupakan data yang didapatkan dari pakar sesuai dengan kejadian yang ada di lapangan sebanyak 15 kasus selama pakar melakukan penelitian dan kunjungan ke berbagai wilayah. Pada Tabel 4.5 ditampilkan data yang akan diolah.

Tabel 4.5 Data Wilayah Oleh Pakar

No	Nama Daerah	Parameter					Keputusan Pakar
		Jenis Irigasi	Kondisi Drainase	Kondisi Hamparan	Kadar Organik	Tekstur Tanah	
1	Lintau Buo, Tanah Datar	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedikit	Halus (Liat)	Tidak Cocok
2	Tanjung Emas, Tanah Datar	Irigasi Tadah Hujan	Lancar	Datar	Sedikit	Kasar (Pasir)	Tidak Cocok
3	X Koto Diatas, Solok	Irigasi ½ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok
4	Pulau Punjung, Dharmasraya	Irigasi Teknis	Lancar	Datar	Sedang	Sedang (Lempung)	Cocok
5	Padang Laweh, Dharmasraya	Irigasi ½ Teknis	Lancar	Datar	Sedikit	Kasar (Pasir)	Cocok Bersyarat
6	Tarusan, Pesisir Selatan	Irigasi Tadah Hujan	Terhambat	Datar	Sedikit	Halus (Liat)	Tidak Cocok
7	Surantih, Pesisir Selatan	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedang	Sedang (Lempung)	Cocok Bersyarat
8	Balai Selasa, Pesisir Selatan	Irigasi Teknis	Lancar	Datar	Sedang	Sedang (Lempung)	Cocok
9	Muaro Bungo, Jambi	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedikit	Halus (Liat)	Tidak Cocok
10	Pagar Alam, Sumatera Selatan	Irigasi ½ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok
11	Taram, Lima Puluh Kota	Irigasi ½ Teknis	Lancar	Datar	Sedikit	Kasar (Pasir)	Cocok Bersyarat

12	Tarantang, Lima Puluh Kota	Irigasi ½ Teknis	Terhambat	Cekung (Lembah)	Sedang	Halus (Liat)	Tidak Cocok
13	Kinali, Pasaman Barat	Irigasi Teknis	Lancar	Datar	Tinggi	Sedang (Lempung)	Cocok
14	Kapar, Pasaman Barat	Irigasi Teknis	Lancar	Datar	Sedang	Kasar (Pasir)	Cocok Bersyarat
15	Padang Aro, Solok Selatan	Irigasi ½ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok

Data-data yang terdapat pada Tabel 4.5 selanjutnya akan diolah, lalu dilakukan kembali uji akurasi. Adapun hasil perbandingan yang di dapat dari pengolahan data-data dimasukkan ke dalam Tabel 4.6 sebagai berikut.

Tabel 4.6 Hasil Perbandingan Data

No	Nama Daerah	CF	Keputusan Pakar	Keputusan Aplikasi	Keterangan
1	Lintau Buo, Tanah Datar	0.6486	Tidak Cocok	Tidak Cocok	Sama
2	Tanjung Emas, Tanah Datar	0.1727	Tidak Cocok	Tidak Cocok	Sama
3	X Koto Diatas, Solok	0.9616	Cocok	Cocok	Sama
4	Pulau Punjung, Dharmasraya	0.9616	Cocok	Cocok	Sama
5	Padang Laweh, Dharmasraya	0.6175	Cocok Bersyarat	Tidak Cocok	Berbeda
6	Tarusan, Pesisir Selatan	-0.5347	Tidak Cocok	Tidak Cocok	Sama
7	Surantih, Pesisir Selatan	0.8631	Cocok Bersyarat	Cocok Bersyarat	Sama
8	Balai Selasa, Pesisir Selatan	0.9616	Cocok	Cocok	Sama
9	Muaro Bungo, Jambi	0.6485	Tidak Cocok	Tidak Cocok	Sama
10	Pagar Alam, Sumatera Selatan	0.9616	Cocok	Cocok	Sama
11	Taram, Lima Puluh Kota	0.6175	Cocok Bersyarat	Tidak Cocok	Berbeda

12	Tarantang, Lima Puluh Kota	-0.2305	Tidak Cocok	Tidak Cocok	Sama
13	Kinali, Pasaman Barat	0.9796	Cocok	Cocok	Sama
14	Kapar, Pasaman Barat	0.9063	Cocok Bersyarat	Cocok Bersyarat	Sama
15	Padang Aro, Solok Selatan	0.9616	Cocok	Cocok	Sama

Dari data pada Tabel 4.6 dapat dilihat terdapat 2 hasil keputusan yang berbeda (berwarna merah) antar pakar dengan aplikasi. Hal ini dikarenakan pada kedua data tersebut memiliki nilai CF yang rendah (menurut aplikasi). Parameter kondisi hamparan yang datar, kadar organik yang sedikit serta tekstur tanah yang berpasir membuat nilai CF menjadi rendah, sehingga nilai CF berada pada kondisi *threshold* "Cocok Bersyarat". Sedangkan ahli/pakar memiliki penilaian sendiri yang tidak bisa dilakukan oleh aplikasi.

Berdasarkan Tabel 4.6, terdapat 2 perbedaan keputusan dari total 15 data yang tersedia sehingga tingkat akurasi dari aplikasi yang dirancang yaitu :

$$\% \text{ Akurasi} = \frac{\text{Jumlah kasus yang benar}}{\text{Jumlah Kasus Keseluruhan}} \times 100 \% = \frac{13}{15} \times 100 \% = 86,7\% \dots \dots (2)$$

Setelah diujikan, didapatkan hasil akurasi sebesar 86,7%. Apabila data yang ada pada Tabel 4.4 dan Tabel 4.6 digabungkan, maka jumlah total keseluruhan data sebanyak 45 data. Dari 45 data yang diujikan, terdapat 4 data dengan keputusan yang berbeda dan 41 data dengan keputusan yang sama. Sehingga apabila dihitung tingkat akurasinya akan menjadi :

$$\% \text{ Akurasi} = \frac{\text{Jumlah kasus yang benar}}{\text{Jumlah Kasus Keseluruhan}} \times 100 \% = \frac{41}{45} \times 100 \% = 91,11\% \dots \dots (3)$$

Dari hasil pengolahan seluruh 45 data yang dilakukan, didapat akurasi sebesar 91,11% (persamaan 3). Dengan tingkat akurasi yang cukup tinggi, dapat

disimpulkan bahwa pengolahan data menggunakan sistem yang dirancang berhasil.

4.4 Hasil Pengujian Fungsionalitas & Tampilan Aplikasi

Pada tahap ini menjelaskan bagaimana hasil dari pengedaran kuesioner yang telah dijelaskan pada subbab 3.5. Adapun hasil dari kuesioner akan dibagi menjadi 2 bagian, yaitu kepuasan terhadap fungsionalitas aplikasi dan kepuasan terhadap tampilan aplikasi.

4.4.1 Hasil Kepuasan Fungsionalitas

Berdasarkan subbab 2.5, untuk menghitung tingkat persentase kepuasan responden terhadap fungsionalitas aplikasi, maka dilakukan suatu perhitungan menggunakan metode Likert. Seluruh perhitungan menggunakan Metode Likert dapat dilihat pada Lampiran F-1.

1. Tani / Kelompok Tani

Setelah dilakukan perhitungan pada semua jawaban pertanyaan yang ada, maka didapatkan persentase kepuasan untuk setiap pertanyaan. Hasil persentase jawaban untuk seluruh pertanyaan dapat dilihat pada Tabel 4.7 berikut.

Tabel 4.7 Hasil Kuesioner Fungsionalitas Tani

	No	Nama Responden	Jawaban Pertanyaan ke-					
			Ke - 1	Ke - 2	Ke - 3	Ke - 4	Ke - 5	Ke - 6
Tenaga Ahli	1	Yeldi Mardi	1	1	1	3	1	2
	2	Rahmawati	1	1	1	3	1	2
	3	Busrialman	1	1	1	2	1	1
	4	Hendri	1	1	2	3	1	1
	5	Afrizul	1	2	1	3	1	2
	6	Mulyati	1	1	1	3	1	1
	7	Suhatman	1	1	1	3	1	1
	8	Asmalinda	1	1	1	2	1	2
	9	Joni	1	1	1	2	1	2
	10	Muchlis	1	2	1	2	2	2
			100%	93,33%	96,67%	86,7%	96,67%	88,00%
			Rata-rata =				93,56%	

Pada Tabel 4.7 dapat terlihat jawaban-jawaban yang diberikan oleh responden pada setiap pertanyaannya. Pada pertanyaan ke-1 (Apakah aplikasi terinstall dengan baik ?), semua responden menjawab dengan jawaban 1 (Ya), sehingga persentase kepuasan pertanyaan 1 adalah 100 %. Sedangkan pertanyaan ke-2 (Bagaimana respon aplikasi saat dibuka ?), terdapat 2 orang yang menjawab dengan jawaban 2 (Kurang Cepat). Hal ini dapat dikarenakan *Smartphone* Android yang digunakan responden memiliki *memory* yang kecil dan hampir penuh. Sehingga persentase kepuasan responden untuk pertanyaan 2 sebesar 93,33%. Untuk pertanyaan ke-3 (Apakah semua fungsi aplikasi berjalan dengan baik ?), terdapat 1 orang responden yang menjawab dengan jawaban 2 (Kurang). Hal ini dapat dikarenakan layar *smartphone* Android responden memiliki masalah. Sehingga persentase kepuasan responden untuk pertanyaan ke-3 adalah 96,67%.

Pada pertanyaan ke-4 (Apakah terdapat *error* pada Android saat menjalankan aplikasi ?), terdapat 4 orang yang menjawab dengan jawaban 2 (Sedikit) dan 6 orang yang menjawab dengan jawaban 3 (Tidak). Hal ini dapat dikarenakan proses instalasi aplikasi yang tidak benar atau kapasitas RAM *smartphone* android responden yang kecil dan hampir penuh. Sehingga persentase kepuasannya menjadi 86,7 %. Untuk pertanyaan ke-5 (Apakah aplikasi berfungsi dengan baik ?), terdapat 1 orang responden yang menjawab dengan jawaban 2 (Kurang Baik) dan 9 orang menjawab dengan jawaban 1 (Baik). Sehingga didapatkan persentase kepuasan sebesar 96,67 %. Pada pertanyaan ke-6 (Seberapa puas dalam menggunakan aplikasi secara keseluruhan), terdapat 4 orang responden yang menjawab dengan jawaban 1 (Sangat Puas) dan 6 orang yang

menjawab dengan jawaban 2 (Puas). Sehingga didapatkanlah persentase kepuasan responden sebesar 88 %.

Dari seluruh persentase yang didapat dari masing-masing jawaban pertanyaan, maka didapatkan nilai rata-rata persentase kepuasan responden dari tani/kelompok tani terhadap fungsionalitas aplikasi sebesar 93,56%. Dari hasil tersebut, dapat disimpulkan bahwa responden dari tani/kelompok tani puas terhadap fungsionalitas aplikasi yang dirancang.

2. Tenaga Ahli

Setelah dilakukan perhitungan pada semua jawaban pertanyaan yang ada, maka didapatkan persentase kepuasan untuk setiap pertanyaan. Hasil persentase jawaban untuk seluruh pertanyaan dapat dilihat pada Tabel 4.8 berikut.

Tabel 4.8 Hasil Kuesioner Fungsionalitas Ahli

	No	Nama Responden	Jawaban Pertanyaan ke-					
			Ke - 1	Ke - 2	Ke - 3	Ke - 4	Ke - 5	Ke - 6
Tenaga Ahli	1	Jefri Chandra, SP	1	1	1	3	1	2
	2	Ir. Erdiman	1	1	1	3	1	2
	3	Murni Kurniaty, STP, MP	1	1	1	2	1	2
	4	Elviana Anwar	1	3	1	3	2	2
	5	Aldilani	1	1	1	3	1	2
	6	Ikhwatul Khairiyah, SP	1	1	1	2	1	2
	7	Gunawan, SE	1	1	1	3	1	2
	8	Syamsul Bahri	1	1	1	3	1	2
	9	Mahdalena, SP	1	1	1	3	2	1
	10	Niedalina, IR, MSi	1	1	1	2	1	2
			100%	93%	100,00%	90%	93%	82,00%
			Rata-rata =				93,11%	

Pada Tabel 4.8 dapat terlihat jawaban-jawaban yang diberikan oleh responden pada setiap pertanyaannya. Pada pertanyaan ke-1 (Apakah aplikasi terinstall dengan baik ?), semua responden menjawab dengan jawaban 1 (Ya), sehingga persentase kepuasan pertanyaan 1 adalah 100 %. Sedangkan pertanyaan ke-2 (Bagaimana respon aplikasi saat dibuka ?), terdapat 2 orang yang menjawab dengan jawaban 3 (Lambat). Hal ini dapat dikarenakan *Smartphone* Android yang digunakan responden memiliki *memory* yang kecil dan sudah penuh. Sehingga persentase kepuasan responden untuk pertanyaan 2 sebesar 93,33%. Untuk pertanyaan ke-3 (Apakah semua fungsi aplikasi berjalan dengan baik ?), seluruh responden menjawab dengan jawaban 1 (Ya). Sehingga persentase kepuasan responden untuk pertanyaan ke-3 adalah 100 %.

