

**SISTEM SMART CALLING DI WARUNG TELEPON ASRAMA
MA'HADUL QUR'AN PONDOK PESANTREN SALAFIYAH
SYAFI'YAH SUKOREJO**

SKRIPSI



Oleh:
BADRUS SHOLEH
NPM : 2021501009

**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS IBRAHIMI
SITUBONDO
2025**

**SISTEM SMART CALLING DI WARUNG TELEPON ASRAMA
MA'HADUL QUR'AN PONDOK PESANTREN SALAFIYAH
SYAFI'YAH SUKOREJO**

SKRIPSI



Oleh:

BADRUS SHOLEH

NPM : 2021501009

**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS IBRAHIMY
SITUBONDO**

2025

HALAMAN JUDUL**SISTEM *SMART CALLING* DI WARUNG TELEPON ASRAMA
MA'HADUL QUR'AN PONDOK PESANTREN SALAFIYAH
SYAFI'YAH SUKOREJO****SKRIPSI**

Diajukan untuk Memenuhi Salah Satu Persyaratan dalam Menyelesaikan Program Sarjana (S-1) Pada Program Studi Ilmu Komputer Fakultas Sains dan Teknologi

Universitas Ibrahimy

Oleh:

BADRUS SHOLEH

2021501009

**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS IBRAHIMY
SITUBONDO
2025**

PERNYATAAN KEASLIAN TULISAN

Saya Yang Bertanda tangan di bawah ini:

Nama : Badrus Sholeh
NPM/NIRM : 2021501009
Program Studi : Ilmu Komputer
Fakultas : Fakultas Sains dan Teknologi

Menyatakan dengan sebenarnya, bahwa tugas akhir/skripsi ini secara keseluruhan adalah hasil penelitian atau karya saya sendiri, kecuali pada bagian-bagian yang dirujuk sebagai sumber referensi dan disebutkan dalam daftar Pustaka. Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa tugas akhir/skripsi ini hasil plagiasi, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Situbondo, 30 Juli 2025

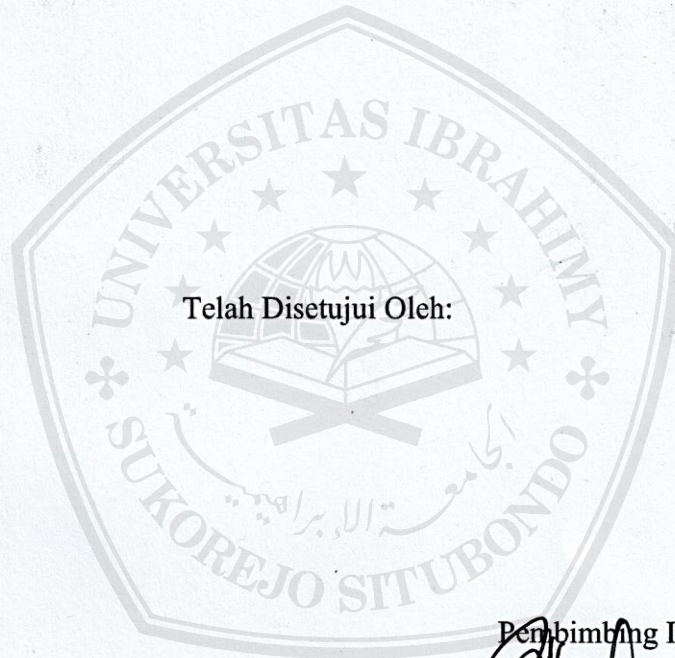
Saya Yang Menyatakan,



Badrus Sholeh

PERSETUJUAN PEMBIMBING

Nama : Badrus Sholeh
NPM/NIRM : 2021501009
Judul : Sistem *Smart Calling* Di Warung Telepon Asrama
Ma'hadul Qur'an Pondok Pesantren Salafiyah Syafi'iyah
Sukorejo



Telah Disetujui Oleh:

Pembimbing I

Adi Susanto, M.Kom
NIDN:708079204

Pembimbing II

Zaehol Fatah, M.Kom
NIDN:0715057801

PENGESAHAN

SKRIPSI

**SISTEM SMART CALLING DI WARUNG TELEPON ASRAMA
MA'HADUL QUR'AN PONDOK PESANTREN SALAFIYAH
SYAFI'YAH SUKOREJO**

BADRUS SHOLEH
NPM : 2021501009

Telah dipertahankan di depan dewan penguji Sidang/Munaqasyah Skripsi pada
Kamis 14 Agustus 2025 sebagai salah satu syarat memperoleh gelar Sarjana
(S.Kom) pada Fakultas Sains dan Teknologi Universitas Ibrahimi

Tim Penguji,

Ketua Sidang,

Dr. Ach. Khumaidi, M.P.
NIDN: 0722049001

Sekretaris Sidang,

Khairur Rosiqin, S.Kom
NIDN: -

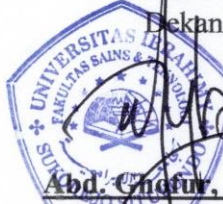
Penguji I,

Farihin Lazim, M. Tr.T
NIDN: 0711099201

Penguji II,

A. Hamdani, M.Kom
NIDN: 0730118806

Mengetahui
Bekan,



Abd. Chafur, M. Kom
NIDN: 0711088303

MOTTO

JANGAN PERNAH MENYERAH, KITA PASTI BISA !



HALAMAN PERSEMBAHAN

Dengan segenap kerendahan hati dan rasa syukur yang dalam, karya sederhana ini penulis persembahkan kepada:

1. Ayahanda tercinta, Aba Saiful Bahri, sosok yang selalu menjadi teladan dalam kesabaran, yang tak pernah lelah memberi nasihat kehidupan. Ibunda tersayang, Alm. Umi Nining Wasiah, dan Mama Ririn Darianti, dua sosok yang dengan kasih tulus dan doanya, menjadi pelita dalam gelap, penguat di kala rapuh. Sungguh tiada kata yang mampu melukiskan ketulusan cinta kalian.
2. Keluargaku yang penuh cinta, Om Agos, Kakek, Nenek, dan seluruh keluarga besar di Bondowoso dan Jember, yang telah memberi tawa, doa, dan semangat yang tak pernah putus dalam setiap detak perjuangan ini.
3. Adik-adikku tersayang, Zakiatus Sholihah, Alfiatul Hasanah, dan Alm. Fatiyatur Rohmah, kalian adalah alasan untuk terus menjadi teladan, cahaya kecil yang menginspirasi dalam diam.
4. Seluruh sahabat, teman seperjuangan, dan keluarga pengabdian, yang pernah hadir di kehidupan ini. Teruntuk Postelkom, LPMI, Al-khidmah, Pendidikan MQ, Iksass Bondowoso, *Robot Station*, SSDC, dan berbagai organisasi yang pernah menjadi tempat singgah, Terima kasih, dari kalian aku belajar arti loyalitas, dan kerja keras. Di sanalah jati diri ini ditempa, karakter dibentuk, dan ilmu pengetahuan penulis dibangun.

Dan tak lupa, kepada semua jiwa dan tangan yang terlibat dalam perjalanan panjang ini, baik yang terlihat maupun tersirat, yang pernah menjadi bagian dari proses tumbuhku, semoga Allah mencatat setiap kontribusi kalian dalam lembaran hidupku ini.

Skripsi ini adalah bukti cinta, perjuangan, pengorbanan, dan bukti bahwa setiap luka pun bisa berubah menjadi ilmu pengetahuan. Semoga bermanfaat untuk peradaban, dan dikenang sepanjang zaman.

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT, Tuhan Yang Maha Esa, atas limpahan rahmat, taufik, dan hidayah-Nya sehingga penulis dapat menyelesaikan penyusunan skripsi yang berjudul: “Sistem *Smart Calling* Di Warung Telepon Asrama Ma’hadul Qur’an Pondok Pesantren Salafiyah Syafi’iyah Sukorejo” dengan baik dan lancar tanpa kendala yang berarti.

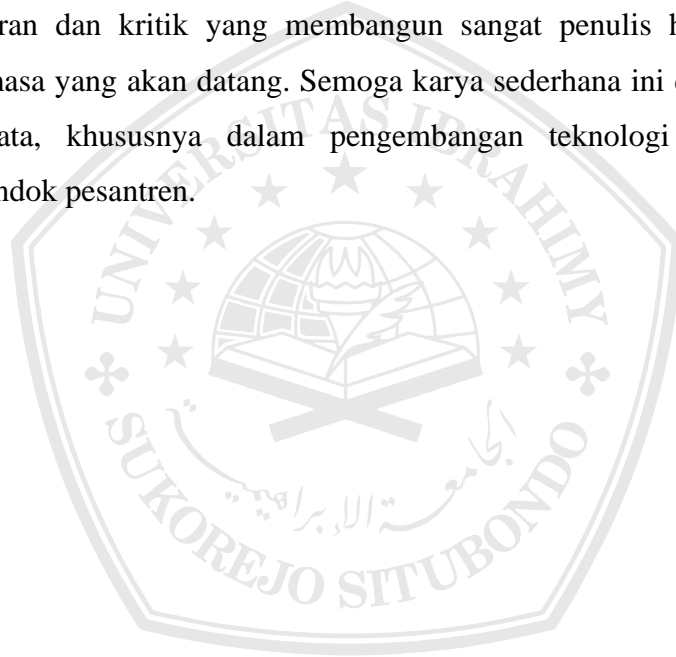
Skripsi ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer pada Program Studi Ilmu Komputer, Fakultas Sains dan Teknologi Universitas Ibrahimi Situbondo. Penelitian ini lahir dari keprihatinan terhadap proses pencatatan layanan telepon manual di lingkungan pesantren, serta keinginan untuk mengintegrasikan teknologi sebagai solusi yang cerdas, efisien, dan berorientasi pada kemudahan pelayanan santri.

Dalam perjalanan penyusunan skripsi ini, penulis banyak menerima arahan, motivasi, dan bantuan dari berbagai pihak. Untuk itu, dengan penuh kerendahan hati, penulis menyampaikan terima kasih dan penghargaan yang sebesar-besarnya kepada:

1. KHR. Ach. Azaim Ibrahimi, M.H, selaku Pengasuh Pondok Pesantren Salafiyah Syafi’iyah Sukorejo Situbondo, atas bimbingan spiritual dan teladan kepemimpinan.
2. KH. Ach. Fadlail, S.H., M.H, selaku Rektor Universitas Ibrahimi Situbondo, yang telah memfasilitasi proses pendidikan penulis selama masa studi.
3. Abd. Ghofur, M.Kom, selaku Dekan Fakultas Sains dan Teknologi, yang selalu mendorong semangat inovasi mahasiswa.
4. Farihin Lazim, M.Tr.T., selaku Ketua Program Studi Ilmu Komputer, yang senantiasa memberikan motivasi akademik kepada mahasiswa bimbingannya.
5. Adi Susanto, M.Kom dan Zaehol Fatah, M.Kom, selaku dosen pembimbing, yang dengan sabar memberikan arahan, evaluasi, dan bimbingan ilmiah hingga skripsi ini dapat terselesaikan.

6. Seluruh dosen Fakultas Sains dan Teknologi Universitas Ibrahimi yang telah membagikan ilmu, pengalaman, dan inspirasi selama proses perkuliahan hingga masa tugas akhir.
7. Seluruh staf akademik dan administrasi yang telah membantu kelancaran berbagai keperluan administrasi skripsi dan perkuliahan.
8. Postelkom MQ Putra, sebagai tempat penelitian sekaligus mitra uji sistem yang telah memberikan dukungan penuh baik dalam bentuk data, fasilitas, maupun masukan sistem secara langsung.

Penulis menyadari bahwa skripsi ini masih memiliki keterbatasan. Oleh karena itu, saran dan kritik yang membangun sangat penulis harapkan demi perbaikan di masa yang akan datang. Semoga karya sederhana ini dapat memberi kontribusi nyata, khususnya dalam pengembangan teknologi informasi di lingkungan pondok pesantren.



DAFTAR ISI

HALAMAN JUDUL	ii
PERNYATAAN KEASLIAN TULISAN.....	iii
PERSETUJUAN PEMBIMBING	iv
PENGESAHAN	v
MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xiv
DAFTAR SEGMENT PROGRAM	xv
DAFTAR LAMPIRAN.....	xvi
ABSTRAK.....	xvii
ABSTRACT	xviii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah	3
1.3 Rumusan Masalah	4
1.4 Batasan Masalah.....	4
1.5 Tujuan Penelitian	4
1.6 Manfaat Penelitian	4
1.7 Metode Penelitian.....	5
1.7.1. Jenis Penelitian.....	5
1.7.2. Teknik Pengumpulan Data.....	5
1.7.3. Metode Pengembangan Sistem	6
1.8 Sistematika Pembahasan	9
BAB II TINJAUAN PUSTAKA	11

2.1	Penelitian Terdahulu	11
2.2	Landasan Teori.....	15
2.3	Pemodelan	16
2.4	Perangkat Lunak yang Digunakan	18
BAB III ANALISIS DAN PERANCANGAN SISTEM		22
3.1	Gambaran Umum Obyek Penelitian	22
3.1.1	Keadaan Sistem yang Berjalan	22
3.1.2	Kelebihan Sistem	22
3.1.3	Kelemahan Sistem.....	23
3.2	Alur Proses	23
3.2.1	Identifikasi dan Analisis Proses Bisnis	23
3.2.2	Identifikasi dan Analisis Kebutuhan	26
3.2.3	Identifikasi dan Analisis Alternatif Solusi	31
3.3	Desain Sistem.....	32
3.3.1	Diagram Blok Sistem	32
3.3.2	Desain Proses	32
3.3.3	Identifikasi dan Desain <i>Database</i>	36
3.3.4	Desain <i>User Interface</i>	38
BAB IV IMPLEMENTASI SISTEM.....		42
4.1	Kontruksi Sistem.....	42
4.1.1.	Kebutuhan Sistem	42
4.1.2.	Instalasi Sistem	43
4.1.3.	Segmen Program	47
4.2	Skenario Pengujian.....	55
4.3	Pengujian.....	57
4.3.1.	Cara Kerja Sistem	57
4.3.2.	Hasil Pengujian	60
4.4	<i>Maintenace</i>	62

BAB V PENUTUP	64
5.1 Kesimpulan	64
5.2 Saran.....	64
DAFTAR PUSTAKA	65
LAMPIRAN-LAMPIRAN.....	68
CURRICULUM VITAE	75



DAFTAR GAMBAR

Gambar 1.1 Metode dalam Waterfall [12].	7
Gambar 3. 1 Diagram Blok Sistem	32
Gambar 3. 2 <i>Flowchat</i> Diagram Alir <i>User</i>	33
Gambar 3. 3 <i>Flowchat</i> Diagram Alir Admin	34
Gambar 3. 4 Comtext Diagram	35
Gambar 3. 5 Data Flow Diagram (DFD)	36
Gambar 3. 6 <i>Conseptua Data Model (CDM)</i>	38
Gambar 3. 7 <i>Phisical Data Model (PDM)</i>	38
Gambar 3. 8 Ilustrasi Desain <i>User</i>	39
Gambar 3. 9 Ilustrasi Desain Data Transaksi	39
Gambar 3. 10 Ilustrasi Desain Data <i>History</i>	40
Gambar 3. 11 Ilustrasi Desain Data Pelanggan	40
Gambar 4. 1 Tampilan Jendela User	58
Gambar 4. 2 Tampilan Jendela Login Petugas/admin	58
Gambar 4. 3 Tampilan Jendela Transaksi	59
Gambar 4. 4 Tampilan Jendela History	59
Gambar 4. 5 Tampilan Jendela Pelanggan	59
Gambar 4. 6 Jendela Pengaturan Petugas dan Pengaturan Hak Akses	60

DAFTAR TABEL

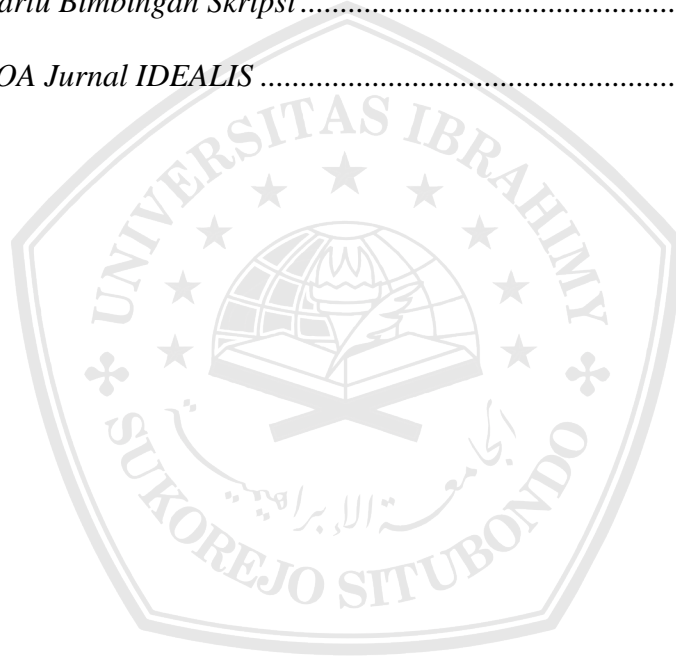
Tabel 2. 1 Simbol-simbol <i>Flowchart</i>	17
Tabel 3. 1 Identifikasi Alternatif Solusi	31
Tabel 3. 2 Tabel Database Transaksi	36
Tabel 3. 3 <i>Tabel Database History</i> Transaksi.....	37
Tabel 3. 4 <i>Tabel Database History</i> Transaksi.....	37
Tabel 4.1 Kebutuhan Perangkat Keras Sistem.....	42
Tabel 4.2 Kebutuhan Perangkat Keras Sistem.....	43
Tabel 4.3 Skenario Pengujian Sistem Smart Calling	56
Tabel 4.4 Hasil Pengujian Sistem Smart Calling.....	61

DAFTAR SEGMENT PROGRAM

Segmen Program 4. 1 Buat Database Otomatis	44
Segmen Program 4. 2 Kirim data ke Server (admin)	45
Segmen Program 4. 3 Kirim ulang data tertunda ke Server (admin)	46
Segmen Program 4. 4 Timer dan Pengaturan Tarif Otomatis	47
Segmen Program 4. 5 Automasi Klik ikon Panggilan	48
Segmen Program 4. 6 Automasi Mengetik Nomor Telepon	48
Segmen Program 4. 7 Pendeteksi Otomatis Tombol Akhiri Panggilan	48
Segmen Program 4. 8 Simpan data Sementara saat gagal terhubung ke server....	49
Segmen Program 4. 9 Validasi NIS sebelum Panggilan	49
Segmen Program 4. 10 Refresh otomatis di Menu Transaksi	50
Segmen Program 4. 11 API Penerima Transaksi dari User	51
Segmen Program 4. 12 Pemindahan Transaksi ke History	51
Segmen Program 4. 13 Pemindahan Transaksi ke History	53
Segmen Program 4. 14 Verifikasi Database Pelanggan	54
Segmen program 4. 15 Struktur Tabel Info	54
Segmen program 4. 16 Pengaturan Hak Akses Petugas	55

DAFTAR LAMPIRAN

Lampiran 1. Bukti Telah Melaksanakan Penelitian	68
<i>Lampiran 2. Transkrip Wawancara dengan petugas</i>	<i>69</i>
<i>Lampiran 3. Transkrip Wawancara dengan Santri</i>	<i>70</i>
<i>Lampiran 4. Buku Hutang piutang & Pencatatan Manual Wartel.....</i>	<i>71</i>
<i>Lampiran 5. Surat Keterangan Hasil Pemeriksaan Plagiasi.....</i>	<i>72</i>
<i>Lampiran 6. Kartu Bimbingan Skripsi</i>	<i>73</i>
<i>Lampiran 7. LOA Jurnal IDEALIS</i>	<i>74</i>



ABSTRAK

Badrus Sholeh, 2025. **Sistem *Smart Calling* Di Warung Telepon Asrama Ma'hadul Qur'an Pondok Pesantren Salafiyah Syafi'iyah Sukorejo**, Skripsi. Program Studi Ilmu Komputer, Universitas Ibrahimi. Pembimbing; (I) Adi Susanto, M.Kom, (II) Zaehol Fatah, M.Kom.

