RANCANG BANGUN SISTEM BEL MIKROKONTROLER BERBASIS ANDROID PADA SMP SETIA BUDI SUNGAILIAT

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan
Sarjana Terapan Politeknik Manufaktur Negeri Bangka Belitung

Diusulkan oleh

Rangga Febri Hadi Tanjung	NIM 1062124
Zacky Alfarizi	NIM 1062130

POLITEKNIK MANUFAKTUR NEGERI
BANGKA BELITUNG
TAHUN 2023/2024

LEMBAR PENGESAHAN

RANCANG BANGUN SISTEM BEL MIKROKONTROLER BERBASIS ANDROID PADA SMP SETIA BUDI SUNGAILIAT

Oleh :

Rangga Febri Hadi Tanjung / 1062124
Zacky Alfarizi / 1062130

Laporan akhir ini telah disetujui dan disahkan sebagai salah satu syarat kelulusan Program Sarjana Terapan Politeknik Manufaktur Negeri Bangka Belitung

Menyetujui,

Pembimbing 1	Pembimbing 2
Yang Agita Rindri, S.Kom.,M.Eng.	Linda Fujiyanti, S.T., M.T.I.
NIP.198609282022032003	NIP.198109262014042001
Penguji 1	**Penguji 2**
Riki Afriansyah , S.T., M.T.	Elisa Mayang Sari, S.Pd., M.Pd.
NIP. 199004042019031013	NIP.19951128202203018

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan di bawah ini :

Nama Mahasiswa	:	Rangga Febri Hadi Tanjung	NPM	:	1062124
Nama Mahasiswa	:	Zacky Alfarizi	NPM	:	1062130

Dengan Judul : Rancang Bangun Sistem Bel Mikrokontroler Berbasis Android Pada SMP Setia Budi Sungailiat

Menyatakan bahwa laporan akhir ini adalah hasil kerja saya sendiri dan bukan merupakan plagiat. Pernyataan ini saya buat dengan sebenarnya dan bila ternyata dikemudian hari ternyata melanggar pernyataan ini, saya bersedia menerima sanksi yang berlaku.

Sungailiat, 24 July 2024

Nama Mahasiswa	Tanda Tangan
1. Rangga Febri Hadi Tanjung	.....
2. Zacky Alfarizi	.....

ABSTRAK

Pada dunia Pendidikan, bel adalah salah satu inovasi penting yang digunakan untuk menginformasikan pergantian waktu atau peristiwa penting. Sistem bel mikrokontroler Berbasis Android ini menjadi suatu bentuk pengembangan dari inovasi bel tersebut. Penelitian ini dilakukan di SMP Setia budi Sungailiat dengan tujuan untuk meningkatkan efisiensi dan efektivitas penggunaan bel sekolah serta mengembangkan aplikasi yang mudah digunakan dikarenakan berdasarkan pengamatan secara langsung menunjukkan bahwa penggunaan bel listrik di sekolah tersebut kurang efisien. Dalam beberapa situasi, saat guru atau staf tidak ada di ruangan, bel sering kali dibunyikan oleh siswa, yang membuat penggunaannya menjadi kurang efektif. Melalui pendekatan studi kasus dan metode prototype, penelitian ini mengembangkan aplikasi yang memungkinkan pengontrolan bel sekolah melalui aplikasi android. Aplikasi ini dilengkapi dengan fitur untuk atur jadwal bel yang akan dibunyikan serta atur waktu real time secara otomatis. Hasil penelitian menunjukkan bahwa sistem mampu menyimpan jadwal bel dan membunyikannya tanpa delay sesuai dengan waktu yang ditentukan serta sesuai dengan suara yang telah dipilih dengan jangkauan efektif sejauh 6 meter. Hasil pengujian kuisioner terhadap responden memperoleh nilai persentase sebesar 80%.

