

SKRIPSI

***Decision Support System (DSS) Memprediksi Kerusakan HP di
Pioneer Service Menggunakan Algoritma C4.5 Berbasis Web***

DISUSUN OLEH

MAISSY MASITOH SIHOMBING

2009010127



UMSU

Unggul | Cerdas | Terpercaya

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA**

MEDAN

2024

LEMBARAN PENGESAHAN

LEMBARAN PENGESAHAN

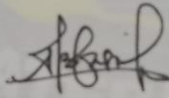
Judul Skripsi : *DECISION SUPPORT SYSTEM (DSS) MEMPREDIKSI KERUSAKAN HP DI PIONEER SERVICE MENGGUNAKAN ALGORITMA C4.5 BERBASIS WEB*

Nama Mahasiswa : MAISSY MASITOH SIHOMBING

NPM : 2009010127

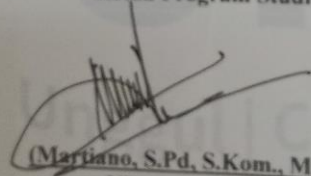
Program Studi : SISTEM INFORMASI

Menyetujui
Dosen Pembimbing



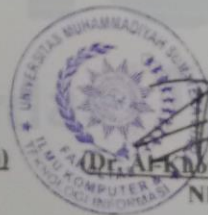
(Mhd Basri, S.Si, M.Kom)
NIDN. 0111078802

Ketua Program Studi



(Martiano, S.Pd, S.Kom., M.Kom)
NIDN. 0128029302

Dekan



(Dr. Al-Khwarizmi, S.Kom., M.Kom.)
NIDN. 0127099201

PERNYATAAN ORISINALITAS

PERNYATAAN ORISINALITAS

*DECISION SUPPORT SYSTEM (DSS) MEMPREDIKSI KERUSAKAN HP
DI PIONEER SERVICE MENGGUNAKAN ALGORITMA C4.5
BERBASIS WEB*

SKRIPSI

Saya menyatakan bahwa karya tulis ini adalah hasil karya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya.

Medan, 07 Juni 2024

Yang membuat pernyataan



MAISSY MASITOH SIHOMBING

**PERNYATAAN PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN
AKADEMIS**

**PERNYATAAN PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN
AKADEMIS**

Sebagai sivitas akademika Universitas Muhammadiyah Sumatera Utara, saya bertanda tangan dibawah ini:

Nama : MAISSY MASITOH SIHOMBING
NPM : 2009010127
Program Studi : Sistem Informasi
Karya Ilmiah : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Muhammadiyah Sumatera Utara Hak Bebas Royalti Non-Eksekutif (*Non-Exclusive Royalty free Right*) atas penelitian skripsi saya yang berjudul:

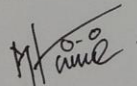
**DECISION SUPPORT SYSTEM (DSS) MEMPREDIKSI KERUSAKAN HP
DI PIONEER SERVICE MENGGUNAKAN ALGORITMA C4.5
BERBASIS WEB**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksekutif ini, Universitas Muhammadiyah Sumatera Utara berhak menyimpan, mengalih media, memformat, mengelola dalam bentuk database, merawat dan mempublikasikan Skripsi saya ini tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis dan sebagai pemegang dan atau sebagai pemilik hak cipta.

Demikian pernyataan ini dibuat dengan sebenarnya.

Medan, 05 Juni 2024

Yang membuat pernyataan



MAISSY MASITOH SIHOMBING
NPM. 2009010127

RIWAYAT HIDUP

DATA PRIBADI

Nama Lengkap : Maissy Masitoh Sihombing
Tempat dan Tanggal Lahir : Medan, 19 Oktober 2001
Alamat Rumah : Sumbul
Telepon/Faks/HP : 089509687213
E-mail : maissymasitoh9@gmail.com
Instansi Tempat Kerja :
Alamat Kantor : -

DATA PENDIDIKAN

SD : SD Negeri 030355 Parratusan TAMAT: 2013
SMP : SMP Negeri 1 Sumbul TAMAT: 2016
SMA : SMK Swasta Ris Maduma Sumbul TAMAT: 2019

Decision Support System (DSS) Memprediksi Kerusakan HP di Pioneer Service Menggunakan Algoritma C.45 Berbasis Web

ABSTRAK

Skripsi ini mengembangkan Sistem Pendukung Keputusan (SPK) untuk memprediksi kerusakan smartphone di Pioneer Service menggunakan algoritma C4.5. Seiring dengan pentingnya smartphone dalam kehidupan modern dan peningkatan frekuensi penggunaannya, insiden kerusakan pun meningkat. Metode diagnostik manual yang ada saat ini di Pioneer Service memakan waktu yang lama, sehingga mempengaruhi kepuasan pelanggan. SPK yang diusulkan memanfaatkan algoritma C4.5 untuk menganalisis pola dalam data kerusakan historis dan memprediksi masalah potensial berdasarkan gejala yang diamati. Sistem ini membantu teknisi dengan menyediakan diagnostik yang lebih cepat, sehingga meningkatkan efisiensi layanan dan kepuasan pelanggan. Implementasinya melibatkan penggunaan teknologi web dan Python untuk perhitungan backend, memastikan solusi yang kuat dan dapat diskalakan.

Kata Kunci : Sistem Pendukung Keputusan, Algoritma C.45, Kerusakan Smartphone, Python, Teknologi Web

***Decision Support System (DSS) for Predicting Smartphone Malfunctions at
Pioneer Service Using the C4.5 Algorithm Web-Based***

ABSTRACT

This thesis develops a Decision Support System (DSS) to predict smartphone malfunctions at Pioneer Service using the C4.5 algorithm. As smartphones become indispensable in modern life and their usage frequency increases, incidents of malfunctions also rise. The current manual diagnostic methods at Pioneer Service are time-consuming, thus affecting customer satisfaction. The proposed DSS utilizes the C4.5 algorithm to analyze patterns in historical malfunction data and predict potential issues based on observed symptoms. This system aids technicians by providing quicker diagnostics, thereby enhancing service efficiency and customer satisfaction. The implementation involves the use of web technologies and Python for backend calculations, ensuring a robust and scalable solution.

Keyword : *Decision Support System, C4.5 Algorithm, Smartphone Malfunctions, Python, Web Technologies.*

KATA PENGANTAR



Alhamdulillah, puji syukur kehadiran Allah SWT atas segala limpahan taufiq, rahmat hidayah, serta inayah-Nya, sehingga penulis dapat menyusun dan menyelesaikan laporan skripsi dengan judul “*Decision Support System (DSS) Memprediksi Kerusakan HP di Pioneer Service Menggunakan Algoritma C4.5 Berbasis Web*”. Skripsi ini disusun untuk memenuhi syarat dalam menyelesaikan studi dan memperoleh gelar sarjana untuk program studi Sistem Informasi Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara.

Dalam pelaksanaan penelitian sampai pembuatan skripsi ini, Banyak hal yang dipetik ketika menyusun laporan tugas akhir ini. Begitu pula dengan berbagai kendala yang muncul dan memberikan manfaat dikemudian hari. Semua itu tak lepas dari peran orang-orang disekitar saya yang selalu memberikan dukungan dan motivasi dalam penyusunan laporan tugas akhir saya , Melalui kesempatan ini, penulis menyampaikan rasa terima kasih yang sebesar-besarnya dan penghargaan yang setinggi-tingginya kepada :

1. Bapak Prof. Dr. Agussani, M.AP., Selaku Rektor Universitas Muhammadiyah Sumatera Utara (UMSU).
2. Bapak Dr. Al-Khowarizmi, S.Kom., M.Kom. Selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
3. Bapak Martiano S.pd, S.Kom., M. Kom Selaku Ketua Program Studi Sistem Informasi.

4. Bapak Halim Maulana, S.T, M.Kom., Selaku Wakil Dekan 1 Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
5. Bapak Mhd Basri, S.Si, M.Kom Selaku Dosen Pembimbing saya yang selalu meluangkan waktu untuk membimbing saya dan memberikan pengingat untuk selalu konsisten mengerjakan skripsi.
6. Ibu Fatma Sari Hutagalung, M.Kom Selaku Dosen Penguji saya yang Telah memberikan kritik dan saran kepada saya sehingga terbentuk skripsi yang baik.
7. Mama yang senantiasa membantu dan mendukung saya melalui Doa yang Dipanjatkan.
8. Ayah yang senantiasa mendoakan saya dari jauh sana.
9. Saudara - saudari saya Nisa, Ainun, Dinda, Ihwa, Dan Dika yang senantiasa memberikan semangat kepada penulis sampai saat ini.
10. Kepada Npm 2009010111 yang selalu menerima emosi saya, menggantikan beban saya dan ikut serta dalam perjalanan saya sejak kuliah hingga saat ini, terimakasih telah hadir.
11. Kepada Nathan Tjoe A On yang membuat saya semangat dalam melakukan apa pun untuk menyelesaikan tugas akhir ini.
12. kepada Lee Dong Hyuk terimakasih telah menjadi motivasi saya untuk selalu kuat dalam menjalani dunia perkuliahan.
13. Kepada Maissy Masitoh terimakasih telah berjuang dari semester awal hingga semester akhir dan selalu kuat dalam segala hal walaupun diremehkan orang atau di sepelekan orang-orang, yang telah memaksimalkan segala hal dengan maksimal dan baik, yang telah sabar dan

mengikuti segala alur hidup yang rumit. terimakasih telah bertahan dalam dunia yang keras ini.

Dalam menyelesaikan tugas akhir ini saya sebagai peneliti tentu mempunyai banyak kekurangan dan kesalahan baik disengaja maupun tidak disengaja. Maka dari itu, saya akan sangat menghargai setiap masukan dan kritik untuk membangun diri saya agar lebih baik kedepannya. Akhir kata, saya memohon maaf sebesarbesarnya dan berharap tugas akhir ini dapat berguna dan memberikan manfaat.

Terimakasih Saya Ucapkan

Medan, 07 Juni 2024

Penyusun

Maissy Masitoh S

Npm : 2009010127

DAFTAR ISI

DAFTAR ISI.....	ii
DAFTAR TABEL.....	iv
DAFTAR GAMBAR	v
BAB I PENDAHULUAN	1
1.1. Latar Belakang Masalah.....	1
1.2. Rumusan Masalah	3
1.3. Batasan Masalah.....	3
1.4. Tujuan Penelitian	3
1.5. Manfaat Penelitian	4
BAB II LANDASAN TEORI	5
2.1. Sistem Pendukung Keputusan (<i>Decision Support System</i>)	5
2.2. <i>Decision Tree</i> / Pohon Keputusan.....	6
2.3. Algoritma C.45.....	7
2.4. Kerusakan pada <i>Handphone</i>	8
2.5. Hypertext Markup Language (HTML)	9
2.6. Hypertext Preprocessor (PHP).....	9
2.7. MySQL.....	9
2.8. <i>Javascript</i>	10
2.9. <i>Python</i>	10
2.10. <i>Library</i>	11
2.11. <i>Pandas</i>	11
2.12. <i>Scikit-Learn</i>	12
2.13. <i>Confusion Matrix</i>	12
2.14. Pra-Pemrosesan Data	13
2.15. Unified Modelling Language (UML).....	14
BAB III METODE PENELITIAN.....	20
3.1. Pengumpulan Data	20
3.2. Pra-Pemrosesan Data	21
3.3. Implementasi Algoritma C.45	24
3.4. Evaluasi Model	26
3.5. Bagan Alir (<i>Flowchart</i>).....	26
3.6. Desain Sistem.....	27
3.7. Desain Rancangan Program	42

3.8.	Penulisan Kode Program Sistem.....	47
3.9.	Pengujian Program.....	48
3.10.	Lokasi Penelitian.....	49
3.11.	Jadwal Penelitian.....	49
BAB IV HASIL DAN PEMBAHASAN		50
4.1.	Tampilan Hasil.....	50
4.2.	Perhitungan Algoritma C4.5	57
4.3.	Melatih Model Algoritma C.45 Pada <i>Google Collab</i>	65
4.4.	Kelebihan dan Kekurangan Program	74
BAB V KESIMPULAN DAN SARAN.....		76
5.1.	Kesimpulan	76
5.2.	Saran.....	76
LAMPIRAN.....		80

DAFTAR TABEL

Tabel 2. 1 Simbol Dalam Use Case	15
Tabel 2. 2 Simbol Activity Diagram.....	16
Tabel 2. 3 Tabel Simbol Sequence Diagram.....	17
Tabel 3. 1 Struktur Tabel User.....	39
Tabel 3. 2 Struktur Tabel Kerusakan	39
Tabel 3. 3 Struktur Tabel Atribut Kerusakan.....	40
Tabel 3. 4 Struktur Tabel Konsultasi Kerusakan	40
Tabel 3. 5 Struktur Tabel Detail Konsultasi.....	41
Tabel 3. 6 Jadwal Penelitian.....	49
Tabel 4. 1 Tabel Data Pemeriksaan.....	57
Tabel 4. 2 Tabel Daftar Kerusakan berdasarkan Jumlah kasus	57
Tabel 4. 3 Sebaran kerusakan berdasarkan data.....	59
Tabel 4. 4 Data Information Gain Pada Percabangan Pertama.....	60
Tabel 4. 5 Distribusi kerusakan berdasarkan atribut sebelumnya.....	62
Tabel 4. 6 Tabel hasil gain untuk cabang berikutnya.....	62
Tabel 4. 7 Perbandingan Label Sebenarnya dan Hasil Prediksi.....	70
Tabel 4. 8 Tabel Pengujian Sistem Login	81
Tabel 4. 9 Tampilan Pengujian Dashboard.....	81
Tabel 4. 10 Tampilan Pengujian Data Atribut	82
Tabel 4. 11 Tampilan Pengujian Data Kerusakan.....	82
Tabel 4. 12 Tabel Pengujian Data Konsultasi dan Prediksi	83
Tabel 4. 13 Tabel Pengujian Sistem Data User.....	84

DAFTAR GAMBAR

Gambar 2. 1 Komponen dalam Class Diagram.....	19
Gambar 3. 1 Kode Import Library Pandas	23
Gambar 3. 2 Kode identifikasi dan menghapus nilai data yang hilang.....	24
Gambar 3. 3 Flowchart prediksi kerusakan HP	27
Gambar 3. 4 Class Diagram <i>Decision Support System</i> (DSS) Memprediksi Kerusakan HP di Pioneer Service Menggunakan Algoritma C.45 Berbasis Web	29
Gambar 3. 5 Activity Diagram Login	30
Gambar 3. 6 Activity Diagram Kelola Data User	31
Gambar 3. 7 Activity Diagram Kelola Data Kerusakan	32
Gambar 3. 8 Activity Diagram Kelola Data Atribut	33
Gambar 3. 9 Activity Diagram Konsultasi Kerusakan.....	34
Gambar 3. 10 Sequence Diagram Login	35
Gambar 3. 11 Sequence Diagram Data User	36
Gambar 3. 12 Sequence Diagram Data Kerusakan.....	36
Gambar 3. 13 Sequence Diagram Kelola Atribut Kerusakan	37
Gambar 3. 14 Sequence Diagram Kelola Data Konsultasi	38
Gambar 3. 15 Rancangan Form Login.....	42
Gambar 3. 16 Rancangan menu dashboard.....	43
Gambar 3. 17 Rancangan menu tambah atribut	43
Gambar 3. 18 Rancangan menu data atribut	44
Gambar 3. 19 Rancangan menu tambah kerusakan	44
Gambar 3. 20 Rancangan Menu Data Kerusakan	45
Gambar 3. 21 Rancangan menu konsultasi dan hasilkan prediksi	45

Gambar 3. 22 Rancangan Form Perhitungan C4.5	46
Gambar 3. 23 Rancangan menu tambah user.....	46
Gambar 4. 1 Use Case Diagram Decision Support System (DSS) Memprediksi Kerusakan HP di Pioneer Service Menggunakan Algoritma C.45 Berbasis Web	28
Gambar 4. 2 Tampilan Menu Login.....	50
Gambar 4. 3 Tampilan Menu Tambah Atribut.....	51
Gambar 4. 4 Tampilan Menu Data Atribut	51
Gambar 4. 5 Tampilan Menu Tambah Kerusakan	52
Gambar 4. 6 Tampilan menu Data Kerusakan	53
Gambar 4. 7 Tampilan menu Tambah Konsultasi dan Prediksi.....	54
Gambar 4. 8 Tampilan Menu Data Konsultasi.....	54
Gambar 4. 9 Tampilan menu Detail Konsultasi.....	55
Gambar 4. 10 Tampilan Menu Perhitungan C4.5	56
Gambar 4. 11 Tampilan Menu Tambah User.....	56
Gambar 4. 12 Tampilan menu data user	57
Gambar 4. 13 Dataset yang diberi ciri-ciri kerusakan.....	56
Gambar 4. 14 Dataset Kerusakan HP setelah atribut diberi data numerik.....	57
Gambar 4. 15 Pohon keputusan pada node awal.....	61
Gambar 4. 16 Pohon keputusan cabang kedua.....	61
Gambar 4. 17 Gambar Pohon keputusan setelah pemilihan cabang kedua.....	64
Gambar 4. 18 Pohon keputusan setelah mendapatkan label kerusakan IC Mikroprosesor.....	65
Gambar 4. 19 Memasukkan Pustaka Python yang digunakan	66
Gambar 4. 20 Memuat data yang digunakan	66

Gambar 4. 21 Tampilan data setelah dimuat ke dataframe pandas.....	67
Gambar 4. 22 Implementasi Pemisahan Data	67
Gambar 4. 23 Implementasi Ubah Kategori teks menjadi fitur angka.....	68
Gambar 4. 24 Tampilan data latih.....	68
Gambar 4. 25 Implementasi Decision Tree Classifier	69
Gambar 4. 26 Implementasi Fitur numerik pada data uji.....	69
Gambar 4. 27 Implementasi Algoritma terhadap data uji	69
Gambar 4. 28 Implementasi menghitung Performa Model.....	71
Gambar 4. 29 Precision, Recall, dan F1 Score.....	72
Gambar 4. 30 Gambar pohon keputusan yang dihasilkan	73
Gambar 4. 31 Gambar keseluruhan Pohon Keputusan	74

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Service handphone adalah suatu perbaikan dimana *handphone* tersebut terdapat kerusakan atau yang harus diperbaiki . Pada era digital dan modern saat ini, penggunaan *handphone* menjadi perangkat yang penting dalam kehidupan sehari-hari. Penggunaan *handphone* dengan frekuensi yang cukup tinggi mengakibatkan meningkatnya potensi kerusakan, sehingga layanan perbaikan *handphone*, memiliki peran penting dalam mengatasi kerusakan-kerusakan yang terjadi. *Pioneer Service* menjadi salah satu jasa layanan perbaikan *handphone*, setiap harinya, tempat ini menerima berbagai macam keluhan kerusakan *handphone* , yang berbeda-beda setiap harinya. Kerusakan *handphone* tentunya disebabkan oleh banyak jenis faktor. Dengan banyaknya faktor yang mempengaruhi kerusakan pada *handphone*, dibutuhkan waktu yang cukup lama untuk menentukan jenis kerusakan pada *handphone* terkait. Hal ini tentu dapat mengurangi kepuasan pelanggan terhadap pelayanan *Pioneer Service*, terlebih ketika pelanggan membutuhkan *handphone* kembali berfungsi kembali dengan cepat.

Sebagai solusi masalah tersebut, dibutuhkan sebuah Sistem Informasi yang dapat memprediksi kerusakan pada ponsel pintar. Sistem Informasi dapat didefinisikan sebagai pengintegrasian unsur sistem dan unsur informasi yang merupakan elemen-elemen yang berinteraksi untuk menghasilkan informasi bagi penggunaannya (Rahman et al., 2022). Saat ini, banyak instansi, perusahaan, atau tempat usaha lainnya yang menggunakan sistem informasi dikarenakan memiliki

beragam manfaat untuk kegiatan operasional perusahaan, dalam kasus Pioneer Service, dibutuhkan sistem yang dapat memprediksi kerusakan pada *Handphone*, salah satu jenis Sistem Informasi yang dapat dibuat sesuai kebutuhan Pioneer Service adalah Sistem Pendukung Keputusan (*Decision Support System*).

Sistem Pendukung Keputusan (*Decision Support System*) merupakan sistem yang membantu pengambilan keputusan dalam organisasi atau perusahaan yang didesain untuk memfasilitasi proses pengambilan keputusan dengan menyediakan informasi yang terstruktur dan relevan. (Sarwandi et al., 2023). Dalam menganalisa data yang dijadikan dalam bentuk informasi dan rekomendasi, Sistem Pendukung Keputusan membutuhkan Algoritma yang tepat. Algoritma memiliki pengertian sebagai sekumpulan tindakan untuk menyelesaikan suatu persoalan. Dengan algoritma yang tepat, sistem pendukung keputusan dapat menghasilkan keputusan dengan menggunakan data dan informasi yang tersedia. (Khairani Aritonang et al., 2022)

Salah satu algoritma yang dapat digunakan untuk sistem pendukung keputusan ini adalah adalah Algoritma C45, Algoritma C.45 merupakan algoritma yang digunakan untuk membangun pohon keputusan. pohon keputusan dibuat menggunakan struktur dimana terdapat simpul yang mendeksripsikan atribut-atribut, setiap cabang menggambarkan hasil dari atribut yang diuji dan setiap daun menggambarkan kelas. Algoritma ini juga menghitung *information gain* dari atribut untuk menentukan seberapa informatifnya dalam pemisahan data agar lebih optimal, dan menghitung entropy untuk heterogenitas data. Berdasarkan uraian masalah diatas, maka penulis membuat sistem pendukung keputusan (*Decision Support System*) dengan menggunakan algoritma C4.5 untuk memprediksi kerusakan pada ponsel pintar dengan harapan dapat membuka peluang untuk peningkatan efisiensi dalam kegiatan operasional.