Layanan komunikasi santri di Asrama Ma'hadul Qur'an masih menggunakan pencatatan transaksi manual, kurang efisien, dan tidak terarsip dengan baik, yang berakibat pada rawannya kesalahan data. Penelitian ini bertujuan untuk merancang dan mengimplementasikan Sistem *Smart Calling* berbasis Python yang dapat mengotomatisasi layanan wartel di asrama Ma'hadul Qur'an. Sistem ini terdiri dari dua bagian: sisi *user* (digunakan oleh santri) dan sisi admin/petugas. Santri cukup menginput NIS dan nomor tujuan untuk melakukan panggilan melalui WhatsApp Desktop yang terintegrasi secara otomatis. Sistem mencatat durasi panggilan dan menghitung tarif sesuai jenis panggilan (audio/video call). Sementara itu, petugas dapat mengelola transaksi, histori, data pelanggan, dan pengaturan sistem melalui antarmuka grafis. Pengembangan sistem menggunakan metode *Waterfall*, dengan tahapan pengumpulan kebutuhan, analisis, perancangan, implementasi, dan pengujian. Pengujian dilakukan dengan metode *black box* dan menunjukkan bahwa seluruh fungsi berjalan sesuai dengan kebutuhan. Sistem ini meningkatkan efisiensi operasional wartel, mengurangi beban administrasi manual, serta meminimalkan risiko kesalahan pencatatan. Dengan sistem ini, layanan panggilan santri menjadi lebih modern, terdokumentasi, dan siap dikembangkan ke skala yang lebih luas di lingkungan pesantren maupun lembaga serupa.

Kata kunci: *Smart Calling*, Python, *WhatsApp Desktop*, *Automasi Panggilan*, *Sistem Wartel Digital*

ABSTRACT

Badrus Sholeh, 2025. **Smart Calling System at the Dormitory Telephone Booth of Ma'hadul Qur'an, Salafiyah Syafi'iyah Sukorejo Islamic Boarding School**, Undergraduate Thesis. Computer Science Study Program, Ibrahimi University. Advisors: (I) Adi Susanto, M.Kom, (II) Zaehol Fatah, M.Kom.

The communication service for students at the Ma'hadul Qur'an dormitory still relies on manual transaction recording, which is inefficient and poorly archived, resulting in a high risk of data errors. This study aims to design and implement a Python-based Smart Calling System that automates telephone services (wartel) in the Ma'hadul Qur'an dormitory. The system consists of two main components: the user side (used by students) and the admin/staff side. Students simply input their student ID (NIS) and the destination number to initiate a call via WhatsApp Desktop, which is integrated automatically. The system records call duration and calculates the fee based on the type of call (audio or video). Meanwhile, staff can manage transactions, history, customer data, and system settings through a graphical user interface. The system was developed using the Waterfall method, which includes requirements gathering, analysis, design, implementation, and testing. The testing phase applied the black-box method and confirmed that all system functions operate according to the intended requirements. This system significantly improves the operational efficiency of the telephone service, reduces manual administrative workloads, and minimizes the risk of recording errors. With this system, student calling services become more modern, well-documented, and scalable for implementation in broader Islamic boarding schools or similar institutions.

Keywords: Smart Calling, Python, WhatsApp Desktop, Call Automation, Digital Wartel System

BAB I PENDAHULUAN

1.1 Latar Belakang

Teknologi digital kian berkembang pesat, segala terobosan baru terus dibuat, menuntut banyak sektor untuk berjalan lebih cepat, termasuk dalam sektor media komunikasi [1]. Sebagai manusia, komunikasi menjadi kebutuhan vital dalam kehidupan sehari-hari. Tanpa adanya alat komunikasi akan menghambat pada rotasi kehidupan sosial masyarakat. Media komunikasi juga sangat dibutuhkan dalam sektor pendidikan, termasuk di lingkungan pondok pesantren [2].

Pondok Pesantren Salafiyah Syafi'iyah Sukorejo merupakan lembaga pendidikan Islam yang cukup besar di Indonesia, pesantren ini telah menampung hingga puluhan ribu santri, untuk menunjang pendidikan yang nyaman, pesantren memastikan komunikasi antar santri dan orang tua terjalin dengan baik [3]. Termasuk di Asrama Ma'hadul Qur'an yang merupakan salah satu cabang dari pesantren ini. Maka telepon menjadi sarana yang sangat dibutuhkan. Bagi santri, telepon menjadi media untuk mengetahui kabar keluarga dari rumah, begitu pun sebaliknya, bagi keluarga di rumah telepon menjadi media untuk mengetahui kabar anaknya yang sedang menuntut ilmu agama di pesantren.

Penggunaan telepon di pesantren tidak seperti pada umumnya di masyarakat. Regulasi pesantren tidak memperbolehkan santrinya menggunakan *handphone* (Hp) agar mereka lebih fokus untuk belajar [4]. Pihak pesantren akan memberikan sanksi kepada santri yang melanggar aturan ini. sebagai gantinya, pesantren

menyediakan warung telepon untuk memfasilitasi komunikasi antar santri dan keluarga di rumah.

Warung Telepon (Wartel) merupakan wadah yang memfasilitasi santri Asrama Ma'hadul Qur'an berkomunikasi dengan keluarga. Adanya Wartel di pesantren memberikan banyak dampak positif, komunikasi antar santri dan keluarga di rumah dapat terjalin dengan baik, santri akan terarut dalam berkomunikasi, dan larangan pesantren untuk penggunaan *handphone* tetap berjalan [3]. Selain itu, Wartel juga menjadi salah satu penyumbang pendapatan keuangan pesantren melalui jasa teleponnya. Wartel menerapkan tarif berdasarkan durasi waktu.

Saat ini, perangkat telepon yang digunakan di Asrama Ma'hadul Qur'an masih menggunakan perangkat telepon rumah, salah satunya Ezitel 6588. Perangkat telepon ini memiliki banyak kekurangan; durasi telepon santri tidak bisa terkontrol oleh petugas, dan petugas tidak mengetahui berapa durasi telepon santri tersebut. Rekapitulasi pendapatan tidak berdasarkan data telepon santri, semua pencatatan laporan keuangan dilakukan manual, hal ini tentu sangat menyulitkan dan memberikan data yang kurang akurat [5]. Tidak terkontrolnya durasi telepon akan berdampak pada semakin maraknya kecurangan pengguna jasa telepon, sehingga rekapitulasi pendapatan yang dilakukan oleh petugas hanya direka-reka saja, tanpa ada data yang jelas.

Melihat problematika di atas, maka perlu adanya solusi dan pembenahan sistem di Wartel. Salah satu upaya yang dapat dilakukan adalah dengan mengimplementasikan teknologi berbasis sistem otomatisasi untuk mendukung

pengelolaan Wartel yang lebih efisien. Pengembangan sistem berbasis otomatisasi telah terbukti mampu meningkatkan akurasi data, mengurangi potensi kesalahan manual, dan memberikan efisiensi dalam operasional [6]. Oleh karena itu, penelitian ini bertujuan untuk merancang dan mengimplementasikan Sistem *Smart Calling*, sebuah sistem informasi otomatisasi wartel yang dibangun menggunakan bahasa pemrograman Python, antarmuka grafis berbasis Tkinter, dan integrasi dengan WhatsApp Desktop melalui modul automasi PyAutoGUI. Sistem ini dirancang untuk mencatat durasi panggilan secara *real-time*, menghitung tarif secara otomatis, serta menyimpan dan mengelola data transaksi secara digital menggunakan database SQLite [7].

Berbeda dengan penelitian sebelumnya yang berfokus pada sistem pemantauan panggilan berbasis GSM atau penggunaan aplikasi berbasis Android [8], penelitian ini mengusung pendekatan baru yang lebih sesuai untuk konteks di lingkungan pendidikan yang tertutup seperti pesantren: yakni integrasi sistem desktop dengan WhatsApp Desktop sebagai media komunikasi utama. Inilah yang menjadi gap penelitian sekaligus kontribusi orisinal dalam studi ini.

1.2 Identifikasi Masalah

Berdasarkan latar belakang di atas, maka dapat ditentukan identifikasi masalah yang diperoleh dalam penelitian ini, diantaranya :

- a. Pemantauan durasi telepon santri tidak bisa dilakukan oleh petugas dan berdampak pada kecurangan pengguna (santri).
- b. Rekapitulasi pendapatan tidak memiliki data yang akurat, sehingga petugas hanya bisa mereka-reka saja.

1.3 Rumusan Masalah

Berdasarkan identifikasi masalah di atas, maka dapat dirumuskan pokok permasalahan yaitu “Bagaimana cara membangun Sistem *Smart Calling* di Asrama Ma’hadul Qur’an Salafiyah Syafi’iyah Sukorejo”

1.4 Batasan Masalah

Agar penelitian ini lebih fokus dan tidak terjadi penyimpangan masalah, maka dilakukan beberapa batasan permasalahan dalam penyusunan skripsi ini, diantaranya :

- a. Sistem hanya aplikasi pihak ketiga yang terkonfigurasi dan bergantung pada aplikasi *Whatsapp Desktop*.
- b. Sistem dirancang dalam skala prototipe dan diuji coba hanya di Asrama Ma’hadul Qur’an.
- c. Sistem dirancang hanya untuk melakukan *monitoring* durasi telepon dan rekapitulasi pendapatan.

1.5 Tujuan Penelitian

Terdapat dua tujuan utama dalam penelitian ini, diantaranya :

- a. Untuk memantau durasi telepon santri.
- b. Melakukan rekapitulasi pendapatan berdasarkan data durasi santri metelepon.

1.6 Manfaat Penelitian

Penelitian *Smart Calling* ini akan memberikan banyak manfaat, diantaranya:

- a. Meminimalisir kecurangan dalam penggunaan jasa telepon di Wartel
- b. Durasi telepon santri dapat dipantau petugas
- c. Rekapitulasi pendapatan Wartel lebih akurat

d. Waktu telepon santri akan lebih mudah dikontrol

1.7 Metode Penelitian

1.7.1. Jenis Penelitian

Karena penelitian ini mencakup pengembangan sistem baru berbasis teknologi (Sistem *Smart Calling* yang terintegrasi dengan aplikasi pihak ketiga WhatsApp), maka pendekatan R&D (*Research and Development*) sangat relevan. Penelitian ini melibatkan tahapan analisis kebutuhan, desain prototipe, implementasi, dan evaluasi sistem [7].

1.7.2. Teknik Pengumpulan Data

Untuk mendukung penelitian dan pengembangan Sistem *Smart Calling* di Asrama Ma'hadul Qur'an, digunakan beberapa teknik pengumpulan data sebagai berikut:

a. Studi Literatur

Peneliti mengkaji referensi berupa jurnal, buku, dan artikel yang relevan dengan penelitian ini, terutama yang membahas sistem komunikasi berbasis teknologi, manajemen wartel, dan penggunaan aplikasi pihak ketiga seperti *WhatsApp desktop*.

b. Observasi

Peneliti melakukan pengamatan langsung terhadap aktivitas komunikasi di wartel Asrama Ma'hadul Qur'an. Observasi ini mencakup proses penggunaan telepon oleh santri, sistem pengaturan durasi telepon, serta cara rekapitulasi pendapatan yang dilakukan oleh petugas.

c. Wawancara

Peneliti mewawancarai pihak-pihak terkait, seperti pengelola wartel dan santri. Tujuan wawancara adalah untuk menggali informasi mendalam mengenai kebutuhan komunikasi, kendala yang dihadapi, serta harapan terhadap sistem baru.

d. Uji Coba Sistem

Setelah sistem prototipe selesai dirancang, dilakukan uji coba langsung di wartel Asrama Ma'hadul Qur'an. Data yang dikumpulkan meliputi durasi telepon, ketepatan pencatatan pendapatan, dan status bayar santri.

1.7.3. Metode Pengembangan Sistem

Penelitian ini mengadopsi metode pengembangan sistem *Waterfall*, yaitu suatu model pengembangan perangkat lunak yang bersifat sistematis, berurutan, terdokumentasi dengan baik, dan sangat cocok untuk proyek dengan kebutuhan yang telah ditetapkan sejak awal serta sumber daya pengembang yang terbatas [10]. Dibandingkan dengan pendekatan iteratif seperti *prototyping* atau *agile*, *Waterfall* lebih memungkinkan pengendalian proses yang linear dan terstruktur, sehingga meminimalisir perubahan selama pengembangan [11]. Gambar 1 menunjukkan model pengembangan sistem *Waterfall* yang digunakan dalam penelitian ini. Model ini terdiri dari tahapan pengumpulan kebutuhan, analisis, desain sistem, pembangunan, dan pengujian, yang dilaksanakan secara berurutan untuk menjaga konsistensi pengembangan sistem [12]. Setiap tahapan harus diselesaikan secara tuntas sebelum berlanjut ke tahap berikutnya.



Gambar 1.1 Metode dalam *Waterfall* [12].

a. *Requirement Gathering* (Pengumpulan Kebutuhan)

Tahapan awal dalam penelitian ini dilakukan dengan mengumpulkan data empiris melalui observasi langsung, wawancara pihak terkait, dan studi literatur [13]. Observasi dilakukan di lokasi wartel Asrama Ma'hadul Qur'an untuk mengamati proses operasional yang berlangsung, mulai dari tata cara santri melakukan panggilan hingga pencatatan transaksi oleh petugas.

b. *Analysis* (Analisis Kebutuhan Sistem)

Setelah data kebutuhan dikumpulkan, tahap selanjutnya adalah analisis sistem. Pada fase ini dilakukan kajian mendalam untuk merumuskan kebutuhan fungsional dan non-fungsional yang akan diterapkan dalam sistem. Kebutuhan fungsional meliputi kemampuan sistem dalam mencatat durasi panggilan secara otomatis, mengintegrasikan panggilan melalui WhatsApp Desktop, mencatat dan menghitung tarif layanan secara *real-time*, serta menyimpan histori transaksi ke dalam *database*. Sedangkan kebutuhan non-fungsional mencakup kemudahan penggunaan sistem oleh petugas, kecepatan pemrosesan data, serta keamanan dan keandalan sistem. Selain itu, diidentifikasi pula jenis pengguna, yaitu santri sebagai *user*, serta petugas/admin sebagai pengelola sistem. Hasil analisis ini dituangkan ke dalam spesifikasi kebutuhan sistem yang menjadi dasar dalam proses desain.

c. *System Design* (Perancangan Sistem)

Tahap desain dilakukan untuk merancang solusi teknis berdasarkan spesifikasi yang telah dianalisis sebelumnya. Desain ini mencakup rancangan antarmuka pengguna, alur proses kerja sistem, serta struktur basis data. Rancangan antarmuka (UI) disusun agar mudah digunakan, terutama oleh petugas yang tidak memiliki latar belakang teknis. Selain itu, dibuat juga *flowchart*, *context diagram*, dan *data flow diagram* (DFD) untuk menggambarkan alur sistem secara menyeluruh dan hubungan antar entitas di dalamnya. Struktur basis data dirancang menggunakan *Entity Relationship Diagram* (ERD) agar data yang disimpan dapat dikelola dengan baik.

d. *Construction* (Pembangunan Sistem)

Pada tahap pembangunan sistem, seluruh rancangan sistem diwujudkan ke dalam bentuk aplikasi nyata. Bahasa pemrograman yang digunakan adalah *Python*, dengan antarmuka grafis dibangun menggunakan pustaka *Tkinter*. Integrasi dengan aplikasi pihak ketiga, yakni *WhatsApp Desktop*, dilakukan menggunakan modul *PyAutoGUI* yang memungkinkan sistem untuk mengontrol antarmuka secara otomatis. Sistem ini juga dilengkapi dengan fitur pencatatan durasi dan perhitungan tarif secara otomatis, serta penyimpanan transaksi ke dalam *database SQLite*.

e. *Testing* (Pengujian Sistem)

Tahap terakhir adalah pengujian sistem yang bertujuan untuk memastikan bahwa seluruh fitur berfungsi sesuai dengan kebutuhan. Pengujian dilakukan secara langsung di lingkungan wartel Asrama Ma'hadul Qur'an dalam skala

terbatas. Metode yang digunakan adalah *black-box testing*, yaitu metode pengujian yang difokuskan pada hasil keluaran tanpa memeriksa struktur kode program. Sistem diuji untuk beberapa aspek utama, seperti validasi input santri, keakuratan pencatatan durasi, ketepatan penghitungan tarif, serta keandalan koneksi dengan *WhatsApp Desktop*.

1.8 Sistematika Pembahasan

Sistematika pembahasan dalam penulisan skripsi ini disusun agar memudahkan pembaca dalam memahami alur berpikir dan langkah-langkah yang dilakukan dalam penelitian serta pengembangan sistem. Adapun sistematika penulisan dalam skripsi ini terdiri dari lima bab utama yang dijabarkan sebagai berikut:

a. Bab I: Pendahuluan

Bab ini memuat uraian umum mengenai latar belakang masalah yang melandasi dilakukannya penelitian, identifikasi masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, serta sistematika pembahasan. Bagian ini memberikan gambaran awal tentang alasan dan arah dari penelitian yang dilakukan.

b. Bab II: Tinjauan Pustaka dan Landasan Teori

Bab ini membahas teori-teori yang relevan dan mendukung penelitian, termasuk penjelasan mengenai sistem informasi, komunikasi digital, pemrograman *Python*, automasi dengan *PyAutoGUI*, serta integrasi sistem dengan *WhatsApp Desktop*. Selain itu, bab ini juga menyajikan hasil kajian terhadap

penelitian terdahulu yang berkaitan dengan topik ini sebagai pembanding dan penguat latar belakang penelitian.

c. Bab III: Metodologi Penelitian

Bab ini menjelaskan pendekatan penelitian yang digunakan, yaitu model pengembangan sistem berbasis metode *Waterfall*. Di dalamnya dijabarkan tahapan-tahapan seperti pengumpulan kebutuhan, analisis sistem, perancangan sistem, implementasi, dan pengujian. Selain itu, dijelaskan juga tentang identifikasi kebutuhan sistem, perancangan antarmuka, serta perangkat lunak dan perangkat keras yang digunakan.

d. Bab IV: Hasil dan Pembahasan

Bab ini menyajikan hasil implementasi dari sistem yang telah dikembangkan. Di dalamnya mencakup tampilan antarmuka program, penjelasan fungsionalitas tiap menu, serta hasil pengujian sistem. Pembahasan dilakukan dengan mengaitkan hasil implementasi terhadap tujuan penelitian dan permasalahan yang telah dirumuskan.

e. Bab V: Kesimpulan dan Saran

Bab terakhir ini berisi kesimpulan dari hasil penelitian yang telah dilakukan, yang merangkum pencapaian terhadap tujuan penelitian. Selain itu, disampaikan pula saran-saran yang dapat menjadi bahan perbaikan dan pengembangan lebih lanjut terhadap sistem yang telah dibuat.