Kata Kunci: Inovasi, Bel, Android, Mikrokontroler

ABSTRACT

In the world of education, the bell is one of the important innovations used to inform the turn of time or important events. This Android-based microcontroller bell system is a form of development of the bell innovation. This research was conducted at SMP Setia budi Sungailiat with the aim of improving the efficiency and effectiveness of using the school bell and developing an application that is easy to use because based on direct observation, it shows that the use of electric bells at the school is less efficient. In some situations, when teachers or staff are not in the room, the bell is often rung by students, which makes its use less effective. Through a case study approach and prototype method, this research developed an application that allows controlling the school bell through an android application. The application is equipped with features to set the bell schedule to be rung as well as set the real time time automatically. The results show that the system is able to save the bell schedule and ring it according to the specified time and according to the selected sound. The results of questionnaire testing on respondents obtained a percentage value of 80%.

Keywords: Innovation, Bell, Android,Micro Controller

DAFTAR ISI

LEMBAR PENGESAHAN ii

PERNYATAAN BUKAN PLAGIAT iii

DAFTAR ISI vi

DAFTAR GAMBAR viii

DAFTAR TABEL ix

BAB I PENDAHULUAN 1

1.1 Latar Belakang 1
1.2 Rumusan Masalah 2
1.3 Tujuan 2

BAB II DASAR TEORI 3

2.1 Tinjauan Pustaka 3
2.2 *Internet of Things* (IoT) 6
2.3 Teknologi Bel 7
2.4 Aplikasi Berbasis Android 9

BAB III METODE PELAKSANAAN 12

3.1 Model Rancangan 12
3.1.1 Metode Pelaksanaan Studi Kasus 12
3.1.2 Metode *Prototype* 13
3.1.3 *Use Case Diagram* 14
3.2 Analisis Kebutuhan 15
3.2.1. Hardware 15
3.2.2. *Software* 15
3.3 Perancangan Sistem 16
3.4 *Activity Diagram* 17

BAB IV PEMBAHASAN 19

4.1 Perancangan Aplikasi dan Alat 19
4.1.1 *Requirements* 19
4.1.2 *Quick Design* 20
4.1.3 *Build Prototype* 22

4.1.4 User Evaluation 22
4.1.5 Refining Prototype 23
4.1.6 Implement and Maintain 24
4.2 Pembuatan alat dan pemrograman sistem 25
4.2.1 Rangkaian modul RTC DS3231 25
4.2.2 Rangkaian Modul DF Player 25
4.2.3 Pemrograman alat 26
4.3 Perangkat Keras (*Hardware)* 29
4.4 Tampilan Aplikasi 29
4.4.1. Tampilan *Home* 29
4.4.2. Tampilan Menu Tambah Jadwal 30
4.4.3. Tampilan Edit 31
4.5 Pengujian 31
4.5.1. Pengujian Delay 32
4.5.2. Hasil Pengujian Kuesioner 33
4.5.3 Pengujian Jarak 35
BAB V KESIMPULAN DAN SARAN 36
5.1. Kesimpulan 36
5.2. Saran 36
DAFTAR PUSTAKA 38
LAMPIRAN 1 41
LAMPIRAN 2 43
LAMPIRAN 3 45

DAFTAR GAMBAR

Gambar 3.1 *Flowchart* Studi Kasus 13
Gambar 3.2 *Use Case Diagram* 14
Gambar 3.3 Perancangan Sistem 16
Gambar 3.4 *Activity Diagram* 17
Gambar 4.1 *Home Page* 20
Gambar 4.2 *Menu Bell* 20
Gambar 4.3 *Menu Add Alarm* 21
Gambar 4.4 *Menu Database* 21
Gambar 4.5 *Build Prototype* 22
Tabel 4.1 *User Evaluation* 23
Gambar 4.7 *Refining Prototype* 24
Gambar 4.8 *Implement and Maintain* 24
Gambar 4.9 Rangkaian Modul RTC DS3231 25
Gambar 4.10 Rangkaian Modul DfPlayer 26
Gambar 4.11 Library Mikrokontroler 26
Gambar 4.12 Inisialisasi Objek 27
Gambar 4.13 Komunikasi JSON 27
Gambar 4.14 *Setup* 28
Gambar 4.15 Fungsi *Looping* 28
Gambar 4.16 *Hardware* 29
Gambar 4.17 Tampilan *Home* 30
Gambar 4.18 Tampilan Tambah Jadwal 31
Gambar 4.19 Tampilan Edit 31