1.2. Rumusan Masalah

Berdasarkan masalah yang diuraikan diatas, adapun rumusan masalah dari penulis adalah bagaimana membangun Sistem Pendukung Keputusan (*Decision Support System*) Memprediksi Kerusakan HP di Pioneer Service Menggunakan Algoritma C.45.

1.3. Batasan Masalah

Dalam penelitian ini, terdapat beberapa batasan yang perlu ditetapkan agar dapat memberikan fokus yang jelas terhadap pengembangan Sistem Pendukung Keputusan (*Decision Support System*) berbasis Algoritma C.45 ini, Batasan masalah tersebut antara lain :

1. Penelitian ini hanya akan fokus pada pengembangan Sistem Pendukung Keputusan (*Decision Support System*) untuk memprediksi kerusakan pada *Handphone* di Pioneer Service.
2. Faktor-Faktor yang mempengaruhi kerusakan *Handphone* dapat diidentifikasi terbatas pada variabel yang relevan sesuai dengan data kerusakan yang sudah tercatat di *Pioneer Service*.
3. Aplikasi berbasis web yang dikembangkan dalam penelitian hanya untuk memprediksi beberapa jenis kerusakan hp saja. Aplikasi yang dibangun menggunakan bahasa pemrograman php untuk *front-end* dan beberapa fungsionalitas aplikasi, MySQL untuk basis data, dan *python* untuk melakukan prediksi kerusakan,

1.4. Tujuan Penelitian

Penelitian ini bertujuan untuk mencapai beberapa target berikut ini

1. Mengembangkan Sistem Pendukung Keputusan (*Decision Support System*) berbasis algoritma C.45 yang dapat memprediksi kerusakan pada *handphone* dengan akurasi yang baik.
2. Mengidentifikasi faktor faktor yang mempengaruhi kerusakan pada *Handphone* agar dapat dimasukkan kedalam Sistem Pendukung keputusan
3. Meningkatkan efisiensi operasional dalam layanan perbaikan *handphone* pada Pioneer Service, khususnya dalam proses mengetahui kerusakan dari perangkat, sehingga memberikan pelayanan yang lebih cepat dan optimal kepada pelanggan *Pioneer Service*.

1.5. Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut :

1. Meningkatkan Kepuasan Pelanggan dengan terciptanya Sistem Pendukung Keputusan yang dapat membuat layanan perbaikan *handphone* lebih cepat dan efisien.
2. Membantu teknisi perbaikan *handphone* dengan informasi yang lebih akurat terkait jenis kerusakan, sehingga mempercepat proses perbaikan
3. Berkontribusi pada peningkatan produktivitas dan efisiensi layanan perbaikan *Handphone* sesuai dengan tuntutan era digital dan modern saat ini.

BAB II

LANDASAN TEORI

2.1. Sistem Pendukung Keputusan (*Decision Support System*)

Keputusan adalah suatu pilihan satu alternatif dari beberapa alternatif penyelesaian masalah untuk mengakhiri atau menyelesaikan masalah tersebut (Pribadi et al., 2020). Sementara itu, sistem pendukung keputusan, merupakan sebuah sistem yang dirancang untuk membantu pengambilan keputusan di dalam organisasi atau perusahaan. Sistem ini bertujuan memfasilitasi proses pengambilan keputusan dengan menyediakan informasi yang terstruktur dan relevan (Sarwandi et al., 2023).

Tujuan utama dari penggunaan sistem pendukung keputusan adalah untuk meningkatkan efisiensi pengambilan keputusan. Dengan adanya sistem ini, pengambil keputusan dapat menganalisis data dengan lebih baik dan cepat, memungkinkan mereka untuk mengambil keputusan dengan lebih cepat pula. Sistem pendukung keputusan bukan hanya sebuah alat untuk menyajikan data, tetapi juga memberikan info mendalam melalui teknik analisis yang beragam, seperti analisis statistik, pemodelan prediktif, dan metode seperti *Decision Tree*.

Sistem pendukung keputusan bekerja dengan memanfaatkan data, model, dan teknik analisis tertentu. Data yang digunakan dapat berasal dari sumber seperti basis data, internet, atau sistem informasi lainnya. Penggunaan berbagai teknik analisis, termasuk *Decision Tree*, memungkinkan sistem ini menyajikan informasi yang dapat membantu pengambil keputusan memahami dampak dari setiap alternatif keputusan yang mungkin diambil.

2.2. *Decision Tree* / Pohon Keputusan

Diantara beberapa metode atau teknik yang dapat digunakan untuk membuat prediksi dalam Sistem Pendukung Keputusan adalah metode *decision tree* atau pohon keputusan. Pohon Keputusan dapat diartikan sebagai suatu cara untuk memprediksi atau mengklarifikasi, yang mana pohon Keputusan dapat membagi kumpulan data yang besar menjadi himpunan-himpunan *record* yang lebih kecil dengan menerapkan serangkaian aturan keputusan (Hafizan & Putri, 2020).

Komponen-komponen dalam pohon keputusan pada umumnya adalah sebagai berikut :

1. Simpul (*Nodes*)

Dalam pohon keputusan terdapat tiga jenis simpul :

a. Simpul Akar

Simpul ini mewakili pilihan yang akan membagi semua data menjadi dua atau lebih kelompok

b. Simpul Internal

Simpul ini adalah pilihan yang tersedia pada suatu titik dalam pohon dan terhubung dengan simpul induk dan simpul anak

c. Simpul Daun

Simpul ini mewakili hasil akhir dari kombinasi keputusan atau peristiwa

2. Cabang (*Branches*)

Cabang menghubungkan satu simpul pohon ke simpul pohon lainnya dalam ruang lingkup pohon keputusan, dan cabang akan merepresentasikan aliran keputusan dari satu simpul pohon ke dalam satu simpul lainnya

Dengan kata lain, pohon keputusan bekerja dengan membentuk pohon keputusan yang dapat disimpulkan aturan-aturan klasifikasi tertentu, salah satu algoritma yang menerapkan pohon keputusan adalah algoritma C.45 (Azwanti, 2019).

2.3. Algoritma C.45

Algoritma C.45 digunakan untuk membentuk pohon keputusan dan dapat memodelkan hasil temuan data penting sehingga pengetahuan atau informasi yang ditemukan mudah untuk dipahami (Yani & Tanjung, 2021). Secara umum langkah langkah yang dilakukan dalam membangun sebuah pohon keputusan dengan algoritma C.45 adalah sebagai berikut :

1. Memilih atribut sebagai akar
2. Buat cabang untuk setiap nilai
3. Membagi kasus dari tiap cabang
4. Ulangi langkah yang sama untuk setiap cabang.

Proses pemilihan atribut akar sendiri, didasarkan pada suatu nilai yang dinyatakan sebagai nilai *information gain* tertinggi dari atribut-atribut yang sudah ada. *Information Gain* adalah nilai yang digunakan untuk dapat mengukur efektivitas suatu atribut dalam mengklasifikasikan data, untuk mengetahui nilai dari *Information Gain*, kita diharuskan menghitung Entropy terlebih dahulu. Dijelaskan juga bahwa, kita biasanya memakai *Entropy* sebagai parameter atau acuan untuk mengetahui keberagaman (heterogenitas) dari data. semakin beragam sampel pada data yang digunakan, maka nilai *Entropy* akan besar juga (Suyanto, 2021).

Nilai *entropy* dapat dihitung menggunakan rumus dibawah ini :

$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{i=1}^n \frac{|S_i|}{|S|} * \text{Entropy}(S_i) \dots\dots\dots (2.1)$$

Dimana :

S : Himpunan kasus

A : Atribut

n : jumlah partisi atribut A

|S_i| : Jumlah kasus pada partisi ke-i

|S| : Jumlah kasus dalam S

Untuk perhitungan *entropy* sendiri, dapat dilihat pada persamaan dibawah ini

$$\text{Entropy}(S) = \sum_{i=1}^n - p_i * \log_2 p_i \dots\dots\dots (2.2)$$

Dimana :

S : Kumpulan Kasus

N : Jumlah partisi S

P_i : proporsi dari S_i terhadap S

2.4. Kerusakan pada *Handphone*

Kerusakan pada *Handphone* adalah perubahan yang terjadi yang mengakibatkan menurunnya *Handphone* dari kondisi awalnya, kerusakan *Handphone* tidak hanya terfokus kepada perubahan pada fungsional sistem yang tidak optimal, kerusakan juga bisa terjadi pada fisik *Handphone* yang membuat *Handphone* tidak dapat berfungsi sebagaimana mestinya (Nur Arif et al., 2021).

Beberapa jenis kerusakan HP yang terjadi antara lain dapat dikelompokkan sebagai berikut :

1. Kerusakan *IC Power*
2. Kerusakan IC Microprocessor
3. Kerusakan Sistem Operasi
4. Kerusakan IC Power Amplifier
5. Kerusakan Pada Baterai (*Battery Drop*)
6. Kerusakan IC *eMMC*
7. Kerusakan IC *Charging*

2.5. Hypertext Markup Language (HTML)

HTML atau *HyperText Markup Language* adalah Bahasa *markup* sebagai dasar untuk menciptakan halaman web. Fungsinya untuk mengatur struktur dari sebuah situs web. HTML juga digunakan untuk menandai bagian mana yang berperan sebagai judul, isi, daftar tabel, dan lainnya. (Pratama, 2020).

2.6. Hypertext Preprocessor (PHP)

PHP adalah bahasa pemrograman skrip yang terintegrasi dengan HTML dan dieksekusi di sisi server, sementara yang dikirim ke browser hanya hasil akhirnya. Saat pengguna internet mengakses situs yang menggunakan PHP, server akan memproses semua perintah PHP terlebih dahulu dan kemudian mengirimkan hasilnya dalam format HTML ke browser pengguna. Dengan demikian, kode PHP asli tidak terlihat di browser. PHP dapat digunakan untuk mengambil informasi dari formulir web dan menggunakannya untuk berbagai fungsi (Aniqsa, 2019).

2.7. MySQL

MySQL merupakan perangkat lunak basis data relasi atau *Relational Database Management System* (RDBMS) yang diterapkan menggunakan lisensi GPL (*General Public License*). MySQL memiliki kinerja, kecepatan proses yang tidak

kalah dengan database-database besar lainnya (Aniqsa, 2019). Beberapa pertimbangan dalam memilih MySQL antara lain :

1. Kecepatan
2. Mudah Digunakan
3. Terbuka
4. Kapabilitas
5. Konektifitas dan Keamanan

2.8. Javascript

Javascript adalah Bahasa pemrograman yang digunakan dalam pengembangan website agar lebih dinamis dan interaktif. Javascript dapat meningkatkan fungsionalitas pada halaman website dengan fitur fitur yang disediakan olehnya

2.9. Python

Python adalah salah satu Bahasa pemrograman yang populer, yang dibuat pertama kali pada tahun 1991 oleh Guido van Rossum. Bahasa Pemrograman Python sendiri dapat digunakan dalam pengembangan aplikasi web, pengembangan software, hingga mennyelesaikan permasalahan matematika dari yang dasar dan kompleks dan bisa digunakan untuk pengembangan algoritma prediksi.

Keunggulan menggunakan Python adalah, dapat digunakan di berbagai platform, memiliki *syntax* yang mudah dan *simple*, dan dapat digunakan dalam pemrograman procedural, pemrograman berorientasi objek, maupun pemrograman fungsional.

2.10. *Library*

Library dalam bidang data berarti kumpulan sumber daya yang digunakan oleh program komputer yang biasanya digunakan untuk pengembangan perangkat lunak, hal ini termasuk data konfigurasi, dokumentasi, data bantuan, kode dan fungsi program yang telah ditulis sebelumnya. Dengan kata lain, *library* merujuk pada kumpulan implementasi perilaku program yang ditulis dalam bahasa pemrograman tertentu.

Dengan menggunakan *library*, Ketika ingin menulis program Tingkat yang lebih tinggi, kita dapat menggunakan *library* untuk panggilan beberapa fungsi dan perilaku dalam system, hal ini lebih menguntungkan alih-alih mengimplementasikan fungsi sistem tersebut dari awal berulang kali (Sudewo, 2022). Dalam pengolahan data, terdapat berbagai macam *library* yang dapat membantu analisis dan manipulasi data dengan lebih efisien dan efektif. Diantaranya adalah *Pandas* dan *Scikit-Learn*.

2.11. *Pandas*

Pandas adalah *library* pada *Python* yang paling sering dipakai untuk menganalisis data. Cara kerja *pandas* cukup menarik, Data berupa format CSV atau SQL diubah menjadi objek dalam *Python* dengan bentuk baris dan kolom yang disebut sebagai *Dataframe*. Objek *Dataframe* ini akan terlihat sangat mirip dengan tabel yang terdapat dalam aplikasi pengolahan statistika pada umumnya.

Penggunaan *Library Pandas* juga dapat membantu dalam verifikasi data, pengolahan data, normalisasi data, penggabungan dan penyatuan data, inspeksi data, serta membuat dan menyimpan data .

2.12. *Scikit-Learn*

Scikit-Learn adalah *library* dalam bahasa pemrograman *python* yang membantu melakukan proses pada data ataupun melakukan pengolahan data dan pelatihan model pada beberapa kebutuhan dalam lingkup *data science* seperti klasifikasi, *clustering*, dan lain-lain. *Library* ini sangat populer dikarenakan banyak sekali model pembelajaran mesin yang dapat dipanggil menggunakan *scikit-learn* (Bisa AI Academy, 2023).

2.13. *Confusion Matrix*

Confusion Matrix adalah tabel yang menyatakan klasifikasi jumlah uji yang benar dan jumlah uji yang salah (Normawati & Prayogi, 2021). *Confusion Matrix* sendiri memiliki empat istilah :

1. *True Negative* (TN) yang berarti model klasifikasi memprediksi data ada di kelas *negative* namun yang sebenarnya data memang ada di kelas *negative*.
2. *True Positive* (TP) yang berarti model klasifikasi memprediksi data ada di kelas positif namun sebenarnya memang data berada di kelas positif.
3. *False Negative* (FN) yang berarti model klasifikasi memprediksi data ada di kelas *negative* namun yang sebenarnya data ada di kelas positif.
4. *False Positive* (FP) yang berarti model klasifikasi memprediksi data ada di kelas positif namun yang sebenarnya data ada di kelas *negative*.

Dari definisi *confusion matrix*, beberapa poin poin dalam *confusion matrix* digunakan untuk menghitung *precision*, *recall*, dan *f1 score*. *Precision* adalah perbandingan antara *True Positive* (TP) dengan banyaknya data yang diprediksi positif, secara matematis dapat dilihat dibawah ini

$$Precision = \frac{TP}{TP+FP} \dots\dots\dots (2.3)$$

Untuk *recall* sendiri adalah perbandingan antara *true positive* (TP) dengan banyaknya data yang sebenarnya positif. Dapat dinyatakan secara matematis seperti dibawah ini

$$Recall = \frac{TP}{TP+FN} \dots\dots\dots (2.4)$$

Sedangkan *F1 Score* adalah nilai tengah dari *precision* dan *recall*. Nilai terbaik dari *F1 Score* adalah 1 dan nilai terburuknya adalah 0, secara matematis dapat dituliskan seperti dibawah ini

$$\frac{1}{F1} = \frac{1}{2} \left(\frac{1}{precision} + \frac{1}{recall} \right) \dots\dots\dots (2.5)$$

Nilai *F1 Score* yang baik menandakan bahwa model klasifikasi kita punya *precision* dan *recall* yang baik (S. Setiawan, 2020).

2.14. Pra-Pemrosesan Data

Pra-pemrosesan data adalah tahapan untuk mengolah data mentah dengan menghilangkan beberapa permasalahan yang mengganggu saat pemrosesan data. Hal tersebut dikarenakan oleh data yang formatnya tidak konsisten. Melalui proses ini, pemodelan algoritma C.45 akan berjalan lebih efektif dan efisien. Adapun tahapan tahapan dalam pra-pemrosesan data adalah :

1. Pertama-tama dalam tahap awal *preprocessing* data, langkah yang esensial adalah melakukan data cleaning. Proses ini melibatkan pemilihan kembali data mentah untuk menghilangkan entri yang tidak lengkap, tidak relevan, atau tidak akurat. Dengan melakukan ini, kita dapat menghindari kesalahpahaman saat melakukan analisis terhadap data tersebut.
2. Langkah berikutnya adalah *data integration*, yang diperlukan karena *preprocessing* data melibatkan penggabungan data dari berbagai sumber ke

dalam satu dataset. Penting untuk memastikan bahwa data dari berbagai sumber memiliki format yang seragam.

3. Setelah itu, kita melanjutkan ke tahap transformasi data. Sebagaimana dijelaskan sebelumnya, data yang berasal dari sumber yang berbeda mungkin memiliki format yang beragam. Oleh karena itu, perlu dilakukan penyesuaian format agar seluruh data yang terkumpul memiliki struktur yang seragam, memudahkan proses analisis data.
4. Tahap terakhir dalam *preprocessing* data adalah mengurangi jumlah data, yang dikenal sebagai *data reduction*. Tujuan utamanya adalah mengurangi sampel data tanpa mengubah hasil analisis. Terdapat tiga teknik yang dapat diterapkan pada tahap ini, yaitu pengurangan dimensi (*dimensionality reduction*), pengurangan jumlah (*numerosity reduction*), dan kompresi data (Binus Student Corner, 2022).

2.15. Unified Modelling Language (UML)

Unified Modelling Language (UML) adalah bahasa pemodelan visual yang digunakan untuk menentukan, memvisualisasikan, membangun berbagai aspek dari sebuah sistem perangkat lunak. UML sendiri memiliki fungsi sebagai alat untuk menangkap pemahaman mengenai sistem yang perlu dibangun.

Bayangkan sistem sebagai kelompok objek yang berbeda yang bekerja sama untuk melakukan pekerjaan yang berguna untuk pengguna. Bagian struktur statis mendefinisikan jenis objek yang penting untuk sistem dan bagaimana mereka terkait satu sama lain. Bagian perilaku dinamis menggambarkan bagaimana objek berubah seiring waktu dan berkomunikasi satu sama lain untuk mencapai tujuan tertentu.

Dengan memodelkan sistem dari berbagai sudut pandang yang terkait, kita dapat memahami sistem tersebut untuk berbagai keperluan (Rumbaugh et al., 2021).

Alat bantu yang digunakan dalam perancangan sistem menggunakan Unified Modelling Language (UML) adalah sebagai berikut :


1. *Use Case Diagram*

Use Case Diagram menunjukkan bagaimana perilaku suatu sistem ketika digunakan oleh orang lain, *Use Case Diagram* memecah fungsionalitas sistem menjadi tindakan yang bermakna bagi pengguna (*Actor*) yang menggunakan sistem. Pengguna (*Actor*) mencakup manusia, serta sistem dan proses komputer lainnya).

Sedangkan *Actor* adalah representasi ideal dari orang, proses, atau objek eksternal yang berinteraksi dengan suatu sistem, subsistem, atau kelas. Untuk identifikasi *Actor*, harus ditentukan tugas tugas yang berkaitan dengan peran pada konteks sistem. Orang atau sistem bisa muncul dalam beberapa peran, perlu diketahui jika *Actor* berinteraksi dengan *use case*.

Selain *Actor*, terdapat simbol simbol lain yang bisa digunakan didalam *Use Case* untuk memodelkan fungsi apa saja dari sistem yang dibangun beserta hubungan antar fungsinya. Adapun simbol-simbol dalam *Use Case* antara lain sebagai berikut :

Tabel 2. 1 Simbol Dalam *Use Case*

Gambar	Function	Keterangan
	Asosiasi	digambarkan dengan garis yang menunjukkan siapa atau apa


		yang terlibat dalam interaksi secara langsung.
«extend» ----->	Extend	Perluasan dari <i>use case</i> lain ketika beberapa kondisi dalam <i>use case</i> terpenuhi
«include» ----->	Include	Merupakan penambahan perilaku tambahan dalam suatu <i>use case</i>
→	Use case generalization	Hubungan antara suatu use case umum dengan use case yang lebih spesifik.



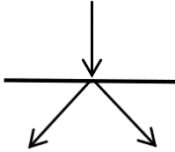
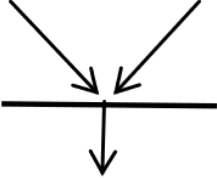
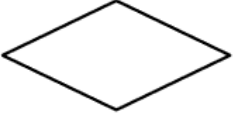
(Sumber : Rumbaugh et al., 2021).

2. Diagram Aktivitas (*Activity Diagram*)

Diagram aktivitas (*Activity Diagram*) menggambarkan aliran kerja dari sebuah sistem. Diagram aktivitas dapat mencakup cabang dan bercabangnya control dalam sebuah sistem yang berjalan secara bersamaan. Cabang cabang ini mewakili aktivitas yang dapat dilakukan secara bersamaan (Rumbaugh et al., 2021). Simbol terkait adalah sebagai berikut :

Tabel 2. 2 Simbol *Activity Diagram*

Gambar	Function	Keterangan
	<i>Start Point</i>	Ditempatkan pada posisi sudut kiri atas yang menandakan awal mula dari aktifitas

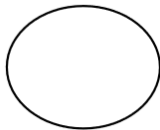
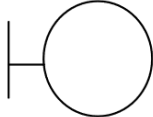
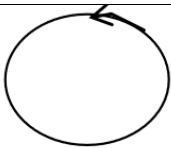

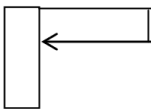
	<i>End Point</i>	Penanda akhir dari suatu aktifitas
	<i>Activities</i>	Proses yang terjadi dalam aplikasi
	<i>Fork</i>	Menggambarkan kegiatan yang dilakukan secara bersamaan dalam aplikasi
	<i>Join</i>	Menandakan pemecahan aplikasi dalam bagian bagian yang lebih kecil
	<i>Decisions Point</i>	Menandakan pilihan untuk mengambil keputusan jika dan maka

(Sumber : Rumbaugh et al., 2021).