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Sejumlah penelitian terdahulu digunakan sebagai referensi dalam penelitian ini, untuk memperkuat dasar teori dan mendukung pengembangan Sistem *Smart Calling* di Asrama Ma'hadul Qur'an, diantaranya;

1. Penerapan Paket Tkinter, *Subproses*, dan OS pada Pembuatan Aplikasi Manajemen Jaringan dengan Python

Permasalahan dalam penelitian ini disebabkan banyaknya pengguna layanan jaringan internet, sedangkan jumlah teknisinya kurang memadai. Hal ini menyebabkan kesulitan dalam pemeliharaan jaringan, terutama bagi instansi yang membutuhkan jaringan untuk operasional harian. Dibutuhkan sebuah sistem otomatisasi untuk membantu pengelolaan jaringan agar lebih efisien. Penelitian ini mengkaji penggunaan Python, yang dikenal sebagai bahasa pemrograman interpretatif dengan pustaka yang luas, seperti Tkinter untuk GUI dan *subprocess* untuk mengakses dan menjalankan perintah terminal. Pustaka ini memungkinkan pengembangan aplikasi dengan antarmuka ramah pengguna, dan membantu dalam tugas seperti ping. Aplikasi ini dibangun menggunakan Python dengan menerapkan Tkinter untuk membuat antarmuka grafis.

Aplikasi berhasil memenuhi fungsi utama, untuk Melakukan ping dan nslookup dengan data *input* berupa *IP address* atau alamat *website*,

Menampilkan informasi jaringan, koneksi aktif, dan proses jaringan. Membuka halaman web melalui antarmuka yang sederhana [9].

Penelitian ini juga relevan dengan "*Sistem Smart Calling*" karena menunjukkan bagaimana Python, terutama Tkinter, dapat digunakan untuk membangun antarmuka yang sederhana namun fungsional. Teknik yang diterapkan, seperti subprocess untuk otomatisasi tugas, dapat diadaptasi untuk mengelola panggilan telepon, mencatat durasi panggilan, dan menghitung biaya panggilan secara otomatis di sistem *Smart Calling*. Hal ini memperkuat efisiensi dan kemudahan operasional dalam pengelolaan komunikasi di pesantren.

2. Perancangan Chatbot WhatsApp *Natural Language Processing* untuk Pelayanan Digital Universitas Nusa Putra

Di lokasi penelitian ini memerlukan sistem yang dapat memberikan informasi akademik dan non-akademik secara efisien kepada mahasiswa dan masyarakat umum. Keterbatasan sumber daya manusia dalam memberikan layanan informasi manual menyebabkan perlunya solusi otomatis yang dapat merespons pertanyaan pengguna dengan cepat dan akurat. Penelitian ini mengkaji penggunaan chatbot sebagai *asisten virtual* yang mampu berinteraksi dengan pengguna melalui pesan teks. Teknologi *Natural Language Processing* (NLP) diterapkan untuk memungkinkan chatbot memahami dan merespons pertanyaan dalam bahasa alami. Bahasa pemrograman Python digunakan bersama dengan PHP untuk membangun sistem ini, memanfaatkan framework seperti Laravel, Node.js, dan NLTK untuk mendukung fungsionalitas chatbot.

Peneliti merancang dan mengimplementasikan chatbot berbasis WhatsApp dengan teknologi NLP. Sistem ini dibangun menggunakan kombinasi PHP dan Python, di mana PHP/Laravel menangani pesan WhatsApp, API, sesi pengguna, dan penyimpanan data, sementara Python melakukan analisis NLP dan pemrosesan bahasa alami. Chatbot ini dirancang untuk menjawab pertanyaan terkait informasi akademik dan non-akademik dalam bahasa Indonesia dan Inggris, dengan fitur *quick replies* dan *input* teks langsung. Pengujian fungsionalitas menggunakan black-box testing menunjukkan bahwa chatbot memiliki tingkat keakuratan sekitar 90% dalam merespons pertanyaan pengguna. Selain itu, survei kepuasan pengguna menunjukkan bahwa 87,9% responden puas dengan kinerja chatbot, terutama dalam hal kecepatan respon dan akurasi informasi yang diberikan. Hasil ini menunjukkan bahwa sistem chatbot berbasis WhatsApp dengan teknologi NLP layak diimplementasikan untuk meningkatkan kualitas layanan digital di Universitas Nusa Putra [15].

Dalam konteks 'Sistem *Smart Calling*', pendekatan serupa dapat diterapkan untuk mengotomatisasi proses panggilan atau interaksi dengan pengguna. Penggunaan Python memungkinkan fleksibilitas dalam pemrosesan data dan logika bisnis, sementara integrasi dengan WhatsApp menyediakan antarmuka yang familiar bagi pengguna.

3. Pengembangan *Virtual Assistant Chatbot* Berbasis WhatsApp

Jurnal ini berawal dari sebuah permasalahan bahwa mahasiswa sering mengalami kesulitan untuk memperoleh informasi perkuliahan, seperti informasi akademik, administrasi, atau jadwal dosen. Hambatan ini diperparah

oleh pandemi Covid-19, yang membatasi komunikasi tatap muka dan memaksa perguruan tinggi untuk mengadopsi metode komunikasi jarak jauh yang lebih efektif. Jurnal ini menyoroti teknologi *chatbot* sebagai solusi berbasis kecerdasan buatan (AI) yang dirancang untuk memberikan informasi dan layanan komunikasi secara otomatis. *Chatbot* terintegrasi dengan WhatsApp menggunakan teknologi *Twilio API*, memanfaatkan popularitas WhatsApp sebagai aplikasi komunikasi yang banyak digunakan, dengan fitur-fitur yang mudah diakses. Penelitian dalam jurnal ini menggunakan metode *Extreme Programming* (XP) untuk mengembangkan chatbot berbasis WhatsApp. Langkah-langkah utamanya meliputi: *Planning, Design, Coding, dan Testing*.

Penelitian ini memberikan hasil bahwa Chatbot yang dikembangkan dapat memberikan respons 24/7 dengan waktu respons yang cepat, sehingga mahasiswa dapat memperoleh informasi kapan saja tanpa perlu datang langsung ke kampus. Pengujian menunjukkan bahwa sistem bekerja dengan akurasi 100% sesuai skenario pengujian [16].

Penelitian ini relevan dengan "Sistem *Smart Calling*" yang dikembangkan untuk komunikasi di Asrama Ma'hadul Qur'an. Sama seperti chatbot berbasis WhatsApp, Sistem *Smart Calling* dapat menggunakan teknologi serupa untuk mempermudah komunikasi antara santri dan keluarga. Penggunaan Python sebagai bahasa utama dan integrasi dengan WhatsApp Desktop dapat memberikan solusi komunikasi yang efisien, mudah diakses, dan dapat diandalkan di lingkungan pesantren.

2.2 Landasan Teori

Landasan teori pada penelitian ini mencakup konsep-konsep yang relevan dengan pengembangan Sistem *smart calling* di Wartel Asrama Ma'hadul Qur'an, diantaranya:

a. Sistem Informasi

Sistem informasi adalah kombinasi dari teknologi informasi dan aktivitas manusia yang mendukung operasi, manajemen, dan pengambilan keputusan dalam suatu organisasi [17]. Pada penelitian ini, sistem informasi digunakan untuk memantau durasi telepon, mencatat pendapatan wartel secara otomatis, serta mengintegrasikan penggunaan aplikasi WhatsApp.

b. Smart Calling

Konsep "*Smart Calling*" merujuk pada sistem komunikasi cerdas, yang mampu memantau, mencatat, dan mengelola panggilan telepon secara otomatis [18]. Sistem ini memanfaatkan teknologi berbasis *software* dan *hardware* untuk meningkatkan efisiensi pengelolaan wartel.

c. Pemrograman Python

Python adalah bahasa pemrograman tingkat tinggi yang sering digunakan untuk pengembangan sistem berbasis GUI (*Graphical User Interface*). Modul-modul seperti Tkinter, Pyautogui, dan Sqlite3 memungkinkan pembuatan aplikasi desktop yang interaktif dan terintegrasi dengan basis data [14].

d. Konsep Automasi dengan Pyautogui dan Time

Modul Pyautogui memungkinkan sistem melakukan *automasi* berbasis GUI, seperti mengontrol aplikasi WhatsApp Desktop untuk mencatat durasi

panggilan. Sementara *modul Time* digunakan untuk penghitungan durasi telepon secara *real-time* [19]. Konsep ini mendukung sistem agar bekerja lebih efisien dan meminimalkan kesalahan manual.

e. Aplikasi WhatsApp Desktop

WhatsApp Desktop adalah aplikasi yang memungkinkan pengguna untuk mengakses WhatsApp melalui komputer atau laptop, baik dengan sistem operasi Windows maupun macOS [18].

f. Database Management dengan SQLite

SQLite adalah sistem manajemen basis data relasional (RDBMS) yang bersifat ringan, *self-contained*, dan *serverless*. SQLite menyimpan seluruh data dalam satu file yang dapat diakses langsung tanpa memerlukan instalasi server terpisah, sehingga sangat cocok untuk aplikasi yang membutuhkan penyimpanan data lokal [19].









2.3 Pemodelan

Dalam penelitian ini, akan dibuat rancangan sistem informasi secara menyeluruh, mulai dari alur proses, struktur data, hubungan antar entitas, hingga interaksi pengguna dengan sistem. Berikut penjelasan spesifiknya.

a. Flowchart

Dalam konteks sistem informasi, *flowchart* digunakan untuk menunjukkan langkah-langkah dalam suatu proses atau prosedur yang terjadi dalam sistem. Setiap langkah dalam proses digambarkan dengan bentuk simbol tertentu (seperti persegi panjang untuk proses, belah ketupat untuk keputusan, dan panah untuk alur) yang membuatnya mudah dipahami [20].

Tabel 2. 1 Simbol-simbol *Flowchart*

No.	Simbol Flowchart	Nama	Arti Simbol Flowchart
1		<i>Terminator</i>	Awal atau akhir konsep(prosedur)
2		<i>Process</i>	Proses operasional
3		<i>Document</i>	Dokumen atau laporan berupa print out
4		<i>Decision</i>	Kaputusa atau sub poin. Garis yang terhubung dengan bentuk <i>secision</i> merujuk pada situasi-situasi yang berbeda sesuai dengan keputusan yang digambarkan
5		<i>Input/Output</i>	Input dan output(contohnya, input; feedback dari pelanggan, output;desain produk baru)
6		<i>On-Page Reference/Connector</i>	Penghubung alur dalam halaman yang sama
7		<i>Off-Page Reference/Off-Page Connector</i>	Penghubung alur dalam halaman yang berbeda
8		<i>Flow</i>	Arah alur dalam konsep (prosedur)

b. DFD (*Data Flow Diagram*)

Data Flow Diagram (DFD) adalah alat pemodelan yang digunakan untuk menggambarkan aliran data dalam sistem informasi. DFD menunjukkan bagaimana data diproses dan dipindahkan antar komponen sistem. DFD dibagi

menjadi beberapa level (Level 0, Level 1, dst.) yang menggambarkan detail sistem secara bertahap, dari gambaran umum hingga lebih mendalam.

c. ERD (*Entity Relationship Diagram*)

Entity Relationship Diagram (ERD) adalah diagram yang digunakan untuk menggambarkan hubungan antar entitas dalam basis data. ERD memberikan gambaran tentang bagaimana data disusun dan diorganisir dalam sistem, serta hubungan antar entitas tersebut, seperti satu-ke-banyak atau banyak-ke-banyak.

d. UI (*User Interface*)

UI (*User Interface*) merujuk pada desain dan antarmuka pengguna dari aplikasi atau sistem informasi. Pemodelan UI bertujuan untuk merancang tampilan aplikasi yang akan digunakan oleh pengguna, agar interaksi dengan sistem menjadi mudah, intuitif, dan efisien. UI melibatkan desain elemen-elemen visual, seperti tombol, *form input*, menu navigasi, dan *layout*.

2.4 Perangkat Lunak yang Digunakan

Dalam proses pengembangan Sistem *Smart Calling* di Warung Telepon Asrama Ma'hadul Qur'an, dibutuhkan sejumlah perangkat lunak (*software*) pendukung yang memiliki peran penting dalam membangun, menguji, dan mengelola sistem. Pemilihan perangkat lunak disesuaikan dengan kebutuhan pengembangan sistem desktop, kemudahan integrasi dengan aplikasi pihak ketiga seperti WhatsApp Desktop, serta dukungan terhadap pemrograman GUI dan basis data lokal. Berikut adalah perangkat lunak yang digunakan dalam penelitian ini:

a. Python

Python merupakan bahasa pemrograman utama yang digunakan dalam pengembangan sistem ini. Python dipilih karena bersifat *open source*, fleksibel, dan memiliki banyak pustaka (library) yang mendukung kebutuhan pengembangan antarmuka pengguna (GUI), pengolahan data, serta automasi aplikasi lain [21]. Versi Python yang digunakan dalam penelitian ini adalah Python 3.11.

b. Tkinter

Tkinter adalah pustaka GUI bawaan dari Python yang digunakan untuk membangun antarmuka grafis dalam aplikasi desktop. Dengan Tkinter, sistem *Smart Calling* dapat menyediakan tampilan yang interaktif, ringan, dan mudah digunakan oleh petugas wartel. Komponen seperti *form input*, tombol, dan tabel data dikembangkan menggunakan Tkinter.

c. PyAutoGUI

PyAutoGUI merupakan library Python yang digunakan untuk melakukan *automasi* kontrol antarmuka (*GUI automation*). Dalam sistem ini, PyAutoGUI dimanfaatkan untuk mengotomatisasi interaksi dengan aplikasi WhatsApp Desktop, seperti membuka jendela panggilan video atau audio, serta mencatat waktu mulai dan waktu selesai panggilan secara otomatis.

d. SQLite

SQLite digunakan sebagai sistem manajemen basis data (DBMS) yang bersifat ringan dan tidak memerlukan instalasi server terpisah. Seluruh data pelanggan, transaksi, dan histori panggilan disimpan dalam database SQLite.

Penggunaan SQLite sangat sesuai untuk aplikasi desktop dengan skala pengguna terbatas namun membutuhkan pengelolaan data yang terstruktur.

e. Visual Studio Code (VS Code)

Visual Studio Code digunakan sebagai editor kode utama dalam pengembangan sistem. Editor ini menyediakan fitur seperti *highlight sintaks*, integrasi terminal, dan ekstensi pendukung Python yang membantu mempercepat proses pengembangan, *debugging*, dan dokumentasi kode program.

f. WhatsApp Desktop

WhatsApp Desktop adalah aplikasi resmi dari WhatsApp yang dijalankan di sistem operasi Windows. Aplikasi ini berfungsi sebagai media komunikasi utama yang digunakan santri untuk melakukan panggilan kepada keluarga. Sistem *Smart Calling* diintegrasikan dengan aplikasi ini melalui *automasi* antarmuka menggunakan PyAutoGUI.

g. Microsoft Excel

Microsoft Excel digunakan dalam proses ekspor data transaksi dan histori ke format xlsx (*Microsoft Excel Open XML Spreadsheet*). Data hasil panggilan dan pembayaran santri dapat diekspor dari sistem *Smart Calling* untuk keperluan *backup* data atau pelaporan keuangan secara manual.

h. PowerDesigner (opsional/lampiran ERD)

PowerDesigner adalah alat bantu pemodelan data yang digunakan dalam perancangan basis data, khususnya untuk membuat *Entity Relationship Diagram* (ERD). Meskipun dalam skripsi ini model ERD tidak menjadi fokus utama, *PowerDesigner* tetap digunakan untuk memvisualisasikan struktur relasi antar

entitas dalam database jika diperlukan sebagai lampiran atau pendukung analisis struktur data.

Dengan kombinasi perangkat lunak di atas, sistem *Smart Calling* dapat dibangun secara efektif dan efisien, serta mampu menjawab kebutuhan pengelolaan layanan wartel berbasis *automasi* di lingkungan pondok pesantren.



BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Gambaran Umum Obyek Penelitian

Wartel di Asrama Ma'hadul Qur'an (MQ) merupakan penyedia fasilitas komunikasi santri dengan keluarga di rumah. Santri dapat menghubungi keluarga atau kerabat yang berada jauh di luar kompleks pesantren melalui wartel. Jasa wartel dihitung melalui durasi per-menit.

3.1.1 Keadaan Sistem yang Berjalan

Saat ini wartel Ma'hadul Qur'an menggunakan fasilitas telepon rumah Ezitel 6588. Telepon ini bisa menggunakan *sim card* semua operator. Namun untuk *sim card* XL sering mengalami gangguan, sehingga seluruh perangkat teleph hanya menggunakan Telkomsel. Perangkat bisa telepon jika *sim card* sudah terisi pulsa, atau paket telepon, tanpa dua hal ini perangkat tidak bisa melakukan panggilan. Telepon Ezitel 6588 hanya melimitasi panggilan sampai dua jam. Tidak ada pengaturan khusus di telepon ini untuk membatasi durasi telepon. Telepon ini tidak bisa dikontrol petugas, durasi panggilan sepenuhnya hanya diketahui pengguna. Selain itu, *History* panggilan dapat dengan mudah dihapus, sehingga petugas tidak bisa melihat jejak *History* yang telah dihapus pengguna.

3.1.2 Kelebihan Sistem

Penggunaan Telepon Ezitel 6588 memiliki kelebihan tanpa harus berpangku ke aplikasi pihak ketiga, cukup ada pulsa atau paket telepon sistem ini sudah bisa berjalan.

3.1.3 Kelemahan Sistem

Penggunaan Telepon Ezitel 6588 dan sistem yang berjalan memiliki banyak kekurangan saat diaplikasikan di wartel Ma'hadul Qur'an, diantaranya:

1. Durasi dan *History* telepon tidak bisa dipantau petugas
2. Rekapitulasi pendapatan wartel tidak dapat dicatat dengan akurat berdasarkan durasi telepon santri
3. Banyak kecurangan penggunaan jasa telepon yang dilakukan pengguna sehingga mengakibatkan kerugian

3.2 Alur Proses

3.2.1 Identifikasi dan Analisis Proses Bisnis

Identifikasi dan Analisis Proses Bisnis adalah untuk memahami dan memetakan alur kerja serta aktivitas yang terlibat dalam sistem *Smart Calling* ini. Langkah tersebut penting untuk mengidentifikasi area yang memerlukan perbaikan, memastikan efisiensi operasional, dan mendukung pengembangan sistem yang lebih efektif [16].

a. Identifikasi Proses Bisnis

Ada beberapa poin yang perlu diidentifikasi dalam kegiatan bisnis yang berhubungan dengan sistem *Smart Calling* dalam penelitian ini. Poin-poin di dalamnya menjadi komponen penting bagi sistem, di antaranya;

1. Form Panggilan (*user/santri*)
2. Pengolahan data transaksi
3. *History* Transaksi
4. Pengolahan data pelanggan (*santri*)

5. Pengaturan Aplikasi

6. Menu Informasi

- b. Identifikasi Proses Bisnis

Analisis proses bisnis bertujuan untuk menganalisis dan menggambarkan aktivitas-aktivitas utama yang terjadi dalam sistem *Smart Calling* serta peran masing-masing aktor dalam alur layanan wartel di Asrama Ma'hadul Qur'an. Dalam Analisis proses bisnis ini dibedakan menjadi dua jenis pengguna, yaitu petugas (admin/operator) dan *user* (santri).