DAFTAR TABEL

Tabel 4.1 *User Evaluation* 23
Tabel 4.2 Pengujian Delay 32
Tabel 4.3 Pertanyaan Kuesioner 33
Tabel 4.4 Hasil Kuesioner 34
Tabel 4,5 Hasil Pengujian Jarak 35

BAB I
PENDAHULUAN

1.1 Latar Belakang

Bel atau Lonceng Sekolah adalah suatu perangkat yang sering ditemukan di sekolah. Perangkat ini merupakan alat bantu untuk memberikan instruksi kepada seluruh murid dan guru dalam melaksanakan kegiatan tertentu[1]. Awalnya sekolah menggunakan lonceng tradisional, namun sekarang kebanyakan sekolah sudah menggunakan bel listrik. Saat ini, bel sekolah yang tersedia sudah semakin canggih, seperti memiliki fitur otomatisasi yang memungkinkan petugas atau guru tidak perlu lagi memencet bel setiap masuk, istirahat, ganti jam pelajaran, ataupun saat waktu pulang[2].

Pemilihan judul ini didasari dari pengamatan secara langsung yang dimana pemakaian bel listrik di sekolah tersebut kurang efisien dikarenakan guru-guru atau staf tidak selalu berada di ruang guru. Dalam beberapa kondisi, ketika guru atau staff tidak ada di ruangan, bel akan dibunyikan oleh siswa yang dimana hal tersebut tidak efisien terhadap penggunaan bel sekolah. Selain itu, perangkat bel listrik hanya mampu menampilkan suara *buzzer* atau serangkaian nada yang ada pada mesin bel, namun tidak memiliki arti atau informasi tertentu sehingga sulit untuk mengerti oleh seluruh siswa maupun guru di sekolah[1]

Selain itu, meskipun perangkat bel listrik yang ada saat ini sudah dilengkapi dengan kemampuan untuk memutar suara dari file MP3 sebagai pengganti suara buzzer, masih terdapat beberapa kendala yang menghambat fungsionalitasnya. Salah satunya adalah kebutuhan untuk mengatur jadwal bel secara manual oleh operator. Proses ini memerlukan waktu dan perhatian ekstra, terutama ketika ada perubahan jadwal mendadak seperti kegiatan khusus atau peringatan hari besar. Ketergantungan pada operator ini meningkatkan risiko kesalahan pengaturan jadwal, yang pada akhirnya dapat mengganggu kelancaran aktivitas sekolah.

Melalui proyek Rancangan Aplikasi *Smart Bell* ini diharapkan dapat meningkatkan efisiensi waktu, mendukung fleksibilitas penggunaan ruang kelas, dan memberikan

kontribusi positif terhadap pengelolaan waktu pembelajaran di SMP Setia Budi Sungailiat.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah di jelaskan, terdapat beberapa masalah yang perlu diselesaikan yaitu:

1. Bagaimana merancang sistem bel mikrokontroler berbasis android dapat mudah dipahami dan digunakan oleh pengguna?
2. Bagaimana mengembangkan sistem bel dapat digunakan dimanapun dengan efisien untuk SMP Setia Budi Sungailiat?
3. Bagaimana membuat perangkat bel mikrokontroler dapat terkoneksi dengan aplikasi android sebagai *remote*?
4. Bagaimana memastikan aplikasi mampu menyesuaikan jadwal bel secara otomatis berdasarkan perubahan yang terjadi di kalender akademik atau jadwal pelajaran SMP Setia Budi?
5. Bagaimana mengoptimalkan kinerja perangkat keras dan perangkat lunak bel agar tetap stabil dan andal dalam kondisi operasional sehari-hari?