3. *Sequence Diagram*

Sequence Diagram menjelaskan perilaku objek objek yang ada didalam *use case* dengan menjelaskan siklus hidup dari objek serta pesan pesan atau data yang dikirim oleh objek tersebut. Simbol terkait dalam *sequence diagram* meliputi :

Tabel 2. 3 Tabel Simbol *Sequence Diagram*

Simbol	Function	Makna Simbol
	<i>Entity Class</i>	Bagian dari sistem yang membentuk gambaran awal dari sistem
	<i>Boundary Class</i>	Berisi kumpulan kelas yang berinteraksi dengan satu atau lebih <i>actor</i> dalam sistem
	<i>Control Class</i>	Suatu entitas yang menampung logika pada aplikasi yang memiliki kewajiban langsung terhadap entitas lain
	<i>Message</i>	Simbol mengirim pesan antar <i>class</i>
	<i>Recursive</i>	Menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri

(Sumber : Rumbaugh et al., 2021).

4. *Class Diagram* (Diagram Kelas)

Class diagram adalah salah satu jenis diagram struktur dalam UML yang dengan jelas menggambarkan struktur serta deskripsi kelas, atribut, metode, dan hubungan antara setiap objek. Class diagram bersifat statis, artinya diagram ini

tidak menunjukkan apa yang terjadi saat kelas-kelas tersebut berinteraksi, melainkan menjelaskan jenis hubungan yang ada. *Class Diagram* disusun dalam komponen-komponen yang dapat dilihat pada gambar dibawah ini.



Gambar 2. 1 Komponen dalam Class Diagram

(Sumber : R. Setiawan, 2021)

Berikut merupakan penjelasan komponen-komponen diatas :

1. Komponen Atas

Komponen ini berisi nama dari kelas dalam diagram, perlu diketahui bahwa setiap kelas memiliki nama yang unik dan tidak sama.

2. Komponen Tengah

Komponen ini memuat atribut-atribut dari sebuah kelas, yang menjelaskan karakter serta kualitas dan isi dari suatu kelas yang juga menyertakan tipe nilai

3. Komponen Bawah

Komponen ini mencakup operasi dalam kelas yang ditampilkan dalam bentuk *list* daftar. Daftar operasi terkait menggambarkan bagaimana interaksi kelas terhadap data yang dimasukkan.

BAB III

METODE PENELITIAN

Metode penelitian diartikan sebagai cara ilmiah untuk mendapatkan data dengan tujuan dan kegunaan tertentu (Ramdhan, 2021). Penelitian ini akan melalui beberapa tahapan.

3.1. Pengumpulan Data

Data yang diperlukan dalam penelitian ini adalah data-data kerusakan HP yang tersedia di tempat penelitian, beberapa Teknik yang penulis gunakan dalam pengumpulan data adalah sebagai berikut :

1. Observasi / Pengamatan Langsung

Penulis melakukan pengamatan langsung ke Pioneer Service untuk mendapatkan data-data yang diperlukan, yang mana data yang paling diperlukan dalam penelitian ini adalah data perbaikan *handphone* dalam pioneer service dalam 1 tahun terakhir.

2. Wawancara / Tanya Jawab

Penulis melakukan wawancara dan diskusi tanya jawab kepada pihak teknisi di Pioneer Service untuk mengetahui jenis-jenis kerusakan yang biasanya terjadi pada *Handphone*, apa yang menyebabkan kerusakan tersebut terjadi, dan rekomendasi solusi perbaikan apabila kerusakan terjadi.

3. Sampel / Contoh

Penulis mengambil sampel data kerusakan Hp yang berguna bagi penelitian ini. Sampel yang diambil kemudian dianalisa menggunakan perhitungan matematis manual, untuk melihat bagaimana data dapat bekerja menggunakan algoritma yang digunakan.

4. Studi Literatur

Sebagai bahan pendukung penelitian, untuk bahan referensi dan landasan teori dalam penelitian, penulis mencari studi literatur mengenai tinjauan pustaka yang berkaitan dengan penelitian penulis.

3.2. Pra-Pemrosesan Data

Data yang tersedia oleh *Pioneer Service* biasanya merupakan data mentah yang masih harus diperiksa, dibersihkan dan transformasi terlebih dahulu sebelum data di proses dengan Algoritma C.45, Adapun tahapan-tahapan pra-pemrosesan data yang dilalui adalah

1. Verifikasi Data

Setelah data dikumpulkan dari *Pioneer Service*, Langkah pertama yang dilakukan adalah verifikasi data. Tahap ini akan mencakup identifikasi dari keseluruhan data dan penanganan yang terjadi apabila ada nilai yang hilang, nilai duplikat, atau tidak valid. Beberapa data yang tidak lengkap atau tidak konsisten perlu diperbaiki atau dihapus.

2. Transformasi Data

Data yang telah dibersihkan selanjutnya perlu ditransformasi kedalam format yang sesuai agar bisa dianalisa menggunakan algoritma C.45, hal

ini melibatkan konversi format data, normalisasi data atau standarisasi nilai-nilai dalam dataset.

3. Deteksi dan Penanganan *outlier*

Outlier atau pencilan dapat mempengaruhi hasil analisis. *Outlier* adalah titik data yang sangat tinggi atau sangat rendah dibandingkan dengan titik data terdekat dalam kumpulan data yang dikerjakan. *Outlier* merupakan nilai ekstrem yang sangat menonjol dari keseluruhan pola nilai dalam Kumpulan data (Dionysia, 2021). Oleh karena itu, perlu dilakukan deteksi *outlier* untuk penanganannya. Beberapa *library* dalam python seperti *scikit-learn* dan *pandas* serta visualisasi menggunakan *matplotlib* dapat digunakan untuk membantu identifikasi *outlier*, dan kemudian *outlier* tersebut dapat dihapus atau ditangani sesuai kebutuhan.

4. Pemilihan Fitur (*Feature Selection*)

Dalam beberapa kasus analisa data, tidak semua fitur yang ada dalam dataset diperlukan untuk analisis. Oleh karena itu, pemilihan fitur dilakukan untuk memilih fitur-fitur yang paling relevan atau berpengaruh terhadap variabel target yang ingin diprediksi.

Beberapa proses diatas tentunya akan sangat panjang jika dilakukan secara manual. Bahasa pemrograman *python* menyediakan *library pandas* yang dapat membantu pemrosesan data menjadi lebih cepat, berikut beberapa langkah pemrosesan data yang tersedia dalam *library pandas* :

1. Mengimpor *Library Pandas*

Langkah ini pertama kali dilakukan agar dapat menggunakan fitur-fitur yang disediakan *pandas* untuk pemrosesan data, adapun potongan kode yang

digunakan dalam mengimpor *Library Pandas* dalam lingkungan pengembangan *Python* dapat dilihat pada gambar dibawah ini :

```
import pandas as pd
```

Gambar 3. 1 Kode Import Library Pandas

Potongan kode di atas adalah contoh penggunaan pernyataan import untuk mengimpor library *Pandas* dengan alias *pd*. Alias digunakan untuk mempermudah pemanggilan saat *pandas* digunakan dalam pemrosesan data.

2. Memasukkan data dalam format csv

Pandas memberikan kemudahan untuk memasukkan data kedalam lingkungan pengembangan *python*, adapun memasukkan data ke dalam lingkungan pengembangan model menggunakan bahasa *python* dan *library pandas* dapat dilihat dalam potongan kode dibawah ini :

```
# Memuat file CSV ke dalam DataFrame  
data = pd.read_csv('nama_file.csv')
```

Setelah kode ini dieksekusi, data dari file CSV akan dimuat ke dalam objek *DataFrame* yang disebut *data*, yang dapat Anda gunakan untuk melakukan berbagai operasi pemrosesan data menggunakan *Pandas*.

3. Verifikasi data

Pemeriksaan apakah ada data yang hilang atau data tidak valid dapat dipermudah dengan menggunakan *library pandas*. Pengecekan nilai yang hilang dalam suatu data dapat dilakukan dengan potongan kode yang terdapat di gambar 3.3 dibawah ini. Kode tersebut juga secara otomatis menghapus nilai yang hilang tersebut.

```
# Misalnya, identifikasi nilai yang hilang
data.dropna(inplace=True)
```

Gambar 3. 2 Kode identifikasi dan menghapus nilai data yang hilang

3.3. Implementasi Algoritma C.45

Dataset yang sudah dipersiapkan kemudian dianalisis menggunakan algoritma C.45, Secara umum langkah langkah yang dilakukan dalam membangun sebuah pohon keputusan dengan algoritma C.45 adalah sebagai berikut :

1. Memilih atribut sebagai akar
2. Buat cabang untuk setiap nilai
3. Membagi kasus dari tiap cabang
4. Ulangi langkah yang sama untuk setiap cabang.

Adapun rumus metode C.45 adalah sebagai berikut :

$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{i=1}^n \frac{|S_i|}{|S|} * \text{Entropy}(S_i) \quad (3.1)$$

Dimana :

S : Himpunan kasus

A : Atribut

n : jumlah partisi atribut A

|S_i| : Jumlah kasus pada partisi ke-i

|S| : Jumlah kasus dalam S

Untuk perhitungan *entropy* sendiri, dapat dilihat pada persamaan dibawah ini

$$\text{Entropy}(S) = \sum_{i=1}^n - p_i * \log_2 p_i \dots \dots \dots (3.2)$$

Dimana :

S : Kumpulan Kasus

N : Jumlah partisi S

P_i : proporsi dari S_i terhadap S

Adapun dalam analisis dataset menggunakan Algoritma C.45, penulis menggunakan beberapa *library* dalam python untuk analisis data dan membangun model pohon Keputusan (*Decision Tree*) menggunakan algoritma C.45. Berikut adalah tahapan implementasinya :

1. Memisahkan data latih dan data uji

Dataset yang telah disiapkan perlu dibagi menjadi data latih (*training data*) dan data uji (*testing data*). Data latih digunakan untuk melatih algoritma C.45 dan data uji digunakan untuk menguji seberapa baik model yang dilatih oleh Algoritma C.45 dapat melakukan prediksi terhadap data baru yang belum pernah dilihat sebelumnya.

Langkah ini merupakan salah satu Langkah penting dalam evaluasi model, dalam pengerjaan pembagian data, penulis menggunakan *library train_test_split* yang tersedia di *scikit-learn*. *Train_test_split* adalah salah satu metode yang dapat digunakan untuk evaluasi performa model klasifikasi. Metode dalam *scikit-learn* ini membagi data menjadi dua bagian, yaitu data yang dilatih dan data yang digunakan untuk menjadi data uji (Afifah, 2023).

2. Membangun model algoritma C.45

Untuk membangun dan menguji model Algoritma C.45 menggunakan *python*, kita perlu implementasikan rumus perhitungan *entropy* untuk setiap himpunan kasus (S) dan setiap partisi dari himpunan kasus. Setelah itu, kita implementasi perhitungan gain menggunakan nilai *entropy* yang didapatkan sebelumnya,

perhitungan matematika ini dapat lebih mudah dilakukan menggunakan *library python* yang bernama *numpy*.

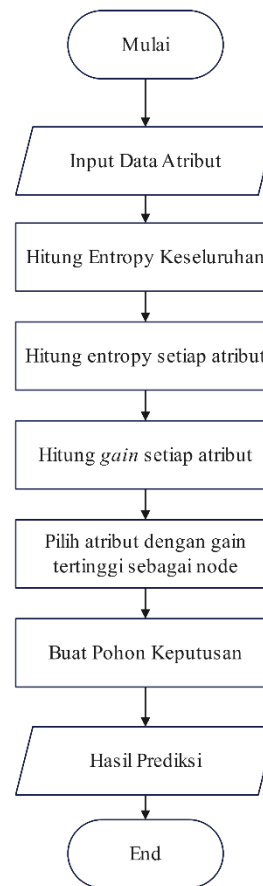
Algoritma yang dibangun kemudian memilih atribut dengan *gain* tertinggi, dan kemudian mulai membangun pohon Keputusan, Langkah perhitungan *entropy* dan *gain* akan terus diulangi hingga kriteria-kriteria dalam membangun pohon keputusan tercapai, seperti mencapai kedalaman maksimum atau tidak ada atribut lagi yang dapat digunakan untuk membagi himpunan data.

3.4. Evaluasi Model

Model prediksi algoritma yang dibuat kemudian diuji menggunakan data uji yang dipisah dari proses pemisahan data sebelumnya. Hasil prediksi model akan dievaluasi dengan metrik-metrik tertentu, seperti tingkat akurasi, tingkat presisi (*Precision*), *recall*, atau uji *F1 Score* untuk menilai kinerja model prediksi algoritma C.45 terhadap data-data baru yang belum pernah dilihat algoritma sebelumnya. Ketika sudah menemukan model algoritma C.45 yang dapat memprediksi dengan baik, model yang ada akan disimpan untuk kemudian digunakan dalam sistem pendukung Keputusan yang dibangun.

3.5. Bagan Alir (*Flowchart*)

Berikut adalah bagan alir dalam memprediksi kerusakan pada *handphone* :



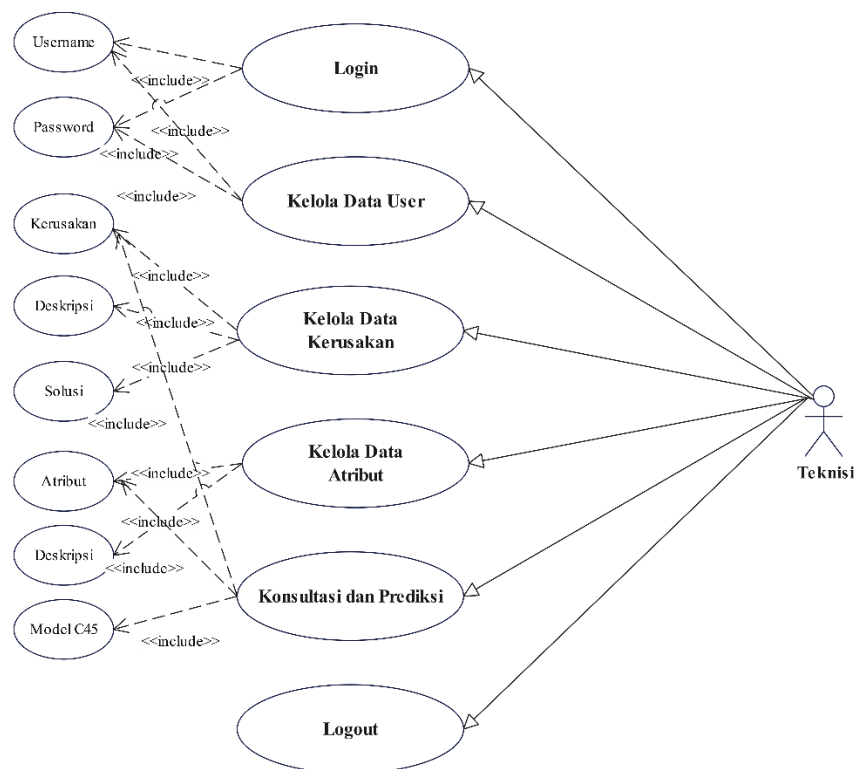
Gambar 3. 3 Flowchart prediksi kerusakan HP

3.6. Desain Sistem

Dalam proses ini, sistem akan didesain sedemikian rupa sesuai dengan kebutuhan-kebutuhan fitur yang harus tersedia dalam Sistem Pendukung Keputusan yang dibuat. Proses ini berfokus pada beberapa perancangan seperti struktur data dan arsitektur perangkat lunak yang dibuat dengan pemodelan UML seperti *use case diagram*, *class diagram*, *activity diagram*, dan *diagram* serta rancangan tampilan sistem yang akan dibuat dan rancangan basis data beserta masing masing relasi antar tabelnya.

3.6.1. Use Case Diagram

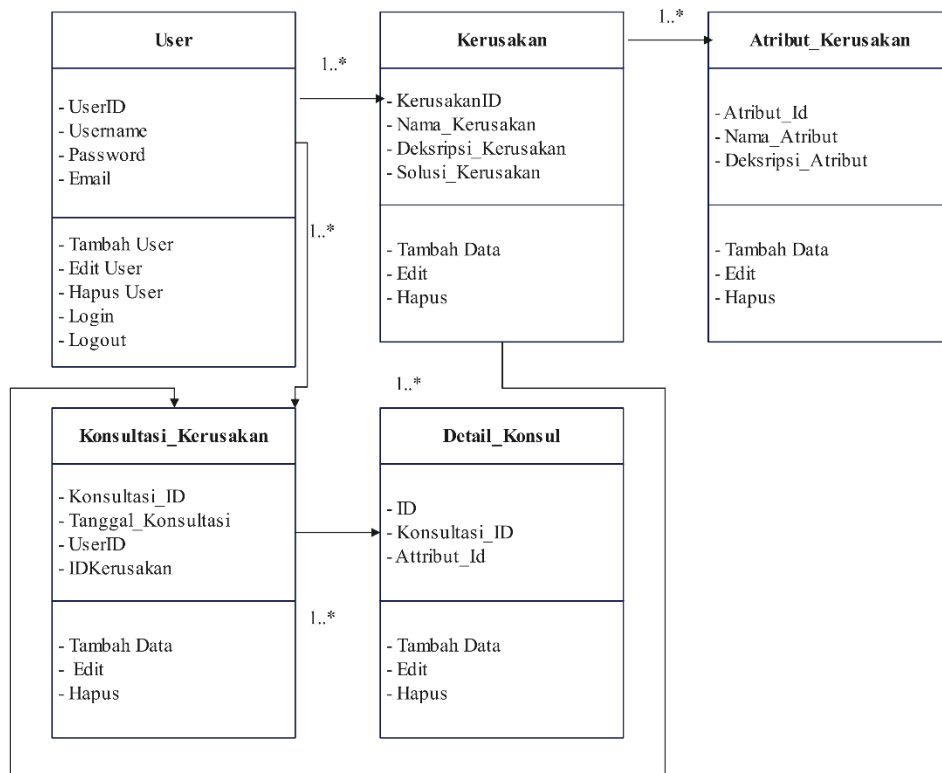
Proses sistem yang direncanakan untuk Sistem Pendukung Keputusan dalam memprediksi kerusakan hp pada layanan Pioneer Service melibatkan satu peran aktor, yaitu Teknisi, yang memiliki akses ke berbagai fitur. Fitur-fitur tersebut mencakup login, akses Dashboard Admin, manajemen data pengguna (Admin), penginputan data Teknisi, manajemen data kerusakan, penginputan dan pengeditan data kerusakan, manajemen data atribut, penginputan dan pengeditan data atribut, pelatihan model C.45, manajemen data konsultasi kerusakan, penginputan data hasil kerusakan, tampilan hasil, pembuatan laporan, dan logout. Tampilan *Use Case Diagram* dapat terlihat dalam gambar 3.7 dibawah ini



Gambar 4. 1 Use Case Diagram Decision Support System (DSS) Memprediksi Kerusakan HP di Pioneer Service Menggunakan Algoritma C.45 Berbasis Web

3.6.2. Class Diagram

Class Diagram adalah sebuah spesifikasi dalam pengembangan sistem yang ketika dipanggil akan mengelompokkan objek-objek yang merupakan inti dari pengembangan desain sistem serta hubungan relasi antar komponen didalamnya. Berikut *Class Diagram* dari sistem yang penulis rancang yang dapat dilihat pada gambar 3.7 dibawah ini



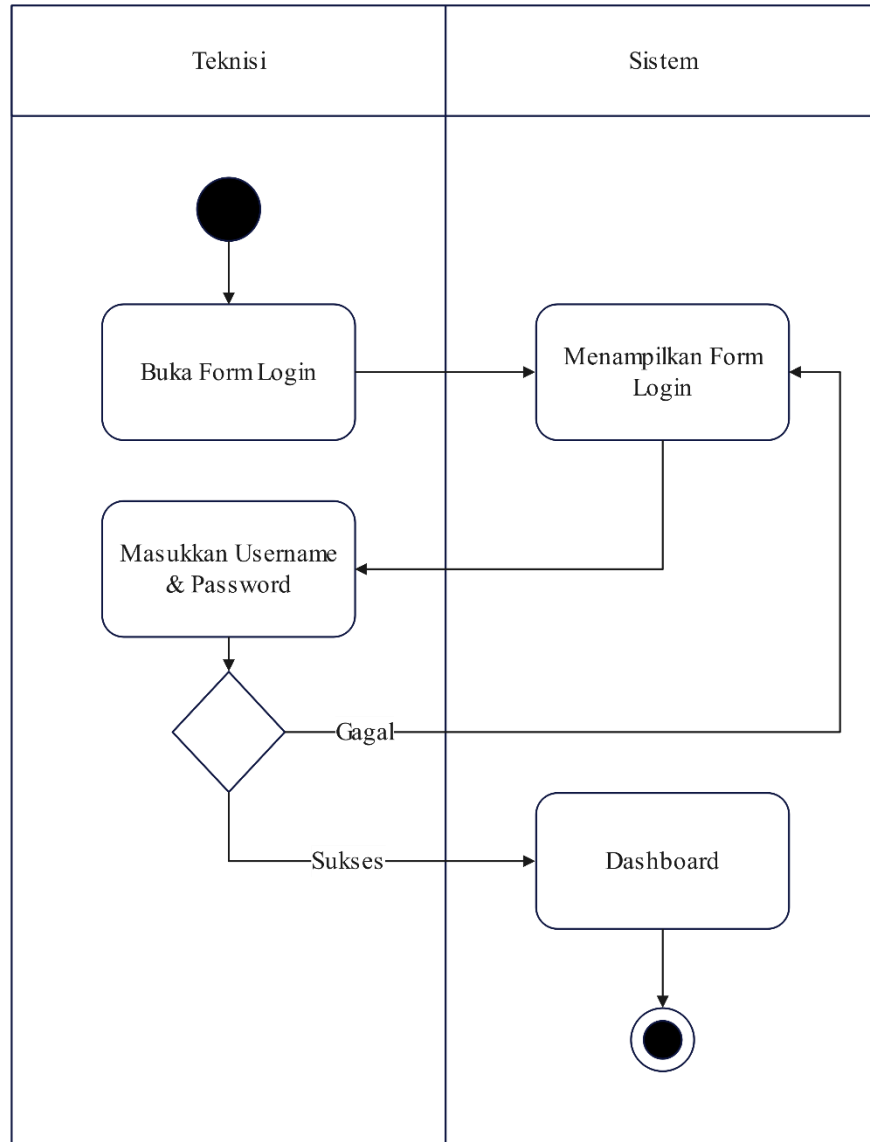
Gambar 3. 4 Class Diagram *Decision Support System (DSS)* Memprediksi Kerusakan HP di Pioneer Service Menggunakan Algoritma C.45 Berbasis Web