1. Proses Bisnis untuk Petugas (Admin/Operator)

Petugas memiliki akses masuk ke dalam sistem melalui fitur *login*. Fitur *login* akan mengidentifikasi apakah petugas biasa atau admin. Setelah berhasil masuk, petugas akan diarahkan ke *dashboard* utama yang berisi beberapa menu utama untuk mengelola operasional wartel, antara lain:

- a) Menu Transaksi

Berfungsi untuk menampilkan data panggilan santri yang masuk secara otomatis dari *form user*. Di sini petugas dapat melihat durasi, tarif panggilan, serta melakukan proses pembayaran. Setelah transaksi dilakukan, data akan dipindahkan ke menu histori.

- b) Menu *History* Transaksi

Menampilkan seluruh data transaksi yang telah selesai dibayar. Petugas dapat melihat riwayat pembayaran, status lunas, serta menghapus atau mengekspor data ke dalam *file Excel* untuk keperluan laporan atau *backup*.

c) Menu Pelanggan

Digunakan untuk mengelola data santri sebagai pelanggan wartel. Di sini petugas dapat menambah, mengedit, atau menghapus data pelanggan yang terdiri dari NIS, nama, asrama, dan alamat. Hanya NIS yang bersifat unik dan tidak dapat diedit karena menjadi *primary key*.

d) Menu Info

Menyediakan informasi atau pengumuman terkait layanan wartel yang dapat ditampilkan kepada pengguna atau petugas. Konten informasi ini dapat berupa jadwal layanan, pengumuman sistem, atau ketentuan tertentu.

e) Menu Pengaturan (khusus admin)

Menu ini hanya dapat diakses oleh admin dengan hak akses tertinggi. Pengaturan ini mencakup konfigurasi sistem seperti tarif panggilan (audio/video), logo aplikasi, dan informasi kontak layanan seperti WhatsApp dan sosial media. Admin juga dapat mengatur hak akses pengguna lain dari menu ini.

Setiap aktivitas dalam sistem petugas dilakukan secara *real-time* dan terdokumentasi dalam basis data SQLite agar data tetap terjaga dan dapat diakses kembali.

2. Proses Bisnis untuk User (Santri)

Santri sebagai pengguna akhir sistem memiliki akses terbatas pada form panggilan. Hal ini bertujuan agar fokus fungsi *user/santri* hanya untuk melakukan panggilan saja, tidak yang lain. Selain itu pembatasan akses ini juga

berfungsi untuk memudahkan penggunaan. Maka, pada form ini, santri cukup menginput:

- a) NIS yang sudah terdaftar di sistem
- b) Nomor WhatsApp tujuan (yang harus aktif dan tersedia)

Setelah data divalidasi, sistem akan memberikan pilihan jenis panggilan yaitu *Audio Call* atau *Video Call*, masing-masing dengan tarif berbeda per menit. Setelah panggilan dilakukan, sistem akan mencatat durasi panggilan secara otomatis, menghitung tarif sesuai waktu berjalan, dan menyimpan data ke dalam sistem transaksi.

Santri tidak memiliki akses ke menu lainnya seperti history, pelanggan, atau pengaturan. Sistem ini dirancang agar sederhana dan fokus pada layanan panggilan, namun tetap mencatat aktivitas secara digital dan terintegrasi dengan WhatsApp Desktop menggunakan PyAutoGUI.

3.2.2 Identifikasi dan Analisis Kebutuhan

Di bagian ini akan mengidentifikasi dan menganalisis jenis kebutuhan dan proses-proses yang akan dilakukan sistem. Selain itu, kebutuhan fungsional dan non fungsional yang berisi informasi-informasi di dalam sistem akan dipaparkan di sini, berikut detailnya.

a. Kebutuhan Fungsional

Kebutuhan fungsional menggambarkan fungsi-fungsi utama yang harus dimiliki oleh sistem agar dapat berjalan sesuai dengan tujuan dan kebutuhan pengguna. Dalam pengembangan Sistem *Smart Calling* di Warung Telepon Asrama Ma'hadul Qur'an, sistem dirancang untuk dapat digunakan oleh dua

jenis pengguna, yaitu petugas/admin dan user (santri). Berdasarkan hasil implementasi dan analisis dari kode sistem yang telah dibuat, berikut adalah kebutuhan fungsional yang diidentifikasi:

1. Kebutuhan Fungsional untuk Petugas/Admin

Sistem harus memiliki fungsi-fungsi berikut:

- a) *Login* Pengguna: Sistem menyediakan fitur login dengan *username* dan *password* untuk membatasi akses berdasarkan peran pengguna (admin atau petugas).
- b) Manajemen Transaksi Panggilan: Petugas dapat melihat daftar transaksi panggilan yang masuk dari *user*, termasuk informasi seperti NIS, nomor telepon, tipe panggilan, durasi, dan tarif yang dihitung otomatis. Transaksi ini dapat diproses menjadi status lunas atau belum lunas.
- c) Pembersihan Otomatis Transaksi Gagal: Transaksi dengan durasi pendek atau tarif nol secara otomatis dihapus setiap beberapa menit.
- d) *History* Transaksi: Sistem mencatat semua riwayat panggilan yang telah dibayar, lengkap dengan informasi durasi, tarif, status pembayaran, dan nama petugas.
- e) Pengelolaan Data Pelanggan (Santri): Petugas dapat menambah, mengedit, atau menghapus data santri (NIS, nama, asrama, nomor telepon).
- f) Pengelolaan Info Aplikasi: Admin dapat membuat, mengedit, dan menghapus informasi seperti jadwal layanan atau pengumuman penting untuk ditampilkan di aplikasi.

- g) Pengaturan Tarif Panggilan: Admin dapat mengubah tarif panggilan untuk *audio call* dan *video call* melalui menu pengaturan.
- h) Pengelolaan Hak Akses: Admin dapat mengatur hak akses setiap pengguna terhadap fitur tertentu seperti ekspor data, akses histori, atau fitur pelanggan.
- i) Ekspor Data: Petugas dengan hak akses tertentu dapat mengekspor histori transaksi atau data pelanggan ke format *Excel*.

2. Kebutuhan Fungsional untuk *User* (Santri)

Sistem juga menyediakan fitur khusus untuk *user* (santri) dalam bentuk aplikasi terpisah:

- a) Formulir Panggilan Sederhana: *User* hanya perlu memasukkan NIS dan nomor telepon tujuan untuk melakukan panggilan.
- b) Validasi NIS: Sistem akan mengecek apakah NIS yang dimasukkan terdaftar di database server.
- c) Pilihan Tipe Panggilan: *User* dapat memilih antara *audio call* dan *video call*. Masing-masing memiliki tarif berbeda per menit.
- d) Integrasi Otomatis dengan WhatsApp Desktop: Sistem akan mengontrol aplikasi WhatsApp Desktop secara otomatis untuk memulai panggilan menggunakan PyAutoGUI.
- e) *Timer* dan Penghitung Tarif Otomatis: Selama panggilan berlangsung, sistem menampilkan durasi waktu secara *real-time* dan menghitung tarif berdasarkan durasi dan jenis panggilan.

- f) Penyimpanan Data Otomatis ke Server: Setelah panggilan selesai, data akan dikirim otomatis ke server admin. Jika gagal, data akan disimpan sementara di database lokal dan dikirim ulang saat koneksi kembali normal.
- g) Pendeteksi Tombol WhatsApp Hilang: Sistem akan memantau keberadaan tombol “Akhir Panggilan” di WhatsApp Desktop. Jika tombol hilang atau panggilan tertutup, sistem akan menghentikan panggilan secara otomatis dan mencatat data.

b. Analisa kebutuhan non fungsional

Kebutuhan non-fungsional merupakan aspek-aspek pendukung yang tidak secara langsung berkaitan dengan fitur atau fungsi utama sistem, namun sangat penting untuk menjamin kualitas, performa, dan keandalan sistem [17]. Dalam pengembangan Sistem *Smart Calling* di Warung Telepon Asrama Ma’hadul Qur’an, kebutuhan non-fungsional dirancang untuk menunjang kelancaran operasional wartel serta memastikan kenyamanan pengguna dalam mengoperasikan sistem.

Adapun kebutuhan non-fungsional yang diidentifikasi adalah sebagai berikut:

1. Kemudahan Penggunaan (*Usability*)

Sistem dirancang dengan antarmuka yang sederhana, bersih, dan intuitif agar mudah digunakan oleh petugas wartel dan santri, tanpa memerlukan pelatihan teknis yang kompleks. Untuk santri, form input disederhanakan

hanya menjadi dua kolom utama, yakni NIS dan nomor WhatsApp, sedangkan petugas mendapatkan akses menu yang rapi dan tersusun berdasarkan peran.

2. Keandalan Sistem (*Reliability*)

Sistem harus tetap dapat berjalan meskipun terjadi gangguan jaringan, terutama pada saat mengirimkan data ke server. Untuk itu, sistem dilengkapi dengan fitur penyimpanan data sementara di database lokal jika koneksi ke server gagal, yang kemudian akan dikirim ulang secara otomatis saat koneksi pulih.

3. Keamanan (*Security*)

Sistem dilengkapi dengan login pengguna untuk membatasi akses ke menu administrasi dan pengaturan penting. Pengguna juga memiliki hak akses yang berbeda sesuai dengan perannya (admin, petugas, user), yang diatur dalam sistem pengaturan akses.

4. Kinerja Sistem (*Performance*)

Aplikasi desktop ini harus mampu merespons input pengguna secara *real-time*, terutama saat melakukan panggilan, mencatat durasi, dan menghitung tarif. Proses perhitungan tarif dan deteksi tombol WhatsApp dilakukan secara paralel agar tidak mengganggu antarmuka.

5. Portabilitas (*Portability*)

Sistem dibangun menggunakan Python dan berjalan sebagai aplikasi desktop tanpa memerlukan instalasi server terpisah. Hal ini memungkinkan sistem untuk dijalankan di komputer-komputer standar wartel yang memiliki sistem operasi Windows, dengan kebutuhan spesifikasi minimum.

6. Integrasi Sistem (*System Integration*)

Salah satu kebutuhan penting adalah kemampuan sistem untuk terintegrasi dengan aplikasi pihak ketiga, yakni WhatsApp Desktop. Integrasi ini dilakukan melalui *automasi* visual menggunakan PyAutoGUI, yang memungkinkan sistem mengendalikan WhatsApp untuk melakukan panggilan secara otomatis dan memantau status panggilan.

3.2.3 Identifikasi dan Analisis Alternatif Solusi

Bagian ini akan mengidentifikasi dan menganalisis berbagai alternatif solusi yang mungkin diterapkan untuk mengatasi masalah atau kekurangan yang ada dalam sistem yang akan dibuat. Proses identifikasi ini bertujuan untuk memberikan gambaran yang jelas mengenai berbagai pilihan yang tersedia, serta memeriksa kelebihan dan kekurangan dari masing-masing alternatif solusi. Setiap alternatif solusi akan dipertimbangkan berdasarkan faktor-faktor seperti biaya terjangkau, kesesuaian dengan kebutuhan sistem, serta kemudahan implementasinya. Tabel berikut akan menyajikan alternatif solusi untuk setiap permasalahan yang dihadapi dalam pengembangan sistem.

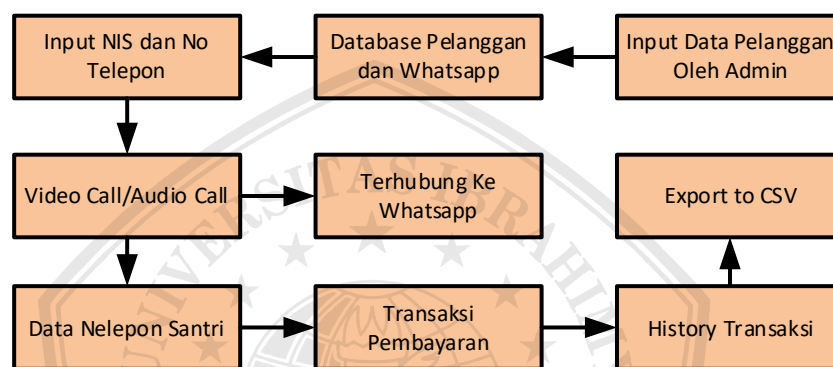
Tabel 3. 1 Identifikasi Alternatif Solusi

No.	Identifikasi	Alternatif Solusi
1	Sistem Panggilan Tidak Berfungsi	Pendataan menggunakan Jendela History Transaksi
2	Perangkat <i>User</i> tidak dapat dikendalikan	Pengaktifan aplikasi Veyon (untuk <i>monitoring</i> sementara disaat darurat)
3	Alat yang digunakan	Komputer/tablet Windows10 64 bit
4	<i>Bacup</i> data	Microsoft <i>Excel</i>

3.3 Desain Sistem

3.3.1 Diagram Blok Sistem

Diagram blok sistem adalah representasi grafis yang menunjukkan komponen utama dari suatu sistem dan interaksi atau hubungan antara komponen-komponen tersebut[18]. Dalam penelitian ini diagram blok dibuat untuk mempermudah dan memperjelas proses berjalannya sistem.



Gambar 3. 1 Diagram Blok Sistem

Proses dimulai dengan *Input* NIS dan Nomor Telepon, Admin mengelola data ini melalui *Input* Data Pelanggan dan menyimpannya dalam Database Pelanggan. Santri dapat memilih *Video call/Audio call*, yang akan terhubung ke WhatsApp. Durasi panggilan dicatat dan diproses dalam Transaksi Pembayaran, lalu disimpan dalam *History* Transaksi, serta dapat diekspor melalui fitur *Export to Excel*.

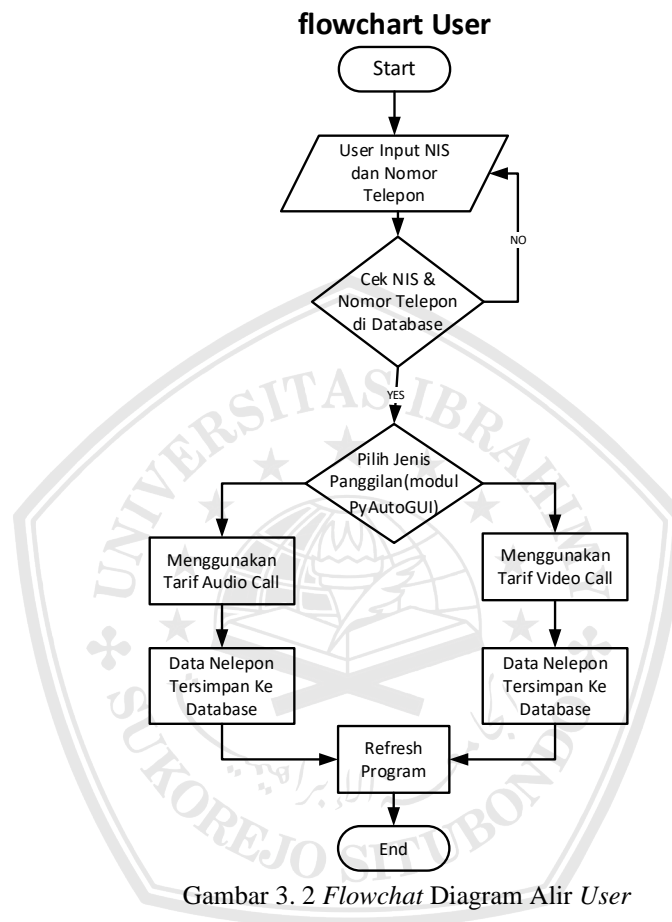
3.3.2 Desain Proses

a. Identifikasi Desain Proses Menggunakan *Flowchart*

Flowchart adalah diagram visual yang menggambarkan urutan langkah-langkah atau proses dalam suatu sistem. *Flowchart* menggunakan simbol-simbol

grafis dan panah untuk menunjukkan alur kerja dari satu langkah ke langkah berikutnya [19].

1. Flowchart Diagram Alir *User*

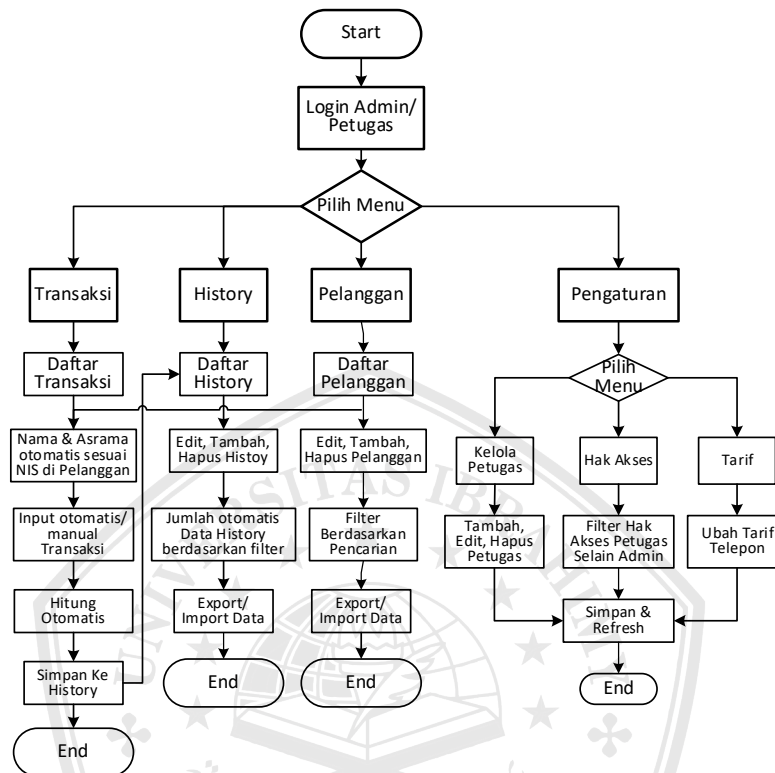


Gambar 3. 2 Flowchat Diagram Alir *User*

Gambar di atas merupakan *flowchat* untuk menunjukkan diagram alir bagi *user*. *User* (santri) melakukan *input* NIS dan nomor telepon sebelum menelepon. NIS santri harus sudah terdaftar sebagai pelanggan, dan nomor telepon harus terdaftar di Whatsapp. Jika sudah terdaftar maka santri bisa memilih untuk melakukan *video call* atau *audio call*. Jika *video call* harga per-menit Rp.1000, jika *audio call* harga Rp.500 per-menit. Setelah menelepon data tersebut akan tersimpan ke database. Lalu sistem akan *merefresh* ulang aplikasi dan kembali ke semula.