Pada pertanyaan ke-4 (Apakah terdapat *error* pada Android saat menjalankan aplikasi ?), terdapat 3 orang yang menjawab dengan jawaban 2 (Sedikit) dan 7 orang yang menjawab dengan jawaban 3 (Tidak). Hal ini dapat dikarenakan proses instalasi aplikasi yang tidak benar atau kapasitas RAM *smartphone* android responden yang kecil dan hampir penuh. Sehingga persentase kepuasannya menjadi 90 %. Untuk pertanyaan ke-5 (Apakah aplikasi berfungsi dengan baik ?), terdapat 2 orang responden yang menjawab dengan jawaban 2 (Kurang Baik) dan 8 orang menjawab dengan jawaban 1 (Baik). Sehingga didapatkan persentase kepuasan sebesar 93 %. Pada pertanyaan ke-6 (Seberapa puas dalam menggunakan aplikasi secara keseluruhan), terdapat 1 orang responden yang menjawab dengan jawaban 1 (Sangat Puas) dan 9 orang yang menjawab dengan jawaban 2 (Puas). Sehingga didapatkanlah persentase kepuasan responden sebesar 82 %.

Dari seluruh persentase yang didapat dari masing-masing jawaban pertanyaan, maka didapatkan nilai rata-rata kepuasan responden dari tani/kelompok tani terhadap fungsionalitas aplikasi sebesar 93,11%. Dari hasil tersebut, dapat disimpulkan bahwa responden dari tenaga ahli puas terhadap fungsionalitas aplikasi yang dirancang.

Apabila hasil dari persentase kepuasan semua responden (tani/kelompok tani dan tenaga ahli) terhadap fungsionalitas aplikasi digabungkan, yaitu sebesar 93,56% dan 93,11%. Maka rata-rata persentase kepuasannya sebesar 93,33%, sehingga dapat disimpulkan bahwa semua responden puas terhadap fungsionalitas aplikasi yang dibuat.

4.4.2 Hasil Kepuasan Tampilan

Untuk menghitung tingkat persentase kepuasan responden terhadap tampilan aplikasi, maka dilakukan suatu perhitungan menggunakan metode Likert. Seluruh perhitungan menggunakan Metode Likert dapat dilihat pada Lampiran F-2.

1. Tani / Kelompok Tani

Setelah dilakukan perhitungan pada semua jawaban pertanyaan yang ada, maka didapatkan persentase kepuasan untuk setiap pertanyaan. Hasil persentase jawaban untuk seluruh pertanyaan dapat dilihat pada Tabel 4.9.

Pada Tabel 4.9 dapat terlihat jawaban-jawaban yang diberikan oleh responden pada setiap pertanyaannya. Pada pertanyaan ke-1 (Seberapa menarik tampilan aplikasi ?), terdapat 3 orang yang menjawab dengan jawaban 1 (Sangat Menarik), 5 orang yang menjawab dengan jawaban 2 (Menarik) dan 2 orang yang menjawab dengan jawaban 3 (Biasa Saja). Hal ini dapat dikarenakan masih

kurang menariknya tampilan aplikasi bagi responden. Sehingga persentase kepuasan pertanyaan 1 menjadi 82 %. Untuk pertanyaan ke-2 (Sudah cukup membantukah tampilan aplikasi ?), seluruh responden menjawab dengan jawaban 1 (Sudah). Sehingga persentase kepuasan responden untuk pertanyaan 2 sebesar 100 %. Pada pertanyaan ke-3 (Apakah masih terdapat kekurangan pada tampilannya ?), seluruh responden menjawab dengan jawaban 1 (Tidak). Sehingga persentase kepuasan responden untuk pertanyaan ke-3 adalah 100 %.

Tabel 4.9 Hasil Kuesioner Tampilan Tani

	No	Nama Responden	Jawaban Pertanyaan ke-		
			1	2	3
Tani/Kelompok Tani	1	Yeldi Mardi	2	1	1
	2	Rahmawati	1	1	1
	3	Busrialman	1	1	1
	4	Hendri	1	1	1
	5	Afrizul	3	1	1
	6	Mulyati	2	1	1
	7	Suhatman	2	1	1
	8	Asmalinda	3	1	1
	9	Joni	2	1	1
	10	Muchlis	2	1	1
			82%	100%	100%
			Rata-rata =		94%

Dari seluruh persentase yang didapat dari masing-masing jawaban pertanyaan, maka didapatkan nilai rata-rata kepuasan responden dari tani/kelompok tani terhadap tampilan aplikasi sebesar 94%. Dari hasil tersebut, dapat disimpulkan bahwa responden dari tani / kelompok tani puas terhadap tampilan aplikasi yang dirancang.

2. Tenaga Ahli

Setelah dilakukan perhitungan pada semua jawaban pertanyaan yang ada, maka didapatkan persentase kepuasan untuk setiap pertanyaan. Hasil persentase jawaban untuk seluruh pertanyaan dapat dilihat pada Tabel 4.10 berikut.

Tabel 4.10 Hasil Kuesioner Tampilan Ahli

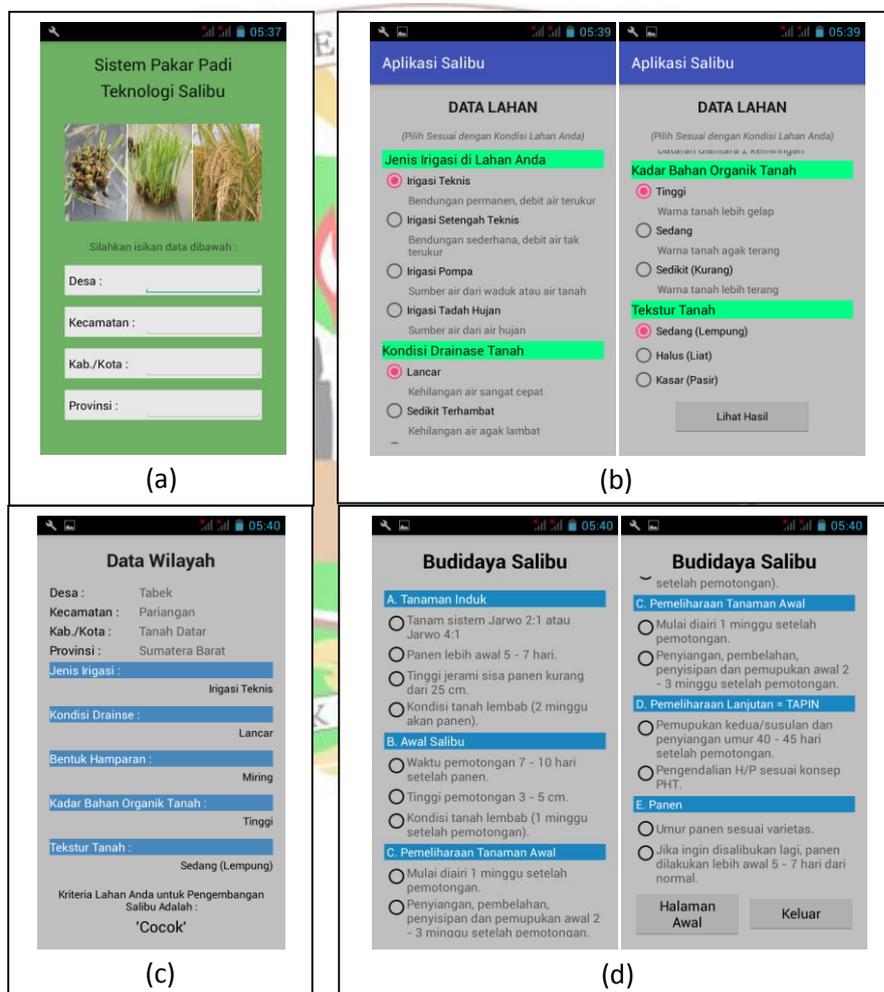
	No	Nama Responden	Jawaban Pertanyaan ke-		
			1	2	3
Tani/Kelompok Tani	1	Jefri Chandra, SP	2	1	1
	2	Ir. Erdiman	1	1	2
	3	Murni Kurniaty, STP, MP	3	2	1
	4	Elviana Anwar	3	2	1
	5	Aldilani	2	1	2
	6	Ikhwatul Khairiyah, SP	2	1	1
	7	Gunawan, SE	2	1	1
	8	Syamsul Bahri	2	1	1
	9	Mahdalena, SP	2	1	1
	10	Niedalina, IR, Msi	2	1	2
			78%	93%	85%
			Rata-rata =		87%

Pada Tabel 4.10 dapat terlihat jawaban-jawaban yang diberikan oleh responden pada setiap pertanyaannya. Pada pertanyaan ke-1 (Seberapa menarik tampilan aplikasi ?), terdapat 1 orang yang menjawab dengan jawaban 1 (Sangat Menarik), 7 orang yang menjawab dengan jawaban 2 (Menarik) dan 2 orang yang menjawab dengan jawaban 3 (Biasa Saja). Hal ini dapat dikarenakan masih kurang menariknya tampilan aplikasi bagi responden. Sehingga persentase kepuasan pertanyaan 1 menjadi 78 %. Untuk pertanyaan ke-2 (Sudah cukup membantukah tampilan aplikasi ?), terdapat 8 orang yang menjawab dengan jawaban 1 (Sudah) dan 2 orang yang menjawab dengan jawaban 2 (Belum Membantu). Sehingga persentase kepuasan responden untuk pertanyaan 2 sebesar 93 %. Pada pertanyaan ke-3 (Apakah masih terdapat kekurangan pada tampilannya ?), terdapat 7 responden yang menjawab dengan jawaban 1 (Tidak) dan 3 orang responden yang menjawab dengan jawaban 2 (Ada). Sehingga persentase kepuasan responden untuk pertanyaan ke-3 adalah 85 %.

Dari seluruh persentase yang didapat dari masing-masing jawaban pertanyaan, maka didapatkan nilai rata-rata kepuasan responden dari tenaga ahli terhadap tampilan aplikasi sebesar 87%. Dari hasil tersebut, dapat disimpulkan

bahwa responden dari tenaga ahli cukup puas terhadap tampilan aplikasi yang dirancang.

Apabila hasil dari persentase kepuasan semua responden (tani/kelompok tani dan tenaga ahli) terhadap tampilan aplikasi digabungkan, yaitu sebesar 94% dan 87%. Maka rata-rata persentasenya sebesar 90,5%, sehingga dapat disimpulkan bahwa semua responden puas terhadap tampilan aplikasi yang dibuat. Pada Gambar 4.2 dapat dilihat hasil dari tampilan yang dirancang.



Gambar 4.2 Tampilan Aplikasi Keseluruhan

4.5 Hasil Penggunaan Pada Sistem Android

Setelah aplikasi diujikan dan dijalankan, maka dilihat bagaimana hasil penggunaan sistem yang digunakan oleh aplikasi. Maka didapatkan hasil sebagai berikut :

1. Minimal OS Android v4.1 (*Jelly Bean*).
2. Memori internal yang digunakan = 5.89 MB (Gambar 4.3a)
3. RAM yang digunakan = 6.3 MB (Gambar 4.3b)
4. Ukuran file aplikasi = 3.5 MB (Gambar 4.3c)



Gambar 4.3 Hasil Penggunaan Aplikasi : (a) Memori yang digunakan, (b) RAM yang digunakan, (c) Ukuran file

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan pengujian dan analisa Aplikasi Sistem Pakar Padi Salibu, maka dapat diambil kesimpulan sebagai berikut :

1. Sistem pakar yang telah dirancang menggunakan metode *Certainty Factor* telah dapat memberikan hasil keputusan yang cukup baik dengan akurasi yang diperoleh yaitu sebesar 91,11 % dari total keseluruhan pengujian sebanyak 45 kasus / data.
2. *Rule-rule* yang didapatkan dengan *threshold* nilai CF setelah melakukan pengujian dan pelatihan pada 30 data adalah sebagai berikut :
 - *Rule 1* : Keputusan “Cocok” jika nilai $CF > 0.9275$
 - *Rule 2* : Keputusan “Cocok Bersyarat” jika nilai $0.76 < CF \leq 0.9275$
 - *Rule 3* : Keputusan “Tidak Cocok” jika nilai $CF \leq 0.76$
3. Fungsi-fungsi yang terdapat pada aplikasi yang dirancang bekerja dengan baik dengan persentase kepuasan responden sebesar 93,33%.
4. Tampilan aplikasi yang dirancang sudah cukup baik, dengan persentase kepuasan responden sebesar 90,5%.

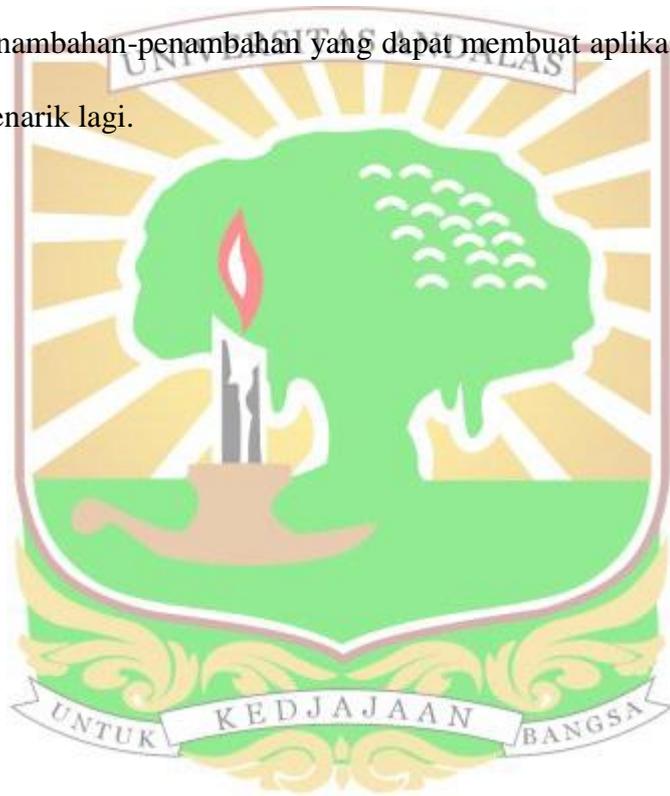
5.2 Saran

Berdasarkan kesimpulan diatas, penulis memberikan saran sebagai berikut:

1. Aplikasi yang dirancang masih belum sempurna dapat dilihat dari akurasi yang belum mencapai 100%, oleh sebab itu sebaiknya coba

gunakan metode sistem pakar lain untuk merancang aplikasi yang serupa.