3.6.3. Activity Diagram

Rangkaian kegiatan pada setiap terjadi suatu aktivitas dalam sistem digambarkan pada beberapa *Activity Diagram* dibawah ini :

1. Activity Diagram Login

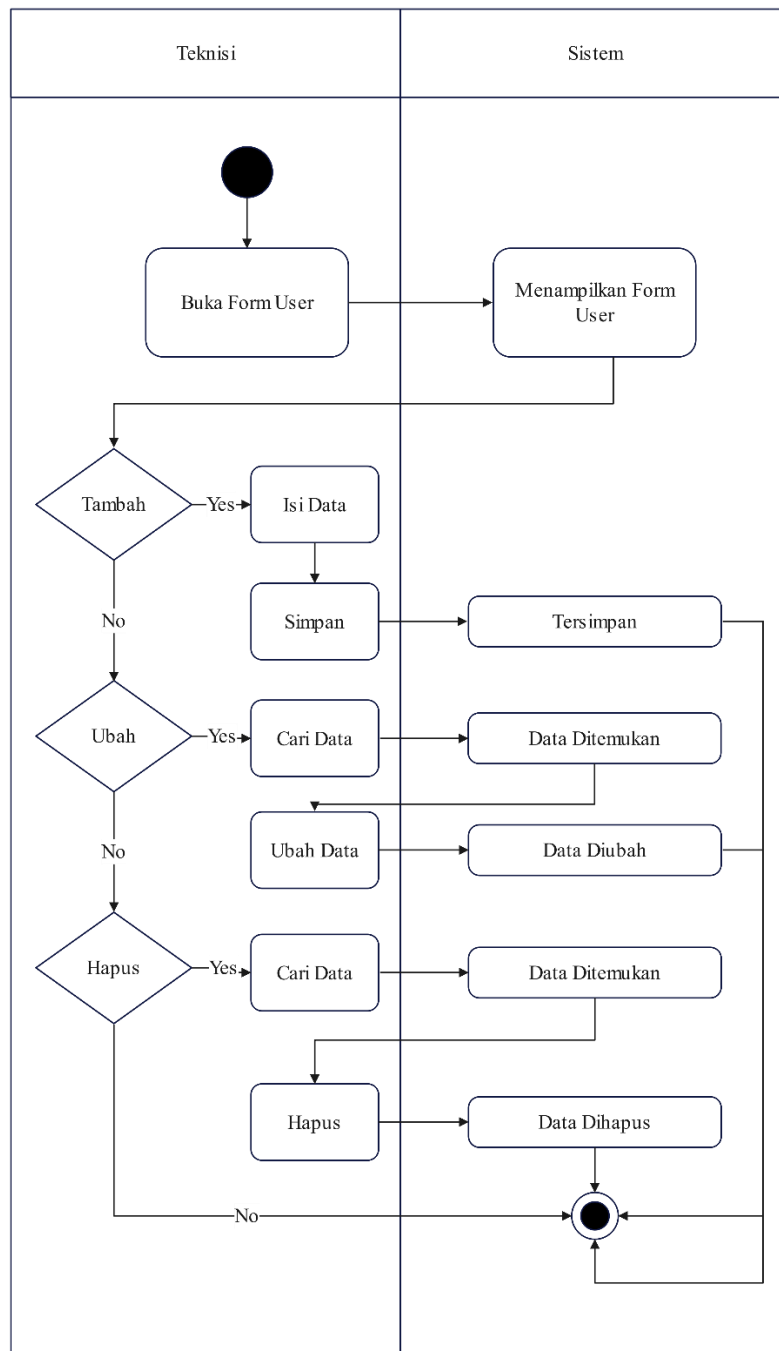
Aktivitas yang dilakukan saat pengguna teknisi melakukan login ke sistem dapat dilihat seperti gambar 3.8 dibawah ini :



Gambar 3. 5 Activity Diagram Login

2. Activity Diagram Kelola Data User

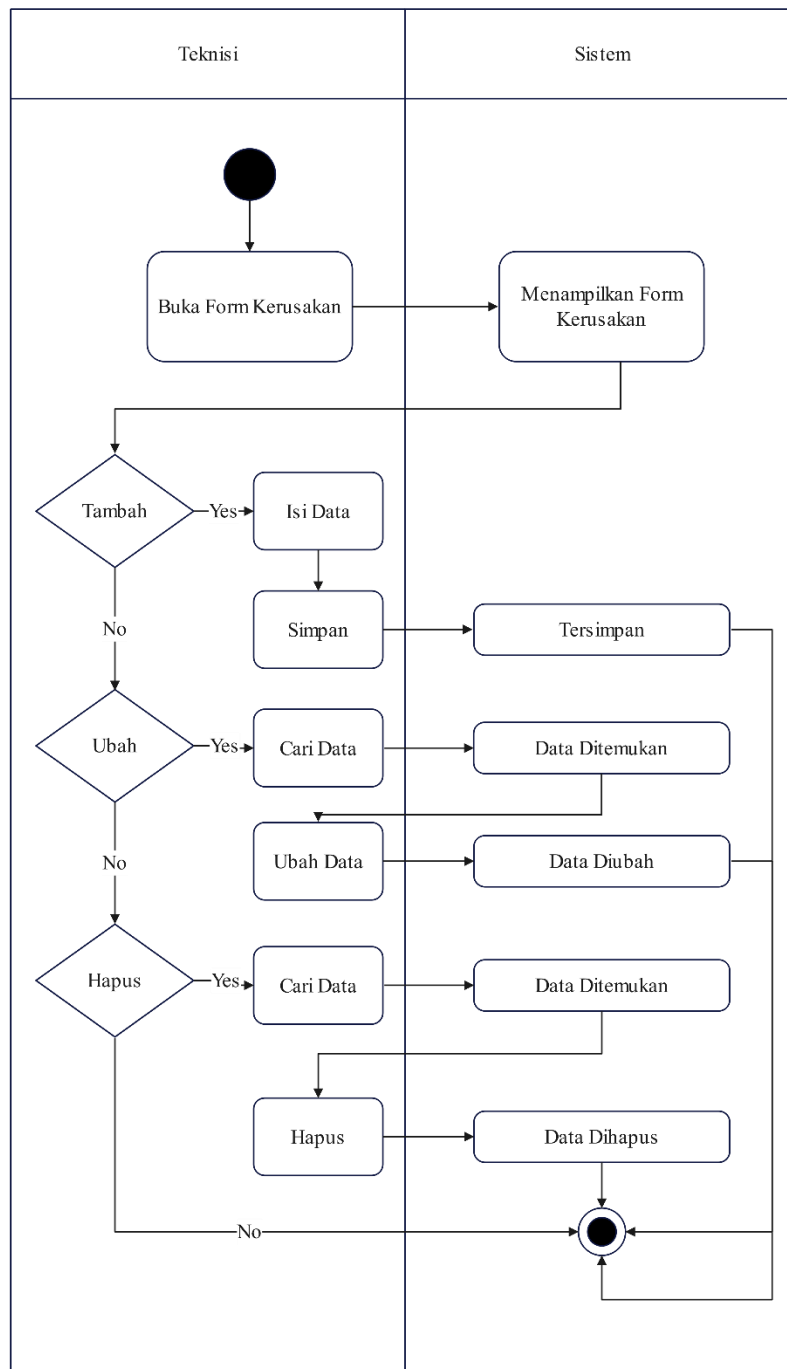
Aktivitas yang dilakukan saat pengguna teknisi mengelola data *user* dapat dilihat seperti gambar 3.9 dibawah ini :



Gambar 3. 6 Activity Diagram Kelola Data User

3. Activity Diagram Kelola Data Kerusakan

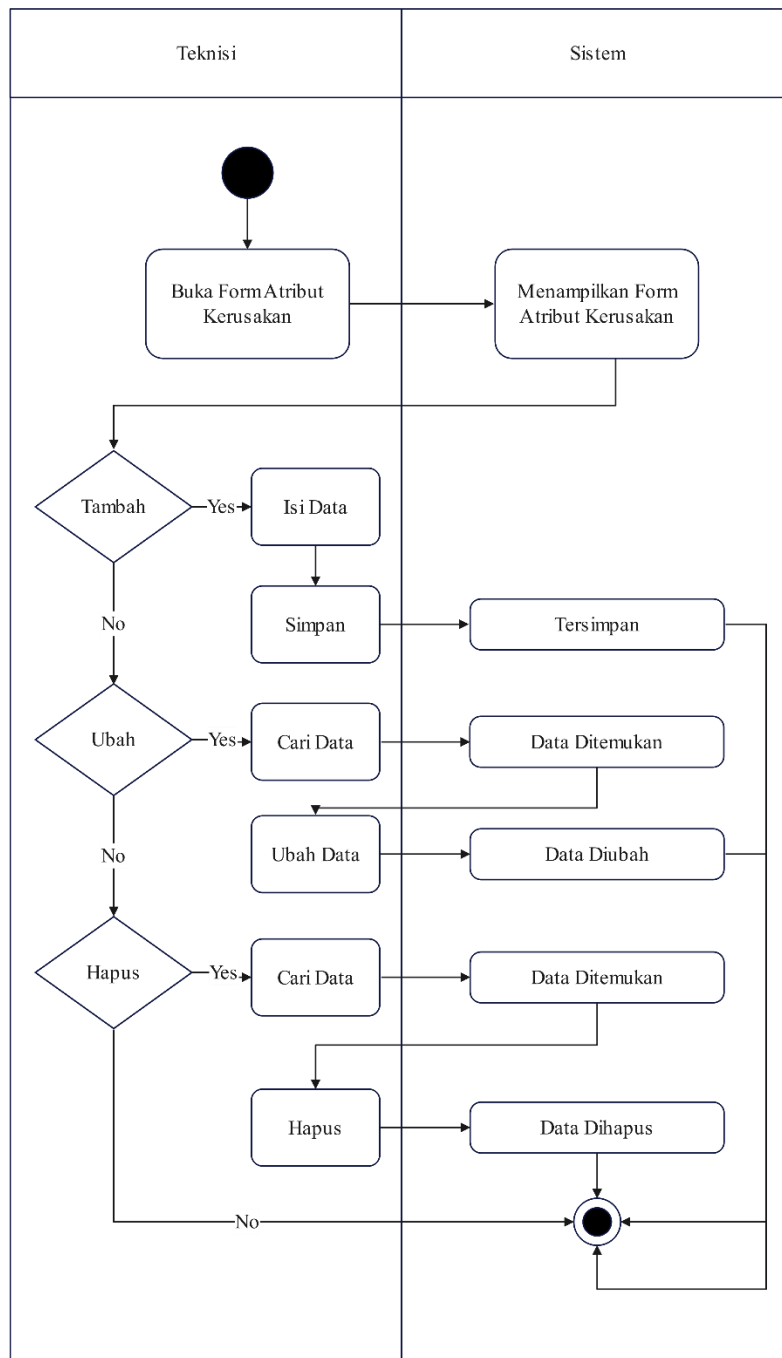
Aktivitas yang dilakukan saat pengguna teknisi mengelola data kerusakan dapat dilihat seperti gambar 3.10 dibawah ini :



Gambar 3.7 Activity Diagram Kelola Data Kerusakan

4. Activity Diagram Kelola Data Atribut Kerusakan

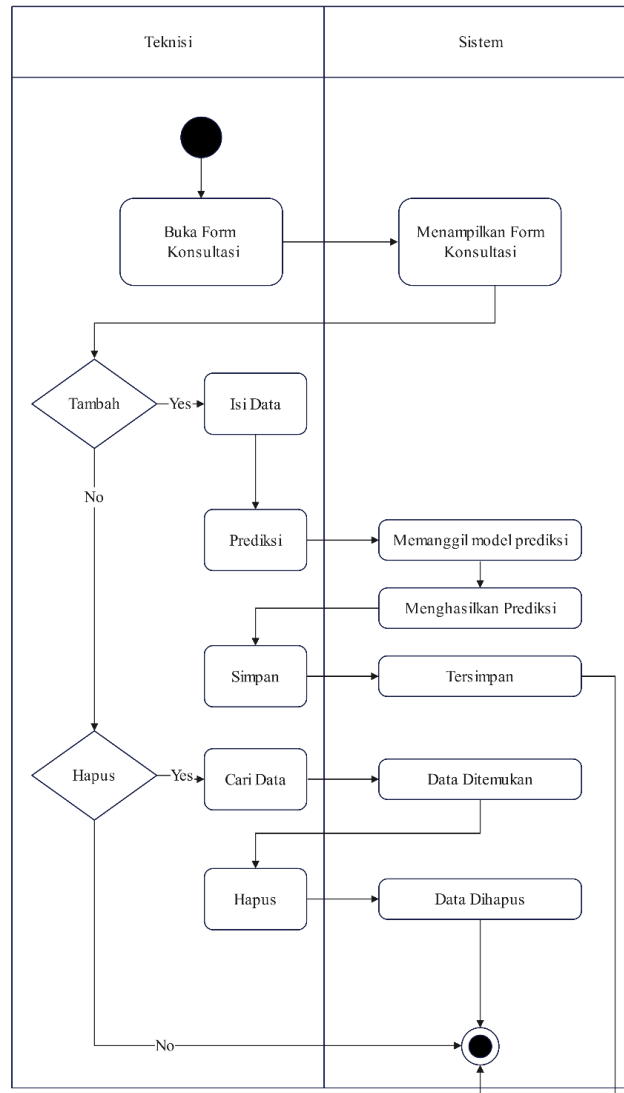
Aktivitas yang dilakukan saat pengguna teknisi mengelola atribut dari masing-masing kerusakan dapat dilihat seperti gambar 3.11 dibawah ini :



Gambar 3. 8 Activity Diagram Kelola Data Atribut

5. Activity Diagram Kelola Data Konsultasi Kerusakan

Aktivitas yang dilakukan saat pengguna teknisi mengelola data konsultasi kerusakan dapat dilihat seperti gambar 3.14 dibawah ini :



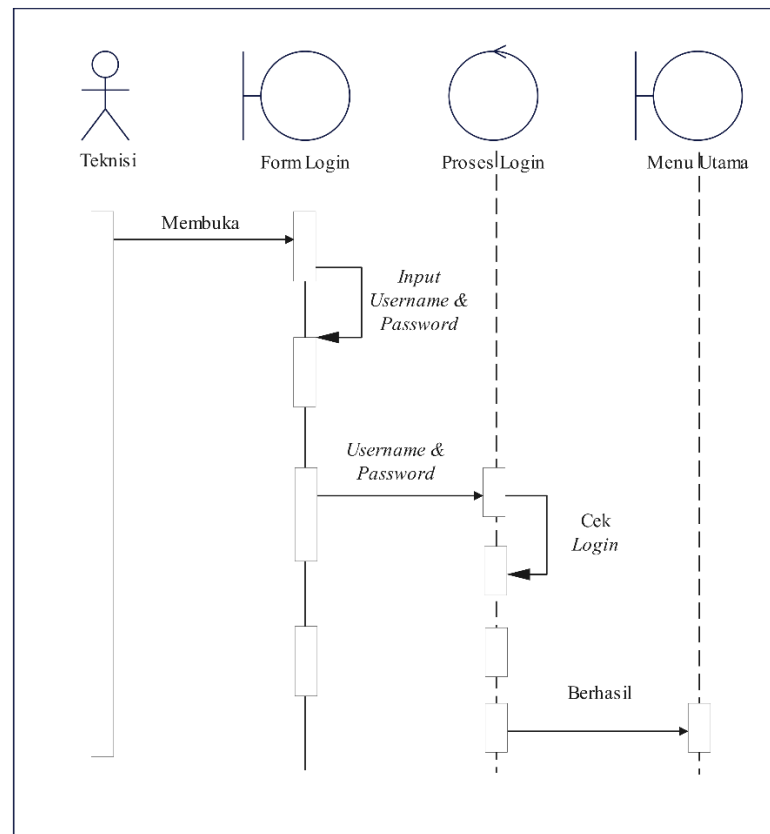
Gambar 3. 9 Activity Diagram Konsultasi Kerusakan

3.6.4. Sequence Diagram

Rangkaian kegiatan pada setiap terjadinya *event* dalam sistem digambarkan pada *sequence diagram* berikut :

1. Sequence Diagram Login

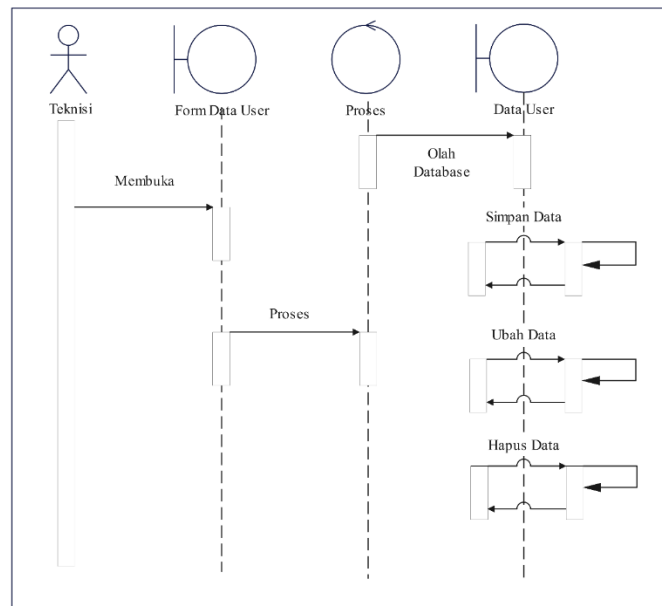
Dalam diagram ini, Teknisi memasukkan kredensial untuk *login*. Jika *Username* dan *Password* benar, Teknisi akan mendapatkan akses ke menu utama sistem. *Sequence Diagram* proses login dapat dilihat dalam gambar 3.16 dibawah ini :



Gambar 3. 10 Sequence Diagram Login

2. Sequence Diagram Kelola Data User

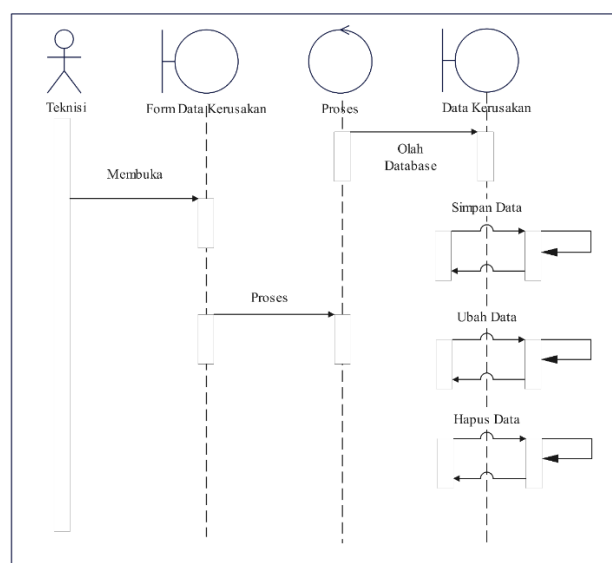
Dalam diagram ini, Teknisi menggunakan form untuk memproses data pengguna, Teknisi dapat memilih untuk menyimpan, mengubah atau menghapus data dalam basis data. *Sequence Diagram* proses kelola data *User* dapat dilihat dalam gambar 3.17 dibawah ini :



Gambar 3. 11 Sequence Diagram Data User

3. *Sequence Diagram* Kelola Data Kerusakan

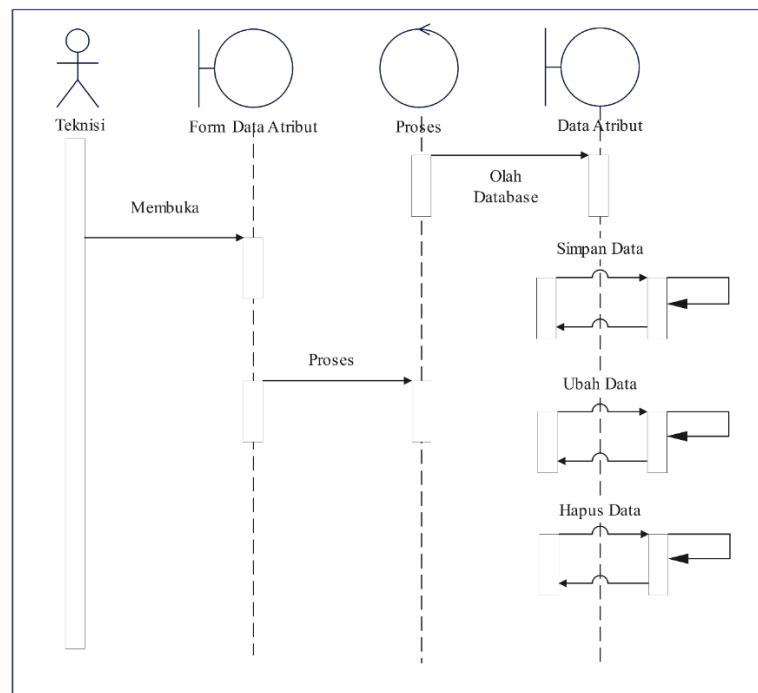
Dalam diagram ini, Teknisi mengelola data kerusakan dengan opsi untuk menambah, mengedit atau menghapus entri daftar kerusakan dari basis data. *Sequence Diagram* Kelola Data Kerusakan dapat dilihat dalam Gambar 3.18 dibawah ini