1.3 Tujuan

Berdasarkan rumusan masalah yang telah dijelaskan, tujuan dari penelitian ini adalah:

1. Membuat rancangan sistem bel mikrokontroler berbasis android yang mudah digunakan oleh pengguna.
2. Membuat perangkat bel mikrokontroler yang dapat terkoneksi dengan aplikasi android.
3. Memodifikasi bel saat ini menjadi bel mikrokontroler berbasis android yang fleksibel dan efisien untuk digunakan di SMP Setia Budi Sungailiat.
4. Mengembangkan fitur otomatisasi pada sistem bel mikrokontroler yang dapat menyesuaikan jadwal bel berdasarkan perubahan jadwal pelajaran atau kalender akademik di SMP Setia Budi.
5. Mengoptimalkan kinerja perangkat keras dan perangkat lunak bel agar tetap stabil dan andal dalam penggunaan sehari-hari di lingkungan sekolah.

BAB II
DASAR TEORI

2.1 Tinjauan Pustaka

Penulis membuat tinjauan pustaka dengan merujuk pada penelitian sebelumnya yang relevan dengan topik penelitian ini. Di bawah ini terdapat hasil dari penelitian yang telah dilakukan sebelumnya:

Tabel 2.1 Tabel Tinjauan Pustaka

No	Judul	Hasil
a)	Aplikasi bel sekolah otomatis berbasis *Arduino* dilengkapi dengan output suara [1]	Penelitian ini bertujuan untuk meningkatkan efisiensi dan kemudahan dalam mengelola waktu belajar mengajar di sekolah melalui pengembangan sistem bel sekolah otomatis. Pembuatan bel otomatis dilakukan dengan menggunakan perangkat arduino yang dilengkapi dengan komponen lain sebagai output suara.
b)	Rancang bangun aplikasi *mobile* notifikasi berbasis *android* untuk mendukung kinerja di instansi pemerintahan [3]	Penelitian ini mengulas tentang pengembangan aplikasi Android yang bertujuan untuk mempermudah pegawai dalam menerima informasi dan notifikasi secara otomatis. Aplikasi ini menggunakan *Google Cloud Messaging* dan *MySQL* untuk mendukung pengelolaan data.
c)	Bel Sekolah Otomatis Berbasis *Arduino* yang Dikontrol Menggunakan Aplikasi *Mobile* [2]	Fokus penelitian ini adalah pada pengembangan *prototype* aplikasi bel sekolah yang memanfaatkan teknologi IoT untuk mempermudah pengelolaan jadwal bel. Alat ini masih menggunakan *buzzer* sebagai output suara dan

		menggunakan aplikasi *Mobile* sebagai alat kontrolnya.
d)	Rancang bangun bel sekolah otomatis berbasis *personal computer* (PC) [4]	Penelitian ini membahas tentang pengembangan bel sekolah yang dapat dibunyikan secara otomatis melalui komputer dan akan meringankan petugas untuk membunyikan bel sekolah setiap harinya.
e)	Rancang Bangun Sistem Bel Sekolah otomatis Berbasis *Mikrokontroler* menggunakan ISD 4003 [5].	Penelitan ini membahas pengembangan bel otomatis dengan menggunakan ISD 4003. Hasil dari penelitian ini menunjukkan bahwa sistem secara keseluruhan mampu menampilkan jadwal sekolah yang berlangsun. Speaker dapat mengeluarkan suara yang sesuai dengan perintah yang ditampilkan oleh LCD.
f)	Perancangan jam digital dan sistem bel otomatis pada sekolah dengan teknik counter berbasis mikrokontroler [6]	Penelitian ini mengembangkan sistem bel sekolah otomatis dengan mengimplementasikan teknik *counter* pada pengoperasian jadwal pembunyian bel dengan akurat dan presisi.
g)	Perancangan otomatisasi bel sekolah dengan *autopower* mengunakan *interface* berbasis *android* [7]	Studi ini bukan membuat bel yang otomatis tapi merancang sistem yang dapat mengontrol bel yang ada di sekolah agar dapat dibunyikan melalui aplikasi *mobile*.
h)	Sistem bel otomatis terprogram berbasis *raspberry pi* [8]	Penelitian ini berhasil merancang sistem pengotomatisan bel menggunakan *raspberry pi* yang terprogram dengan