2. Untuk pengembangan tugas akhir selanjutnya, aplikasi dapat ditambahkan dengan beberapa fitur lainnya seperti : penambahan GPS untuk mengetahui lokasi secara otomatis, tambahan menu penyimpanan, serta tampilan yang dibuat lebih menarik.
3. Tampilan aplikasi masih kurang menarik, sehingga diperlukan penambahan-penambahan yang dapat membuat aplikasi menjadi lebih menarik lagi.



DAFTAR PUSTAKA

- [1] Badan Pusat Statistik. <http://bps.go.id>
- [2] Erdiman. 2012. *Teknologi Salibu Meningkatkan Produktivitas Lahan (3-6 Ton/Ha/Tahun) dan Pendapatan Petani (Rp.15-25 Juta/Tahun)*. (Balai Pengkajian Teknologi Pertanian Sumatera Barat).
- [3] Susilawati, 2011. *Agronomi Ratus Genotipe – Genotipe Padi Potensial Untuk Lahan Pasang Surut*. Disertasi Pasca Sarjana Institut Pertanian Bogor, 94 Hal.
- [4] Rivaldi. 2015. *Pertumbuhan dan Hasil Padi (Oryza Sativa L.) Salibu Varietas Hibrida Pada Tinggi dan Waktu Penggenangan*. Padang: Universitas Taman Siswa
- [5] Puspitasari, Denok. 2010. *Sistem Pakar Diagnosa Diabetes Nefropathy Dengan Metode Certainty Factor Berbasis Web dan Mobile*. Surabaya : Politeknik Elektronik Negeri Surabaya.
- [6] Das, Subrata. 2014. *Computational Business Analytics*. Boca Raton: CRC Press.
- [7] BBPADI. *Panduan Teknologi Budidaya Padi Salibu*. Dikutip pada bulan Mei 2016 dari :
<http://bbpadi.litbang.pertanian.go.id/index.php/publikasi/panduan-teknis/content/item/235-panduan-teknologi-budidaya-padi-salibu>
- [8] Oktaviani. *Analisis Uji Komparasi Sistem Operasi pada Android dan Blackberry*. Diakses pada Mei 2016 dari :
<http://oktaviani.staff.gunadarma.ac.id/Publications/files/2134/android.pdf>
- [9] Khotimah, Nurul. *Perkembangan dan Versi Android dari Waktu ke Waktu*. Diakses pada Mei 2016 dari :
<http://ilmuti.org/2014/03/07/perkembangan-dan-versi-android-dari-waktu-kewatu/>
- [10] Mahardika, Gede. 2012. *Pengembangan Sistem Pakar Diagnosa Hama dan Penyakit Tanaman Padi Menggunakan Metode Certainty Factor Berbantuan Metode Kuantifikasi Pertanyaan*. Singaraja: Universitas Pendidikan Ganesha.
- [11] Amalia, Nanda. 2015. *Rancang Bangun Sistem Pakar Identifikasi Penyakit Fisik Akibat Kerja Dengan Metode Certainty Factor*. Surabaya: Universitas Negeri Surabaya.

- [12] Budi, Gregorius S, Rolly Intan. *Probabilitas Penggunaan Premis untuk menentukan Certainty Factor dari Rule*. Surabaya: Universitas Kristen Petra.
- [13] Anonim. 2013. *Metode Certainty Factor (CF)*. Diakses pada bulan mei 2016 dari: <http://www.metode-algoritma.com/2013/06/contoh-certainty-factor-cf.html>
- [14] TWOH&Co. 2014. *Belajar Membuat Aplikasi Android Menggunakan Android Studio*. Diakses pada bulan Mei 2016 dari : <http://www.twoh.co/2014/09/belajar-membuat-aplikasi-android-menggunakan-android-studio/>
- [15] Humala, Arief. 2012. *Pembuatan Aplikasi Pencarian Halte Transjakarta Terdekat Berbasis Android*. Jakarta: Universitas Gunadarma.
- [16] A. W, Ebranda dkk. 2013. *Penerapan Metode Naive Bayes Untuk Sistem Klasifikasi SMS Pada Smartphone Android*. Palembang: STMIK-MDP
- [17] Nugraha L, Hendra. 2015. *Perancangan Penunjuk Rute Pada Kendaraan Pribadi Menggunakan Aplikasi Mobile GIS Berbasis Android Yang Terintegrasi Pada Google Maps*. Manado: Universitas Sam Ratulangi
- [18] Dwi L, Yunus. 2012. *Implementasi Android SDK Dalam Pembuatan Aplikasi Mobile "Kalkulator MIPA" Untuk Sekolah Menengah Pertama*. Yogyakarta: STMIK AMIKOM
- [19] Sugiono. 2012. *Metode Penelitian Kuantitatif, Kualitatif dan R & D*. Bandung : ALFABETA
- [20] Handayani, Febria Sri. 2014. *Contoh Terapan Perhitungan Manual dan Analisa Hasil Kuesioner Menggunakan Skala Likert*. Diakses pada bulan September 2016 dari : <http://news.palcomtech.com/contoh-terapan-perhitungan-manual-dan-analisa-hasil-kuesioner-menggunakan-skala-likert/>



LAMPIRAN



Lampiran A

Hasil Pembobotan Oleh Pakar

KUESIONER BUDIDAYA PADI SALIBU

Responden (pakar) yang terhormat,

Saya Rifky Syafrizal Mahasiswa Teknik Elektro Universitas Andalas sedang menyusun Tugas Akhir bermaksud melakukan penelitian dengan judul **"IMPLEMENTASI METODE CERTAINTY FACTOR UNTUK MENILAI KELAYAKAN BUDIDAYA PADI SALIBU BERBASIS ANDROID"**. Oleh sebab itu, perkenankanlah saya untuk meminta sedikit waktu Anda untuk membantu penelitian saya dengan mengisi kuisisioner ini.

a. Petunjuk Pengisian

Pada lembar Kuesioner ini akan diberikan beberapa parameter yang berkaitan mengenai budidaya penanaman padi salibu dan diharapkan kepada responden (pakar) untuk memilih pilihan kondisi (*term*) yang disediakan dan mengisi pada kolom nilai masing-masing parameter yang ada.

b. Pilihan Kondisi (*term*)

Dalam mengisi Kuesioner ini, diberikan beberapa pilihan kepada responden (pakar) berupa kondisi-kondisi agar lebih memudahkan dalam penilaian.

Kondisi	Nilai
Pasti Tidak Ada	1
Hampir Pasti Tidak Ada	2
Kemungkinan Besar Tidak Ada	3
Mungkin Tidak Ada	4
Tidak Tahu	5
Mungkin Ada	6
Kemungkinan Besar Ada	7
Hampir Pasti Ada	8
Pasti Ada	9

c. Pertanyaan

Pada parameter-parameter berikut, seberapa besar kepastian dari pengaruh tiap parameter pada suatu lahan sehingga cocok dan dapat diterapkannya Budidaya Padi Salibu ?

d. Kuesioner

No	Parameter	Pilihan Parameter	Nilai
1	Jenis Irigasi	• Irigasi Teknis	8
		• Setengah Teknis	6
		• Irigasi Pompa	5
		• Tadah Hujan	2
2	Kondisi Drainase Tanah	• Lancar	8
		• Sedikit Terhambat	6
		• Terhambat	3
3	Kondisi Hamparan Lahan	• Miring	8
		• Datar	6
		• Cekung (Lembah)	3
4	Kadar Organik Tanah	• Tinggi	8
		• Sedang	6
		• Rendah (Kurang)	4
5	Tekstur Tanah	• Sedang (Lempung)	7
		• Halus (Liat)	6
		• Kasar (Pasir)	3



Lampiran B

B-1. Data Pengujian dan Pelatihan (30 Data)

B-2. Data Pengolahan Lanjutan (15 Data)

Lampiran B-1

Variasi Data Wilayah dan Kesesuaian Pengembangan Teknologi Salibu

No	Nama Daerah	Parameter					Hasil Lapangan
		Jenis Irigasi	Kondisi Drainase	Kondisi Hamparan	Kadar Organik	Tekstur Tanah	
1	Sei Tarab, Batasangkar	Irigasi Teknis	Lancar	Miring	Tinggi	Sedang (Lempung)	Cocok
2	Nagari Tabek, Tanah Datar	Irigasi ½ Teknis	Lancar	Miring	Tinggi	Sedang (Lempung)	Cocok
3	Rambatan, Tanah Datar	Irigasi Tadah Hujan	Lancar	Miring	Tinggi	Sedang (Lempung)	Tidak Cocok
4	Lima Kaum, Tanah Datar	Irigasi Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok
5	Batipuh, Tanah Datar	Irigasi ¼ Teknis	Lancar	Miring	Tinggi	Sedang (Lempung)	Cocok
6	X Koto, Tanah Datar	Irigasi ¼ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok
7	Singkarak, Solok	Irigasi Pompa	Sedikit Terhambat	Datar	Tinggi	Halus (Liat)	Cocok Bersyarat
8	Sumani, Solok	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Halus (Liat)	Cocok Bersyarat
9	Selayo, Solok	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Sedang (Lempung)	Cocok Bersyarat
10	Talang, Solok	Irigasi Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok
11	Cupak, Solok	Irigasi ¼ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok
12	Sungai Lasi, Solok	Irigasi ¼ Teknis	Sedikit Terhambat	Datar	Sedang	Kasar (Pasir)	Cocok Bersyarat
13	Parabek, Agam	Irigasi ¼ Teknis	Lancar	Miring	Tinggi	Sedang (Lempung)	Cocok
14	Matur, Agam	Irigasi ¼ Teknis	Lancar	Datar	Sedang	Kasar (Pasir)	Cocok Bersyarat
15	Lubuk Basung, Agam	Irigasi ¼ Teknis	Lancar	Datar	Sedang	Sedang (Lempung)	Cocok
16	Biaro, Agam	Irigasi ¼ Teknis	Lancar	Datar	Sedang	Kasar (Pasir)	Cocok
17	Kuranji, Padang	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedikit (Kurang)	Halus (Liat)	Tidak Cocok
18	Koto Tangah, Padang	Irigasi Teknis	Lancar	Datar	Sedang	Sedang (Lempung)	Cocok
19	Pauh, Padang	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Sedang (Lempung)	Cocok

20	Lubuk Alung, Padang Pariaman	Irigasi Teknis	Sedikit Terhambat	Cekung (Lembah)	Sedang	Kasar (Pasir)	Tidak Cocok
21	Buayan, Padang Pariaman	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Halus (Liat)	Cocok Bersyarat
22	Kasang, Padang Pariaman	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedang	Halus (Liat)	Cocok Bersyarat
23	Sitiung, Dharmasraya	Irigasi Teknis	Sedikit Terhambat	Datar	Sedikit (Kurang)	Sedang (Lempung)	Cocok Bersyarat
24	Sijunjung	Irigasi Tadah Hujan	Sedikit Terhambat	Datar	Sedang	Halus (Liat)	Tidak Cocok
25	Karawang	Irigasi Teknis	Terhambat	Datar	Sedang	Halus (Liat)	Tidak Cocok
26	Sukamandi	Irigasi Teknis	Terhambat	Datar	Sedikit (Kurang)	Halus (Liat)	Tidak Cocok
27	Purwakarta	Irigasi Teknis	Lancar	Miring	Tinggi	Halus (Liat)	Cocok
28	Brebes	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Sedang (Lempung)	Cocok
29	Lombok	Irigasi ½ Teknis	Sedikit Terhambat	Miring	Tinggi	Sedang (Lempung)	Cocok
30	Muara Enim	Irigasi Pompa	Sedikit Terhambat	Datar	Tinggi	Sedang (Lempung)	Cocok Bersyarat

Padang, 18 Agustus 2016

(Ir. Erdiman)

Lampiran B-2

Data Wilayah dan Kesesuaian Pengembangan Teknologi Padi Salibu

No	Nama Daerah	Parameter					Hasil Pakar
		Jenis Irigasi	Kondisi Drainase	Kondisi Hamparan	Kadar Organik	Tekstur Tanah	
1	Lintau Buo, Tanah Datar	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedikit	Halus (Liat)	Tidak Cocok
2	Tanjung Emas, Tanah Datar	Irigasi Tadah Hujan	Lancar	Datar	Sedikit	Kasar (Pasir)	Tidak Cocok
3	X Koto Diatas, Solok	Irigasi ½ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok
4	Pulau Punjung, Dharmasraya	Irigasi Teknis	Lancar	Datar	Sedang	Sedang (Lempung)	Cocok
5	Padang Laweh, Dharmasraya	Irigasi ½ Teknis	Lancar	Datar	Sedikit	Kasar (Pasir)	Cocok Bersyarat
6	Tarusan, Pesisir Selatan	Irigasi Tadah Hujan	Terhambat	Datar	Sedikit	Halus (Liat)	Tidak Cocok
7	Surantih, Pesisir Selatan	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedang	Sedang (Lempung)	Cocok Bersyarat
8	Balai Selasa, Pesisir Selatan	Irigasi Teknis	Lancar	Datar	Sedang	Sedang (Lempung)	Cocok
9	Muaro Bungo, Jambi	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedikit	Halus (Liat)	Tidak Cocok
10	Pagar Alam, Sumatera Selatan	Irigasi ½ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok
11	Taram, Lima Puluh Kota	Irigasi ½ Teknis	Lancar	Datar	Sedikit	Kasar (Pasir)	Cocok Bersyarat
12	Tarantang, Lima Puluh Kota	Irigasi ½ Teknis	Terhambat	Cekung (Lembah)	Sedang	Halus (Liat)	Tidak Cocok
13	Kinali, Pasaman Barat	Irigasi Teknis	Lancar	Datar	Tinggi	Sedang (Lempung)	Cocok