Gambar 3. 12 Sequence Diagram Data Kerusakan

4. *Sequence Diagram* Kelola Data Atribut Kerusakan

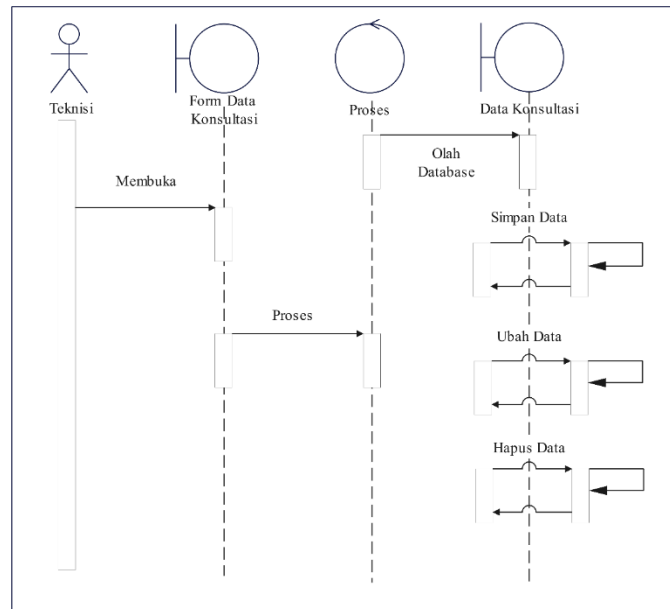
Dalam diagram ini, Teknisi mengelola data atribut kerusakan dengan opsi untuk menambah, mengedit atau menghapus daftar atribut kerusakan dari basis data. Tampilan *Sequence Diagram* Kelola Data Atribut Kerusakan terdapat dalam Gambar 3.19 dibawah ini



Gambar 3. 13 Sequence Diagram Kelola Atribut Kerusakan

5. *Sequence Diagram* Kelola Data Konsultasi

Dalam diagram ini, Teknisi mengelola data atribut kerusakan dengan opsi untuk menambah, mengedit atau menghapus daftar konsultasi dari basis data. *Sequence Diagram* Kelola Data Konsultasi dapat dilihat dalam Gambar 3.22 dibawah ini



Gambar 3. 14 Sequence Diagram Kelola Data Konsultasi

3.6.5. Desain *Database* dan Tabel

Struktur tabel dan basis data dalam sistem dibutuhkan agar sistem berjalan dengan baik dan data-data yang diproses dalam penggunaan sistem dapat disimpan dengan baik juga. Berikut merupakan rancangan basis data dan tabel dari sistem pendukung keputusan yang dibuat :

1. Struktur Tabel *User*

Tabel User berfungsi untuk menyimpan data pengguna yang menggunakan sistem, termasuk Username dan Password yang digunakan untuk melakukan proses login ke sistem. Struktur tabel ini dapat dilihat pada tabel dibawah :

Nama Basis Data : dss_c45

Nama Tabel : tb_user

Tabel 3. 1 Struktur Tabel User

Nama Field	Tipe Data	Ukuran	Index
user_id	INT	5	Primary Key, Auto Increment
Username	VARCHAR	10	-
Password	VARCHAR	10	-
Email	TEXT	50	-

2. Struktur Tabel Kerusakan

Tabel Kerusakan digunakan untuk menyimpan data data kerusakan yang terjadi di *Handpohone* mulai dari nama, deskripsi, hingga solusi dari kerusakan tersebut.

Selengkapnya struktur tabel dapat dilihat dibawah ini :

Nama Basis Data : dss_c45

Nama Tabel : tb_kerusakan

Tabel 3. 2 Struktur Tabel Kerusakan

Nama Field	Tipe Data	Ukuran	Index
KerusakanID	INT	5	Primary Key, Auto Increment
Nama_kerusakan	TEXT	-	-
Deskripsi_kerusakan	TEXT	-	-
Solusi_kerusakan	TEXT	-	-

3. Struktur Tabel Atribut Kerusakan

Tabel Atribut Kerusakan digunakan untuk menyimpan data-data atribut atau gejala yang terjadi di *handphone* yang berkaitan dengan kerusakan di tabel sebelumnya, selengkapnya dapat dilihat di tabel dibawah ini :

Nama Basis Data : dss_c45

Nama Tabel : tb_atribut

Tabel 3. 3 Struktur Tabel Atribut Kerusakan

Nama Field	Tipe Data	Ukuran	Index
id_atribut	INT	5	Primary Key, Auto Increment
nama_atribut	TEXT	-	-
deskripsi_atribut	TEXT	-	-

4. Struktur Tabel Konsultasi Kerusakan

Tabel Konsultasi Kerusakan digunakan untuk menyimpan data utama Konsultasi setiap perangkat yang diperiksa dan diprediksi kerusakannya menggunakan algoritma C.45, tabel ini akan berelasi dengan tabel lain untuk melengkapi informasinya. Struktur tabel dapat terlihat dibawah ini :

Nama Basis Data : dss_c45

Nama Tabel : tb_konsultasi

Tabel 3. 4 Struktur Tabel Konsultasi Kerusakan

Nama Field	Tipe Data	Ukuran	Index
id_konsultasi	INT	5	Primary Key, Auto Increment
tanggal	DATETIME	-	-
user_id	INT	-	<i>Foreign Key</i> dari tabel User
kerusakan_id	Int	-	<i>Foreign Key</i> dari tabel kerusakan

5. Struktur Tabel Detail Konsultasi

Tabel Detail Konsultasi berisi daftar pemilihan atribut-atribut kondisi kerusakan yang dialami oleh *Handphone* tabel ini berelasi langsung dengan tabel konsultasi kerusakan sebagai *detail* bagi apa saja kondisi kerusakan yang dialami oleh *Handphone*, struktur tabel ini dapat terlihat dibawah :

Nama Basis Data : dss_c45

Nama Tabel : tb_detailrusak

Tabel 3. 5 Struktur Tabel *Detail* Konsultasi


Nama Field	Tipe Data	Ukuran	Index
id_detail	INT	5	Primary Key, Auto Increment
id_konsultasi	INT	-	<i>Foreign Key</i> dari Tabel Konsultasi
id_atribut	INT	-	

3.7. Desain Rancangan Program

Tahapan ini merupakan tahapan untuk merancang tampilan antar muka pengguna dari aplikasi yang akan dibangun. Tujuannya adalah untuk menghasilkan pedoman mengenai bagaimana tampilan dan interaksi antarmuka pengguna yang akan dibuat. Rancangan terkait adalah sebagai berikut :

1. Rancangan tampilan menu *login*

Rancangan tampilan menu *login* dapat dilihat pada gambar dibawah ini :

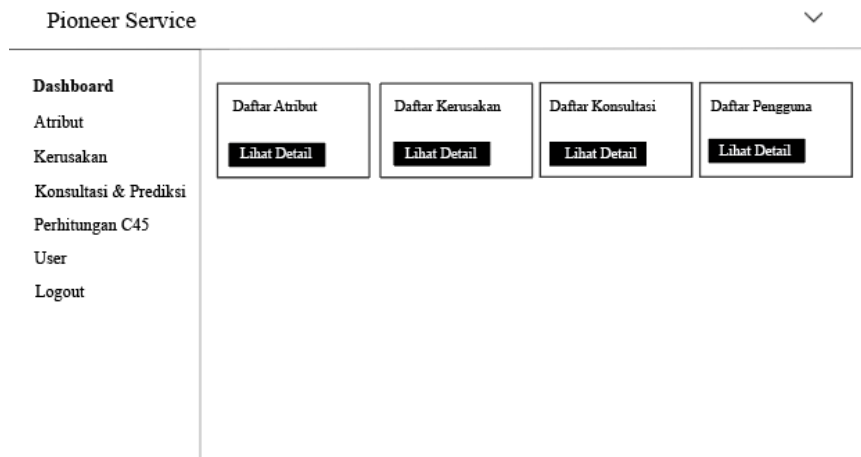


The image shows a login form titled "Selamat Datang, Teknisi". It contains two input fields: "Username" and "Password". Below the password field is a black button with the text "Login" in white. The form is enclosed in a rectangular border.

Gambar 3. 15 Rancangan Form Login

2. Rancangan tampilan menu *dashboard*

Rancangan dashboard merupakan halaman utama atau pusat kontrol dari aplikasi, didalamnya terdapat ringkasan informasi atau navigasi cepat ke berbagai fitur dalam aplikasi web, berikut merupakan rancangannya :



Gambar 3. 16 Rancangan menu dashboard

3. Rancangan tampilan menu tambah atribut

Rancangan tampilan tambah atribut dapat dilihat pada gambar dibawah ini :

Gambar 3. 17 Rancangan menu tambah atribut

4. Rancangan *Form* Data Atribut

Data atribut berisi gejala gejala yang dialami oleh perangkat, rancangan *form* tersebut dapat dilihat dalam gambar dibawah ini :

Pioneer Service

Dashboard

Atribut

Kerusakan

Konsultasi & Prediksi

Perhitungan C45

User

Logout

Data Atribut **Tambah Atribut**

Show Entries Search

No	Nama Atribut	Deskripsi	Action

Gambar 3. 18 Rancangan menu data atribut

Rancangan tampilan tambah atribut dapat dilihat pada gambar dibawah ini :

5. Rancangan tampilan menu tambah kerusakan

Rancangan menu tambah kerusakan dapat dilihat dalam gambar dibawah ini :

Pioneer Service

Dashboard

Atribut

Kerusakan

Konsultasi & Prediksi

Perhitungan C45

User

Logout

Tambah Kerusakan **Data Kerusakan**

Tambah Otomatis

Choose File No File Chosen

Upload & Simpan Data

Tambah Manual

Nama Kerusakan

Deskripsi Kerusakan

Solusi Kerusakan

Tambah Data

Gambar 3. 19 Rancangan menu tambah kerusakan

6. Rancangan tampilan menu data kerusakan

Rancangan menu data kerusakan dapat dilihat dalam gambar dibawah ini :

Pioneer Service ▼

Dashboard

Atribut

Kerusakan

Konsultasi & Prediksi

Perhitungan C45

User

Logout

Data Kerusakan **Tambah Kerusakan**

Show Entries Search

No	Nama Kerusakan	Deksripsi Kerusakan	Solusi Kerusakan	Action

Gambar 3. 20 Rancangan Menu Data Kerusakan

7. Rancangan tampilan menu data konsultasi dan prediksi

Dalam menu ini, pengguna akan menginput atribut gejala dari sebuah perangkat, sebelum data konsultasi disimpan, pengguna akan menyuruh sistem untuk melakukan prediksi menggunakan algoritma C45. Rancangan menu data konsultasi dan prediksi sebagai berikut

Pioneer Service ○

Dashboard

Atribut

Kerusakan

Konsultasi & Prediksi

Perhitungan C45

User

Logout

Input Data Konsultasi **Data Konsultasi**

Tanggal Nama Perangkat

Nama Pelanggan

Pilih atribut kerusakan

+

+

Lakukan Prediksi

Hasil Prediksi

Simpan Data

Gambar 3. 21 Rancangan menu konsultasi dan hasilkan prediksi

8. Rancangan Perhitungan C4.5

Form ini bermanfaat untuk melatih data dengan algoritma C4.5, form ini juga menghasilkan pohon keputusan dan performa akurasi dari algoritma tersebut, berikut merupakan rancangan *form* perhitungan C4.5

Pioneer Service	
Dashboard	Proses Data
Atribut	Proses Data
Kerusakan	Lakukan Perhitungan C45
Konsultasi & Prediksi	Proses Perhitungan
Perhitungan C45	
User	
Logout	

Gambar 3. 22 Rancangan Form Perhitungan C4.5

9. Rancangan menu tambah user

Form ini bermanfaat untuk menambahkan pengguna yang dapat mengakses sistem secara keseluruhan, rancangan *form* dapat dilihat pada gambar dibawah ini

Pioneer Service	
Dashboard	Tambah User
Atribut	Data User
Kerusakan	Username
Konsultasi & Prediksi	<input type="text"/>
Perhitungan C45	Password
User	<input type="text"/>
Logout	Email
	<input type="text"/>
	Tambah Data

Gambar 3. 23 Rancangan menu tambah user

3.8. Penulisan Kode Program Sistem

Pada tahap ini untuk pembuatan sistem diimplementasikan kedalam kode program, Pembangunan sistem aplikasi sendiri akan menggunakan bahasa pemrograman, *library*, dan *tools* sebagai berikut :

1. Pelatihan model klasifikasi algoritma C.45

Bahasa Pemrograman, *Library*, dan *Tools* yang digunakan dalam melatih model klasifikasi algoritma C.45 adalah sebagai berikut :

Tabel 3. 6 Tabel teknologi untuk pengolahan data

Bahasa Pemrograman/Library/Tools	Kegunaan
Python	Bahasa pemrograman yang digunakan untuk mengembangkan algoritma C.45
Pandas	Library Python yang digunakan untuk manipulasi dan transformasi data.
Scikit-Learn	Library Python yang digunakan untuk transformasi data serta pengujian akurasi algoritma pembelajaran mesin.
Google Colab	Platform online untuk menjalankan kode Python dan memproses algoritma dalam lingkungan cloud.

2. Pengembangan Aplikasi Web

Bahasa Pemrograman, *Library*, dan *Tools* yang digunakan dalam mengembangkan aplikasi web dan mengintegrasikannya dengan model C.45 yang telah dilatih adalah sebagai berikut :

Tabel 3. 7 Tabel Teknologi pengembangan web

Bahasa Pemrograman/Library/Tools	Kegunaan
HTML	Untuk merancang kerangka halaman web
Bootstrap	<i>Framework</i> untuk mengembangkan antar muka pengguna di aplikasi web
PHP	Bahasa Pemrograman <i>Server Side</i> untuk menghasilkan konten dinamis pada halaman web
MySQL	untuk menyimpan dan mengelola data dalam sebuah basis data
JavaScript	Untuk validasi formulir, pengiriman data antara mysql dan php, serta pengiriman data untuk dikirim ke API
Flask	<i>Framework</i> python yang dapat digunakan untuk membangun API untuk agar dapat menyambungkan model C45 ke aplikasi Web
XAMPP	Untuk membuat dan menguji situs web secara lokal

3.9. Pengujian Program

Setelah sistem selesai dibangun, Pengujian harus dilakukan dengan menguji masing masing fungsionalitas dari fitur-fitur didalam aplikasi, pengajuan praktek dilakukan dengan menggunakan aplikasi *web* dengan menggunakan *localhost*. Hal ini berguna untuk menemukan *bug* dan memastikan bahwa aplikasi berfungsi sesuai dengan yang diharapkan.

BAB IV

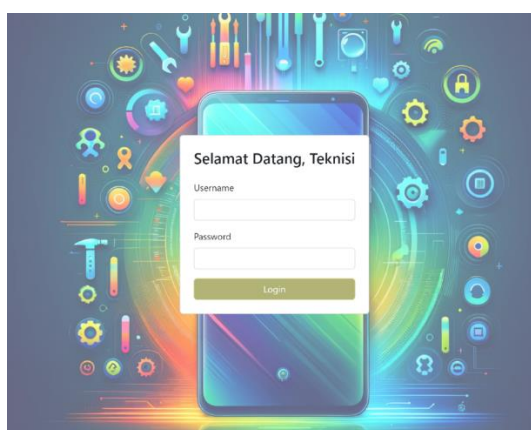
HASIL DAN PEMBAHASAN

4.1. Tampilan Hasil

Sistem Pendukung Keputusan memprediksi kerusakan HP menggunakan Algoritma C.45 dibangun dengan integrasi model algoritma C4.5 yang dilatih melalui *Google Collab*. Sistem ini memungkinkan pengguna untuk mengambil prediksi dari model menggunakan bantuan dari *Flask API* pada python, dan disertai dengan berbagai fitur untuk mencatat data kerusakan, atribut, dan data konsultasi untuk dapat digunakan selanjutnya, berikut merupakan hasil tampilan implementasi sistem.

1. Tampilan Login

Tampilan login bertujuan untuk meninjau masuk seorang teknisi dalam aplikasi yang dibuat. Peninjauan masuk menggunakan *username* dan *password* yang divalidasi dari data yang tersimpan di database agar dapat membuat aktifitas digital di aplikasi lebih aman. Tampilan form *login* dapat dilihat pada gambar dibawah ini :



Gambar 4. 2 Tampilan Menu Login

2. Tampilan Menu Tambah Atribut

Tampilan menu tambah atribut bertujuan untuk menambahkan atribut kerusakan beserta deskripsinya kedalam aplikasi sistem, tampilan menu tambah atribut dapat dilihat dalam gambar berikut ini :

Gambar 4. 3 Tampilan Menu Tambah Atribut

3. Tampilan Menu Data Atribut

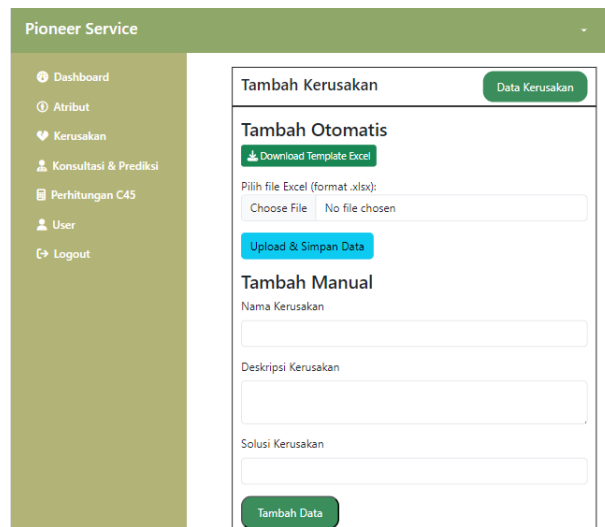
Tampilan menu data atribut berisi data-data atribut yang sudah tersimpan dalam sistem, terdapat juga navigasi edit dan hapus yang dapat memudahkan pengeditan data apabila memiliki kesalahan penginputan atau atribut yang tidak digunakan lagi, tampilan data atribut dapat dilihat pada gambar dibawah ini :

No	Nama Atribut	Deskripsi	Action
1	Kesalahan sistem yang terus menerus muncul	-	Edit Hapus
2	Perangkat Sering Mati Mendadak	tidak bisa menyala	Edit Hapus
3	suara atau bunyi aneh saat perangkat dinyalakan	tidak ada suara	Edit Hapus

Gambar 4. 4 Tampilan Menu Data Atribut

4. Tampilan Menu Tambah Kerusakan

Tampilan menu tambah kerusakan bertujuan untuk menambahkan data kerusakan beserta deskripsinya kedalam aplikasi sistem, agar mudah dalam melakukan prediksi dari model . tampilan menu tambah kerusakan dapat dilihat dalam gambar berikut ini :



Gambar 4. 5 Tampilan Menu Tambah Kerusakan

5. Tampilan Menu Data Kerusakan

Tampilan ini berguna untuk menampilkan data kerusakan yang dapat diprediksi oleh sistem beserta deskripsi dan solusinya, terdapat juga fitur edit data jika ingin mengubah data yang sudah ada dan hapus data jika terdapat data kerusakan yang tidak ingin digunakan lagi, berikut merupakan tampilan dari data kerusakan

Pioneer Service

Dashboard
Atribut
Kerusakan
Konsultasi & Prediksi
User
Logout

Data Kerusakan Tambah Kerusakan

Show: 3 entries Search:

No	Nama Kerusakan	Deskripsi Kerusakan	Solusi Kerusakan	Action
1	Kerusakan Battery Drop	Battery drop mengacu pada penurunan mendadak pada kapasitas baterai, sering terjadi ketika baterai cepat habis meski baru diisi daya.	Baterai yang mengalami drop biasanya perlu diganti. Juga disarankan untuk menggunakan charger asli untuk menghindari masalah serupa di masa depan.	Edit Hapus
2	Kerusakan Sistem Operasi	Kerusakan pada sistem operasi bisa disebabkan oleh file sistem yang rusak, kegagalan update, atau malware. Gejalanya termasuk boot loop, crash, atau kinerja perangkat yang lambat.	Masalah sistem operasi sering bisa diatasi dengan melakukan hard reset atau reinstalasi sistem operasi. Pastikan untuk backup data sebelum melakukan tindakan ini.	Edit Hapus
3	Kerusakan IC Charging	IC Charging mengontrol pengisian baterai perangkat. Masalah dengan IC ini bisa menyebabkan perangkat tidak mengisi daya atau mengisi daya	Ganti IC Charging dengan yang baru. Karena ini melibatkan komponen yang sangat kecil, penggantian harus dilakukan oleh teknisi	Edit Hapus

Gambar 4. 6 Tampilan menu Data Kerusakan

6. Tampilan Menu Tambah Konsultasi dan Prediksi

Tampilan ini berguna agar teknisi dapat mencatat konsultasi pelanggan mengenai kerusakan HP terkait, dalam menu ini, teknisi akan memasukkan gejala gejala dari Handphone terkait. Setelah itu, tombol lakukan prediksi akan menghubungkan sistem dengan model C.45 menggunakan flask untuk menghasilkan prediksi kerusakan terkait, berikut merupakan tampilan menu tambah konsultasi dan prediksi