2. Flowchart Diagram Alir Admin (Petugas)

flowchart Petugas/Admin



Gambar 3.3 Flowchat Diagram Alir Admin

Gambar di atas merupakan *flowchat* diagram alir dari menu Admin.

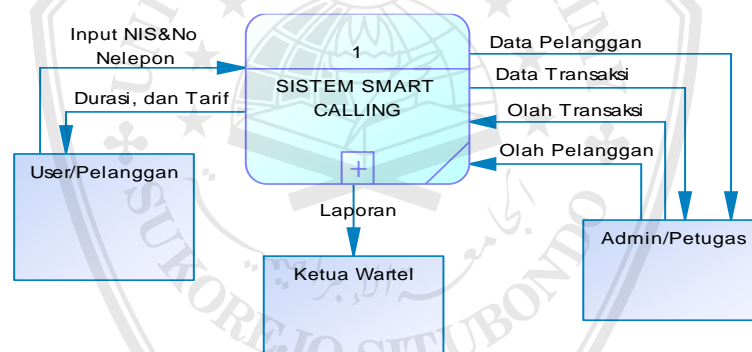
Awal dari program ini adalah mengambil data dari database transaksi yang telah dilakukan inputan oleh santri. Kemudian data tersebut diolah untuk dilakukan transaksi. Jika data dilakukan pembayaran maka data akan diproses untuk dipindah ke database *History* Transaksi. Jika data tidak dilakukan pembayaran, maka data akan tetap berada di *form* transaksi hingga dilakukan pembayaran. Setelah data dipindah maka sistem akan *merefresh* ulang dan sistem kembali ke semula.

b. Pemodelan Sistem

Penelitian ini menggunakan pemodelan sistem *Data Flow Diagram* (DFD). *Data Flow Diagram* (DFD) adalah alat visual yang digunakan untuk menggambarkan aliran informasi dalam suatu sistem. DFD menyajikan representasi grafis tentang bagaimana data bergerak melalui sistem, mulai dari input, proses, hingga output [20].

1. Context Diagram

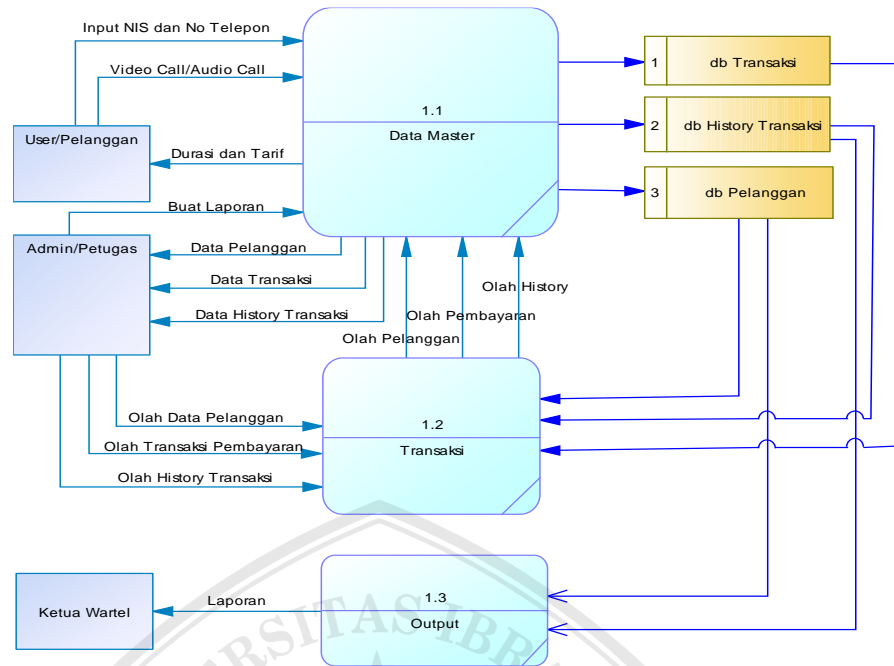
Context Diagram pada sistem *Smart Calling* ini akan memberikan gambaran umum tentang alur sistem yang akan dibangun. Penggambaran elemennya menggunakan beberapa entitas seperti gambar di bawah.



Gambar 3. 4 Context Diagram

2. Data Flow Diagram

Data Flow Diagram adalah turunan dari proses *Context Diagram*. Penggambaran desain prosesnya lebih terperinci dari *Context Diagram*, karena menambahkan detail proses yang terjadi pada sistem [21]. Gambaran detailnya sebagaimana pada gambar berikut.



Gambar 3. 5 Data Flow Diagram (DFD)

3.3.3 Identifikasi dan Desain Database

a. Identifikasi Tabel Database

Bagian ini akan mengidentifikasi apa saja yang akan ada dalam database menggunakan tabel, fungsinya untuk menggambarkan kebutuhan database.

1. Identifikasi Database Transaksi

Tabel 3. 2 Tabel Database Transaksi

No	Nama	Tipe Data	Panjang
1	Id transaksi (key)	integer	20
2	waktu	Date	-
3	User	text	20
4	nis	Integer	12
5	Nama	Text	30
6	Asrama	Text	15
7	No. telepon	Integer	15
8	Tipe	Text	15
9	Durasi	Time	-
10	tarif	text	20

2. Identifikasi Database *History* Transaksi

Tabel 3. 3 Tabel Database *History* Transaksi

No	Nama	Tipe Data	Panjang
1	Id History transaksi (key)	integer	20
2	waktu	Date	-
3	User	text	20
4	nis	Integer	12
5	Nama	Text	30
6	Asrama	Text	15
7	No. telepon	Integer	15
8	Tipe	Text	15
9	Durasi	Time	-
10	Tarif	text	20
11	Status	Text	15

3. Identifikasi Database Pelanggan

Tabel 3. 4 Tabel Database *History* Transaksi

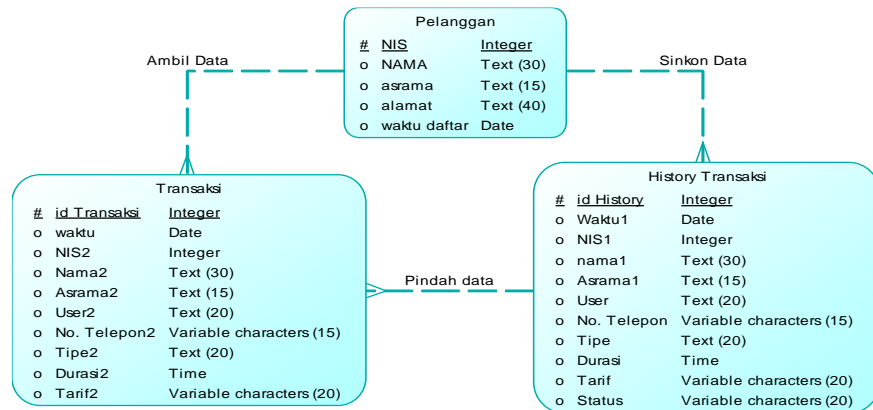
No	Nama	Tipe Data	Panjang
1	NIS (key)	integer	20
5	Nama	Text	30
6	Asrama	Text	15
7	Waktu Daftar	Date	-
8	Alamat	Text	40

b. Pemodelan Database

Dalam pemodelan database akan dibuat ERD (*Entity Relationship Diagram*), hal ini untuk mempermudah saat menggambarkan data yang memiliki relasi dalam bentuk sebuah desain. Di sini akan terdiri dari *Conseptual Data Model* (CDM) dan *Phisical Data Model* (PDM).

1. *Conseptual Data Model* (CDM)

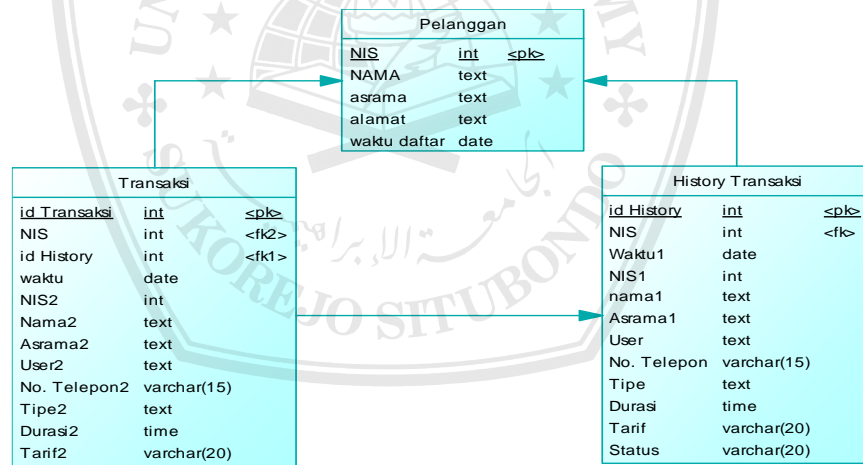
Pada bagian ini CDM merupakan suatu keseluruhan struktur logis dari database. CDM dari perancangan sistem ini seperti berikut.



Gambar 3. 6 Conceptua Data Model (CDM)

2. Physical Data Model (PDM)

Pada bagian ini PDM merupakan organisasi fisik dalam suatu format grafik yang menghasilkan catatan modifikasi dalam rancangan database. PDM dari perancangan sistem *Smart Calling* ini seperti gambar di bawah berikut.



Gambar 3. 7 Physical Data Model (PDM)

3.3.4 Desain User Interface

Desain *User Interface* (UI) adalah proses perancangan tampilan dan interaksi sistem yang berfokus pada keindahan dan kemudahan penggunaan, termasuk penentuan tata letak, pemilihan warna, dan elemen visual lainnya untuk memastikan pengguna dapat berinteraksi dengan sistem secara intuitif [22].

1. Ilustrasi Desain *User*



The image shows a user registration form with a black background and white text. At the top, it says 'USER01'. Below that is a label 'NIS' followed by a white input field. Then, a label 'NOMOR TELEPON' followed by another white input field. Below the input fields, the text '00.00.00' and 'Rp. 0' is displayed. At the bottom, there are two buttons: a yellow 'Audio Call' button and a red 'Video Call' button.

Gambar 3. 8 Ilustrasi Desain *User*

Jendela user adalah awal dari input data telepon santri. Menu ini akan terkonfigurasi dengan Aplikasi Whatsapp Desktop. User/santri harus input NIS dan Nomor Telepon. NIS harus sudah terdaftar terlebih dahulu di sistem, dan Nomor telepon harus terdaftar di Whatsapp. Jika sudah terdaftar, santri bisa memilih tipe telepon, apakah *Audio call* atau *Video call*. Durasi akan berjalan jika telepon sedang berlangsung dan Tarif berjalan otomatis mengikuti durasi telepon.

2. Ilustrasi Desain Transaksi



The image shows a transaction data table with a black background and white text. The table has columns for 'id', 'Waktu', 'User', 'NIS', 'Nomor Telepon', 'Durasi', 'Tarif', and 'Status'. There are two rows of data. The first row has id 1, Waktu 2024-12-23 15:32:11, User User01, NIS 2015010077, Nomor Telepon 082230387800, Durasi 00.20.12, Tarif Rp.10.000, and Status with an unchecked checkbox. The second row has id 2, Waktu 2024-12-23 15:32:11, User User01, NIS 2015010077, Nomor Telepon 082230387800, Durasi 00.20.12, Tarif Rp.10.000, and Status with a checked checkbox. A modal window is overlaid on the table, titled 'SIWARTEL' and 'LUNAS'. It contains fields for 'NIS' (2021017766), 'NAMA' (BADRUS SHOLEH), 'ASRAMA' (MQ.13), 'TAGIHAN' (Rp. 10.000), and 'BAYAR/FS' (Rp. 10.000). A yellow 'BAYAR' button is at the bottom right of the modal.

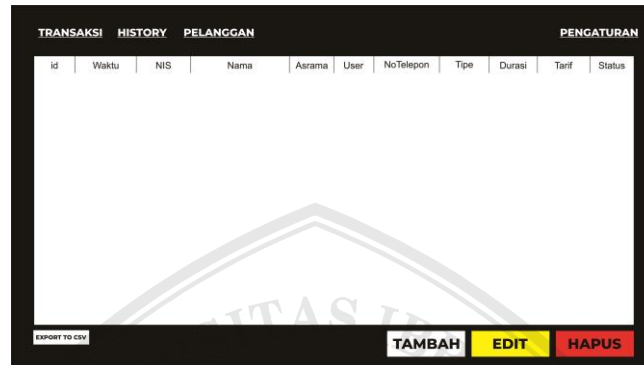
id	Waktu	User	NIS	Nomor Telepon	Durasi	Tarif	Status
1	2024-12-23 15:32:11	User01	2015010077	082230387800	00.20.12	Rp.10.000	<input type="checkbox"/>
2	2024-12-23 15:32:11	User01	2015010077	082230387800	00.20.12	Rp.10.000	<input checked="" type="checkbox"/>

Gambar 3. 9 Ilustrasi Desain Data Transaksi

Jendela Transaksi akan mengelola data yang dilakukan di menu User. Semua inputan yang dilakukan User akan masuk ke Transaksi untuk dilakukan pembayaran. Untuk melakukan pembayaran Admin/petugas tinggal mengklik

dua kali data yang akan dibayar, maka sistem akan menampilkan jendela detail pembayaran, yang berisi NIS, Nama, Asrama, Tagihan, Input pembayaran, dan Total. Jika lunas maka muncul Lunas di kolom Total.

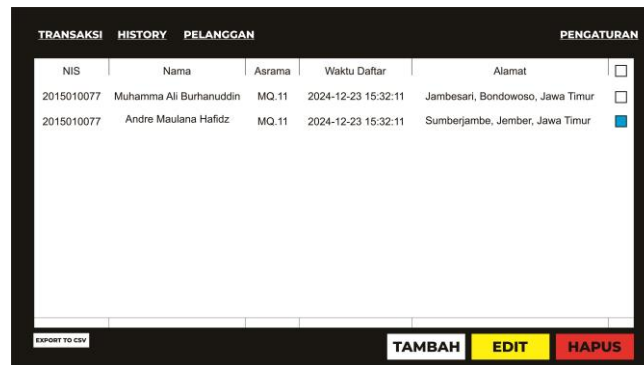
3. Ilustrasi Desain *History*



Gambar 3. 10 Ilustrasi Desain Data *History*

Jendela *History* akan menjadi tempat menampung semua data yang telah dilakukan pembayaran di data transaksi. Di sini data yang disimpan menjadi lebih kompleks. Selain menampung data dari Transaksi, di *History* juga bisa melakukan tambah data secara manual, untuk mengantisipasi terjadinya Error di menu Transaksi. Selain itu Admin juga bisa melakukan Edit data, dan Hapus data. Data-data ini juga bisa diekspor ke *Excel* dan dibuka di Microsoft *Excel*.

4. Ilustrasi Desain Data Pelanggan



Gambar 3. 11 Ilustrasi Desain Data Pelanggan

Jendela data pelanggan hanya dapat diakses oleh admin/petugas. Di sini petugas bisa melakukan input data, edit data, dan tambah data. Data-data yang harus di isi meliputi NIS sebagai *primarykey*, Nama, Asrama, Waktu Daftar (otomatis dilakukan sistem), dan Alamat rumah. Pada menu edit yang tidak bisa diedit adalah NIS, sebab NIS merupakan *primarykey*. Sehingga jika ada kesalahan pada NIS maka dapat melakukan tambah data ulang, dan menghapus data yang lama. Data-data di jendela Pelanggan ini juga bisa di *Ekport* ke *Excel* dan bisa dilihat di Microsoft *Excel*.



BAB IV

IMPLEMENTASI SISTEM

4.1 Kontruksi Sistem

4.1.1. Kebutuhan Sistem

Agar sistem *Smart Calling* ini dapat berjalan secara optimal, diperlukan perangkat keras dan perangkat lunak yang sesuai. Sistem ini terdiri dari dua sisi, yaitu sisi petugas/admin (server) dan sisi user (*client*). Berikut adalah kebutuhan sistem minimal dan yang diperlukan:

1. Kebutuhan Perangkat Keras

Dalam kebutuhan Perangkat keras akan dijelaskan dalam tabel. Tabel ini memaparkan spesifikasi minimum dan rekomendasi untuk menjalankan Sistem *Smart Calling* dengan optimal, baik di sisi user (santri/petugas) maupun admin. Setiap komponen berpengaruh terhadap performa dan stabilitas sistem.

Tabel 4.1 Kebutuhan Perangkat Keras Sistem

Komponen	Minimum	Rekomendasi
Prosesor	Intel Core i3 atau setara	Intel Core i5/i7
RAM	4 GB	8 GB
Penyimpanan	128 GB SSD/HDD	256 SSD
Resolusi Layar	1366 x 768	1920 x 1080
Kamera & Mikrofon	Internal/eksternal	Internal HD kamera + <i>mic</i> eksternal
Koneksi	Jaringan LAN/Wi-Fi	LAN stabil

2. Kebutuhan Perangkat Lunak

Kebutuhan Perangkat Lunak menjadi salah satu yang penting untuk di jelaskan. Tabel di bawah menjelaskan perangkat lunak atau aplikasi yang dibutuhkan untuk menjalankan sistem *Smart Calling*, termasuk versi minimum agar fungsionalitas berjalan baik.

Tabel 4.2 Kebutuhan Perangkat Keras Sistem

Perangkat Lunak	Versi Minimum/Keterangan
Sistem Operasi	Windows 10 64-bit
Python	Versi 3.9 atau lebih tinggi
SQLite	Bawaan Python
PyAutoGUI	Library Python untuk automasi GUI
Tkinter	Bawaan Python untuk antarmuka desktop
WhatsApp Desktop	Versi terbaru, terhubung dengan akun resmi

4.1.2. Instalasi Sistem

Tahap instalasi sistem bertujuan untuk menyiapkan lingkungan agar aplikasi Sistem *Smart Calling* dapat berjalan dengan baik pada komputer client (user) maupun komputer petugas/admin. Sistem ini dibangun menggunakan bahasa Python dan membutuhkan beberapa pustaka tambahan serta konfigurasi khusus, seperti integrasi dengan WhatsApp Desktop dan koneksi ke server lokal. Instalasi dilakukan dalam dua sisi, yaitu sisi user (santri) dan sisi petugas/admin.

a. Instalasi di Sisi User (Santri)

1. Install Python. Unduh dan pasang Python versi minimal 3.9 dari situs resmi <https://python.org>. Saat instalasi, centang opsi "*Add Python to PATH*".