14	Kapar, Pasaman Barat	Irigasi Teknis	Lancar	Datar	Sedang	Kasar (Pasir)	Cocok Bersyarat
15	Padang Aro, Solok Selatan	Irigasi ½ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok

Padang, Oktober 2016

(Ir. Erdiman)



Lampiran C

C-1. Hasil Pengujian & Pelatihan (30 Data)

C-2. Hasil Pengolahan Lanjutan

Lampiran C-1

Hasil Pengolahan Aplikasi
Pengembangan Teknologi Salibu

No	Nama Daerah	Parameter						Hasil Lapangan	Hasil Program	Nilai CF
		Jenis Irigasi	Kondisi Drainase	Kondisi Hamparan	Kadar Organik	Tekstur Tanah				
1	Sei. Tarab, Batusangkar	Irigasi Teknis	Lancar	Miring	Tinggi	Sedang (Lempung)	Cocok	Cocok	0,9892	
2	Nagari Tabek, Tanah Datar	Irigasi ½ Teknis	Lancar	Miring	Tinggi	Sedang (Lempung)	Cocok	Cocok	0,9796	
3	Rambatan, Tanah Datar	Irigasi Tadah Hujan	Lancar	Miring	Tinggi	Sedang (Lempung)	Tidak Cocok	Tidak Cocok	0,956	
4	Lima Kaum, Tanah Datar	Irigasi Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok	Cocok	0,9892	
5	Batipuh, Tanah Datar	Irigasi ½ Teknis	Lancar	Miring	Tinggi	Sedang (Lempung)	Cocok	Cocok	0,9796	
6	X Koto, Tanah Datar	Irigasi ½ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok	Cocok	0,9616	
7	Singkarak, Solok	Irigasi Pompa	Sedikit Terhambat	Datar	Tinggi	Halus (Liat)	Cocok Bersyarat	Cocok Bersyarat	0,8937	
8	Sumani, Solok	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Halus (Liat)	Cocok Bersyarat	Cocok Bersyarat	0,9139	
9	Selayo, Solok	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Sedang (Lempung)	Cocok Bersyarat	Cocok	0,9276	
10	Talang, Solok	Irigasi Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok	Cocok	0,9796	

11	Cupak, Solok	Irigasi ¼ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok	Cocok	0,9616
12	Sungai Lasi, Solok	Irigasi ¼ Teknis	Sedikit Terhambat	Datar	Sedang	Kasar (Pasir)	Cocok Bersyarat	Cocok Bersyarat	0,6659
13	Parabek, Agam	Irigasi ¼ Teknis	Lancar	Miring	Tinggi	Sedang (Lempung)	Cocok	Cocok	0,9796
14	Matur, Agam	Irigasi ¼ Teknis	Lancar	Datar	Sedang	Kasar (Pasir)	Cocok Bersyarat	Cocok Bersyarat	0,8231
15	Lubuk Basung, Agam	Irigasi ¼ Teknis	Lancar	Datar	Sedang	Sedang (Lempung)	Cocok	Cocok	0,9276
16	Biaro, Agam	Irigasi ¼ Teknis	Lancar	Datar	Sedang	Kasar (Pasir)	Cocok	Cocok Bersyarat	0,8231
17	Kuranji, Padang	Irigasi ¼ Teknis	Sedikit Terhambat	Datar	Sedikit (Kurang)	Halus (Liat)	Tidak Cocok	Tidak Cocok	0,6485
18	Koto Tangah, Padang	Irigasi Teknis	Lancar	Datar	Sedang	Sedang (Lempung)	Cocok	Cocok	0,9616
19	Paui, Padang	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Sedang (Lempung)	Cocok	Cocok	0,9276
20	Lubuk Ahung, Padang Pariaman	Irigasi Teknis	Sedikit Terhambat	Cekung (Lembah)	Sedang	Kasar (Pasir)	Tidak Cocok	Tidak Cocok	0,4998
21	Buayan, Padang Pariaman	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Halus (Liat)	Cocok Bersyarat	Cocok Bersyarat	0,9139
22	Kasang, Padang Pariaman	Irigasi ¼ Teknis	Sedikit Terhambat	Datar	Sedang	Halus (Liat)	Cocok Bersyarat	Cocok Bersyarat	0,8375
23	Sitiung, Dharmasraya	Irigasi Teknis	Sedikit Terhambat	Datar	Sedikit (Kurang)	Sedang (Lempung)	Cocok Bersyarat	Cocok Bersyarat	0,8433
24	Sijunjung	Irigasi Tadah Hujan	Sedikit Terhambat	Datar	Sedang	Halus (Liat)	Tidak Cocok	Tidak Cocok	0

25	Karawang	Irigasi Teknis	Terhambat	Datar	Sedang	Halus (Liat)	Tidak Cocok	Tidak Cocok	0,7567
26	Sukamandi	Irigasi Teknis	Terhambat	Datar	Sedikit (Kurang)	Halus (Liat)	Tidak Cocok	Tidak Cocok	0,4738
27	Purwakarta	Irigasi Teknis	Lancar	Miring	Tinggi	Halus (Liat)	Cocok	Cocok	0,9872
28	Brebes	Irigasi Teknis	Sedikit Terhambat	Datar	Sedang	Sedang (Lempung)	Cocok	Cocok	0,9276
29	Lombok	Irigasi 1/4 Teknis	Sedikit Terhambat	Miring	Tinggi	Sedang (Lempung)	Cocok	Cocok	0,9616
30	Muara Enim	Irigasi Pompa	Sedikit Terhambat	Datar	Tinggi	Sedang (Lempung)	Cocok Bersyarat	Cocok Bersyarat	0,9105

Jumlah kasus dengan keputusan benar
Jumlah keseluruhan kasus

: 28
: 30

$$\% \text{ Akurasi} = \frac{\text{Jumlah Kasus dengan Keputusan Benar}}{\text{Jumlah Keseluruhan Kasus}} = \frac{28}{30} \times 100\% = 93,33\%$$



Hasil Pengolahan Data
Pengembangan Padi Salibu

No	Nama Daerah	Parameter						Hasil Pakar	Hasil Aplikasi	Nilai CF
		Jenis Irigasi	Kondisi Drainase	Kondisi Hamparan	Kadar Organik	Tekstur Tanah				
1	Lintau Buo, Tanah Datar	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedikit	Halus (Liat)	Tidak Cocok	Tidak Cocok	0.6486	
2	Tanjung Emas, Tanah Datar	Irigasi Tadah Hujan	Lancar	Datar	Sedikit	Kasar (Pasir)	Tidak Cocok	Tidak Cocok	0.1727	
3	X Koto Diatas, Solok	Irigasi ½ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok	Cocok	0.9616	
4	Pulau Punjung, Dharmasraya	Irigasi Teknis	Lancar	Datar	Sedang	Sedang (Lempung)	Cocok	Cocok	0.9616	
5	Padang Laweh, Dharmasraya	Irigasi ½ Teknis	Lancar	Datar	Sedikit	Kasar (Pasir)	Cocok Bersyarat	Tidak Cocok	0.6175	
6	Tarusan, Pesisir Selatan	Irigasi Tadah Hujan	Terhambat	Datar	Sedikit	Halus (Liat)	Tidak Cocok	Tidak Cocok	0	
7	Suranlih, Pesisir Selatan	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedang	Sedang (Lempung)	Cocok Bersyarat	Cocok Bersyarat	0.8631	
8	Balai Selasa, Pesisir Selatan	Irigasi Teknis	Lancar	Datar	Sedang	Sedang (Lempung)	Cocok	Cocok	0.9616	
9	Muaro Bungo, Jambi	Irigasi ½ Teknis	Sedikit Terhambat	Datar	Sedikit	Halus (Liat)	Tidak Cocok	Tidak Cocok	0.6485	

10	Pagar Alam, Sumatera Selatan	Irigasi ½ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok	Cocok	0.9616
11	Taram, Lima Puluh Kota	Irigasi ½ Teknis	Lancar	Datar	Sedikit	Kasar (Pasir)	Cocok Bersyarat	Tidak Cocok	0.6175
12	Tarantang, Lima Puluh Kota	Irigasi ½ Teknis	Terhambat	Cekung (Lembah)	Sedang	Halus (Liat)	Tidak Cocok	Tidak Cocok	0
13	Kinali, Pasaman Barat	Irigasi Teknis	Lancar	Datar	Tinggi	Sedang (Lempung)	Cocok	Cocok	0.9796
14	Kapar, Pasaman Barat	Irigasi Teknis	Lancar	Datar	Sedang	Kasar (Pasir)	Cocok Bersyarat	Cocok Bersyarat	0.9063
15	Padang Aro, Solok Selatan	Irigasi ½ Teknis	Lancar	Miring	Sedang	Sedang (Lempung)	Cocok	Cocok	0.9616

Jumlah kasus dengan keputusan benar
Jumlah keseluruhan kasus

: 13
: 15

$$\% \text{ Akurasi} = \frac{\text{Jumlah Kasus dengan Keputusan Benar}}{\text{Jumlah Keseluruhan Kasus}} = \frac{13}{15} \times 100\% = 86,7\%$$



Lampiran D

Lampiran D-1 Listing Program Aplikasi Android (Login.java)

```
package com.example.rifky.sistempakarpadisaibu;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class Login extends Activity {
    TextView txt_desa, txt_kec, txt_kota, txt_prov;
    EditText form_desa, form_kec, form_kota, form_prov;
    Button login_button;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);

        form_desa = (EditText) findViewById(R.id.editText_desa);
        form_kec = (EditText) findViewById(R.id.editText_kec);
        form_kota = (EditText) findViewById(R.id.editText_kota);
        form_prov = (EditText) findViewById(R.id.editText_prov);

        login_button = (Button) findViewById(R.id.button_login);

        login_button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                EditText etUserName1 = (EditText) findViewById(R.id.editText_desa);
                EditText etUserName2 = (EditText) findViewById(R.id.editText_kec);
                EditText etUserName3 = (EditText) findViewById(R.id.editText_kota);
                EditText etUserName4 = (EditText) findViewById(R.id.editText_prov);
                String strUserName1 = etUserName1.getText().toString();
                String strUserName2 = etUserName2.getText().toString();
                String strUserName3 = etUserName3.getText().toString();
                String strUserName4 = etUserName4.getText().toString();

                if(TextUtils.isEmpty(strUserName1)){
                    etUserName1.setError("Data Desa Masih Kosong");
                    return;
                } else if(TextUtils.isEmpty(strUserName2)){
                    etUserName2.setError("Data Kecamatan Masih Kosong");
                    return;
                } else if(TextUtils.isEmpty(strUserName3)){
                    etUserName3.setError("Data Kota Masih Kosong");
                    return;
                } else if(TextUtils.isEmpty(strUserName4)) {
                    etUserName4.setError("Data Provinsi Masih Kosong");
                    return;
                }
            }
        });
    }
}
```

```
}
```

```
Intent intent = new Intent(getApplicationContext(), MainActivity.class);  
Bundle login = new Bundle();
```

```
login.putString("desa", form_desa.getText().toString());  
login.putString("kecamatan", form_kec.getText().toString());  
login.putString("kota", form_kota.getText().toString());  
login.putString("provinsi", form_prov.getText().toString());
```

```
intent.putExtras(login);  
startActivity(intent);
```

```
}  
});  
}  
}
```