Pioneer Service

- Dashboard
- Atribut
- Kerusakan
- Konsultasi & Prediksi
- Perhitungan C45
- User
- Logout

Input Data Konsultasi

Data Konsultasi

Tambah Otomatis

Download Template Excel

Pilih file Excel (format .xlsx):

Choose File No file chosen

Upload & Simpan Data

Tambah Manual

Tanggal: mm/dd/yyyy

Nama Perangkat: _____

Nama Pelanggan: _____

Silahkan Pilih Atribut Gejala Pada Perangkat

Pilih atribut

Lakukan Prediksi

Hasil Prediksi: _____

Simpan Data

Gambar 4. 7 Tampilan menu Tambah Konsultasi dan Prediksi

7. Tampilan Menu Data Konsultasi dan Prediksi

Tampilan Menu Data Konsultasi berisi daftar konsultasi yang sudah tercatat dalam sistem, terdapat navigasi menuju detail konsultasi untuk melihat masing masing konsultasi secara detail dan hapus konsultasi, tampilan menu data konsultasi dapat dilihat pada gambar dibawah ini :

Pioneer Service

- Dashboard
- Atribut
- Kerusakan
- Konsultasi & Prediksi
- User
- Logout

Data Konsultasi

Tambah Konsultasi

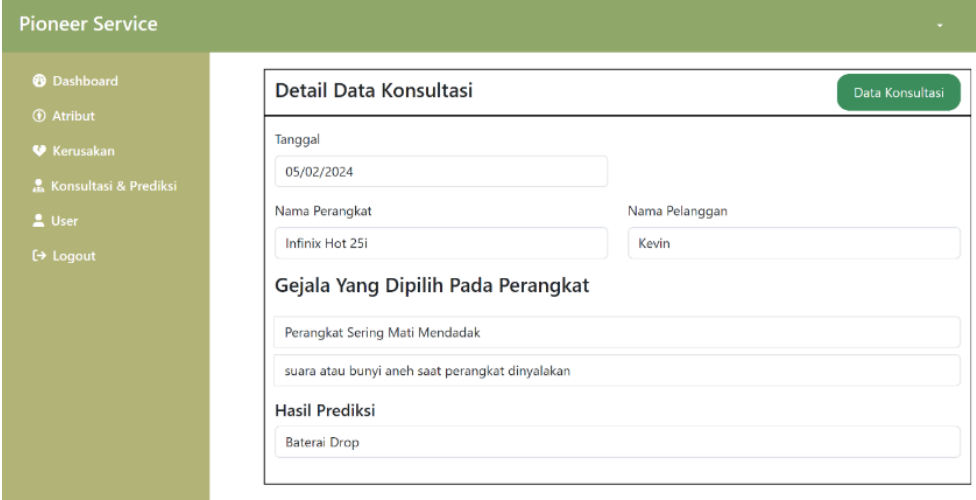
Show 1 entries Search: _____

No	Tanggal	Nama Perangkat	Nama Pelanggan	Action
1	2024-05-02	Infinitix Hot 25i	Kevin	Detail Hapus

Gambar 4. 8 Tampilan Menu Data Konsultasi

8. Tampilan Menu Detail Konsultasi

Tampilan menu detail konsultasi menampilkan lebih jelas rincian dari setiap data konsultasi beserta hasil prediksi kerusakannya, tampilan menu Detail konsultasi dapat dilihat pada gambar dibawah ini :



The screenshot displays the 'Detail Data Konsultasi' form within the Pioneer Service application. The form is structured as follows:

- Header:** 'Detail Data Konsultasi' with a 'Data Konsultasi' button.
- Tanggal:** Input field containing '05/02/2024'.
- Nama Perangkat:** Input field containing 'Infinix Hot 25i'.
- Nama Pelanggan:** Input field containing 'Kevin'.
- Gejala Yang Dipilih Pada Perangkat:**
 - Input field: 'Perangkat Sering Mati Mendadak'
 - Input field: 'suara atau bunyi aneh saat perangkat dinyalakan'
- Hasil Prediksi:** Input field containing 'Baterai Drop'.

The sidebar menu on the left includes: Dashboard, Atribut, Kerusakan, Konsultasi & Prediksi, User, and Logout.

Gambar 4. 9 Tampilan menu Detail Konsultasi

9. Tampilan Menu Perhitungan C4.5

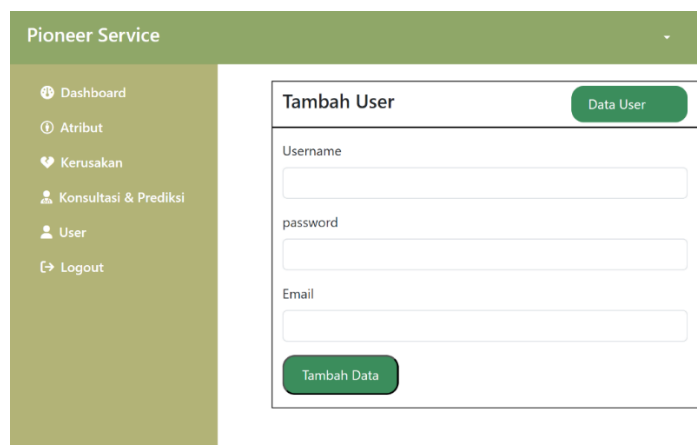
Menu perhitungan C4.5 berfungsi sebagai tempat untuk melatih data konsultasi yang ada menggunakan algoritma C4.5, menu ini akan menghasilkan model pohon keputusan, dan model prediksi C4.5 yang dapat digunakan untuk memprediksi data baru, berikut merupakan tampilan menu perhitungan C4.5



Gambar 4. 10 Tampilan Menu Perhitungan C4.5

10. Tampilan Menu Tambah User

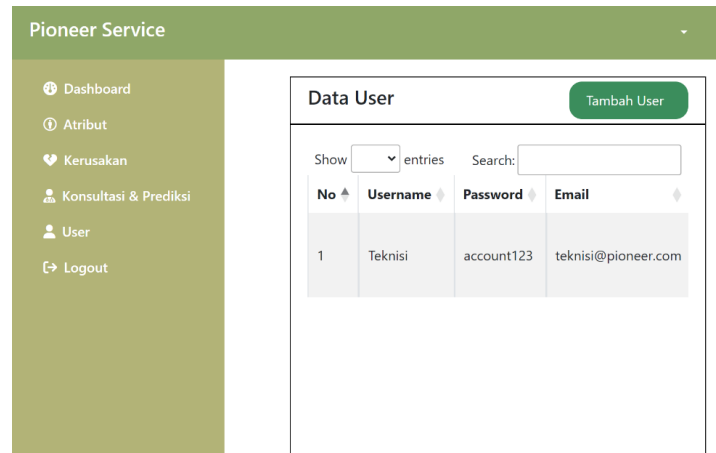
Menu Tambah user berguna untuk menambah data pengguna yang memiliki akses terhadap aplikasi sistem, tampilan menu tambah user dapat dilihat pada gambar dibawah ini :



Gambar 4. 11 Tampilan Menu Tambah User

11. Tampilan Menu Data User

Tampilan menu data user berguna untuk melihat data pengguna yang memiliki akses terhadap sistem, berikut merupakan tampilan dari menu data user :



Gambar 4. 12 Tampilan menu data user

4.2. Perhitungan Algoritma C4.5

Data yang diperoleh dari lokasi penelitian diproses terlebih dahulu. Data yang didapatkan berupa data pemeriksaan servis hp di Pioneer Service pada tahun 2023, data terkait dapat dilihat dalam tabel dibawah ini :

Tabel 4. 1 Tabel Data Pemeriksaan

Tanggal Pemeriksaan	Nama Perangkat	Kerusakan
02/01/2023	Vivo V21	IC Charging
02/01/2023	Samsung Galaxy S21	IC Power
05/01/2023	Realme Narzo 30A	IC Mikroprosesor
09/01/2023	Samsung Galaxy S21	IC eMMC
13/01/2023	Samsung Galaxy S21	IC PA (Power Amplifier)
14/01/2023	Oppo Reno5	IC Charging
15/01/2023	Oppo Reno5	IC eMMC
15/01/2023	Samsung Galaxy A52	IC Mikroprosesor
21/01/2023	Realme C21	Baterai Drop
.....		

Terdapat berbagai macam kerusakan yang bisa dialami oleh perangkat HP, namun dalam penelitian ini, jenis kerusakan HP beserta ciri ciri perangkat mengalami kerusakan terkait yang diimplementasikan kedalam aplikasi adalah sebagai berikut :

1. Kerusakan *IC Power*

Ciri-ciri perangkat yang mengalami kerusakan ini adalah :

- Perangkat tidak menyala
- Tidak ada reaksi saat tombol daya ditekan
- Perangkat sering mati mendadak
- *Overheating* pada area power IC
- Suara atau bunyi aneh saat perangkat dinyalakan

2. Kerusakan IC Power Amplifier

Ciri-ciri perangkat yang mengalami kerusakan ini adalah :

- Suara keluar dari perangkat terdengar sangat lemah
- Distorsi suara atau suara pecah
- Tidak ada suara sama sekali
- Suara terdengar kasar atau tidak stabil
- Perangkat panas saat digunakan dalam waktu lama

3. Kerusakan IC *Charging*

Ciri-ciri perangkat yang mengalami kerusakan ini adalah :

- Perangkat tidak bisa diisi daya
- Indikator pengisian tidak bekerja dengan benar
- Perangkat tidak mendeteksi charger
- Pengisian berlangsung sangat lambat

- *Charger* atau *port* terasa panas saat digunakan

4. Kerusakan IC *eMMC*

Ciri-ciri perangkat yang mengalami kerusakan ini adalah :

- Perangkat lambat merespon
- Aplikasi sering keluar atau *crash*
- Kesalahan saat membuka atau menyimpan file
- Perangkat tidak dapat melakukan update Sistem Operasi
- Data hilang atau tidak terbaca

5. Kerusakan IC Mikroprocessor

Ciri-ciri perangkat yang mengalami kerusakan ini adalah :

- Perangkat sering mengalami hang atau *crash*
- Kinerja perangkat sangat lambat
- Tidak dapat menjalankan aplikasi yang kompleks
- Panas berlebihan pada area mikroprocessor
- Kesalahan sistem operasi yang tidak dapat dijelaskan

6. Baterai Drop

Ciri-ciri perangkat yang mengalami kerusakan ini adalah :

- Baterai cepat habis atau tidak bertahan lama
- Persentase baterai tidak akurat
- Baterai tidak dapat diisi daya sepenuhnya
- Perangkat mati mendadak meskipun baterai masih terisi
- Baterai mengalami pembengkakan atau deformasi

7. Kerusakan Sistem Operasi

Ciri-ciri perangkat yang mengalami kerusakan ini adalah :

- Perangkat sering *crash* atau *restart* sendiri
- Aplikasi tidak dapat dijalankan
- Kesalahan Sistem yang terus menerus muncul
- Perangkat menjadi sangat lambat
- Tidak dapat melakukan update atau instalasi *sistem operasi*

Setiap ciri-ciri kerusakan diatas, akan dimasukkan kedalam data penelitian untuk memudahkan algoritma C45 dalam mempelajari pola pada data pemeriksaan, berikut merupakan tampilan data yang sudah ditambah ciri-ciri kerusakan :

Tanggal	Nama Perangkat	Kesalahan sistem yang terus menerus muncul	Tidak ada reaksi saat tombol daya ditekan	Perangkat lambat merespons	Perangkat tidak menyala	Overheating pada area power IC	Pengisian berlangsung sangat lambat	Aplikasi tidak dapat dijalankan	Kerusakan
02/01/2023	Vivo V21	Tidak	Tidak	Tidak	Tidak	Tidak	Ya	Tidak	IC Charging
02/01/2023	Samsung Galaxy S21	Tidak	Ya	Tidak	Ya	Ya	Tidak	Tidak	IC Power
05/01/2023	Realme Narzo 30A	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	IC Mikroprosessor
09/01/2023	Samsung Galaxy S21	Tidak	Tidak	Ya	Tidak	Tidak	Tidak	Tidak	IC eMMC
13/01/2023	Samsung Galaxy S21	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	IC PA (Power Amplifier)
14/01/2023	Oppo Reno5	Tidak	Tidak	Tidak	Tidak	Tidak	Ya	Tidak	IC Charging
15/01/2023	Oppo Reno5	Tidak	Tidak	Ya	Tidak	Tidak	Tidak	Tidak	IC eMMC
15/01/2023	Samsung Galaxy A52	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	IC Mikroprosessor
21/01/2023	Realme C21	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Baterai Drop
21/01/2023	Oppo Reno5	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	IC Charging
22/01/2023	Vivo V21	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Baterai Drop
28/01/2023	Vivo Y20s	Tidak	Tidak	Ya	Tidak	Tidak	Tidak	Tidak	IC eMMC
28/01/2023	Realme Narzo 30A	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	IC PA (Power Amplifier)
29/01/2023	Realme Narzo 30A	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	IC Mikroprosessor
02/02/2023	Vivo V21	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	IC Mikroprosessor
04/02/2023	Realme C21	Tidak	Tidak	Tidak	Tidak	Tidak	Ya	Tidak	IC Charging
04/02/2023	Oppo A54	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	IC Mikroprosessor

Gambar 4. 13 Dataset yang diberi ciri-ciri kerusakan

Ciri-ciri kerusakan pada perangkat dijadikan atribut dalam dataset, dimana setiap perangkat yang menunjukkan ciri-ciri tersebut ditandai dengan nilai ‘Ya’, sedangkan yang tidak menunjukkan ciri-ciri tersebut dengan nilai ‘Tidak’. Setelah data diolah, data tersebut harus diubah menjadi format numerik untuk memungkinkan pelatihan menggunakan algoritma C.45. Dalam data tersebut, atribut ini direpresentasikan secara biner, dimana 1 menunjukkan keberadaan ciri-ciri tersebut dan 0 menunjukkan sebaliknya, implementasi pelabelan menjadi bentuk numerik untuk setiap atribut dapat dilihat pada gambar dibawah ini

Panas berlebihan pada area mikroprosesor	Kesalahan sistem operasi yang tidak dapat dijelaskan	Kinerja perangkat sangat lambat	...	Suara atau bunyi aneh saat perangkat dinyalakan	perangkat tidak mendeteksi charger	Salah saat menekan layar	Tidak dapat menjalankan aplikasi yang kompleks	Charger atau port terasa panas saat digunakan	Munculnya garis-garis atau bintik-bintik pada layar	Overheating pada area power IC	Pengisian berlangsung sangat lambat	Aplikasi tidak dapat dijalankan	Kerusakan
0	1	1	...	0	0	0	1	0	0	0	0	0	IC Mikroprosesor
0	1	0	...	0	0	0	0	0	0	0	0	0	IC Mikroprosesor
0	0	0	...	1	0	0	0	0	0	1	0	0	IC Power
0	0	0	...	0	0	0	1	0	0	0	0	0	IC Mikroprosesor
0	0	0	...	0	0	0	0	0	0	0	0	0	Kerusakan Sistem Operasi
...
0	0	0	...	1	0	0	0	0	0	0	0	0	IC Power
0	0	0	...	0	0	0	0	0	0	0	0	0	Baterai Drop
0	0	0	...	0	0	0	0	0	0	0	0	0	IC eMMC
0	0	0	...	0	0	0	0	0	0	0	0	0	IC eMMC
0	0	0	...	0	0	0	0	0	0	0	0	0	Kerusakan Sistem Operasi

Gambar 4. 14 Dataset Kerusakan HP setelah atribut diberi data numerik

Dataset terdiri atas 262 baris data kerusakan untuk data latih dan 35 baris data untuk data uji, yang mencakup berbagai atribut yang disebutkan. Sekarang, data latih diatas akan mulai diolah mulai dari perhitungan *Entropy*, *Information Gain*, hingga pembuatan pohon keputusannya. Melalui data latih terkait, berikut merupakan ringkasan yang dapat berguna untuk perhitungan-perhitungan yang dibutuhkan :

Tabel 4. 2 Tabel Daftar Kerusakan berdasarkan Jumlah kasus

Kerusakan	Total Kasus
Kerusakan <i>IC Power</i>	45 Kasus
Kasus Kerusakan IC PA	33 Kasus
Kasus Kerusakan IC eMMC	44 Kasus
Kasus Kerusakan IC <i>Charging</i>	30 Kasus
Kasus Kerusakan IC <i>Mikroprocessor</i>	38 Kasus
Kasus Kerusakan <i>Battery Drop</i>	38 Kasus
Kasus Kerusakan Sistem Operasi	34 Kasus
Total Pemeriksaan	262 Kasus

Sesuai dengan tabel diatas, maka perhitungan *entropy* dapat dilakukan dan hasilnya adalah sebagai berikut :

$$\begin{aligned} \text{Entropy Total} = & - \left(\frac{45}{262} \text{Log}_2 \frac{45}{262} \right) \pm \left(\frac{33}{262} \text{Log}_2 \frac{33}{262} \right) + \\ & - \left(\frac{44}{262} \text{Log}_2 \frac{44}{262} \right) + - \left(\frac{30}{262} \text{Log}_2 \frac{30}{262} \right) + - \left(\frac{38}{262} \text{Log}_2 \frac{38}{262} \right) + \\ & - \left(\frac{38}{262} \text{Log}_2 \frac{38}{262} \right) + - \left(\frac{34}{262} \text{Log}_2 \frac{34}{262} \right) \end{aligned}$$

$$\begin{aligned} \text{Entropy Total} = & 0,43653 + 0,37645 + 0,43227 + 0,3580 + 0,40400 \\ & + 0,40400 + 0,3823 \end{aligned}$$

$$\text{Entropy Total} = \mathbf{2,7940}$$

Nilai *entropy* diatas mencerminkan tingkat keberagaman data yang cukup tinggi, hal ini sesuai dengan data yang memiliki berbagai macam ciri-ciri kerusakan dan jenis kerusakan yang ingin diprediksi. Setelah menghitung jumlah *Entropy* total, selanjutnya, nilai *information gain*, setiap atribut-atribut yang berbeda akan dihitung. Berikut adalah contoh perhitungan manual dalam 1 atribut yang akan dibentuk menjadi sebuah pohon keputusan :

1. Kesalahan Sistem yang terus menerus muncul

Atribut muncul bernilai ya sebanyak 17 kali dan muncul dalam label kerusakan dibawah ini :

- Kerusakan Sistem Operasi: 17 kali

Dikarenakan jumlah muncul sama dengan jumlah atribut ya, maka *entropy* nya bernilai 0

Atribut muncul bernilai tidak sebanyak 245 kali dan tersebar dalam label kerusakan berikut ini :

Tabel 4. 3 Sebaran kerusakan berdasarkan data

Jenis Kerusakan	Jumlah Muncul
Battery Drop	38
IC Charging	30
IC eMMC	44
IC Mikroprosessor	38
IC Power Amplifier	33
IC Power	45
Kerusakan Sistem Operasi	17

Perhitungan Entropy untuk atribut tidak adalah sebagai berikut :

$$\begin{aligned}
 \text{Entropy Total} &= - \left(\frac{38}{245} \text{Log}_2 \frac{38}{245} \right) \pm \left(\frac{30}{245} \text{Log}_2 \frac{30}{245} \right) + \\
 &- \left(\frac{44}{245} \text{Log}_2 \frac{44}{245} \right) + - \left(\frac{38}{245} \text{Log}_2 \frac{38}{245} \right) + - \left(\frac{33}{245} \text{Log}_2 \frac{33}{245} \right) + \\
 &- \left(\frac{45}{245} \text{Log}_2 \frac{45}{245} \right) + - \left(\frac{17}{245} \text{Log}_2 \frac{17}{245} \right) \\
 \text{Entropy Total} &= 0,4170 + 0,3710 + 0,4445 + 0,4170 + 0,3896 + \\
 &0,4490 + 0,2671 = \mathbf{2,7556}
 \end{aligned}$$

Untuk perhitungan *Information Gain* nya adalah

$$\begin{aligned}
 \text{Gain} &= 2.7940 - [(17/262) * 0] + (245/262) * 2.7556] \\
 &= \mathbf{0.2172}
 \end{aligned}$$

Perhitungan gain diatas kemudian dilanjutkan kepada 34 Atribut lain dengan metode perhitungan yang sama, sehingga menghasilkan tabel berisi besar nilai *Information Gain* setiap atribut seperti dibawah ini untuk Node pertama :

Tabel 4. 4 Data *Information Gain* Pada Percabangan Pertama

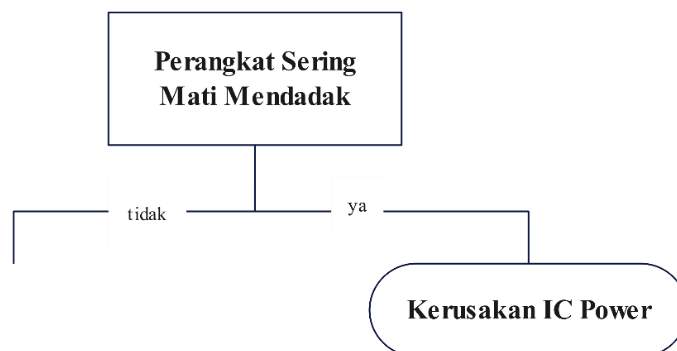
Nama Gejala	Gain
Perangkat Sering Mati Mendadak	0,453406
Suara atau bunyi aneh saat perangkat dinyalakan	0,370887
Tidak dapat menjalankan aplikasi yang kompleks	0,369807
Tidak ada reaksi saat tombol daya ditekan	0,355631
Perangkat Tidak Menyala	0,340718
Panas berlebihan pada area mikroprosesor	0,304091
Data hilang atau tidak terbaca	0,302659
Perangkat tidak bisa diisi daya	0,301806
Perangkat menjadi sangat lambat	0,294461
Aplikasi tidak dapat dijalankan	0,294461
Kesalahan sistem operasi yang tidak dapat dijelaskan	0,288671
Perangkat sering mengalami hang atau crash	0,288671
Overheating pada area power IC	0,284083
Kinerja Perangkat Sangat Lambat	0,273595
Baterai Tidak dapat diisi daya sepenuhnya	0,273595
Suara Terdengar kasar atau tidak stabil	0,2673
Tidak ada suara sama sekali	0,2673
Perangkat Lambat Merespons	0,261349
Perangkat tidak dapat melakukan update Sistem Operasi	0,261349
Perangkat tidak mendeteksi charger	0,249885
Kesalahan saat membuka atau menyimpan file	0,248075
Aplikasi sering keluar atau crash	0,248075
Distorsi suara atau suara pecah	0,23586
Perangkat panas saat digunakan dalam waktu lama	0,23586
Tidak dapat melakukan update atau instalasi OS	0,231615
Persentase baterai tidak akurat	0,230217
Charger atau port terasa panas saat digunakan	0,217536
Pengisian berlangsung sangat lambat	0,217536
Kesalahan Sistem Yang Terus Menerus Muncul	0,21677
Baterai mengalami pembengkakan atau deformasi	0,216314
Suara keluar dari perangkat sangat lemah	0,205802
Baterai cepat habis atau tidak bertahan lama	0,202664
Perangkat sering crash atau restart sendiri	0,202226
Indikator pengisian tidak bekerja dengan benar	0,201934
Perangkat Mati Mendadak meskipun baterai masih terisi	0,189254

Dari nilai *information gain* diatas, gain Perangkat Sering Mati Mendadak adalah yang paling besar. Maka atribut tersebut dijadikan sebagai *node* awal.