2. Install Library Tambahan. Gunakan pip untuk menginstal pustaka berikut:
pip install pyautogui requests
3. Pasang WhatsApp Desktop. Instal aplikasi WhatsApp Desktop dari situs resmi <https://www.whatsapp.com/download>. Login menggunakan akun WhatsApp santri atau admin.
4. Jalankan Script user.py. Aplikasi user dapat dijalankan langsung melalui terminal atau shortcut:python user.py. Antarmuka form akan tampil di pojok kiri atas layar. Santri cukup memasukkan NIS dan nomor WhatsApp tujuan, kemudian memilih jenis panggilan.
5. Inisialisasi Database Lokal (otomatis). Saat pertama dijalankan, script akan memanggil fungsi berikut untuk membuat database SQLite lokal:

```
1. def init_temp_db():
2.     conn = sqlite3.connect(TEMP_DB)
3.     c = conn.cursor()
4.     # Tabel untuk data gagal
5.     c.execute('''CREATE TABLE IF NOT EXISTS pending (
6.                 id INTEGER PRIMARY KEY AUTOINCREMENT,
7.                 waktu TEXT, user TEXT, nis TEXT,nomor_t
8.                 lepon TEXT,
9.                 tipe TEXT, durasi INTEGER, tarif INTEGER
10.            )''')
11.     # Tabel untuk konfigurasi
12.     c.execute('''CREATE TABLE IF NOT EXISTS config (
13.                 key TEXT PRIMARY KEY,value TEXT)''')
```

Segmen Program 4. 1 Buat Database Otomatis

Fungsi ini memastikan bahwa data transaksi tersimpan, terutama saat koneksi ke server terputus.

- b. Instalasi di Sisi Petugas/Admin

1. Instalasi Python dan SQLite. Sama seperti sisi user, admin perlu memasang Python versi 3.9+ dan SQLite yang sudah terintegrasi dalam Python.
2. Pasang Library Pendukung. Selain pyautogui dan requests, pastikan semua library yang digunakan di sipostel.py sudah terinstal:

```
pip install flask openpyxl pyautogui requests
```

3. Konfigurasi File sipostel.py dan user.py. Admin dapat mengatur tarif panggilan, IP server, dan hak akses pengguna melalui konfigurasi lokal atau GUI pengaturan sistem.
 4. Jalankan Server Aplikasi. File sipostel.py bertindak sebagai GUI admin. Admin cukup menjalankannya: `python sipostel.py`
Sistem akan menampilkan dashboard dengan berbagai menu: Transaksi, Pelanggan, Info, Pengaturan, dan Riwayat.
- c. Jalur Komunikasi (API)

Transaksi dari sisi user akan dikirim ke server admin menggunakan metode POST seperti salah satu contoh kode di bawah ini:

```
1. def save_call_data(waktu, user, nis, nomor_telepon, tipe, durasi, tarif):
2.     data = {"waktu": waktu, "user": user, "nis": nis,
3.           "nomor_telepon": nomor_telepon, "tipe": tipe, "durasi": durasi,
4.           "tarif": int(tarif.replace("Rp. ", "").replace(".", ""))}
5.     try:
6.         response = requests.post(f"{get_server_url()}/tambah_transaksi", json=data)
7.         if response.status_code == 200:
8.             print("Transaksi berhasil dikirim ke server admin.")
9.         else:
```

```
10.         raise Exception(response.text)
11.     except Exception as e:
12.         print(f"Gagal kirim data ke server, simpan lokal. E
    rror: {e}")
13.         simpan_data_sementara(data)
```

Segmen Program 4. 2 Kirim data ke Server (admin)

Jika server tidak dapat diakses, data akan disimpan ke pending dan dikirim ulang setiap beberapa detik:

```
1. def kirim_data_tertunda():
2.     conn = sqlite3.connect(TEMP_DB)
3.     c = conn.cursor()
4.     c.execute("SELECT * FROM pending")
5.     rows = c.fetchall()
6.     for row in rows:
7.         id_data, waktu, user, nis, nomor_telepon, tipe, dur
    asi, tarif = row
8.         data = {"waktu": waktu, "user": user, "nis": nis,
9.                 "nomor_telepon": nomor_telepon, "tipe": tipe,
10.                "durasi": durasi, "tarif": tarif}
11.         try:
12.             response = requests.post(f"{get_server_url()}/t
   ambah_transaksi", json=data)
13.             if response.status_code == 200:
14.                 print(f"Berhasil kirim data tertunda ID {id
    _data}")
15.                 c.execute("DELETE FROM pending WHERE id = ?
    ", (id_data,))
16.                 conn.commit()
17.         except Exception as e:
18.             print(f"Gagal kirim ID {id_data}, coba lagi nan
    ti: {e}")
19.     conn.close()
20.     root.after(10000, kirim_data_tertunda) # Ulangi setiap
    10 detik
```

Segmen Program 4. 3 Kirim ulang data tertunda ke Server (admin)

Dengan mengikuti prosedur instalasi di atas, sistem dapat langsung digunakan secara operasional di lingkungan wartel pondok. Proses ini dirancang agar mudah dilakukan oleh operator tanpa memerlukan keahlian teknis tinggi.

4.1.3. Segmen Program

Segmen program adalah bagian penting yang menunjukkan bagaimana sistem dibangun melalui kode-kode Python. Pada sistem *Smart Calling*, terdapat beberapa segmen program utama yang mendukung fungsionalitas sistem, antara lain: penghitungan tarif panggilan, automasi WhatsApp Desktop, pencatatan transaksi, dan pengiriman data ke server.

a. Timer dan Penghitungan Tarif Otomatis

Sistem menghitung durasi panggilan secara *real-time* sejak tombol panggilan ditekan, lalu menghitung tarif berdasarkan jenis panggilan (Audio atau Video). Kode ini digunakan untuk menghitung tarif dengan pembulatan ke kelipatan 500 atau 1000 rupiah tergantung jenis panggilan.

```
1. def calculate_cost(duration_seconds, rate_per_minute, tipe):
2.     minutes = duration_seconds / 60
3.     cost = minutes * rate_per_minute
4.     if tipe == "Video Call":
5.         return round(cost/1000)*1000
6.     elif tipe == "Audio Call":
7.         return round(cost/500)*500
```

Segmen Program 4. 4 *Timer* dan Pengaturan Tarif Otomatis

b. Automasi WhatsApp Desktop

Untuk melakukan panggilan otomatis, sistem mencari tombol dan ikon di WhatsApp Desktop menggunakan *pyautogui*, kemudian menjalankan klik secara otomatis. Automasi ini dijalankan pada sisi user (*user.py*) dan akan otomatis membuka layar WhatsApp, mengetik nomor tujuan, dan memilih jenis panggilan. Berikut contoh automasi tombol Video Call, dan automasi mengetik nomor WhatsApp:

```
1. # Klik tombol video
2. button_location = pyautogui.locateOnScreen('video_buttom.png', confidence=0.8)
3. if not button_location:
4.     messagebox.showerror("Error", "Gambar video_buttom.png tidak ditemukan!")
5.     unlock_controls()
6.     return
7. pyautogui.click(button_location)
```

Segmen Program 4. 5 Automasi Klik ikon Panggilan

```
1. pyautogui.click(number_icon_location)
2. pyautogui.typewrite(phone)
3. time.sleep(2)
```

Segmen Program 4. 6 Automasi Mengetik Nomor Telepon

c. Pendeteksi Otomatis Tombol Akhiri Panggilan

Untuk mendeteksi apakah panggilan sudah berakhir, sistem mendeteksi keberadaan tombol "akhiri_panggilan.png". Jika tombol hilang, sistem otomatis mengakhiri panggilan. Ini membuat sistem dapat berjalan mandiri tanpa perlu campur tangan manual dalam menghentikan panggilan.

```
1. cek_tombol_timer = {"aktif": False}
2. def cek_tombol_hilang():
3.     tombol_ada = False
4.     for _ in range(10): # Cek selama 10 detik
5.         try:
6.             lokasi = pyautogui.locateOnScreen('akhiri_panggilan.png', confidence=0.7)
7.             if lokasi:
8.                 tombol_ada = True
9.                 break
10.        except pyscreeze.ImageNotFoundException:
11.            pass
12.            time.sleep(1)
13.
14.    if not tombol_ada:
15.        print("Tombol akhiri hilang, akhiri otomatis.")
16.    end_call()
```

```
17. cek_tombol_timer["aktif"] = False # Reset flag agar bisa dicek lagi nanti
```

Segmen Program 4. 7 Pendeteksi Otomatis Tombol Akhiri Panggilan

d. Pencatatan dan Pengiriman Transaksi

Setelah panggilan berakhir, data akan dicatat dan dikirim ke server melalui API. Jika gagal, akan disimpan lokal. Fungsi `simpan_data_sementara()` untuk menyimpan data ke SQLite lokal jika gagal kirim ke server.

```
1. def simpan_data_sementara(data):
2.     conn = sqlite3.connect(TEMP_DB)
3.     c = conn.cursor()
4.     c.execute('INSERT INTO pending (waktu, user, nis, nomor_telepon, tipe, durasi, tarif)
5.             VALUES (?, ?, ?, ?, ?, ?, ?)',
6.             (data["waktu"], data["user"], data["nis"], data["nomor_telepon"],
7.             data["tipe"], data["durasi"], data["tarif"]))
8.     conn.commit()
9.     conn.close()
10.
```

Segmen Program 4. 8 Simpan data Sementara saat gagal terhubung ke server

e. Validasi NIS Sebelum Panggilan

Sebelum panggilan dilakukan, sistem memverifikasi apakah NIS santri sudah terdaftar. Jika tidak ditemukan, maka panggilan tidak akan diproses.

```
1. def cek_nis_terdaftar(nis):
2.     global server_ip
3.     try:
4.         response = requests.get(f"{get_server_url()}/cek_nis",
5.                                 params={"nis": nis})
6.         if response.status_code == 200:
7.             data = response.json()
8.             return data["status"] == "found"
9.         else:
10.            return False
11.     except Exception as e:
```

```
12.     messagebox.showerror("Error", f"Gagal cek NIS dari s
       erver:\n{e}")
13.     return False
```

Segmen Program 4. 9 Validasi NIS sebelum Panggilan

Sistem *Smart Calling* pada sisi petugas (admin/operator) dibangun menggunakan bahasa pemrograman Python dengan antarmuka berbasis Tkinter serta didukung API lokal menggunakan Flask untuk menerima transaksi dari sisi user. Aplikasi ini dirancang modular dengan pembagian menu seperti Transaksi, History, Pelanggan, Info, dan Pengaturan. Berikut adalah segmen-segmen program penting yang terdapat di bagian petugas/admin:

a. Menu Transaksi

Menu ini berfungsi menampilkan semua transaksi panggilan yang dikirim oleh user. Data transaksi ditampilkan dalam bentuk tabel (*Treeview*) dan di-refresh otomatis setiap 3 detik. Berikut contoh inisialisasi refresh:

```
1. def auto_refresh():
2.
3.     global auto_refresh_id, refresh_enabled
4.     try:
5.
6.         if refresh_enabled and halaman_aktif == "TRANSAKSI"
           and 'transaksi_app' in globals() and transaksi_app:
7.
8.             transaksi_app.populate_table()
9.             transaksi_app.update_column_width()
10.    except Exception as e:
11.        print("Auto-refresh error:", e)
12.
13.    if refresh_enabled and root and root.winfo_exists():
14.        auto_refresh_id = root.after(3000, auto_refresh)
15.    else:
16.        auto_refresh_id = None
```

Segmen Program 4. 10 Refresh otomatis di Menu Transaksi

b. API Penerima Transaksi

Sisi admin bertindak sebagai server yang menerima data transaksi dari user melalui endpoint POST /tambah_transaksi. Jika API diaktifkan setelah login, data akan disimpan ke database lokal siwartel.db.

```
1. @app.route('/tambah_transaksi', methods=['POST'])
2. def tambah_transaksi():
3.     if not api_enabled:
4.         return jsonify({"status": "error", "message": "API t
   idak aktif"}), 403
5.
6.     data = request.json
7.     try:
8.         with sqlite3.connect(DB_PATH) as conn:
9.             cursor = conn.cursor()
10.            cursor.execute("""
11.                INSERT INTO transaksi (waktu, user, nis, nom
   or_telepon, tipe, durasi, tarif)
12.                VALUES (?, ?, ?, ?, ?, ?, ?)
13.                """, (
14.                    data['waktu'], data['user'], data['nis'],
15.                    data['nomor_telepon'], data['tipe'], data['d
   urasi'], data['tarif']
16.                ))
17.            conn.commit()
18.            return jsonify({"status": "success"}), 200
19.     except Exception as e:
20.         return jsonify({"status": "error", "message": str(e)
   }), 500
```

Segmen Program 4. 11 API Penerima Transaksi dari *User*

c. Pemindahan Transaksi ke History

Saat *logout* atau saat petugas memilih untuk menyelesaikan transaksi, data pada tabel transaksi dipindahkan ke history. Semua data transaksi dilengkapi nama santri dan informasi asrama yang diambil dari tabel pelanggan.

```
1. def selesaikan_semua_transaksi_otomatis():
2.     try:
3.         conn = sqlite3.connect("siwartel.db")
```

```
4.         conn.row_factory = sqlite3.Row
5.         cursor = conn.cursor()
6.
7.         # Ambil semua transaksi
8.         cursor.execute("SELECT * FROM transaksi")
9.         transaksi_rows = cursor.fetchall()
10.
11.        # Ambil semua data pelanggan sekali saja (untuk efis
12.        iensi)
13.        cursor.execute("SELECT nis, nama, asrama FROM pelang
14.        gan")
15.
16.        pelanggan_map = {row["nis"]: (row["nama"], row["asra
17.        ma"]) for row in cursor.fetchall()}
18.
19.        for row in transaksi_rows:
20.            try:
21.                nis = row["nis"]
22.                durasi = int(row["durasi"] or 0)
23.                tagihan = int(row["tarif"] or 0)
24.                waktu = row["waktu"]
25.                tipe = row["tipe"]
26.                user = row["user"]
27.                nomor = row["nomor_telepon"] # bukan "nomor
28.                _telepon"
29.                nama, asrama = pelanggan_map.get(nis, ("-
30.                ", "-"))
31.
32.                bayar = 0
33.                sisa = tagihan
34.                status = "Belum Lunas" if sisa > 0 else "Lun
35.                as"
36.
37.                cursor.execute("""
38.                INSERT INTO history (
39.                    waktu, nis, nama, asrama, user, nomo
40.                    r_telepon, tipe,
41.                    durasi, tagihan, dibayar, sisa, stat
42.                    us, petugas
43.                ) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?,
44.                ?, ?, ?)
45.                """, (
46.                    waktu, nis, nama, asrama, user, nomor, t
47.                    ipe,
48.                    durasi, tagihan, bayar, sisa, status, cu
49.                    rrent_user or "-"
```

```
40.         ))
41.
42.         except Exception as e:
43.             print("Gagal menyimpan transaksi ke history:
", e)
44.             print(" >> data:", dict(row))
45.
46.         cursor.execute("DELETE FROM transaksi")
47.         conn.commit()
48.         conn.close()
49.
50.         print("Semua transaksi diselesaikan otomatis saat lo
gout.")
51.
52.     except Exception as e:
53.         print("Gagal memproses transaksi otomatis:", e)
54.
```

Segmen Program 4. 12 Pemindahan Transaksi ke *History*

d. Menu *History* Transaksi

Menu *History* digunakan untuk melihat semua histori panggilan yang sudah diselesaikan. Akses data dapat dibatasi berdasarkan kolom dan status tertentu sesuai dengan hak akses pengguna (*akses_history*, *akses_status*).

```
1. def get_user_akses_history():
2.     try:
3.
4.         with open("users.json") as f:
5.             users = json.load(f)
6.             for u in users:
7.                 if u.get("username") == current_user:
8.                     if u.get("role") == "admin" and not u.get("t
erapkan_admin", False):
9.                         return ["semua"] # admin tanpa batas
10.                    return u.get("akses_history", [])
11.     except:
12.
13.         pass
14.     return []
```

Segmen Program 4. 13 Pemindahan Transaksi ke *History*

e. Menu Pelanggan

Petugas dapat menambah, mengedit, dan menghapus data santri dari tabel pelanggan. Data ini penting karena saat user menginput NIS, sistem memverifikasi ke database admin untuk mendapatkan nama dan asrama.

```
1. self.cursor.execute(  
2.     "INSERT INTO pelanggan (NIS, Nama, Asrama, WaktuDaftar,  
3.     Alamat) VALUES (?, ?, ?, ?, ?)",  
4.     data  
5. )
```

Segmen Program 4. 14 Verifikasi Database Pelanggan

f. Menu Info

Menu ini menyimpan informasi penting yang bisa ditampilkan kepada pengguna (*user*). Admin dapat menambahkan, mengedit, dan menyembunyikan info. Berikut struktur tabel info:

```
1.     CREATE TABLE IF NOT EXISTS info (  
2.         id INTEGER PRIMARY KEY AUTOINCREMENT,  
3.         tanggal TEXT,  
4.         judul TEXT,  
5.         konten TEXT  
6.     )  
7.     """)  
8. cursor.execute("PRAGMA table_info(info)")  
9. kolom_info = [row[1] for row in cursor.fetchall()]  
10. if "hidden" not in kolom_info:  
11.     cursor.execute("ALTER TABLE info ADD COLUMN hidden INTEG  
12.         ER DEFAULT 0")  
13. conn.commit()  
14. conn.close()
```

Segmen program 4. 15 Struktur Tabel Info

g. Pengaturan Hak Akses Petugas

Sistem memiliki manajemen hak akses berbasis file `users.json`, yang memungkinkan admin membatasi akses tertentu berdasarkan role atau

pengaturan personal. Dengan desain ini, seluruh pengelolaan transaksi dan sistem panggilan dapat dilakukan secara efisien dan terkontrol oleh petugas/admin pondok pesantren, tanpa memerlukan akses ke sisi user.

```
1. def get_user_fitur(kategori):
2.     try:
3.         with open("users.json") as f:
4.             users = json.load(f)
5.         for u in users:
6.             if u.get("username") == current_user:
7.                 role = u.get("role", "")
8.                 terapkan_admin = u.get("terapkan_admin", False)
9.
10.                # Admin tidak dibatasi
11.                if role == "admin" and not terapkan_admin:
12.                    return {"import": True, "tambah": True, "
edit": True, "hapus": True, "history": True, "pelanggan": True}
13.
14.                # Admin dibatasi atau user biasa
15.                return u.get(kategori, {})
16.     except Exception as e:
17.         print(f"[get_user_fitur ERROR]: {e}")
18.     return {}
```

Segmen program 4. 16 Pengaturan Hak Akses Petugas

4.2 Skenario Pengujian

Pengujian sistem dilakukan untuk memastikan bahwa seluruh fitur berjalan sesuai dengan kebutuhan yang telah ditetapkan. Metode yang digunakan dalam pengujian ini adalah *black box testing*, yaitu metode pengujian perangkat lunak yang berfokus pada fungsi sistem dari sisi pengguna, tanpa mengetahui struktur kode program di dalamnya. Tujuan dari pengujian ini adalah untuk memastikan bahwa setiap masukan (input) yang diberikan ke sistem akan menghasilkan keluaran (output) yang sesuai dengan spesifikasi.