Lampiran D-2 Listing Program Aplikasi Android (MainActivity.java)

```
package com.example.rifky.sistempakarpadialibu;

import android.app.ActionBar;
import android.app.AlertDialog;
import android.app.TabActivity;
import android.content.DialogInterface;
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.webkit.WebSettings;
import android.widget.Button;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TabHost;
import android.widget.TextView;
import android.widget.Toast;

import org.w3c.dom.Text;

public class MainActivity extends AppCompatActivity {
    RadioButton rbir1, rbir2, rbir3, rbir4, rbdr1, rbdr2, rbdr3, rbor1, rbor2,
    rbor3, rbhm1, rbhm2, rbhm3, rbtt1, rbtt2, rbtt3;
    TextView txt_hasil1, txt_hasil2, txt_hasil3, txt_hasil4, txt_hasil5,
    txt_kesimpulan, txt_akh1, txt_akh2, txt_akh3;
    TextView txt_saran1, txt_saran2, txt_saran3, txt_saran4, txt_desa, txt_kec,
    txt_kota, txt_prov;
    RadioGroup rgir, rgdr, rgor, rghm, rgtt;
    Button b1;

    TabHost tabHost;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        findViewById();

        rbir1 = (RadioButton) findViewById(R.id.radioButton_irigasi1);
        rbir2 = (RadioButton) findViewById(R.id.radioButton_irigasi2);
        rbir3 = (RadioButton) findViewById(R.id.radioButton_irigasi3);
        rbir4 = (RadioButton) findViewById(R.id.radioButton_irigasi4);

        rbdr1 = (RadioButton) findViewById(R.id.radioButton_drainase1);
        rbdr2 = (RadioButton) findViewById(R.id.radioButton_drainase2);
        rbdr3 = (RadioButton) findViewById(R.id.radioButton_drainase3);
        rbhm1 = (RadioButton) findViewById(R.id.radioButton_hamparan1);
        rbhm2 = (RadioButton) findViewById(R.id.radioButton_hamparan2);
        rbhm3 = (RadioButton) findViewById(R.id.radioButton_hamparan3);
    }
}
```

```
rbor1 = (RadioButton) findViewById(R.id.radioButton_organik1);  
rbor2 = (RadioButton) findViewById(R.id.radioButton_organik2);  
rbor3 = (RadioButton) findViewById(R.id.radioButton_organik3);
```

```
rbtt1 = (RadioButton) findViewById(R.id.radioButton_tekstur1);  
rbtt2 = (RadioButton) findViewById(R.id.radioButton_tekstur2);  
rbtt3 = (RadioButton) findViewById(R.id.radioButton_tekstur3);
```

```
rgir = (RadioGroup) findViewById(R.id.radioGroup_irigasi);  
rgdr = (RadioGroup) findViewById(R.id.radioGroup_drainase);  
rgor = (RadioGroup) findViewById(R.id.radioGroup_organik);  
rghm = (RadioGroup) findViewById(R.id.radioGroup_hamparan);  
rgtt = (RadioGroup) findViewById(R.id.radioGroup_tekstur);
```

```
txt_hasil1 = (TextView) findViewById(R.id.textView_hasil1);  
txt_hasil2 = (TextView) findViewById(R.id.textView_hasil2);  
txt_hasil3 = (TextView) findViewById(R.id.textView_hasil3);  
txt_hasil4 = (TextView) findViewById(R.id.textView_hasil4);  
txt_hasil5 = (TextView) findViewById(R.id.textView_hasil5);  
txt_kesimpulan = (TextView) findViewById(R.id.textView_kesimpulan);
```

```
txt_akh1 = (TextView) findViewById(R.id.textView_hasilakh1);  
txt_akh2 = (TextView) findViewById(R.id.textView_hasilakh2);  
txt_akh3 = (TextView) findViewById(R.id.textView_hasilakh3);
```

```
txt_saran1 = (TextView) findViewById(R.id.textView_che2);  
txt_saran2 = (TextView) findViewById(R.id.textView_che3);  
txt_saran3 = (TextView) findViewById(R.id.textView_che4);  
txt_saran4 = (TextView) findViewById(R.id.textView_che5);
```

```
txt_desa = (TextView) findViewById(R.id.textView_desa);  
txt_kec = (TextView) findViewById(R.id.textView_kec);  
txt_kota = (TextView) findViewById(R.id.textView_kota);  
txt_prov = (TextView) findViewById(R.id.textView_prov);
```

```
Bundle login = getIntent().getExtras();
```

```
txt_desa.setText(login.getCharSequence("desa"));  
txt_kec.setText(login.getCharSequence("kecamatan"));  
txt_kota.setText(login.getCharSequence("kota"));  
txt_prov.setText(login.getCharSequence("provinsi"));
```

```
final double ch1, ch2, ch3, ch4, ch5;
```

```
final double rbir1, rbir2, rbir3, rbir4, rbdr1, rbdr2, rbdr3, rbhm1, rbhm2, rbhm3,  
rbor1, rbor2, rbor3, rbtt1, rbtt2, rbtt3;
```

```
ch1 = 0.8;
```

```
ch2 = 0.8;
```

```
ch3 = 0.8;
```

```
ch4 = 0.8;
```

```
ch5 = 0.6;
```

```
rbir1 = 0.8;
rbir2 = 0.4;
rbir3 = 0.2;
rbir4 = -0.4;
```

```
rbdr1 = 0.8;
rbdr2 = 0.4;
rbdr3 = -0.6;
```

```
rbhm1 = 0.8;
rbhm2 = 0.4;
rbhm3 = -0.6;
```

```
rbor1 = 0.8;
rbor2 = 0.4;
rbor3 = -0.4;
```

```
rbtt1 = 0.6;
rbtt2 = 0.4;
rbtt3 = -0.6;
```

```
//----- Saat Tombol Ditekan -----//
b1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

//----- Kondisi Penentuan Nilai C(H/E) -----//
        int a = rgir.getCheckedRadioButtonId();
        double cf1 = 0;
        double che1;

        if (a == R.id.radioButton_irigasi1) {
            cf1 = rbir1 * ch1;

        } else if (a == R.id.radioButton_irigasi2) {
            cf1 = rbir2 * ch1;

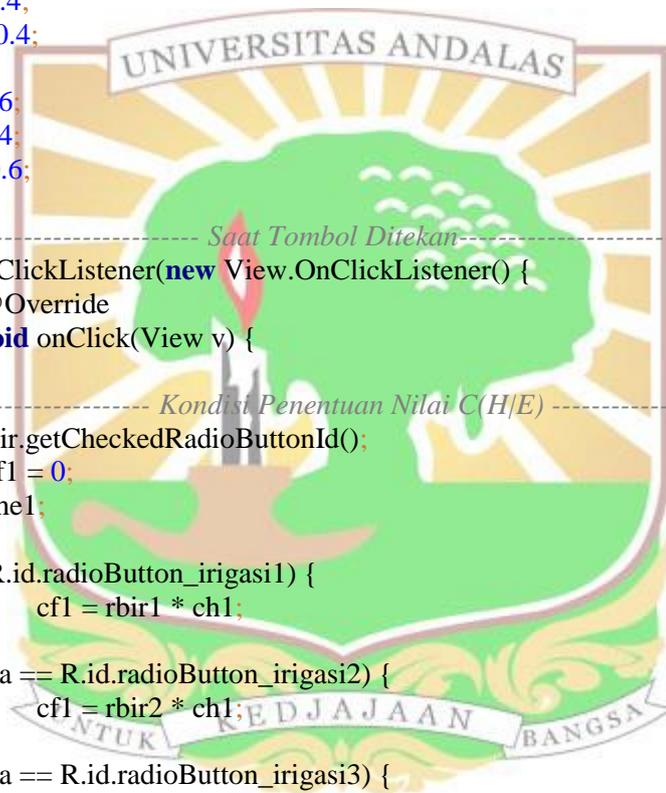
        } else if (a == R.id.radioButton_irigasi3) {
            cf1 = rbir3 * ch1;

        } else if (a == R.id.radioButton_irigasi4) {
            cf1 = rbir4 * ch1;
        }

        che1 = Double.parseDouble(String.valueOf(cf1));
        txt_akh2.setText(String.valueOf(che1));

        int bb = rgdr.getCheckedRadioButtonId();
        double cf2 = 0;
        double che2;

        if (bb == R.id.radioButton_drainase1) {
            cf2 = rbdr1 * ch2;
```



```

} else if (bb == R.id.radioButton_drainase2) {
    cf2 = rbdr2 * ch2;

} else if (bb == R.id.radioButton_drainase3) {
    cf2 = rbdr3 * ch2;

}
    che2 = Double.parseDouble(String.valueOf(cf2));
txt_saran1.setText(String.valueOf(che2));

int c = rghm.getCheckedRadioButtonId();
double cf3 = 0;
double che3;

if (c == R.id.radioButton_hamparan1) {
    cf3 = rbhm1 * ch3;

} else if (c == R.id.radioButton_hamparan2) {
    cf3 = rbhm2 * ch3;

} else if (c == R.id.radioButton_hamparan3) {
    cf3 = rbhm3 * ch3;

}
    che3 = Double.parseDouble(String.valueOf(cf3));
txt_saran2.setText(String.valueOf(che3));

int d = rgor.getCheckedRadioButtonId();
double cf4 = 0;
double che4;

if (d == R.id.radioButton_organik1) {
    cf4 = rbor1 * ch4;

} else if (d == R.id.radioButton_organik2) {
    cf4 = rbor2 * ch4;

} else if (d == R.id.radioButton_organik3) {
    cf4 = rbor3 * ch4;

}
    che4 = Double.parseDouble(String.valueOf(cf4));
txt_saran3.setText(String.valueOf(che4));

int e = rgtt.getCheckedRadioButtonId();
double cf5 = 0;
double che5;

if (e == R.id.radioButton_tekstur1) {
    cf5 = rbtt1 * ch5;

} else if (e == R.id.radioButton_tekstur2) {
    cf5 = rbtt2 * ch5;

```



```

} else if (e == R.id.radioButton_tekstur3){
    cf5 = rbtt3 * ch5;

}
    che5 = Double.parseDouble(String.valueOf(cf5));
txt_saran4.setText(String.valueOf(che5));

//----- Kondisi Kombinasi CFCn -----//

double cfc1 = 0;
double cfcx1, x1a, x2a, abs1a, abs2a, m1;

if (che1 >0 && che2 >0) {
    cfc1 = che1 + che2 - che1 * che2;

} else if (che1 <0 && che2 <0) {
    cfc1 = che1 + che2 + che1 * che2;

} else if (che1 <0 || che2 <0) {
    abs1a = Math.abs(che1);
abs2a = Math.abs(che2);
m1 = Math.min(abs1a, abs2a);
x1a = che1 + che2;
x2a = 1-m1;
cfc1 = x1a/x2a;

}

cfcx1 = Double.parseDouble(String.valueOf(cfc1));

double cfc2 = 0;
double cfcx2, x1b, x2b, abs1b, abs2b, m2;;

if (cfcx1 >0 && che3 >0) {
    cfc2 = cfcx1 + che3 - cfcx1 * che3;

} else if (cfcx1 <0 && che3 <0) {
    cfc2 = cfcx1 + che3 + cfcx1 * che3;

} else if (cfcx1 <0 || che3 <0) {
    abs1b = Math.abs(cfcx1);
abs2b = Math.abs(che3);
m2 = Math.min(abs1b, abs2b);
x1b = cfcx1 + che3;
x2b = 1-m2;
cfc2 = x1b/x2b;

}

cfcx2 = Double.parseDouble(String.valueOf(cfc2));

```

```

double cfc3 = 0;
double cfcx3, abs1c, abs2c, x1c, x2c, m3;

if (cfcx2 >0 && che4 >0) {
    cfc3 = cfcx2 + che4 - cfcx2 * che4;
} else if (cfcx2 <0 && che4 <0) {
    cfc3 = cfcx2 + che4 + cfcx2 * che4;
} else if (cfcx2 <0 || che4 <0) {
    abs1c = Math.abs(cfcx2);
abs2c = Math.abs(che4);
m3 = Math.min(abs1c, abs2c);
x1c = cfcx2 + che4;
x2c = 1-m3;
cfc3 = x1c/x2c;
}

cfcx3 = Double.parseDouble(String.valueOf(cfc3));

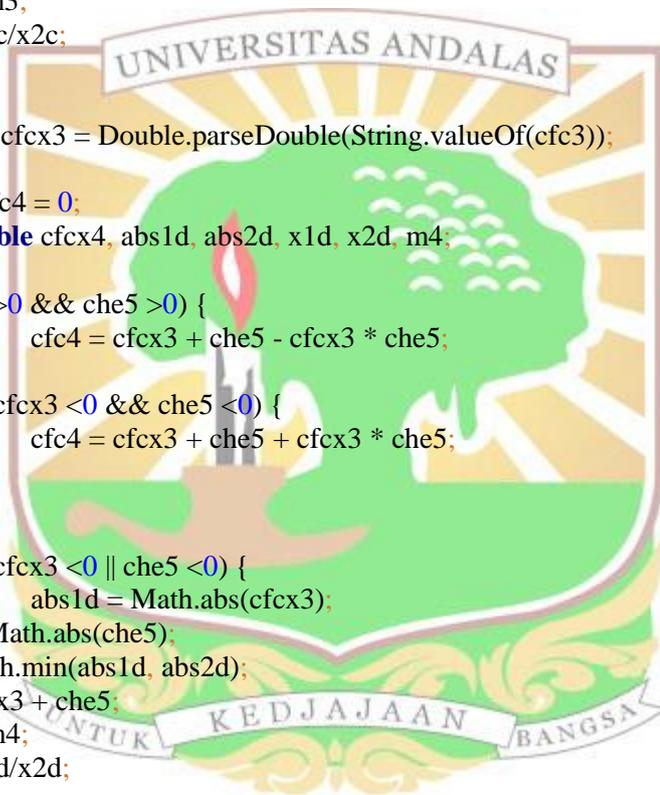
double cfc4 = 0;
final double cfcx4, abs1d, abs2d, x1d, x2d, m4;

if (cfcx3 >0 && che5 >0) {
    cfc4 = cfcx3 + che5 - cfcx3 * che5;
} else if (cfcx3 <0 && che5 <0) {
    cfc4 = cfcx3 + che5 + cfcx3 * che5;
} else if (cfcx3 <0 || che5 <0) {
    abs1d = Math.abs(cfcx3);
abs2d = Math.abs(che5);
m4 = Math.min(abs1d, abs2d);
x1d = cfcx3 + che5;
x2d = 1-m4;
cfc4 = x1d/x2d;
}

cfcx4 = Double.parseDouble(String.valueOf(cfc4));
txt_akh3.setText(String.valueOf(cfc4));

if (che1 <0 || cfcx4 <= 0.76) {
    txt_akh1.setText("Tidak Cocok");
} else if (cfcx4 >0.76 && cfcx4 <= 0.9275) {
    txt_akh1.setText("Cocok Bersyarat");
} else if (cfcx4 >0.9275) {
    txt_akh1.setText("Cocok");
}