Gambar 4. 15 Pohon keputusan pada node awal

Selanjutnya, algoritma C.45 akan dipanggil kembali dengan kumpulan sampel ‘ya’ pada ‘Perangkat sering mati mendadak’ dengan target salah satu dari ketujuh jenis kerusakan. Karena salah satu sampel termasuk pada kelas label “Kerusakan IC Power”. Maka fungsi untuk ya akan berhenti dan membuat simpul tunggal dengan label “Kerusakan IC Power”.



Gambar 4. 16 Pohon keputusan cabang kedua

Selanjutnya perlu memanggil algoritma C.45 dengan kumpulan sampel yang dapat menjadi node dari atribut ‘perangkat sering mati mendadak bernilai ‘tidak’

dengan target salah satu dari ketujuh jenis label kerusakan. ketika atribut “perangkat tidak bisa diisi data” bernilai tidak, distribusi kerusakan pada perangkat terdiri dari jenis-jenis berikut :

Tabel 4. 5 Distribusi kerusakan berdasarkan atribut sebelumnya

Jenis Kerusakan	Jumlah Muncul
Battery Drop	38
IC Charging	30
IC eMMC	44
IC Mikroprosessor	38
IC Power Amplifier	33
IC Power	9
Kerusakan Sistem Operasi	34

Langkah selanjutnya adalah menghitung entropy terbaru untuk keseluruhan data, seperti dibawah ini :

$$\begin{aligned} \text{Entropy Total} = & - \left(\frac{38}{245} \text{Log}_2 \frac{38}{245} \right) \pm \left(\frac{30}{245} \text{Log}_2 \frac{30}{245} \right) + - \left(\frac{44}{245} \text{Log}_2 \frac{44}{245} \right) + \\ & - \left(\frac{38}{245} \text{Log}_2 \frac{38}{245} \right) + - \left(\frac{33}{245} \text{Log}_2 \frac{33}{245} \right) + - \left(\frac{9}{245} \text{Log}_2 \frac{9}{245} \right) + \\ & - \left(\frac{34}{245} \text{Log}_2 \frac{34}{245} \right) \end{aligned}$$

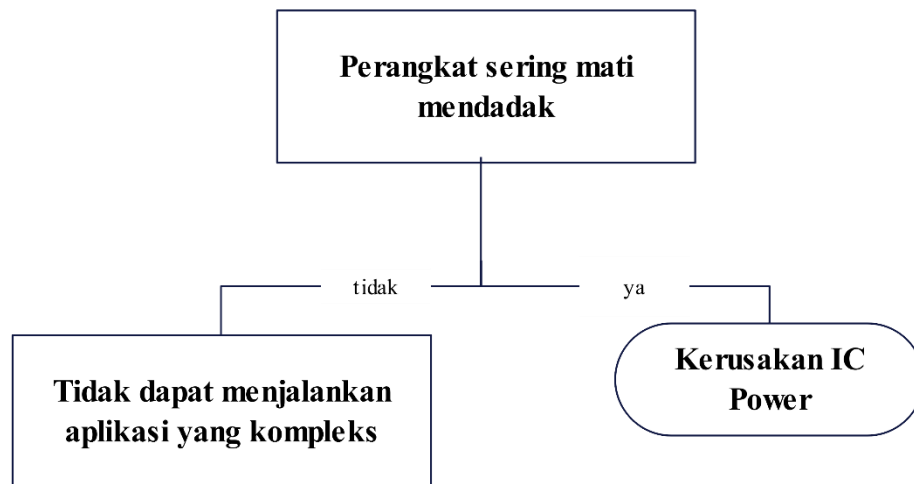
$$\text{Entropy Total} = 2,713$$

Langkah selanjutnya adalah melakukan perhitungan entropi di seluruh atribut yang masih tersedia dan menghitung masing masing gainnya, perhitungan dan pengurutan *gain* dari terbesar dan terkecil dapat dilihat pada tabel dibawah ini

Tabel 4. 6 Tabel hasil *gain* untuk cabang berikutnya

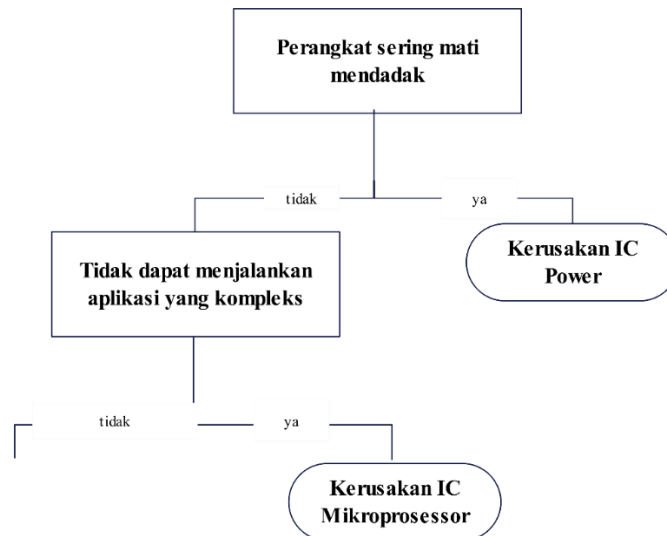
Attribute	Information Gain
Tidak dapat menjalankan aplikasi yang kompleks	0,400645443
Perangkat tidak bisa diisi daya	0,329157665
Panas berlebihan pada area mikroprosesor	0,328687829
Data hilang atau tidak terbaca	0,324924187
Perangkat menjadi sangat lambat	0,319608359
Aplikasi tidak dapat dijalankan	0,319608359
Kesalahan sistem operasi yang tidak dapat dijelaskan	0,311855609
Perangkat sering mengalami hang atau crash	0,311855609
Kinerja perangkat sangat lambat	0,295417833
Baterai tidak dapat diisi daya sepenuhnya	0,295417833
suara terdengar kasar atau tidak stabil	0,29018605
tidak ada suara sama sekali	0,29018605
Perangkat lambat merespons	0,280181102
Perangkat tidak dapat melakukan update Sistem Operasi	0,280181102
perangkat tidak mendeteksi charger	0,272043435
Kesalahan saat membuka atau menyimpan file	0,265832655
Aplikasi sering keluar atau crash	0,265832655
Distorsi suara atau suara pecah	0,255784678
perangkat panas saat digunakan dalam waktu lama	0,255784678
tidak dapat melakukan update atau instalasi OS	0,250863927
Persentase baterai tidak akurat	0,248221339
Charger atau port terasa panas saat digunakan	0,236571406
Pengisian berlangsung sangat lambat	0,236571406
Kesalahan sistem yang terus menerus muncul	0,234670436
Beterai mengalami pembengkakan atau deformasi	0,233125224
Suara keluar dari perangkat terdengar sangat lemah	0,222967917
Indikator pengisian tidak bekerja dengan benar	0,219491848
Perangkat sering crash atau restart sendiri	0,218822125
Baterai cepat habis atau tidak bertahan lama	0,218317731
Perangkat mati mendadak meskipun baterai masih terisi	0,203783792
Perangkat tidak menyala	0,113737411
Suara atau bunyi aneh saat perangkat dinyalakan	0,113737411
Overheating pada area power IC	0,113737411
Tidak ada reaksi saat tombol daya ditekan	0,088851498

Dari nilai *information gain* diatas, atribut **Tidak dapat menjalankan aplikasi yang kompleks** memiliki *gain* tertinggi. Maka atribut tersebut dijadikan sebagai *node* cabang berikutnya, sehingga gambar pohon keputusan menjadi seperti dibawah ini :



Gambar 4. 17 Gambar Pohon keputusan setelah pemilihan cabang kedua

Selanjutnya, algoritma C.45 akan dipanggil kembali dengan kumpulan sampel **ya** pada atribut **Tidak dapat menjalankan aplikasi yang kompleks** dengan target salah satu dari ketujuh jenis kerusakan. Karena salah satu sampel termasuk pada kelas label **Kerusakan IC Mikroprosesor**. Maka fungsi untuk **ya** akan berhenti dan membuat simpul tunggal dengan label **Kerusakan IC Mikroprosesor**. Sehingga gambar pohon keputusan akan menjadi seperti gambar dibawah ini.



Gambar 4. 18 Pohon keputusan setelah mendapatkan label kerusakan *IC Mikroprosesor*

Selanjutnya perlu memanggil algoritma C.45 dengan kumpulan sampel yang dapat menjadi node dari atribut **Tidak dapat menjalankan aplikasi yang kompleks** bernilai **tidak** dengan target salah satu dari ketujuh jenis label kerusakan. Proses rekursif akan terus terjadi hingga seluruh atribut memasuki cabang pohon keputusan.

Pengembangan model prediksi berbasis Algoritma C.45 dikembangkan dengan bahasa *Python*, model dilatih dengan data yang sudah diverifikasi dan ditransformasi sebelumnya untuk membangun pohon keputusan. Model yang sudah dilatih disimpan untuk kemudian dihubungkan sebagai fitur prediksi dalam sistem.

4.3. Melatih Model Algoritma C.45 Pada *Google Collab*

Algoritma C4.5 diimplementasikan menggunakan bahasa pemrograman *Python* dan dilatih dengan dataset yang terbagi menjadi data latih (*training data*) dan data uji (*testing data*). Implementasi algoritma ini dilakukan di *Google Colab*, tempat proses pemeriksaan data, transformasi data, dan pengujian akurasi berlangsung. Berikut adalah detail hasil dari implementasi yang telah dilakukan.

1. Memasukkan Dataset ke *DataFrame* Pandas

Langkah pertama yang dilakukan adalah memasukkan pustaka-pustaka yang dibutuhkan untuk memproses dan menganalisa data penelitian menggunakan *Google Collab*, berikut penerapan memasukkan pustaka terkait :

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
from sklearn.tree import DecisionTreeClassifier, export_graphviz
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, confusion_matrix, classification_report
from sklearn.preprocessing import LabelEncoder
import pydotplus
from IPython.display import Image
import pickle
```

Gambar 4. 19 Memasukkan Pustaka Python yang digunakan

Berikut pustaka pustaka yang dimasukkan :

- **Pandas** digunakan untuk manipulasi dan analisis data.
- **Numpy** digunakan untuk operasi array dan matriks.
- **Scikit-Learn** menyediakan alat untuk pemodelan data termasuk klasifikasi *decision tree*
- **pydotplus** digunakan untuk visualisasi grafik dari pohon keputusan.
- **IPython.display.Image** digunakan untuk menampilkan gambar dalam notebook.
- **Pickle** digunakan untuk menyimpan model.

Langkah berikutnya adalah memuat data yang digunakan untuk melatih dan menguji model *Decision Tree C4.5* menggunakan *dataframe* pada *pandas*, berikut merupakan implementasinya :

```
data = pd.read_csv('data_latih.csv')
```

Gambar 4. 20 Memuat data yang digunakan

Berikut merupakan tampilan data setelah dimasukkan kedalam *dataframe* :

	Tanggal Pemeriksaan	Nama Perangkat	Kesalahan sistem yang terus menerus muncul	Tidak ada reaksi saat tombol daya ditekan	Perangkat lambat merespons	Perangkat tidak menyala	tidak dapat melakukan update atau instalasi OS	Indikator pengisian tidak bekerja dengan benar	Perangkat tidak bisa diisi daya	Panas berlebihan pada area mikroprosesor	...	Perangkat menjadi sangat lambat	Perangkat sering mengalami hang atau crash	Suara atau bunyi aneh saat perangkat dinyalakan
0	2023-04-13	Xiaomi Mi 11	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	...	Tidak	Tidak	Tidak
1	2023-12-15	Vivo V21	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	...	Tidak	Tidak	Tidak
2	2023-09-28	Samsung Galaxy S21	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	...	Tidak	Tidak	Tidak
3	2023-04-17	Xiaomi Redmi Note 10	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	...	Tidak	Tidak	Tidak
4	2023-03-13	Samsung Galaxy S21	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	...	Tidak	Tidak	Tidak
...
257	2023-09-20	Samsung Galaxy S21	Ya	Tidak	Tidak	Tidak	Ya	Tidak	Tidak	Tidak	...	Ya	Tidak	Tidak

Gambar 4. 21 Tampilan data setelah dimuat ke dataframe pandas

Langkah berikutnya adalah melanjutkan proses data dengan menghapus kolom yang tidak digunakan dan memisahkan data menjadi data latih dan data uji, berikut merupakan implementasinya :

```
column_drop = ['Tanggal Pemeriksaan', 'Nama Perangkat']
data = data.drop(columns=column_drop)
X = data.drop('Kerusakan', axis=1)
y = data['Kerusakan']
X_train, X_test, y_train, y_test = train_test_split(X,y, test_size=0.2, random_state = 42)
```

Gambar 4. 22 Implementasi Pemisahan Data

Kode tersebut bertujuan untuk memproses dataset dengan melakukan dua tahap utama: penghapusan kolom yang tidak relevan dan pemisahan data menjadi variabel independen (X) dan variabel target (y) untuk pembelajaran mesin. Pertama, kolom-kolom yang tidak dibutuhkan, yaitu 'Tanggal Pemeriksaan' dan 'Nama Perangkat', dihapus dari dataset menggunakan metode drop dengan argumen columns yang mengacu pada daftar column_drop. Setelah itu, variabel independen (X) diinisialisasi dengan menghapus kolom target 'Kerusakan' dari dataset, sedangkan variabel target (y) diisi dengan kolom 'Kerusakan'.

Selanjutnya, data dipisahkan menjadi data latih dan data uji menggunakan fungsi `train_test_split`, di mana data latih (X_{train} , y_{train}) akan memiliki ukuran 80% dari dataset, sedangkan data uji (X_{test} , y_{test}) akan memiliki ukuran 20%. Langkah berikutnya adalah mengubah kategori teks yang bernilai ‘Ya’ dan ‘Tidak’ didalam data menjadi fitur angka dengan nilai 1 dan 0, langkah pertama diimplementasikan kepada data latih terlebih dahulu, berikut merupakan implementasi kodenya

```
from sklearn.preprocessing import LabelEncoder

label_encoder = LabelEncoder()
for column in X_train.columns:
    X_train[column] = label_encoder.fit_transform(X_train[column])
```

Gambar 4. 23 Implementasi Ubah Kategori teks menjadi fitur angka

Berikut tampilan data latih X_{train} ketika sudah diubah menjadi kategori fitur angka :

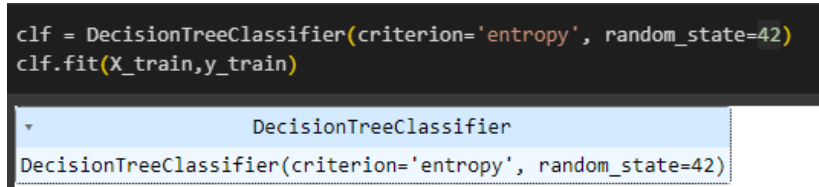
Kesalahan sistem yang terus menerus muncul	Tidak ada reaksi saat tombol daya ditekan	Perangkat lambat merespons	Perangkat tidak menyala	tidak dapat melakukan update atau instalasi OS	Indikator pengisian tidak bekerja dengan benar	Perangkat tidak bisa diisi daya	Panas berlebihan pada area mikroprosesor	Kesalahan sistem operasi yang tidak dapat dijelaskan	Kinerja perangkat sangat lambat	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	1	1	1	...
0	0	0	0	0	0	0	0	0	0	...
1	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	1	1	0	0	0	...
...
0	0	0	0	0	0	0	0	0	0	...
0	1	0	1	0	0	0	0	0	0	...
0	0	1	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...

Gambar 4. 24 Tampilan data latih

Langkah berikutnya adalah menginisialisasi model *Decision Tree Classifier* dengan menggunakan kriteria pengukuran *Entropy* dan *Information Gain* sebagai metrik pemilihan fitur terbaik untuk membangun pohon keputusan. Fitur atribut

yang menghasilkan *Gain* tertinggi akan dipilih untuk membagi data pada setiap tahap dalam pembangunan pohon keputusan. Berikut merupakan implementasinya :

```
clf = DecisionTreeClassifier(criterion='entropy', random_state=42)
clf.fit(X_train,y_train)
```



Gambar 4. 25 Implementasi *Decision Tree Classifier*

2. Hasil Akurasi setelah Implementasi Algoritma C4.5

Setelah dilakukan pelatihan model terhadap data latih, maka langkah selanjutnya adalah menguji performa model algoritma C4.5 yang sudah dibangun menggunakan data uji, namun sebelum model memprediksi data baru, langkah berikutnya adalah mengubah fitur kategori teks dalam setiap ciri-ciri kerusakan pada data latih menjadi fitur numerik, berikut merupakan implementasinya :

```
# Label Encoder pada X_test
for column in X_test.columns:
    X_test[column] = label_encoder.transform(X_test[column])
```

Gambar 4. 26 Implementasi Fitur numerik pada data uji

Implementasi algoritma untuk memprediksi terhadap data uji diimplementasikan dalam kode seperti dibawah ini :

```
y_pred = clf.predict(X_test)
```

Gambar 4. 27 Implementasi Algoritma terhadap data uji

Setelah implementasi dilakukan, langkah selanjutnya adalah membandingkan label kerusakan yang ada pada data uji dengan label yang diprediksi oleh algoritma yang telah dilatih untuk mengetahui performa model, metrik yang digunakan adalah *Accuracy*, *Precision*, *Recall*, dan *F1 Score*. Berikut merupakan perbandingan hasil label antara data uji dan hasil prediksi algoritma.

Tabel 4. 7 Perbandingan Label Sebenarnya dan Hasil Prediksi

Index	Label Sebenarnya	Label Prediksi
125	IC Mikroprosesor	IC Mikroprosesor
129	IC Mikroprosesor	IC Mikroprosesor
82	IC eMMC	IC eMMC
41	IC Charging	IC Charging
7	Baterai Drop	IC Mikroprosesor
137	IC Mikroprosesor	IC Mikroprosesor
126	IC Mikroprosesor	IC Mikroprosesor
69	IC eMMC	IC eMMC
175	IC PA (Power Amplifier)	IC PA (Power Amplifier)
195	IC Power	IC Power
14	Baterai Drop	Baterai Drop
142	IC Mikroprosesor	IC Mikroprosesor
110	IC eMMC	IC eMMC
154	IC PA (Power Amplifier)	IC PA (Power Amplifier)
212	IC Power	IC Power
79	IC eMMC	IC eMMC
44	IC Charging	IC Mikroprosesor
253	Kerusakan Sistem Operasi	Kerusakan Sistem Operasi
58	IC Charging	IC Charging
208	IC Power	IC Mikroprosesor
258	Kerusakan Sistem Operasi	Kerusakan Sistem Operasi
261	Kerusakan Sistem Operasi	Kerusakan Sistem Operasi
28	Baterai Drop	Baterai Drop
159	IC PA (Power Amplifier)	IC PA (Power Amplifier)
70	IC eMMC	IC eMMC
22	Baterai Drop	IC Mikroprosesor
230	Kerusakan Sistem Operasi	Kerusakan Sistem Operasi
91	IC eMMC	IC eMMC
67	IC Charging	IC Charging
247	Kerusakan Sistem Operasi	Kerusakan Sistem Operasi
219	IC Power	IC Power
111	IC eMMC	IC eMMC
221	IC Power	IC Power
Index	Label Sebenarnya	Label Prediksi
46	IC Charging	IC Charging
108	IC eMMC	IC eMMC
145	IC Mikroprosesor	IC Mikroprosesor
176	IC PA (Power Amplifier)	IC PA (Power Amplifier)

234	Kerusakan Sistem Operasi	Kerusakan Sistem Operasi
182	IC PA (Power Amplifier)	IC PA (Power Amplifier)
10	Baterai Drop	Baterai Drop
99	IC eMMC	IC eMMC
156	IC PA (Power Amplifier)	IC PA (Power Amplifier)
127	IC Mikroprosesor	IC Mikroprosesor
18	Baterai Drop	Baterai Drop
246	Kerusakan Sistem Operasi	Kerusakan Sistem Operasi
31	Baterai Drop	Baterai Drop
226	IC Power	IC Power
121	IC Mikroprosesor	IC Mikroprosesor
55	IC Charging	IC Charging
186	IC Power	IC Power
214	IC Power	IC Power
224	IC Power	IC Power
20	Baterai Drop	Baterai Drop

Dari data diatas, terdapat 50 sampel data yang dijadikan sebagai data uji diatas, terdapat 4 data yang memiliki hasil prediksi yang salah, oleh karena itu, *Confussion Matrix* akan dihitung untuk masing masing label kelas, scikit-learn menyediakan fitur untuk mempercepat perhitungan akurasi tersebut, fitur ini akan menghasilkan otomatis jumlah *Precision*, *Recall*, *F1 Score* dan *Accuracy* dari model terkait, berikut implementasinya