Pengujian dilakukan terhadap beberapa fitur utama baik pada sisi user (santri) maupun petugas (admin/operator). Berikut ini adalah skenario pengujian yang telah dirancang dan dilaksanakan:

Tabel 4.3 Skenario Pengujian Sistem *Smart Calling*

No	Fitur yang Diuji	Input yang Diberikan	Input yang Diberikan
1	Login petugas	Username dan password valid	Masuk ke dashboard petugas
2	Login petugas dengan data salah	Username/password salah	Muncul pesan "Login gagal"
3	Form panggilan user	NIS terdaftar dan nomor WhatsApp valid	Sistem membuka WhatsApp dan memulai panggilan
4	Form panggilan user	NIS tidak terdaftar	Muncul pesan "NIS tidak ditemukan"
5	Pemilihan tipe panggilan	Audio Call / Video Call	Tarif dihitung sesuai tipe panggilan
6	Penghitungan durasi & tarif	Panggilan selama 2 menit	Tarif muncul sesuai waktu dan tipe
7	Pengiriman data transaksi	Setelah panggilan selesai	Data transaksi muncul di menu Transaksi petugas
8	Transaksi gagal kirim (offline)	Jaringan mati saat pengiriman	Data disimpan lokal di tabel `pending`
9	Pengiriman ulang otomatis	Sistem online kembali	Data `pending` dikirim ke admin otomatis
10	Validasi input pelanggan	Data pelanggan kosong / duplikat	Muncul notifikasi validasi
11	Menu Riwayat	Klik menu History	Menampilkan semua transaksi lunas
12	Ekspor data transaksi	Klik tombol ekspor Excel	File `riwayat_transaksi.xlsx` berhasil dibuat
13	Pengaturan tarif panggilan	Ubah tarif Video dan Audio di menu Pengaturan	Tarif baru diterapkan ke perhitungan user
14	Akses terbatas crew	Login sebagai crew tanpa hak tertentu	Menu sensitif tidak ditampilkan

Pengujian ini menunjukkan bahwa seluruh fungsi sistem dapat berjalan sesuai dengan yang diharapkan, baik dalam kondisi normal maupun ketika terjadi

gangguan seperti koneksi terputus. Dengan pendekatan black box, pengujian difokuskan pada respon sistem terhadap masukan nyata dari pengguna, bukan pada kode internal sistem.

4.3 Pengujian

Pengujian sistem dilakukan setelah seluruh komponen sistem berhasil diimplementasikan, baik pada sisi petugas maupun sisi *user*. Tujuan pengujian ini adalah untuk memastikan bahwa sistem berjalan sesuai dengan kebutuhan fungsional yang telah dirancang. Pengujian dilakukan secara langsung di lingkungan wartel Pondok Pesantren Ma'hadul Qur'an, menggunakan pendekatan *black box* testing untuk menguji apakah *output* sistem sesuai dengan *input* yang diberikan, tanpa memperhatikan struktur kode internal.

4.3.1. Cara Kerja Sistem

Sistem *Smart Calling* bekerja dalam dua sisi terpisah namun saling terhubung, yaitu sisi user (digunakan oleh santri) dan sisi admin/petugas (digunakan oleh operator wartel). Keduanya berkomunikasi melalui jaringan lokal menggunakan protokol HTTP sederhana.

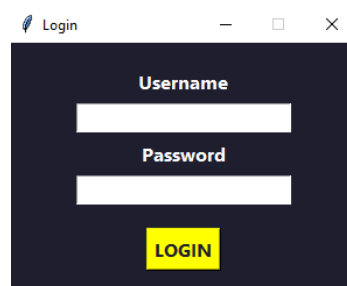
1. Alur Kerja Sisi User (Santri):
 - a. Santri membuka aplikasi user_Siwartel yang menampilkan form sederhana.
 - b. Santri mengisi NIS dan nomor WhatsApp tujuan.
 - c. Sistem melakukan validasi ke server apakah NIS tersebut terdaftar.
 - d. Santri memilih jenis panggilan: *Audio Call* atau *Video Call*.
 - e. Sistem akan membuka WhatsApp Desktop secara otomatis dan memulai panggilan menggunakan PyAutoGUI.

- f. Selama panggilan berlangsung, sistem menampilkan durasi waktu secara *real-time*.
- g. Setelah tombol "Akhir Panggilan" hilang dari layar (dideteksi otomatis), panggilan dianggap selesai.
- h. Sistem menghitung tarif panggilan berdasarkan durasi dan jenis panggilan.
- i. Data panggilan dikirim secara otomatis ke server. Jika gagal, data disimpan lokal dan dikirim ulang saat koneksi kembali.



Gambar 4. 1 Tampilan Jendela *User*

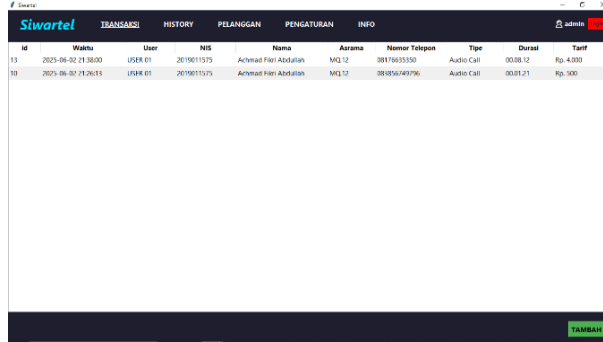
2. Alur Kerja Sisi Petugas/Admin:
 - a. Petugas login ke sistem menggunakan sipostel.py.



Gambar 4. 2 Tampilan Jendela *Login* Petugas/admin

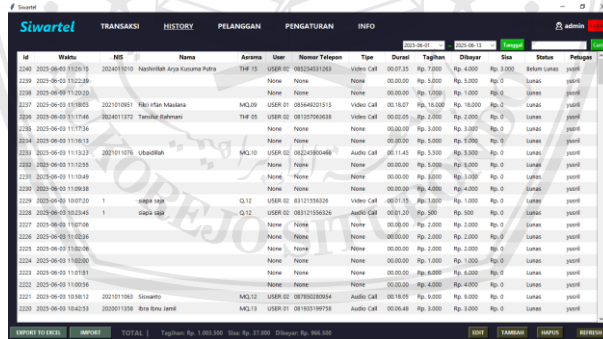
- b. Setelah login, petugas diarahkan ke dashboard utama yang menampilkan menu: Transaksi, History, Pelanggan, Info, dan Pengaturan.

- c. Pada menu Transaksi, petugas dapat melihat daftar panggilan yang masuk dari santri secara *real-time*.



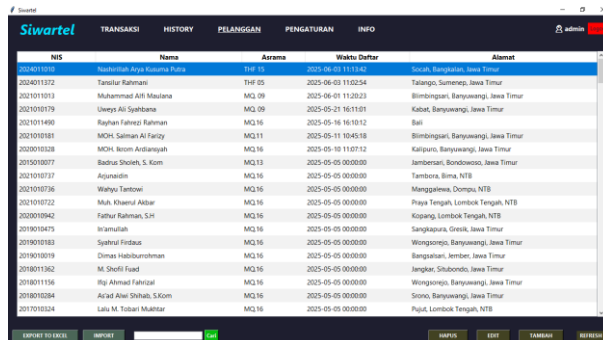
Gambar 4. 3 Tampilan Jendela Transaksi

- d. Petugas dapat menandai panggilan sebagai lunas dan memproses data ke histori.
- e. Pada menu History, petugas dapat melihat semua riwayat panggilan, memfilter berdasarkan tanggal, dan mengekspor data ke Excel.



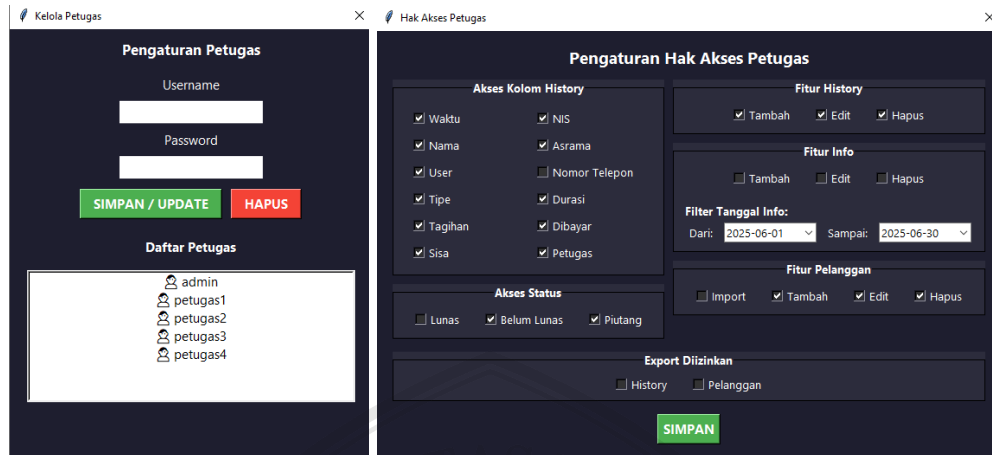
Gambar 4. 4 Tampilan Jendela History

- f. Di menu Pelanggan, petugas dapat mengelola data santri (tambah, edit, hapus).



Gambar 4. 5 Tampilan Jendela Pelanggan

- g. Di menu Pengaturan, admin dapat mengubah tarif panggilan, hak akses pengguna, serta informasi aplikasi.



Gambar 4. 6 Jendela Pengaturan Petugas dan Pengaturan Hak Akses

Dengan alur kerja ini, sistem dapat menggantikan proses pencatatan manual menjadi otomatis dan terdokumentasi secara digital, meningkatkan efisiensi layanan wartel serta mengurangi risiko kesalahan pencatatan.

4.3.2. Hasil Pengujian

Hasil pengujian sistem *Smart Calling* diperoleh melalui pelaksanaan *Black Box Testing* berdasarkan skenario-skenario yang telah dirancang sebelumnya. Pengujian dilakukan untuk setiap fitur utama sistem baik pada sisi petugas maupun *user*. Tujuan dari pengujian ini adalah untuk memastikan bahwa seluruh fungsionalitas sistem dapat bekerja dengan baik sesuai dengan kebutuhan dan perancangan awal. Pengujian dilakukan dalam kondisi nyata di lingkungan wartel pesantren, menggunakan akun WhatsApp aktif serta data santri yang sebenarnya.

Dari seluruh skenario yang diuji, tidak ditemukan fitur yang gagal berfungsi. Semua hasil aktual sesuai dengan ekspektasi dan kebutuhan sistem.

Hal ini menunjukkan bahwa sistem telah memenuhi syarat kelayakan untuk digunakan dalam kegiatan operasional layanan wartel berbasis digital, serta mampu menggantikan proses manual dengan sistem yang otomatis dan terdokumentasi. Berikut adalah tabel rekapitulasi hasil pengujian:

Tabel 4.4 Hasil Pengujian Sistem *Smart Calling*

No	Fitur yang Diuji	Input yang Diberikan	Output yang Diharapkan	Status
1	Login petugas	Username dan password valid	Masuk ke dashboard petugas	Berhasil
2	Login petugas dengan data salah	Username/password salah	Muncul pesan “Login gagal”	Berhasil
3	Form panggilan user	NIS terdaftar dan nomor WhatsApp valid	Sistem membuka WhatsApp dan memulai panggilan	Berhasil
4	Form panggilan user	NIS tidak terdaftar	Muncul pesan “NIS tidak ditemukan”	Berhasil
5	Pemilihan tipe panggilan	Audio Call / Video Call	Tarif dihitung sesuai tipe panggilan	Berhasil
6	Penghitungan durasi & tarif	Panggilan selama 2 menit	Tarif muncul sesuai waktu dan tipe	Berhasil
7	Pengiriman data transaksi	Setelah panggilan selesai	Data transaksi muncul di menu Transaksi petugas	Berhasil
8	Transaksi gagal kirim (offline)	Jaringan mati saat pengiriman	Data disimpan lokal di tabel `pending`	Berhasil
9	Pengiriman ulang otomatis	Sistem online kembali	Data `pending` dikirim ke admin otomatis	Berhasil
10	Validasi input pelanggan	Data pelanggan kosong / duplikat	Muncul notifikasi validasi	Berhasil
11	Menu Riwayat	Klik menu History	Menampilkan semua transaksi lunas	Berhasil
12	Ekspor data transaksi	Klik tombol ekspor Excel	File `riwayat_transaksi.xlsx` berhasil dibuat	Berhasil
13	Pengaturan tarif panggilan	Ubah tarif Video dan Audio di menu Pengaturan	Tarif baru diterapkan ke perhitungan user	Berhasil
14	Akses terbatas crew	Login sebagai crew tanpa hak tertentu	Menu sensitif tidak ditampilkan	Berhasil

4.4 Maintenance

Pemeliharaan sistem (*maintenance*) merupakan tahap penting setelah implementasi dan pengujian sistem selesai dilakukan. Tujuan dari pemeliharaan adalah untuk menjaga agar sistem tetap berjalan dengan baik, menyesuaikan terhadap perubahan kebutuhan pengguna, serta memperbaiki bug atau kendala teknis yang mungkin terjadi dalam jangka panjang. Sistem Smart Calling di Warung Telepon Asrama Ma'hadul Qur'an dirancang agar mudah dalam perawatan dan pengembangannya, baik dari sisi teknis maupun operasional.

Jenis-jenis pemeliharaan yang diterapkan dalam sistem ini antara lain:

1. Pemeliharaan Korektif (*Corrective Maintenance*)

Pemeliharaan ini dilakukan untuk memperbaiki kesalahan (bug) atau malfungsi yang terjadi saat sistem sudah digunakan. Contohnya adalah ketika terjadi error dalam pencatatan durasi panggilan, gagal ekspor file, atau koneksi antara user dan server terputus. Sistem menyediakan log transaksi dan pencatatan error secara lokal yang dapat digunakan untuk menelusuri dan memperbaiki kesalahan secara cepat.

2. Pemeliharaan Adaptif (*Adaptive Maintenance*)

Pemeliharaan adaptif dilakukan jika terjadi perubahan kebutuhan pengguna atau kebijakan pondok pesantren. Misalnya, jika tarif panggilan perlu diubah, admin dapat langsung menyesuaikannya melalui menu Pengaturan tanpa harus memodifikasi kode program secara langsung. Selain itu, penambahan fitur informasi baru, atau pengelolaan hak akses pengguna tambahan juga dapat dilakukan dari antarmuka aplikasi.

3. Pemeliharaan Perfektif (*Perfective Maintenance*)

Jenis pemeliharaan ini dilakukan untuk meningkatkan performa dan kenyamanan penggunaan sistem. Contohnya adalah penyempurnaan tampilan antarmuka, penambahan tombol ekspor cepat, perbaikan posisi tombol di form, hingga pengembangan ke sistem *multi-user* dan *multi-lokasi*. Karena sistem ini dikembangkan menggunakan Python dan struktur modular, proses *upgrade* dapat dilakukan dengan menambahkan modul atau memperbarui file tertentu saja.

4. Pemeliharaan Preventif (*Preventive Maintenance*)

Preventive maintenance bertujuan mencegah terjadinya gangguan sistem di masa depan. Dalam sistem ini, misalnya dengan melakukan *backup* rutin database (siwartel.db) dan file konfigurasi (users.json, tarif.json) agar dapat dipulihkan jika terjadi kerusakan atau kehilangan data. Selain itu, sistem secara berkala juga membersihkan transaksi dengan durasi pendek atau tarif nol untuk mencegah penumpukan data yang tidak valid.

Dengan strategi pemeliharaan ini, sistem *Smart Calling* dapat digunakan secara berkelanjutan dan stabil dalam mendukung layanan komunikasi santri, serta siap dikembangkan lebih lanjut bila dibutuhkan oleh pihak pondok di masa yang akan datang.

BAB V

PENUTUP

5.1 Kesimpulan

Dari hasil perancangan, implementasi, dan pengujian yang telah dilakukan, dapat disimpulkan bahwa sistem *Smart Calling* berhasil dibangun untuk menggantikan proses layanan wartel manual di Asrama Ma'hadul Qur'an. Sistem ini memungkinkan santri melakukan panggilan dengan mudah melalui *form input* sederhana, dan petugas dapat memantau, mencatat, mengelola transaksi secara otomatis serta terstruktur.

Fitur-fitur utama seperti automasi WhatsApp Desktop, penghitungan tarif, histori transaksi, pengelolaan pelanggan, serta pengaturan sistem berjalan sesuai kebutuhan. Pengujian menggunakan metode *black box* menunjukkan bahwa seluruh fungsi sistem berjalan dengan baik dan sesuai harapan.

5.2 Saran

Agar sistem dapat terus bermanfaat dan dikembangkan ke skala yang lebih luas, ada beberapa saran untuk peneliti selanjutnya:

1. Dapat mengembangkan sistem ini menuju versi berbasis Android atau web yang memungkinkan biaya lebih terjangkau.
2. Tingkatkan keamanan sistem, misalnya dengan *otentikasi* ganda untuk admin.

DAFTAR PUSTAKA

- [1] Y. Ababil and L. Tanti, "Rancang Bangun Aplikasi Promosi Penjualan Hp Menggunakan *Augmented Reality* Berbasis Android," *Information Technology and Cyber Crime (ITCC)*, vol. 2, no. 1, 2023, doi: 10.59840/itcc.v2i1.222.
- [2] M. A. Azizi, "Konsep *Technological Determinism* Dalam Penelitian Komunikasi dan Pengaruhnya Terhadap Kehidupan Manusia," *Universal Grace Journal*, vol. 1, no. 1, 2023.
- [3] A. Muchasan, Nur Syam, and Anis Humaidi, "Pemanfaatan Teknologi di Pesantren (Dampak dan Solusi dalam Konteks Pendidikan)," *INOVATIF: Jurnal Penelitian Pendidikan, Agama, dan Kebudayaan*, vol. 10, no. 1, 2024, doi: 10.55148/inovatif.v10i1.849.
- [4] S. Sudahri, "Tradisi Komunikasi Di Pondok Pesantren Tradisional Dan Pondok Pesantren Modern," *Mediakom*, vol. 1, no. 2, 2023, doi: 10.32528/mdk.v1i2.1573.
- [5] I. T. buana and H. Zakaria, "Implementasi Sistem Informasi Pembayaran Spp Berbasis Web Menggunakan Metode *Extreme Programming* Dengan Notifikasi *Whatsapp Gateway*," *Logic: Jurnal Ilmu Komputer dan Pendidikan*, vol. 1, no. 2, 2023.
- [6] S. Wibawa, "Analisis Chatbot Otomatisasi Tugas Administratif dan Manajemen Dalam Lingkungan Digital Dengan Menggunakan Python," *INSANTEK*, vol. 4, no. 1, 2023, doi: 10.31294/insantek.v4i1.2190.
- [7] S. Gustiani, "*Research and Development (R&D) Method as a Model Design in Educational Research and Its Alternatives*," *HOLISTICS JOURNAL*, vol. 11, no. 2, 2019.
- [8] A. Abdul Wahid Sekolah Tinggi Manajemen Informatika dan Komputer Sumedang, "Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi." [Online]. Available: <https://www.researchgate.net/publication/346397070>

- [9] M. Guntur Akbar, H. Witriyono, Y. Apridiyansyah, and D. Abdullah, "Implementation Of The Inter Tk Package, Sub-Process And Os In The Network Management Application Development With Python Programming Language Penerapan Paket Tk Inter, Sub Proses Dan Os Pada Pembuatan Aplikasi Manajemen Jaringan Dengan Bahasa Pemrograman Python," *JURNAL KOMITEK*, vol. 3, no. 1, pp. 187–196, doi: 10.53697/jkomitek.v3i1.
- [10] A. Fergina, S. Alpariji, and A. Sujjada, "Perancangan Chatbot Whatsapp Natural Language Processing Untuk Pelayanan Digital Universitas Nusa Putra," *Jurnal Riset Sistem Informasi Dan Teknik Informatika (JURASIK)*, vol. 9, no. 2, pp. 697–709, 2024, [Online]. Available: <https://tunasbangsa.ac.id/ejurnal/index.php/jurasik>
- [11] S. H. Bariyah and K. A. N. Imania, "Pengembangan Virtual Assistant Chatbot Berbasis Whatsapp Pada Pusat Layanan Informasi Mahasiswa Institut Pendidikan Indonesia - Garut," *JURNAL PETIK*, vol. 8, no. 1, pp. 66–79, Mar. 2022, doi: 10.31980/jpetik.v8i1.1575.
- [12] J. I. H. Pangemanan, "Sistem Informasi Adalah: Ciri, Fungsi, Tujuan, Komponen, dan Contoh," *Media Indonesia*.
- [13] N. Abidin, A. Aini, and M. Izzuddin, "Pengembangan Sistem Informasi Manajemen Keuangan Sekolah berbasis Web menggunakan Whatsapp Gateway," *Jurnal Informatika Terpadu*, vol. 9, no. 2, 2023, doi: 10.54914/jit.v9i2.797.
- [14] M. R. S. Alfarizi, M. Z. Al-farish, M. Taufiqurrahman, G. Ardiansah, and M. Elgar, "Penggunaan Python Sebagai Bahasa Pemrograman untuk Machine Learning dan Deep Learning," *Karya Ilmiah Mahasiswa Bertauhid (KARIMAH TAUHID)*, vol. 2, no. 1, 2023.
- [15] R. Rosaly and A. Prasetyo, "Pengertian Flowchart Beserta Fungsi dan Simbol-simbol Flowchart yang Paling Umum Digunakan," *Nesabamedia*, vol. 2, 2019.