```



```

}

//-----Passing Data to Next Activity -----//

Intent intent = new Intent(getApplicationContext(), Hasil.class);
Bundle b = new Bundle();

int idir = rgir.getCheckedRadioButtonId();
RadioButton rb_irigasi = (RadioButton) findViewById(idir);
b.putString("irigasi", rb_irigasi.getText().toString());

int iddr = rgdr.getCheckedRadioButtonId();
RadioButton rb_drainase = (RadioButton) findViewById(iddr);
b.putString("drainase", rb_drainase.getText().toString());

int idor = rgor.getCheckedRadioButtonId();
RadioButton rb_organik = (RadioButton) findViewById(idor);
b.putString("organik", rb_organik.getText().toString());

int idhm = rghm.getCheckedRadioButtonId();
RadioButton rb_hamparan = (RadioButton) findViewById(idhm);
b.putString("hamparan", rb_hamparan.getText().toString());

int idtt = rgtt.getCheckedRadioButtonId();
RadioButton rb_tekstur = (RadioButton) findViewById(idtt);
b.putString("tekstur", rb_tekstur.getText().toString());

b.putString("kesimpulan", txt_akh1.getText().toString());
b.putString("che", txt_akh2.getText().toString());
b.putString("cfcx", txt_akh3.getText().toString());
b.putString("che2", txt_saran1.getText().toString());
b.putString("che3", txt_saran2.getText().toString());
b.putString("che4", txt_saran3.getText().toString());
b.putString("che5", txt_saran4.getText().toString());

b.putString("desa1", txt_desa.getText().toString());
b.putString("kec1", txt_kec.getText().toString());
b.putString("kota1", txt_kota.getText().toString());
b.putString("prov1", txt_prov.getText().toString());

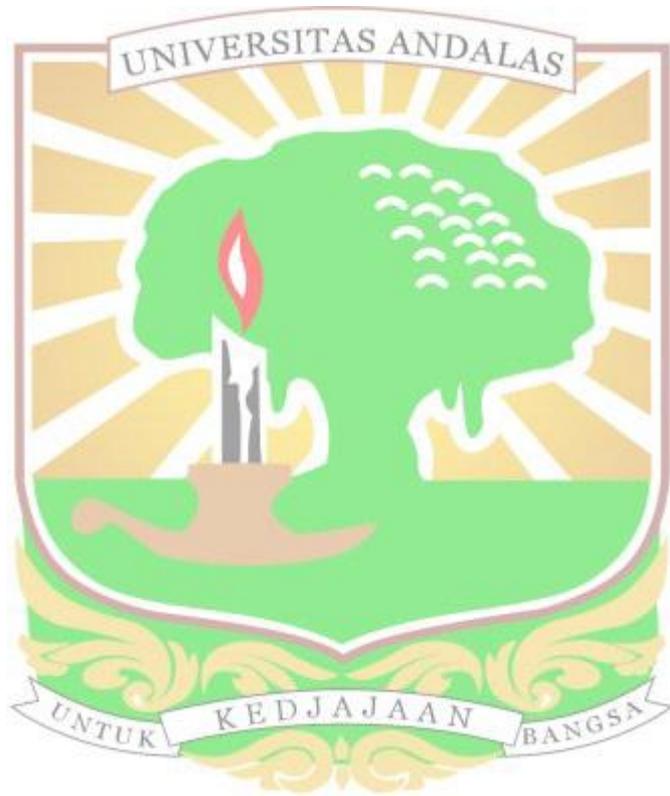
intent.putExtras(b);
startActivity(intent);
}

});
}

private void findAllViewsId(){
b1 = (Button)findViewById(R.id.button_proses);
rgir = (RadioGroup)findViewById(R.id.radioGroup_irigasi);
rgdr = (RadioGroup)findViewById(R.id.radioGroup_drainase);
rgor = (RadioGroup)findViewById(R.id.radioGroup_organik);
rghm = (RadioGroup)findViewById(R.id.radioGroup_hamparan);

```

```
rgtt = (RadioGroup)findViewById(R.id.radioGroup_tekstur);  
txt_akh1 = (TextView)findViewById(R.id.textView_hasilakh1);  
txt_akh2 = (TextView) findViewById(R.id.textView_hasilakh2);  
txt_akh3 = (TextView) findViewById(R.id.textView_hasilakh3);  
txt_saran1 = (TextView) findViewById(R.id.textView_che2);  
txt_saran2 = (TextView) findViewById(R.id.textView_che3);  
txt_saran3 = (TextView) findViewById(R.id.textView_che4);  
txt_saran4 = (TextView) findViewById(R.id.textView_che5);  
  
txt_desa = (TextView) findViewById(R.id.textView_desa);  
txt_kec = (TextView) findViewById(R.id.textView_kec);  
txt_kota = (TextView) findViewById(R.id.textView_kota);  
txt_prov = (TextView) findViewById(R.id.textView_prov);  
  
}  
}
```



Lampiran D-3 Listing Program Aplikasi Android (Hasil.java)

```
package com.example.rifky.sistempakarpadialibu;

import android.app.Activity;
import android.content.Intent;
import android.content.res.Resources;
import android.graphics.Color;
import android.os.Bundle;
import android.text.Html;
import android.view.View;
import android.webkit.WebSettings;
import android.webkit.WebView;
import android.widget.Button;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;

public class Hasil extends Activity {
    RadioButton rbir1, rbir2, rbir3, rbdr1, rbdr2, rbdr3, rbor1, rbor2, rbor3, rbhm1,
    rbhm2, rbtt1, rbtt2;
    TextView txt_hasil1, txt_hasil2, txt_hasil3, txt_hasil4, txt_hasil5, txt_kesimpulan,
    txt_akh1, txt_akh2, txt_akh3, txt_judulmnmjm;
    TextView txt_che1, txt_cfcx4, txt_arahan1, txt_arahan2, txt_arahan3, txt_arahan4,
    txt_desa1, txt_kec1, txt_kota1, txt_prov1;
    RadioGroup rgir, rgdr, rgor, rghm, rgtt;
    Button b2, b4;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_hasil);

        String htmlText = "<html><body style='text-align:justify'> %s </body></html>";
        String myData = "Untuk daerah 'Datar' atau 'Cekung' serta drainase yang 'Sedikit  
Terhambat atau 'Terhambat' maka Dibuatkan Saluran Pembuangan Air.";

        WebView webView = (WebView) findViewById(R.id.webView_arahan1);
        webView.loadData(String.format(htmlText, myData), "text/html", "utf-8");
        webView.setBackgroundColor(0);
        webView.setBackgroundColor(Color.TRANSPARENT);

        final WebSettings webSettings = webView.getSettings();
        webSettings.setDefaultFontSize(12);

        String myData1 = "Bila kadar bahan organik 'Rendah' dan tekstur tanah 'Kasar'  
maka Tambahkan Bahan Organik (Pupuk Kandang / Kompos).";

        WebView webView1 = (WebView) findViewById(R.id.webView_arahan2);
        webView1.loadData(String.format(htmlText, myData1), "text/html", "utf-8");
        webView1.setBackgroundColor(0);
        webView1.setBackgroundColor(Color.TRANSPARENT);

        final WebSettings webSettings1 = webView1.getSettings();
        webSettings1.setDefaultFontSize(12);

        rbir1 = (RadioButton) findViewById(R.id.radioButton_irigasi1);
```

```

rbir2 = (RadioButton) findViewById(R.id.radioButton_irigasi2);
rbir3 = (RadioButton) findViewById(R.id.radioButton_irigasi3);
rbdr1 = (RadioButton) findViewById(R.id.radioButton_drainase1);
rbdr2 = (RadioButton) findViewById(R.id.radioButton_drainase2);
rbdr3 = (RadioButton) findViewById(R.id.radioButton_drainase3);
rbor1 = (RadioButton) findViewById(R.id.radioButton_organik1);
rbor2 = (RadioButton) findViewById(R.id.radioButton_organik2);
rbor3 = (RadioButton) findViewById(R.id.radioButton_organik3);
rbhm1 = (RadioButton) findViewById(R.id.radioButton_hampanan1);
rbhm2 = (RadioButton) findViewById(R.id.radioButton_hampanan2);
rbtt1 = (RadioButton) findViewById(R.id.radioButton_tekstur1);
rbtt2 = (RadioButton) findViewById(R.id.radioButton_tekstur2);

rgir = (RadioGroup) findViewById(R.id.radioGroup_irigasi);
rgdr = (RadioGroup) findViewById(R.id.radioGroup_drainase);
rgor = (RadioGroup) findViewById(R.id.radioGroup_organik);
rghm = (RadioGroup) findViewById(R.id.radioGroup_hampanan);
rgtt = (RadioGroup) findViewById(R.id.radioGroup_tekstur);

txt_hasil1 = (TextView) findViewById(R.id.textView_hasil1);
txt_hasil2 = (TextView) findViewById(R.id.textView_hasil2);
txt_hasil3 = (TextView) findViewById(R.id.textView_hasil3);
txt_hasil4 = (TextView) findViewById(R.id.textView_hasil4);
txt_hasil5 = (TextView) findViewById(R.id.textView_hasil5);
txt_kesimpulan = (TextView) findViewById(R.id.textView_kesimpulan);

txt_judulnmjm = (TextView) findViewById(R.id.textView_mnjlahan);
txt_arahan1 = (TextView) findViewById(R.id.textView_che2);
txt_arahan2 = (TextView) findViewById(R.id.textView_che3);
txt_arahan3 = (TextView) findViewById(R.id.textView_che4);
txt_arahan4 = (TextView) findViewById(R.id.textView_che5);

txt_desa1 = (TextView) findViewById(R.id.editText_desa);
txt_kec1 = (TextView) findViewById(R.id.editText_kec);
txt_kota1 = (TextView) findViewById(R.id.editText_kota);
txt_prov1 = (TextView) findViewById(R.id.editText_prov);
txt_akh1 = (TextView) findViewById(R.id.textView_hasilakh1);
txt_akh2 = (TextView) findViewById(R.id.textView_hasilakh2);
txt_akh3 = (TextView) findViewById(R.id.textView_hasilakh3);

txt_che1 = (TextView) findViewById(R.id.textView_che1);
txt_cfcx4 = (TextView) findViewById(R.id.textView_cfcx4);

Bundle b = getIntent().getExtras();

txt_hasil1.setText(b.getCharSequence("irigasi"));
txt_hasil2.setText(b.getCharSequence("drainase"));
txt_hasil3.setText(b.getCharSequence("organik"));
txt_hasil4.setText(b.getCharSequence("hampanan"));
txt_hasil5.setText(b.getCharSequence("tekstur"));
txt_kesimpulan.setText(b.getCharSequence("kesimpulan"));
txt_che1.setText(b.getCharSequence("che"));
txt_cfcx4.setText(b.getCharSequence("cfcx"));
txt_arahan1.setText(b.getCharSequence("che2"));
txt_arahan2.setText(b.getCharSequence("che3"));
txt_arahan3.setText(b.getCharSequence("che4"));
txt_arahan4.setText(b.getCharSequence("che5"));

```

```
txt_desa1.setText(b.getCharSequence("desa1"));
txt_kec1.setText(b.getCharSequence("kec1"));
txt_kota1.setText(b.getCharSequence("kota1"));
txt_prov1.setText(b.getCharSequence("prov1"));
```

```
String gettxt_che = txt_che1.getText().toString();
String gettxt_cfcx = txt_cfcx4.getText().toString();
```

```
String gettxt_che2 = txt_arahan1.getText().toString();
String gettxt_che3 = txt_arahan2.getText().toString();
String gettxt_che4 = txt_arahan3.getText().toString();
String gettxt_che5 = txt_arahan4.getText().toString();
```

```
double off1 = Double.parseDouble(gettxt_che);
double off2 = Double.parseDouble(gettxt_cfcx);
```

```
double ch2 = Double.parseDouble(gettxt_che2);
double ch3 = Double.parseDouble(gettxt_che3);
double ch4 = Double.parseDouble(gettxt_che4);
double ch5 = Double.parseDouble(gettxt_che5);
```

```
if (ch2 == 0.32000000000000006 && ch3 == 0.32000000000000006){
    webView.setVisibility(View.VISIBLE);
    txt_judulmnjm.setVisibility(View.VISIBLE);
} else if (ch2 == 0.32000000000000006 && ch3 == -0.48){
    webView.setVisibility(View.VISIBLE);
    txt_judulmnjm.setVisibility(View.VISIBLE);
} else if (ch2 == -0.48 && ch3 == 0.32000000000000006){
    webView.setVisibility(View.VISIBLE);
    txt_judulmnjm.setVisibility(View.VISIBLE);
} else if (ch2 == -0.48 && ch3 == -0.48){
    webView.setVisibility(View.VISIBLE);
    txt_judulmnjm.setVisibility(View.VISIBLE);
} if (ch4 == -0.32000000000000006 && ch5 == -0.36){
    webView1.setVisibility(View.VISIBLE);
    txt_judulmnjm.setVisibility(View.VISIBLE);
}
}
```

```
b2 = (Button) findViewById(R.id.button_manajemen);
if (off1 < 0 || off2 <= 0.48) {
    b2.setVisibility(View.INVISIBLE);
    txt_judulmnjm.setVisibility(View.GONE);
    webView.setVisibility(View.INVISIBLE);
    webView1.setVisibility(View.INVISIBLE);
}
b4 = (Button) findViewById(R.id.button_kembali);
b4.setOnClickListener(new View.OnClickListener() {
```

```
@Override
public void onClick(View v) {

    Intent intent4 = new Intent(getApplicationContext(), Login.class);
startActivity(intent4);
}

});

b2.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {

    Intent intent1 = new Intent(getApplicationContext(), Manajemen.class);
startActivity(intent1);

}

});

}
}
```



Lampiran D-4 Listing Program Aplikasi Android (Manajemen.java)

```
import android.app.Activity;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.graphics.Color;
import android.os.Bundle;

import android.view.View;
import android.webkit.WebSettings;
import android.webkit.WebView;
import android.widget.Button;

public class Manajemen extends Activity {

    Button b3, b4;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_manajemen);

        //-----

        b3 = (Button) findViewById(R.id.button_awal);
        b3.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                Intent intent2 = new Intent(getApplicationContext(), Login.class);
                startActivity(intent2);
            }
        });

        b4 = (Button) findViewById(R.id.button_exit);
        b4.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                close();
            }
        });
    }

    public void close() {

        AlertDialog.Builder builder = new AlertDialog.Builder(Manajemen.this);
```

```

builder.setMessage("Yakin Ingin Keluar ?")
    .setCancelable(false)
    .setPositiveButton("Ya",
new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int id) {
    finishAffinity();
System.exit(0);
}
    })
    .setNegativeButton("Tidak", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int id) {
    dialog.cancel();
}
    });
AlertDialog a = builder.create();
a.show();
}
}

```





Lampiran E

Kuesioner Kepuasan Responden

Lampiran F-1 Perhitungan Kepuasan Fungsionalitas

Dalam metode Likert, maka perlu dilakukan penentuan skor/bobot masing-masing jawaban seperti yang tertera pada tabel dibawah.