```
# Menghitung matriks kebingungan dari hasil prediksi
matrix = confusion_matrix(y_test, y_pred)

# Visualisasi matriks kebingungan menggunakan heatmap
plt.figure(figsize=(10, 7)) # Mengatur ukuran gambar
sns.heatmap(matrix, annot=True, fmt='d', cmap='Reds', cbar=False)
plt.title('Confusion Matrix dengan LR') # Judul grafik
plt.xlabel('Predicted Label') # Label sumbu X
plt.ylabel('True Label') # Label sumbu Y
plt.show() # Menampilkan grafik

# Mencetak laporan klasifikasi untuk melihat presisi, recall, dan f1-score
print("Laporan Klasifikasi:\n", classification_report(y_test, y_pred, digits=3))

# Menghitung metrik presisi, recall, dan F1 score dengan rata-rata 'weighted'
tree_precision = precision_score(y_test, y_pred, average='weighted')
tree_recall = recall_score(y_test, y_pred, average='weighted')
tree_f1 = f1_score(y_test, y_pred, average='weighted')

# Mencetak nilai presisi, recall, dan F1 score
print(f"Presisi Terbobot: {tree_precision:.3f}")
print(f"Recall Terbobot: {tree_recall:.3f}")
print(f"F1 Score Terbobot: {tree_f1:.3f}")
```

Gambar 4. 28 Implementasi menghitung Performa Model

Dalam kode diatas, *Confussion Matrix* dihitung menggunakan fungsi *confussion matrix* dari *scikit-learn*. Kode diatas mencetak laporan klasifikasi yang memberikan detail terkait jumlah *Precision*, *Recall*, dan *F1 Score* untuk setiap kelas dalam dataset. Berikut merupakan rincian *Precision*, *Recall*, dan *F1 score* dari algoritma dalam memprediksi menggunakan data uji :

```
Laporan Klasifikasi:
```

	precision	recall	f1-score	support
Baterai Drop	1.000	0.750	0.857	8
IC Charging	1.000	0.833	0.909	6
IC Mikroprosesor	0.667	1.000	0.800	8
IC PA (Power Amplifier)	1.000	1.000	1.000	6
IC Power	1.000	0.889	0.941	9
IC eMMC	1.000	1.000	1.000	9
Kerusakan Sistem Operasi	1.000	1.000	1.000	7
accuracy			0.925	53
macro avg	0.952	0.925	0.930	53
weighted avg	0.950	0.925	0.928	53

Presisi Terbobot: 0.950
 Recall Terbobot: 0.925
 F1 Score Terbobot: 0.928

Gambar 4. 29 Precision, Recall, dan F1 Score

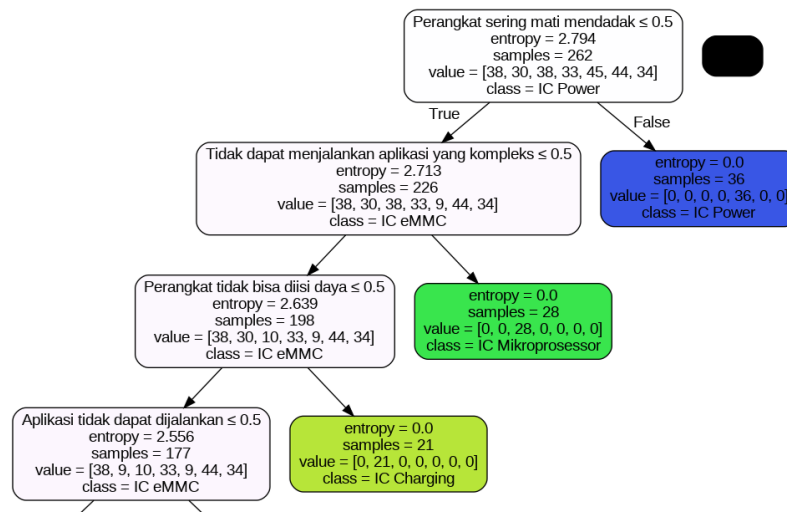
Dari hasil klasifikasi diatas, kesimpulan mengenai performa model yang sudah dilatih adalah sebagai berikut :

1. Model yang dilatih memiliki akurasi 0.925 atau 92,5%, hal ini menunjukkan model memiliki performa umum yang baik dalam memprediksi keseluruhan label di dataset
2. Jumlah presisi, recall, dan F1 score terbobot masing masing adalah 0.950 (95%), 0.925 (92,5%), dan 0.928 (92,8%) yang menegaskan bahwa model memiliki performa baik secara keseluruhan dengan menghitung jumlah prediksi benar setiap sampel untuk masing masing kelas dalam perhitungan.

Kesimpulan ini menunjukkan bahwa model dapat mengklasifikasikan kerusakan HP yang ada dengan benar dengan tingkat kesalahan yang minimal, setelah ini, model siap untuk diintegrasikan kedalam aplikasi sistem.

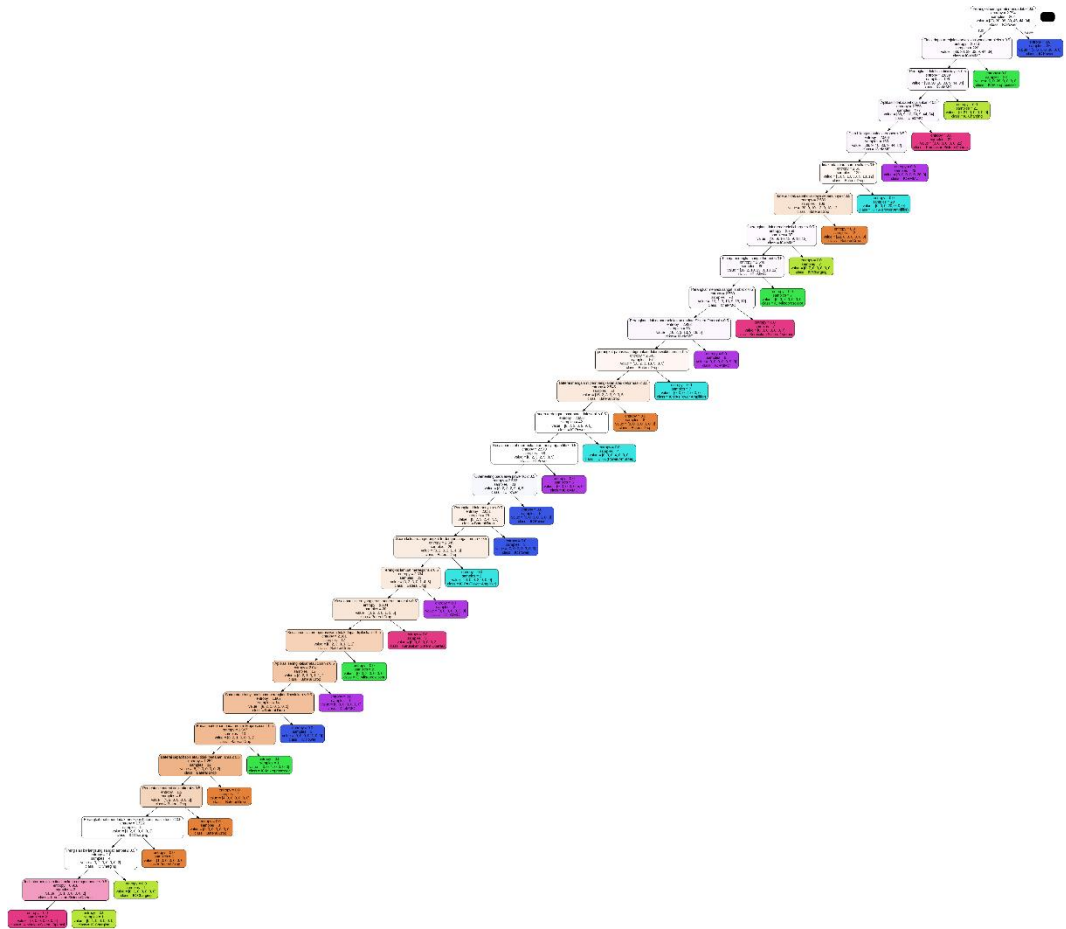
3. Hasil visualisasi Pohon Keputusan

Pohon keputusan yang dihasilkan dari hasil latih Algoritma C.45 dapat dilihat dalam gambar dibawah ini :



Gambar 4. 30 Gambar pohon keputusan yang dihasilkan

Pada node atas, yang merupakan titik awal dari pohon keputusan, atribut **Perangkat Sering Mati Mendadak** mendapatkan entropy tertinggi, dan kelas mayoritas dari atribut tersebut adalah **Kerusakan IC Power**, cabang kiri merupakan percabangan node berikutnya yang mana algoritma memilih atribut **Tidak dapat menjalankan aplikasi yang kompleks**. Terlihat dengan menggunakan algoritma c.45 yang telah dilatih, model dapat menghasilkan pohon keputusan lebih cepat, berikut merupakan gambar keseluruhan dari pohon keputusan terkait.



Gambar 4. 31 Gambar keseluruhan Pohon Keputusan

4.4. Kelebihan dan Kekurangan Program

Adapun kelebihan dan kekurangan dari sistem prediksi kerusakan HP menggunakan algoritma C.45 yang telah dibangun adalah sebagai berikut :

1. Kelebihan Sistem

Adapun beberapa kelebihan sistem yang telah dibangun adalah sebagai berikut :

- a. Dapat menjadikan prediksi kerusakan HP lebih efektif dan efisien dikarenakan sudah diintegrasikan dengan Model Prediksi algoritma C.45 yang dapat memberikan hasil prediksi kerusakan berdasarkan data atribut dengan cepat

- b. Sistem yang dibuat mudah untuk digunakan dan dapat dijalankan di berbagai jenis laptop dengan spesifikasi minimum

2. Kekurangan Sistem

Adapun kekurangan dari sistem yang telah dibangun adalah sebagai berikut :

- a. Sistem tidak memiliki sistem backup data secara otomatis
- b. Model algoritma C.45 dalam sistem masih terintegrasi didalam sistem dan belum secara *cloud* sehingga jika terjadi perubahan atau peningkatan model Algoritma C.45, model harus dimasukkan ulang secara manual kedalam struktur data
- c. Belum mencakup semua jenis kerusakan HP

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil pembahasan dan uji coba yang dilakukan, dapat ditarik kesimpulan sebagai berikut :

1. Pembangunan sistem pendukung keputusan (*Decision Support System*) Memprediksi Kerusakan HP di Pioneer Service Menggunakan Algoritma C.45 Berbasis Web dapat dilakukan dengan mengintegrasikan model algoritma C.45 yang dilatih menggunakan lingkungan pengembangan *Python* di Google Collab. Integrasinya sendiri menggunakan API yang dibangun dengan *Flask*.
2. Algoritma C.45 yang dilatih mendapat akurasi yang baik dan sempurna dan dapat memprediksi kerusakan hp berdasarkan data atribut dan kerusakan yang diambil dari data historis penjualan di *Pioneer Service*
3. Aplikasi ini bertujuan untuk membantu teknisi dalam memprediksi kerusakan HP lebih cepat

5.2. Saran

Adapun yang menjadi saran untuk penelitian kedepannya adalah sebagai berikut :

1. Pada penelitian selanjutnya, pengembangan aplikasi dapat dikembangkan lebih lanjut untuk memprediksi lebih banyak kasus kerusakan HP, sehingga dapat memberikan solusi yang lebih baik
2. Pada penelitian selanjutnya, dapat menambah lebih banyak lagi kriteria atribut kerusakan dan daftar kerusakan. Hal ini bertujuan untuk menghasilkan aplikasi dengan konsistensi dan akurasi yang lebih meningkat lagi.

3. Penelitian selanjutnya dapat mengeksplorasi integrasi aplikasi dengan cloud agar otomatisasi model pembaruan Algoritma C4.5 dengan data baru lebih baik.

DAFTAR PUSTAKA

- Afifah, L. (2023). *Evaluasi Model Machine Learning: Train/Test Split - IlmudataPy*. <https://ilmudatapy.com/evaluasi-model-machine-learning-dengan-train-test-split/>
- Azwanti, N. (2019). *Analisa Algoritma C.45 Untuk Memprediksi Penjualan Motor Pada PT.Capella Dinamik Nusantara Cabang Muka Kuning*. Dalam *Jurnal Ilmiah Ilmu Komputer* (Vol. 13, Nomor 1).
- Binus Student Corner. (2022). *Teknik pre-processing dan classification dalam data science – Master of Industrial Engineering*. Binus University. <https://mie.binus.ac.id/2022/08/26/teknik-pre-processing-dan-classification-dalam-data-science/>
- Bisa AI Academy. (2023). *Bisa AI Academy - Course: Machine Learning Dengan Scikit Learn Python*. <https://bisa.ai/course/detail/MzU3/1>
- Dionysia. (2021). *Apa itu Pencilan? Pengertian dan Cara Menemukan Pencilan dalam Statistik*. <https://www.freecodecamp.org/news/what-is-an-outlier-definition-and-how-to-find-outliers-in-statistics/>
- Hafizan, H., & Putri, A. N. (2020). *Penerapan Metode Klasifikasi Decision Tree Pada Status Gizi Balita Di Kabupaten Simalungun* (Vol. 1, Nomor 2).
- H.M., A. (2019). *Penerapan Metode C45 Dalam Memprediksi Pola Pembelian Bahan Campuran Olahan Karet (Studi Kasus : PT.Anugrah Sibolga Lestari)*. <http://repository.potensi-utama.ac.id/jspui/jspui/handle/123456789/3504>
- Khairani Aritonang, T., Pendidikan Matematika-, P., & Ilmu Tarbiyah dan Keguruan, F. (2022). *Pengenalan Algoritma Pada Pembelajaran Pemrograman Komputer*.
- Normawati, D., & Prayogi, S. A. (2021). Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter. Dalam *Jurnal Sains Komputer & Informatika (J-SAKTI)* (Vol. 5, Nomor 2).
- Nur Arif, S., Syahril, M., Kusnasari, S., & Winata, H. (2021). Sistem Pakar Mendiagnosa Kerusakan Handphone Oppo Dengan Menggunakan Teorema Bayer. *J-Sisco Tech*. <https://doi.org/https://doi.org/10.53513/jsk.v4i1.2626>

- Pratama, A. (2020). *HTML Uncover - Panduan Belajar HTML Untuk Pemula*.
www.duniailkom.com
- Pribadi, D., Amegia Saputra, R., Maulana Hudin, J., & Gunawan. (2020). *Sistem Pendukung Keputusan*.
- Rahman, W., Saudin, L., & Sri Wahyuni, N. (2022). *Bahan Ajar Sistem Informasi Manajemen* www.penerbitwidina.com
- Ramadhan, M. (2021). *Metode Penelitian*.
https://books.google.co.id/books?id=Ntw_EAAAQBAJ&printsec=frontcover&hl=id&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false
- Rumbaugh, James., Jacobson, Ivar., & Booch, Grady. (2021). *The unified modeling language reference manual*. Addison-Wesley.
- Sarwandi, T.Sianturi, L., Astuty Hasibuan, N., Iwan Sudipa, I. G., Syahrizal, M., Alwendi, Mesran, Muqimuddin, Dwi Meilani, B., Wiwik Sri Rahayu Ginanta, N. L., & Fajar Israwan, LM. (2023). *Sistem Pendukung Keputusan*.
https://books.google.co.id/books?hl=id&lr=&id=qmm-EAAAQBAJ&oi=fnd&pg=PP1&dq=sarwandi+et+all+sistem+pendukung+keputusan&ots=NOVvGmGYkg&sig=QiGup_fWGuKgCJJN-41pU9iUnmg&redir_esc=y#v=onepage&q&f=false
- Setiawan, R. (2021). *Memahami Class Diagram Lebih Baik - Dicoding Blog*.
<https://www.dicoding.com/blog/memahami-class-diagram-lebih-baik/>
- Setiawan, S. (2020). *Membicarakan Precision, Recall, dan F1-Score*. Medium.com. <https://stevkarta.medium.com/membicarakan-precision-recall-dan-f1-score-e96d81910354>
- Sudewo, R. (2022). *Library Pendas: Pengertian, Manfaat, Keunggulan | Danacita*.
<https://danacita.co.id/blog/pengertian-manfaat-dan-keunggulan-library-panda/>
- Suyanto. (2021). *Artificial Intelligence Edisi 3*.
- Yani, D., & Tanjung, H. (2021). Optimalisasi Algoritma C4.5 untuk Prediksi Kerusakan Mesin ATM Optimization of the C4.5 Algorithm for Predicting ATM Machine Damage. *12. InfoSys Journal*, 6, 12–21.

LAMPIRAN – PENGUJIAN PROGRAM

Uji coba terhadap sistem bertujuan untuk memastikan bahwa sistem aplikasi yang dibuat sudah berada dalam kondisi siap pakai. Instrumen yang digunakan untuk melakukan pengujian yaitu menggunakan spesifikasi sebagai berikut :

1. Satu unit laptop atau PC dengan spesifikasi sebagai berikut :
 - a. Processor Intel Core I3
 - b. RAM 2 GB
 - c. Hard Drive 500 GB
2. Perangkat lunak dengan spesifikasi sebagai berikut :
 - a. XAMPP
 - b. MySQL Server
 - c. Visual Studio
 - d. Google Collab
 - e. Python dengan terinstall library dan package berikut :
 - *Pandas*
 - *Scikit-Learn*
 - *Flask*
 - *Decision Tree Classifier*
 - *Matplotlib Pyplot*

Skenario pengujian sistem bertujuan untuk melakukan pengujian terhadap hasil dan membandingkan dengan konsep penyelesaian masalah yang telah dirancang sebelumnya, pengujian dilakukan dengan menggunakan *localhost*.

1. Pengujian Login

Tabel 4. 8 Tabel Pengujian Sistem Login

Data Masukan	Hasil yang diharapkan	Pengamatan	Kesimpulan
<i>Username</i> dan <i>Password</i> yang telah ditentukan	Masuk ke tampilan <i>Dashboard</i>	<i>Login</i> Berhasil	[✓] Valid [] Invalid
<i>Username</i> dan <i>Password</i> kosong atau salah	Akan menampilkan “Login Gagal”	<i>Login</i> Gagal	[✓] Valid [] Invalid

2. Pengujian Dashboard

Tabel 4. 9 Tampilan Pengujian Dashboard

Data Masukan	Hasil yang diharapkan	Pengamatan	Kesimpulan
<i>Dashboard</i>	Akan menampilkan menu utama	Terdapat beberapa menu yang dapat diakses	[✓] Valid [] Invalid

3. Pengujian Data Atribut

Tabel 4. 10 Tampilan Pengujian Data Atribut

Data Masukan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Tambah Data Atribut	Menginput dan menampilkan data atribut	Dapat menambah data dan menampilkan data	[✓] Valid [] Invalid
Edit Data Atribut	Mengubah data atribut	Dapat mengubah data atribut	[✓] Valid [] Invalid
Hapus Data Atribut	Menghapus data atribut	Dapat menghapus data atribut	[✓] Valid [] Invalid

4. Pengujian Data Kerusakan

Tabel 4. 11 Tampilan Pengujian Data Kerusakan

Data Masukan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Tambah Data Kerusakan	Menginput dan menampilkan data kerusakan	Dapat menambah data dan menampilkan data	[✓] Valid [] Invalid
Edit Data Kerusakan	Mengubah data kerusakan	Dapat mengubah data kerusakan	[✓] Valid [] Invalid

Hapus Data	Menghapus data	Dapat menghapus	<input checked="" type="checkbox"/> Valid
Kerusakan	kerusakan	data kerusakan	<input type="checkbox"/> Invalid

5. Pengujian Data Konsultasi dan Prediksi

Tabel 4. 12 Tabel Pengujian Data Konsultasi dan Prediksi

Data Masukan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Tambah Data Konsultasi	Menginput dan menampilkan data konsultasi	Dapat menambah data dan menampilkan data	<input checked="" type="checkbox"/> Valid <input type="checkbox"/> Invalid
Prediksi Data	Menghasilkan prediksi berdasarkan atribut yang dipilih	Berfungsi sesuai diharapkan	<input checked="" type="checkbox"/> Valid <input type="checkbox"/> Invalid
Detail Data Konsultasi	Menampilkan rincian konsultasi	Berfungsi sesuai diharapkan	<input checked="" type="checkbox"/> Valid <input type="checkbox"/> Invalid
Hapus Data Konsultasi	Menghapus data konsultasi	Dapat menghapus data konsultasi	<input checked="" type="checkbox"/> Valid <input type="checkbox"/> Invalid

6. Pengujian Data User

Tabel 4. 13 Tabel Pengujian Sistem Data User

Data Masukan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Tambah Data User	Menginput dan menampilkan data user	Dapat menambah data dan menampilkan data	<input checked="" type="checkbox"/> Valid <input type="checkbox"/> Invalid
Edit Data User	Mengubah data user	Dapat mengubah data user	<input checked="" type="checkbox"/> Valid <input type="checkbox"/> Invalid
Hapus Data User	Menghapus data user	Dapat menghapus data user	<input checked="" type="checkbox"/> Valid <input type="checkbox"/> Invalid