- [16] Tawar and P. Y. Pangestu, "Analisis dan Perbaikan Proses Bisnis dalam Perspektif Sistem Informasi," *Bincang Sains dan Teknologi*, vol. 2, no. 01, pp. 30–40, Apr. 2023, doi: 10.56741/bst.v2i01.294.
- [17] A. Riyandi Saputra and D. Adira Titania, "Pengembangan Website Fakultas Teknik Menggunakan *Ramework Codeigniter* Versi 3 Dengan Metode Agile."
- [18] M. Irsyad Awaludin and B. Dwi Waluyo, "Perancangan Sistem Kontrol Suhu Oven Listrik Menggunakan Metode PID Design of Electric Oven *Temperature Control System Using PID Method*," Dec. 2023.
- [19] M. K. Yuyun Khairunisa, Eri MARDiani., S. Kom. ,M. Kom. Nur Rahmansyah, and Agus Ambarwari, "Dasar-Dasar Pemrograman," 2023.
- [20] M. Rahmadan and C. E. Gunawan, "Perancangan Data Flow Diagram Aplikasi Tabungan Sampah Pt Pusri Palembang," 2024.
- [21] M. Rahmadan and C. E. Gunawan, "Perancangan Data Flow Diagram Aplikasi Tabungan Sampah Pt Pusri Palembang," 2024.
- [22] S. Mufti Prasetyo and F. A. Ariestia, "OKTAL : Jurnal Ilmu Komputer dan *Science Mengenal User Interface dan User Experience* dalam Dunia Desain dan Teknologi," vol. 2, no. 10, 2023, [Online]. Available: <https://journal.mediapublikasi.id/index.php/oktal>

LAMPIRAN-LAMPIRAN

Lampiran 1. Bukti Telah Melaksanakan Penelitian



**DIVISI POSTELKOM
ASRAMA MA'HADUL QUR'AN PUTRA
Pondok Pesantren Salafiyah Syafi'iyah
Sukorejo Situbondo**

Sekretariat : Kantor Postelkom Asrama Ma'hadul Qur'an Putra PP. Salafiyah Syafi'iyah Sukorejo Situbondo

Nomor : 0828/ /Postel.MQ.Pa/V/2025

Lamp : -

Hal : **Surat Pengantar Bukti Telah Melakukan Penelitian**

Kepada

Yth. **Pihak Berkepentingan**

di -

Tempat

Assalamu'alaikum Warahmatullahi Wabarakatuh

Salam silaturahmi, semoga kita senantiasa berada di lindungan-Nya dan mendapat barokah atas pengabdian kita di pesantren. Kami selaku pengelola Warung Telepon (Wartel) di Asrama Ma'hadul Qur'an Pondok Pesantren Salafiyah Syafi'iyah Sukorejo, dengan ini menyampaikan bahwa:

Nama : Badrus Sholeh
NPM : 2021501009
Program Studi : Ilmu Komputer
Lembaga : Fakultas Sains dan Teknologi Universitas Ibrahimi

Telah melakukan kegiatan penelitian di lingkungan wartel Asrama Ma'hadul Qur'an Putra. Demikian surat ini kami buat, atas perhatiannya disampaikan terimakasih, *Jazaa Kumullahu Khoiron.*

Wassalamu'alaikum Warahmatullahi Wabarakatuh

Sukorejo, 01 Mei 2025
Koor. Postelkom,

M. Yusril Afandi

Lampiran 2. Transkrip Wawancara dengan petugas

Peneliti : Badrus Sholeh
Narasumber : Ustadz M. Yusril Afandi (Petugas Wartel Asrama Ma'hadul Qur'an Putra)
Tanggal : 29 April 2025
Lokasi : Ruang Wartel Asrama Ma'hadul Qur'an, Sukorejo, Situbondo

Peneliti : Assalamu'alaikum Ustadz, terima kasih sudah meluangkan waktu. Bisa diceritakan sedikit bagaimana sistem wartel di asrama ini berjalan sekarang?

Petugas : Wa'alaikumussalam. Saat ini kami masih menggunakan telepon rumah, salahsatunya tipe Ezitel 6588. Santri yang mau menelepon tinggal datang, melakukan panggilan, lalu melakukan pembayara ke petugas.

Peneliti : Apakah telepon tersebut bisa membatasi durasi panggilan santri?

Petugas : Tidak. Telepon ini hanya membatasi maksimal dua jam panggilan, tapi itu jarang sekali sampai segitu. Tidak ada pengaturan otomatis, jadi durasi sebenarnya hanya diketahui santri, bukan kami sebagai petugas.

Peneliti : Bagaimana cara Petugas mencatat pembayaran atau rekap pendapatan?

Petugas : Kami catat manual berdasarkan pengakuan santri tentang berapa menit mereka menelepon. Kadang kami juga lihat perkiraan waktu, tapi tidak akurat lalu dihitung biayanya. Jika ada yang hutang atau kembaliannya ada di sini kami tulis di buku Hutang Piutang. Kalau bayarnya pas/lunas tidak masuk ke dalam catatan harian, hanya direkap bulanan.

Peneliti : Apakah pernah terjadi kendala atau kecurangan dalam proses ini?

Petugas : Pernah, sering malah. Ada santri yang bilang nelpon cuma lima menit, padahal sebenarnya lebih. Karena kami tidak punya data durasi yang pasti, ya kami hanya bisa percaya. Itu bisa mengurangi pendapatan wartel.

Peneliti : Selain itu, kendala lain apa yang sering para petugas rasakan?

Petugas : History panggilan di telepon bisa dihapus sama santri. Jadi kami tidak punya bukti kalau mereka menelepon lebih lama. Lalu provider selain Telkomsel sangat sulit sekali. Kartu seperti XL beberapa bulan belakangan tidak bisa digunakan di telepon rumah ini, jadi hanya Telkomsel yang normal.

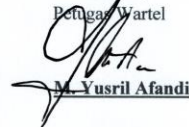
Peneliti : Kalau boleh membayangkan, sistem seperti apa yang Ustadz harapkan untuk memudahkan pekerjaan di wartel?

Petugas : Ya kalau bisa ada sistem yang otomatis mencatat durasi panggilan, menghitung tarif, dan menyimpan data transaksi. Jadi kami tinggal mengawasi dan menerima pembayaran. Kalau bisa juga ada rekap otomatis ke komputer, biar laporan lebih rapi dan tidak perlu menghitung manual.

Peneliti : Terima kasih banyak, Ustadz, atas waktunya dan informasinya.

Petugas : Sama-sama, semoga sistem barunya bisa segera dipakai.

Narasumber,
Petugas Wartel



M. Yusril Afandi

Lampiran 3. Transkrip Wawancara dengan Santri

Peneliti : Badrus Sholeh
Narasumber : Taufiqur Rohman (Santri Asrama Ma'hadul Qur'an No.05)
Tanggal : 30 April 2025
Lokasi : Asrama Ma'hadul Qur'an, Sukorejo, Situbondo

Peneliti : Assalamu'alaikum, terima kasih sudah bersedia diwawancarai. Bisa ceritakan bagaimana biasanya proses menelepon keluarga di wartel ini?

Santri : Wa'alaikumussalam. Biasanya saya datang ke wartel, minta izin petugas, antri, lalu langsung menelepon dari telepon yang ada. Habis itu saya bilang ke petugas berapa lama saya menelepon, baru bayar sesuai hitungan.

Peneliti : Jadi selama menelepon, apakah ada batasan waktu yang memberi tahu kalau waktu sudah habis?

Santri : Tidak ada. Tapi aturannya sampai 15 menit. Kalau kelihatan lama banget yang antri di belakang akan menegur, tapi kalau asyik ngobrol bisa kelewatan. Biasanya sih petugas percaya saja sama kita.

Peneliti : Menurut kamu, ada masalah tidak dengan cara seperti itu?

Santri : Ya kadang kalau kita lupa waktu, bayar jadi lebih banyak. Tapi saya juga pernah dengar ada teman yang ngomongnya menelepon sebentar padahal lama. Jadi mungkin petugas agak susah mengontrol.

Peneliti : Kalau kualitas layanan sekarang, menurut kamu bagaimana?

Santri : Ya lumayan, cuma kadang yang sering susah itu kalau orang tua pakai kartu selain Telkomsel, pakatnya sering habis. Terus kalau mau lihat history panggilan, di teleponnya bisa dihapus, jadi tidak selalu kelihatan riwayatnya.

Peneliti : Kalau misalnya nanti ada sistem baru yang otomatis mencatat durasi panggilan dan menghitung biaya, kamu setuju?

Santri : Setuju banget. Jadi kita sama-sama enak, petugas nggak ribet, kita juga jelas bayar sesuai data, bukan perkiraan.

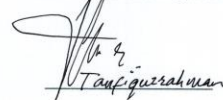
Peneliti : Ada tambahan harapan untuk sistem baru ini?

Santri : Kalau bisa cepat prosesnya, nggak bikin antri lama, dan aman datanya. Dan kalau bisa Video call juga, biar bisa lihat wajah orang tua.

Peneliti : Baik, terima kasih atas waktunya dan masukannya.

Santri : Sama-sama.

Narasumber,
Santri Ma'hadul Qur'an



Lampiran 5. Surat Keterangan Hasil Pemeriksaan Plagiasi

PONDOK PESANTREN SALAFIYAH SYAFI'YAH SUKOREJO
UNIVERSITAS IBRAHIMY
PERPUSTAKAAN IBRAHIMY
N P P . 3 5 1 2 1 4 2 F 2 0 0 6 5 6 7
Jl. KHR. Syamsul Arifin No. 1-2 PO. Box. 2 Kode Pos. 68374 Phone (0338) 432666 Fax. (0338) 453068
SUMBEREJO BANYUPUTIH SITUBONDO JAWA TIMUR

**SURAT KETERANGAN
HASIL PEMERIKSAAN PLAGIASI**

Yang bertanda tangan di bawah ini

Nama : Muhammad Ali Ridla, M.Kom.
Jabatan : Kepala Perpustakaan

Menyatakan dengan sebenarnya bahwa:

NPM : 2021501009
Nama : BADRUS SHOLEH
Fakultas : Sains dan Teknologi
Prodi : Ilmu Komputer
Kecamatan : JAMBESARI DARUS SHOLAH
Kabupaten : KAB. BONDOWOSO
Provinsi : JAWA TIMUR
Judul Skripsi : SISTEM SMART CALLING DI WARUNG
TELEPON ASRAMA MA'HADUL QUR'AN
PONDOK PESANTREN SALAFIYAH
SYAFI'YAH SUKOREJO

Dengan dosen Pembimbing :

1. Zaehol Fatah, M.Kom.
2. Adi Susanto, M. Kom.

Telah dilakukan cek plagiasi di Perpustakaan Universitas Ibrahimi dengan persentase plagiasi terakhir sebesar **15%**.

Demikian Surat Keterangan ini dibuat untuk dipergunakan sebagaimana mestinya.

Sukorejo, 13 Juli 2025
Kepala Perpustakaan,



Muhammad Ali Ridla, M.Kom.



UU ITE No 11 Tahun 2008 Pasal 5 Ayat 1
"Informasi Elektronik dan/atau Dokumen Elektronik
dan/atau hasil cetaknya merupakan alat bukti yang sah."

www.lib.ibrahimiy.ac.id


library@ibrahimiy.ac.id

Perpustakaan Ibrahimi

@ibrahimiy_lib

Lampiran 6. Kartu Bimbingan Skripsi

KARTU BIMBINGAN TUGAS AKHIR / SKRIPSI
FAKULTAS SAINS & TEKNOLOGI
UNIVERSITAS IBRAHIMY
TAHUN AKADEMIK 2024/2025



NPM : 2021501009
 Nama : Badrus Sholeh
 Program Studi : Ilmu Komputer
 Judul TA / Skripsi : Sistem Smart calling di Warung
 Telekomunikasi Asrama Mahasiswa di
 Pondok Pesantren Salafiyah Spektipura

= CATATAN =

1. Dalam penyusunan Laporan TA / Skripsi, mahasiswa harus berkonsultasi dengan pembimbingnya secara bertahap.
2. Pada setiap konsultasi, kartu bimbingan harus dibawa dan diisi oleh pembimbing.
3. Mahasiswa wajib Konsultasi selama penyusunan Laporan TA / Skripsi ke pembimbing Minimal 6 x.
4. Waktu bimbingan dimulai sejak tahapan proposal sampai laporan kegiatan.
5. Skedul TA / Skripsi dapat dilihat pada buku panduan penyusunan Laporan Kegiatan.

Pembimbing I : <u>Adi Susanto, M. Kom</u>				Pembimbing II : <u>Zachol Fatah, M. Kom</u>			
NO	TANGGAL	CATATAN	PARAF	NO	TANGGAL	CATATAN	PARAF
1	08/05/2025	Berulah Bagian Awal Abstrak		1	10/05/2025	Revisi Bab I, II	
2	17/05/2025	Kata bahasa Inggris dimiringkan		2	15/05/2025	Revisi Bab III, IV	
3	05/06/2025	Beri Penjelasan Pada ketidakan Pengingat ke bagian awal di bab IV		3	02/06/2025	ACC Bab I, II Revisi	
4	20/06/2025	Paragraf Judul di halaman 57		4	27/06/2025	Revisi Bab V ACC BAB III	
5	16/07/2025	Harus Alinea paragraf di kesimpulan Harus bagian poin 2 di Summary		5	18/07/2025	Revisi Daftar Pustaka	
6	09/08/2025	ACC SKRIPSI		6	02/08/2025	ACC SKRIPSI	

Lampiran 7. LOA Jurnal IDEALIS



IDEALIS
E-ISSN: 2684-7280

JURNAL IDEALIS (inDonEsiA Journal Information System)
FAKULTAS TEKNOLOGI INFORMASI, UNIVERSITAS BUDI LUHUR
 Jl. Ciledug Raya, RT.10/RW.2, Petukangan Utara, Pesangrahan, Jakarta Selatan,
 Daerah Khusus Ibukota Jakarta, Indonesia 12260
 URL: <https://jorn.fi.budiluhur.ac.id/index.php/IDEALIS/index>, email: idealis.fi@budiluhur.ac.id



SK No. 225/E/KPT/2022, Berlaku
 hingga Vol 8 No.1 Tahun 2025

LETTER OF ACCEPTANCE (LOA)

No. 019/IDEALIS-UBL-FTI/07/25

Jakarta, 13 Juli 2025

Kepada Yth.

Badrus Sholeh, Adi Susanto, Zaehol Fatah

Program Studi Ilmu Komputer, Fakultas Sains dan Teknologi, Universitas Ibrahimiy,
 Situbondo, Indonesia

Terima kasih telah mengirimkan artikel ilmiah untuk diterbitkan di Jurnal IDEALIS dengan data artikel sebagai berikut:

- Judul : Implementasi Sistem Smart Calling Berbasis Python Untuk Otomatisasi Layanan Wartel Di Lingkungan Pesantren
- Penulis : 1. Badrus Sholeh,
 2. Adi Susanto,
 3. Zaehol Fatah

Berdasarkan hasil review, artikel tersebut dinyatakan **DITERIMA** untuk dipublikasikan pada **Jurnal IDEALIS (Indonesia Journal Information System)** pada **Volume 8 Nomor 2 Juli 2025**.

Demikian informasi ini disampaikan, dan atas perhatiannya kami ucapkan terima kasih.

Hormat kami,
 Editor in Chief Jurnal IDEALIS

IDEALIS
INDONESIA JOURNAL INFORMATION SYSTEM



Safitri Juanita, S.Kom, M.T.I
 Scopus ID: 57226491480

CURRICULUM VITAE



Badrus Sholeh lahir di Jember pada 23 Agustus 2002, tumbuh dalam keluarga yang sederhana namun sarat akan nilai-nilai keikhlasan, kerja keras, dan keteguhan. Sejak muda, ia telah menunjukkan ketertarikan yang besar terhadap teknologi dan dunia informasi. Ketertarikan tersebut kemudian membawanya melanjutkan pendidikan tinggi di Program Studi Ilmu Komputer, Fakultas Sains dan Teknologi, Universitas Ibrahimi Situbondo.

Selain aktif di Pesantren dan Kampus, penulis juga dikenal sebagai pribadi yang terlibat dalam berbagai kegiatan organisasi. Ia pernah menjadi bagian dari Postelkom MQ Putra, LPMI, Al-Khidmah, IKSASS Bondowoso, SSDC dan komunitas teknologi seperti Robot Station. Dari sinilah karakter kepemimpinan, disiplin tinggi, dan tanggung jawabnya tumbuh dan berkembang.

Skripsi ini merupakan refleksi dari semangat penulis dalam menggabungkan nilai pengabdian dengan solusi teknologi. Melalui karya ini, ia berharap mampu memberi kontribusi nyata dalam pemanfaatan teknologi di lingkungan pesantren.

Penulis percaya bahwa ilmu adalah jalan pulang menuju cahaya, dan teknologi hanyalah alat, bukan tujuan. Ia bercita-cita menjadi pribadi yang mampu menebar manfaat lewat karya, bukan hanya prestasi.

Di balik layar komputernya, penulis adalah seorang perenung. Di balik baris-baris kode, ada doa-doa sunyi yang ia selipkan. Dan dalam setiap jerih payah, ada wajah keluarga, sahabat, dan masa depan yang selalu menjadi alasan untuk terus diperjuangkan. Karena ia percaya, selagi terus berusaha, semua bisa jadi nyata.