Tabel Pembobotan Jawaban Kuesioner Fungsionalitas

	Jawaban	Bobot
Pertanyaan 1	1 = Ya	2
	2 = Tidak	1
Pertanyaan 2	1 = Respon Cepat	3
	2 = Kurang Cepat	2
	3 = Lambat	1
Pertanyaan 3	1 = Ya	3
	2 = Kurang	2
	3 = Tidak	1
Pertanyaan 4	1 = Ya	1
	2 = Sedikit	2
	3 = Tidak	3
Pertanyaan 5	1 = Baik	3
	2 = Kurang Baik	2
	3 = Tidak	1
Pertanyaan 6	1 = Sangat Puas	5
	2 = Puas	4
	3 = Kurang Puas	3
	4 = Sedikit	2
	5 = Tidak Puas	1

Selanjutnya untuk perhitungannya, berikut ini dibuatkan masing-masing perhitungan 2 sampel jawaban pertanyaan (1 dan 2) sebagai berikut :

1. Tani / Kelompok Tani

a. Pertanyaan 1

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Ya	$10 \times 2 = 20$
Tidak	$0 \times 1 = 0$
Total	20

- Skor Ideal

$$\begin{aligned} \text{Skor Ideal (n)} &= \text{Nilai Skala max} \times \text{Jumlah responden} \\ &= 2 \times 10 \\ &= 20 \end{aligned}$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{20}{20} \times 100\%$$

$$p = 100\%$$

b. Pertanyaan 2

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Respon Cepat	8 x 3 = 24
Kurang Cepat	2 x 2 = 4
Lambat	0 x 1 = 0
Total	28

- Skor Ideal

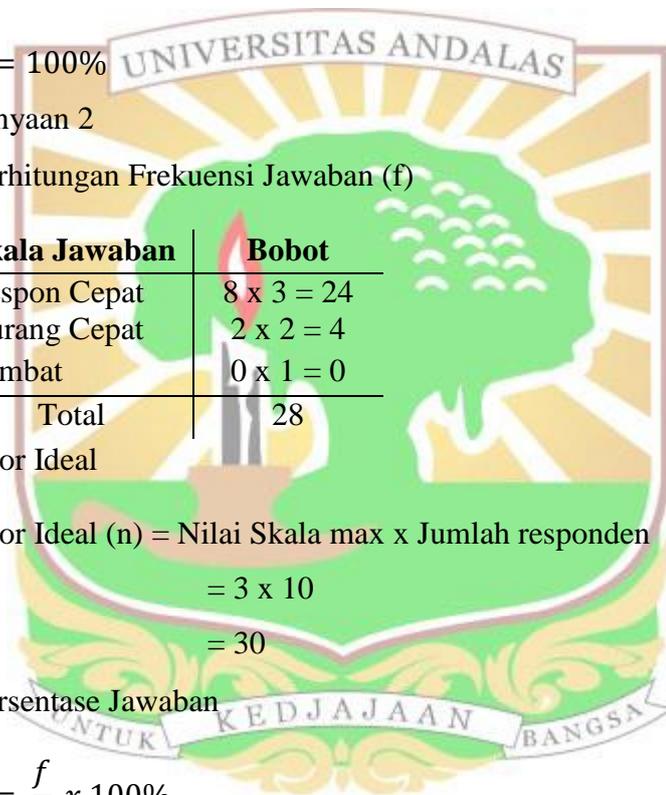
$$\begin{aligned} \text{Skor Ideal (n)} &= \text{Nilai Skala max} \times \text{Jumlah responden} \\ &= 3 \times 10 \\ &= 30 \end{aligned}$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{28}{30} \times 100\%$$

$$p = 93,33\%$$



c. Pertanyaan 3

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Ya	$9 \times 3 = 27$
Kurang	$1 \times 2 = 2$
Tidak	$0 \times 1 = 0$
Total	29

- Skor Ideal

Skor Ideal (n) = Nilai Skala max x Jumlah responden

$$= 3 \times 10 \\ = 30$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{29}{30} \times 100\%$$

$$p = 96,67\%$$

d. Pertanyaan 4

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Ya	$6 \times 3 = 18$
Sedikit	$4 \times 2 = 8$
Tidak	$0 \times 1 = 0$
Total	26

- Skor Ideal

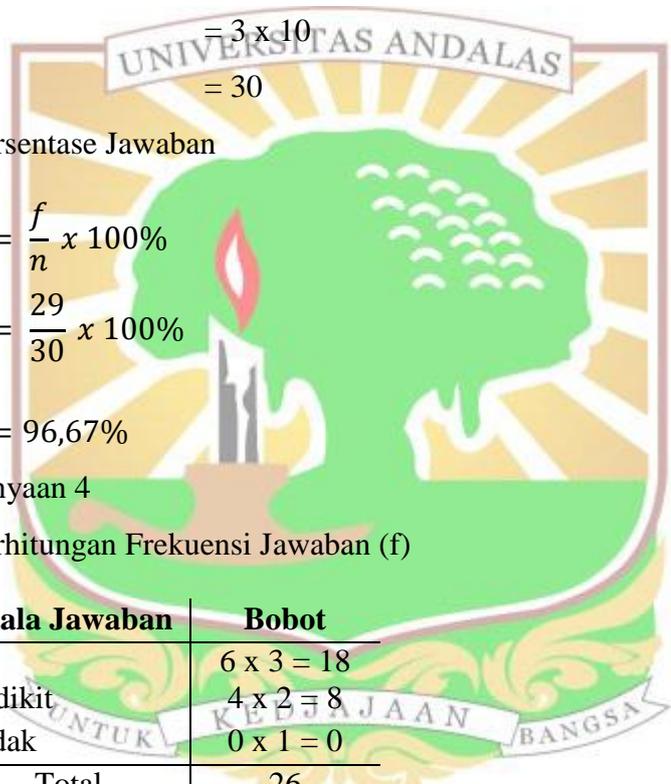
Skor Ideal (n) = Nilai Skala max x Jumlah responden

$$= 3 \times 10$$

$$= 30$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$



$$p = \frac{26}{30} \times 100\%$$

$$p = 86,7\%$$

e. Pertanyaan 5

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Baik	9 x 3 = 27
Kurang Baik	1 x 2 = 2
Tidak	0 x 1 = 0
Total	29

- Skor Ideal

$$\begin{aligned} \text{Skor Ideal (n)} &= \text{Nilai Skala max} \times \text{Jumlah responden} \\ &= 3 \times 10 \\ &= 30 \end{aligned}$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{29}{30} \times 100\%$$

$$p = 96,67\%$$

f. Pertanyaan 6

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Sangat Puas	4 x 5 = 20
Puas	6 x 4 = 24
Kurang Puas	0 x 3 = 0
Sedikit	0 x 2 = 0
Tidak Puas	0 x 1 = 0
Total	44

- Skor Ideal

Skor Ideal (n) = Nilai Skala max x Jumlah responden

$$= 5 \times 10$$

$$= 50$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{44}{50} \times 100\%$$

$$p = 88\%$$

Setelah dilakukan perhitungan pada semua jawaban pertanyaan yang ada, maka didapatkan persentase kepuasan untuk setiap pertanyaan. Hasil persentase jawaban untuk seluruh pertanyaan dapat dilihat pada tabel dibawah.

Tabel Hasil Kuesioner Fungsionalitas Tani

	No	Nama Responden	Jawaban Pertanyaan ke-					
			Ke - 1	Ke - 2	Ke - 3	Ke - 4	Ke - 5	Ke - 6
Tenaga Ahli	1	Yeldi Mardi	1	1	1	3	1	2
	2	Rahmawati	1	1	1	3	1	2
	3	Busrialman	1	1	1	2	1	1
	4	Hendri	1	1	2	3	1	1
	5	Afrizul	1	2	1	3	1	2
	6	Mulyati	1	1	1	3	1	1
	7	Suhatman	1	1	1	3	1	1
	8	Asmalinda	1	1	1	2	1	2
	9	Joni	1	1	1	2	1	2
	10	Muchlis	1	2	1	2	2	2
			100%	93,33%	96,67%	86,7%	96,67%	88,00%
			Rata-rata =				93,56%	

Keterangan :

	Jawaban
Pertanyaan 1	1 = Ya
	2 = Tidak
Pertanyaan 2	1 = Respon Cepat
	2 = Kurang Cepat
	3 = Lambat
Pertanyaan 3	1 = Ya
	2 = Kurang

	3 = Tidak
Pertanyaan 4	1 = Ya
	2 = Sedikit
	3 = Tidak
Pertanyaan 5	1 = Baik
	2 = Kurang Baik
	3 = Tidak
Pertanyaan 6	1 = Sangat Puas
	2 = Puas
	3 = Kurang Puas
	4 = Sedikit
	5 = Tidak Puas

2. Tenaga Ahli

a. Pertanyaan 1

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Ya	10 x 2 = 20
Tidak	0 x 1 = 0
Total	20

- Skor Ideal

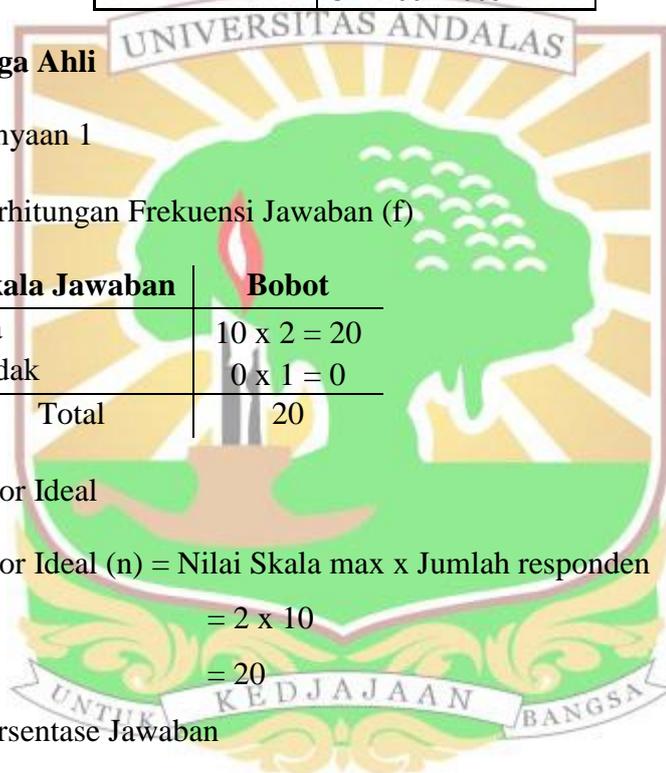
$$\begin{aligned} \text{Skor Ideal (n)} &= \text{Nilai Skala max} \times \text{Jumlah responden} \\ &= 2 \times 10 \\ &= 20 \end{aligned}$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{20}{20} \times 100\%$$

$$p = 100\%$$



b. Pertanyaan 2

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Respon Cepat	$9 \times 3 = 27$
Kurang Cepat	$0 \times 2 = 0$
Lambat	$1 \times 1 = 1$
Total	28

- Skor Ideal

Skor Ideal (n) = Nilai Skala max x Jumlah responden

$$= 3 \times 10$$

$$= 30$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

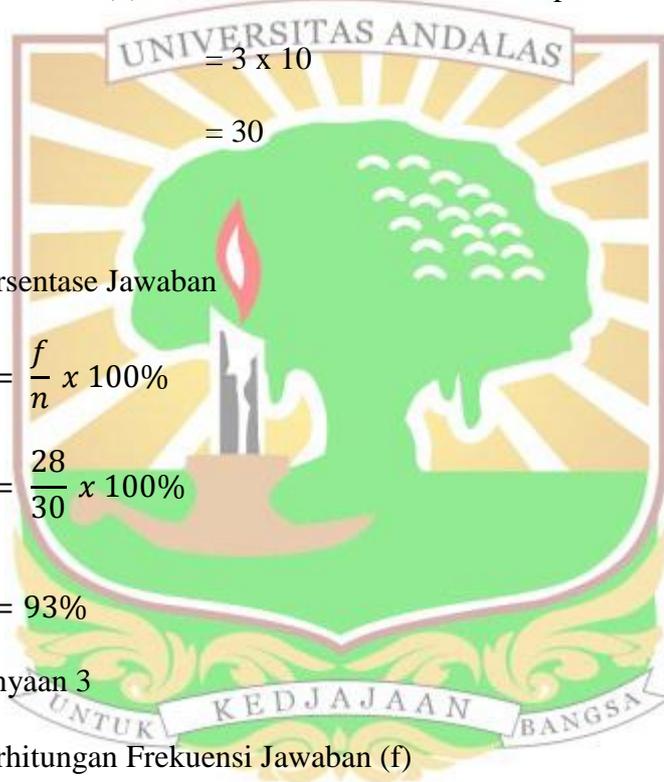
$$p = \frac{28}{30} \times 100\%$$

$$p = 93\%$$

c. Pertanyaan 3

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Ya	$10 \times 3 = 30$
Kurang	$0 \times 2 = 0$
Tidak	$0 \times 1 = 0$
Total	30



- Skor Ideal

$$\begin{aligned} \text{Skor Ideal (n)} &= \text{Nilai Skala max} \times \text{Jumlah responden} \\ &= 3 \times 10 \\ &= 30 \end{aligned}$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{30}{30} \times 100\%$$

$$p = 100\%$$

d. Pertanyaan 4

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Ya	7 x 3 = 21
Sedikit	3 x 2 = 6
Tidak	0 x 1 = 0
Total	27

- Skor Ideal

$$\begin{aligned} \text{Skor Ideal (n)} &= \text{Nilai Skala max} \times \text{Jumlah responden} \\ &= 3 \times 10 \\ &= 30 \end{aligned}$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{27}{30} \times 100\%$$

$$p = 90\%$$

e. Pertanyaan 5

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Baik	$8 \times 3 = 24$
Kurang Baik	$2 \times 2 = 4$
Tidak	$0 \times 1 = 0$
Total	28

- Skor Ideal

Skor Ideal (n) = Nilai Skala max x Jumlah responden

$$= 3 \times 10$$

$$= 30$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{28}{30} \times 100\%$$

$$p = 93\%$$

f. Pertanyaan 6

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Sangat Puas	$1 \times 5 = 5$
Puas	$9 \times 4 = 36$
Kurang Puas	$0 \times 3 = 0$
Sedikit	$0 \times 2 = 0$
Tidak Puas	$0 \times 1 = 0$
Total	41

- Skor Ideal

Skor Ideal (n) = Nilai Skala max x Jumlah responden

$$= 5 \times 10$$

$$= 50$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{41}{50} \times 100\%$$

$$p = 82\%$$

Setelah dilakukan perhitungan pada semua jawaban pertanyaan yang ada, maka didapatkan persentase kepuasan untuk setiap pertanyaan. Hasil persentase jawaban untuk seluruh pertanyaan dapat dilihat pada tabel dibawah.

Tabel Hasil Kuesioner Fungsionalitas Ahli

No	Nama Responden	Jawaban Pertanyaan ke-						
		Ke - 1	Ke - 2	Ke - 3	Ke - 4	Ke - 5	Ke - 6	
Tenaga Ahli	1	Jefri Chandra, SP	1	1	1	3	1	2
	2	Ir. Erdiman	1	1	1	3	1	2
	3	Murni Kurniaty, STP, MP	1	1	1	2	1	2
	4	Elviana Anwar	1	3	1	3	2	2
	5	Aldilani	1	1	1	3	1	2
	6	Ikhwatul Khairiyah, SP	1	1	1	2	1	2
	7	Gunawan, SE	1	1	1	3	1	2
	8	Syamsul Bahri	1	1	1	3	1	2
	9	Mahdalena, SP	1	1	1	3	2	1
	10	Niedalina, IR, MSi	1	1	1	2	1	2
		100%	93%	100,00%	90%	93%	82,00%	
		Rata-rata =					93,11%	

Keterangan :

	Jawaban
Pertanyaan 1	1 = Ya
	2 = Tidak
Pertanyaan 2	1 = Respon Cepat
	2 = Kurang Cepat
	3 = Lambat
Pertanyaan 3	1 = Ya

	2 = Kurang
	3 = Tidak
Pertanyaan 4	1 = Ya
	2 = Sedikit
	3 = Tidak
Pertanyaan 5	1 = Baik
	2 = Kurang Baik
	3 = Tidak
Pertanyaan 6	1 = Sangat Puas
	2 = Puas
	3 = Kurang Puas
	4 = Sedikit
	5 = Tidak Puas



Lampiran F-2 Perhitungan Kepuasan Tampilan

Dalam metode Likert, maka perlu dilakukan penentuan skor/bobot masing-masing jawaban seperti yang tertera pada tabel dibawah.

Tabel Pembobotan Jawaban Kuesioner Tampilan

Keterangan	Jawaban	Bobot
Pertanyaan 1	1 = Sangat Menarik	5
	2 = Menarik	4
	3 = Biasa Saja	3
	4 = Kurang Menarik	2
	5 = Tidak Menarik	1
Pertanyaan 2	1 = Sudah	3
	2 = Belum Membantu	2
	3 = Tidak Membantu	1
Pertanyaan 3	1 = Tidak	2
	2 = Ada	1

Selanjutnya untuk perhitungannya, berikut ini dibuatkan masing-masing perhitungan 2 sampel jawaban pertanyaan (1 dan 2) sebagai berikut :

1. Tani / Kelompok Tani

a. Pertanyaan 1

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Sangat Menarik	$3 \times 5 = 15$
Menarik	$5 \times 4 = 20$
Biasa Saja	$2 \times 3 = 6$
Kurang Menarik	$0 \times 2 = 0$
Tidak Menarik	$0 \times 1 = 0$
Total	41

- Skor Ideal

$$\begin{aligned}\text{Skor Ideal (n)} &= \text{Nilai Skala max} \times \text{Jumlah responden} \\ &= 5 \times 10 \\ &= 50\end{aligned}$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{41}{50} \times 100\%$$

$$p = 82\%$$

b. Pertanyaan 2

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Sudah	10 x 3 = 30
Belum Membantu	0 x 2 = 0
Tidak Membantu	0 x 1 = 0
Total	30

- Skor Ideal

Skor Ideal (n) = Nilai Skala max x Jumlah responden

$$= 3 \times 10$$

$$= 30$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{30}{30} \times 100\%$$

$$p = 100\%$$

a. Pertanyaan 3

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Tidak	$10 \times 2 = 20$
Ada	$0 \times 1 = 0$
Total	20

- Skor Ideal

Skor Ideal (n) = Nilai Skala max x Jumlah responden

$$= 2 \times 10$$

$$= 20$$

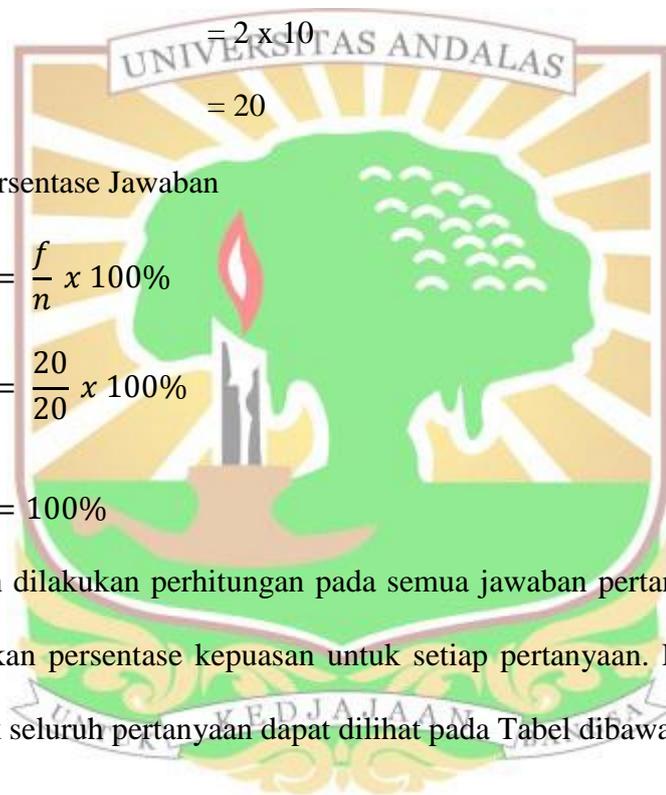
- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{20}{20} \times 100\%$$

$$p = 100\%$$

Setelah dilakukan perhitungan pada semua jawaban pertanyaan yang ada, maka didapatkan persentase kepuasan untuk setiap pertanyaan. Hasil persentase jawaban untuk seluruh pertanyaan dapat dilihat pada Tabel dibawah.



Tabel Hasil Kuesioner Tampilan Tani

	No	Nama Responden	Jawaban Pertanyaan ke-		
			Ke - 1	Ke - 2	Ke - 3
Tani/Kelompok Tani	1	Yeldi Mardi	2	1	1
	2	Rahmawati	1	1	1
	3	Busrialman	1	1	1
	4	Hendri	1	1	1
	5	Afrizul	3	1	1
	6	Mulyati	2	1	1
	7	Suhatman	2	1	1
	8	Asmalinda	3	1	1
	9	Joni	2	1	1
	10	Muchlis	2	1	1
			82%	100%	100%
			Rata-rata = 94%		

Keterangan :



Keterangan	Jawaban	Bobot
Pertanyaan 1	1 = Sangat Menarik	5
	2 = Menarik	4
	3 = Biasa Saja	3
	4 = Kurang Menarik	2
	5 = Tidak Menarik	1
Pertanyaan 2	1 = Sudah Membantu	3
	2 = Belum Membantu	2
	3 = Tidak Membantu	1
Pertanyaan 3	1 = Tidak	2
	2 = Ada	1

2. Tenaga Ahli

a. Pertanyaan 1

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Sangat Menarik	$1 \times 5 = 5$
Menarik	$7 \times 4 = 28$
Biasa Saja	$2 \times 3 = 6$
Kurang Menarik	$0 \times 2 = 0$
Tidak Menarik	$0 \times 1 = 0$
Total	39

- Skor Ideal

$$\begin{aligned} \text{Skor Ideal (n)} &= \text{Nilai Skala max} \times \text{Jumlah responden} \\ &= 5 \times 10 \\ &= 50 \end{aligned}$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{39}{50} \times 100\%$$

$$p = 78\%$$

b. Pertanyaan 2

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Sudah	8 x 3 = 24
Belum Membantu	2 x 2 = 4
Tidak Membantu	0 x 1 = 0
Total	28

- Skor Ideal

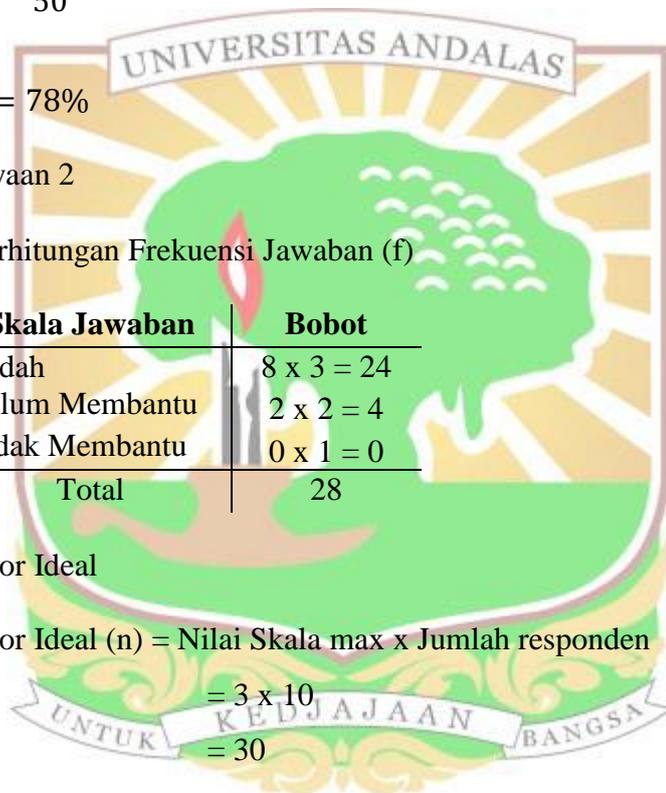
$$\begin{aligned} \text{Skor Ideal (n)} &= \text{Nilai Skala max} \times \text{Jumlah responden} \\ &= 3 \times 10 \\ &= 30 \end{aligned}$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{28}{30} \times 100\%$$

$$p = 93\%$$



c. Pertanyaan 3

- Perhitungan Frekuensi Jawaban (f)

Skala Jawaban	Bobot
Tidak	$7 \times 2 = 14$
Ada	$3 \times 1 = 3$
Total	17

- Skor Ideal

Skor Ideal (n) = Nilai Skala max x Jumlah responden

$$= 2 \times 10$$

$$= 20$$

- Persentase Jawaban

$$p = \frac{f}{n} \times 100\%$$

$$p = \frac{17}{20} \times 100\%$$

$$p = 85\%$$

Setelah dilakukan perhitungan pada semua jawaban pertanyaan yang ada, maka didapatkan persentase kepuasan untuk setiap pertanyaan. Hasil persentase jawaban untuk seluruh pertanyaan dapat dilihat pada Tabel dibawah.



Hasil Kuesioner Tampilan Ahli

	No	Nama Responden	Jawaban Pertanyaan ke-		
			Ke - 1	Ke - 2	Ke - 3
Tani/Kelompok Tani	1	Jefri Chandra, SP	2	1	1
	2	Ir. Erdiman	1	1	1
	3	Murni Kurniaty, STP, MP	1	1	1
	4	Elviana Anwar	1	1	1
	5	Aldilani	3	1	1
	6	Ikhwatul Khairiyah, SP	2	1	1
	7	Gunawan, SE	2	1	1
	8	Syamsul Bahri	2	1	1
	9	Mahdalena, SP	2	1	1
	10	Niedalina, IR, Msi	2	1	2
			78%	93%	85%
			Rata-rata =		87%

Keterangan :



Keterangan	Jawaban
Pertanyaan 1	1 = Sangat Menarik
	2 = Menarik
	3 = Biasa Saja
	4 = Kurang Menarik
	5 = Tidak Menarik
Pertanyaan 2	1 = Sudah Membantu
	2 = Belum Membantu
	3 = Tidak Membantu
Pertanyaan 3	1 = Tidak
	2 = Ada