		sirine yang menjadi output suara dari bel tersebut.
i)	*Prototype* bel cerdas cermat berbasis *arduino uno* [9]	Dalam penelitian ini, peneliti merancang bel yang dibunyikan dengan *push button* yang mana *buzzer* menjadi output suara dari bel tersebut.
j)	Sistem informasi penjadwalan bel sekolah otomatis berbasis mikrokontroler [10].	Penelitian ini berhasil mengembangkan sistem bel sekolah yang mampu menghasilkan output suara dalam format mp3 yang dapat dikontrol melalui aplikasi pada komputer sekolah.

Berdasarkan hasil kajian pustaka yang telah kami kaji, yakni mencakup berbagai penelitian terkait Bel Mikrokontroler Berbasis Andoird. Beberapa fitur unggulan dari penelitian-penelitian tersebut adalah penggunaan yang fleksibel dan mudah untuk diakses, eksplorasi teknologi IoT, serta upaya meningkatkan kenyamanan akses bel sekolah. Namun, ada beberapa kekurangan yang dapat diidentifikasi dari penelitian-penelitian tersebut.Kelemahan yang dapat diidentifikasi dari tinjauan pustaka tersebut meliputi:

a) Beberapa penelitian melakukan pendekatan berbasis IoT yang dikendalikan melalui aplikasi Android mungkin menghadapi masalah keterbatasan jaringan atau ketergantungan pada konektivitas internet yang stabil. Ini bisa menjadi masalah di daerah dengan infrastruktur internet yang buruk. Selain itu, kompleksitas sistem mungkin memerlukan keahlian teknis untuk pengelolaan dan pemeliharaan.
b) Beberapa penelitian membangun sistem yang berbasis mikrokontroler yang dimana sering kali memerlukan pemrograman yang rumit dan mungkin tidak mudah diperbaiki atau ditingkatkan tanpa pengetahuan teknis yang mendalam. Selain itu, perangkat keras mikrokontroler bisa menghadapi masalah ketahanan dan keandalan dalam jangka panjang.
c) Beberapa penelitian menghadapi masalah dalam hal akurasi waktu bel dan membutuhkan kalibrasi yang sering. Keterbatasan dalam daya pemrosesan

dan memori mikrokontroler Arduino juga bisa menjadi kendala jika sistem perlu diperluas atau diintegrasikan dengan fitur tambahan.

d) Beberapa penelitian menggunakan teknologi IoT dalam prototipe ini dapat menghadapi tantangan dalam hal kompatibilitas dan integrasi dengan sistem eksisting di sekolah. Selain itu, prototipe ini mungkin memerlukan biaya implementasi dan perawatan yang lebih tinggi karena ketergantungan pada perangkat IoT khusus.

Sementara itu, penelitian kami memiliki beberapa keunggulan yang signifikan diantaranya adalah akurasi waktu pembunyian bel yang akurat, aplikasi yang mudah digunakan, dan aktifasi tanpa menggunakan internet. Sebagai hasilnya, penelitian kami menawarkan solusi yang lebih maju, aman, dan sesuai dengan lingkungan SMP Setia Budi Sungailiat, memungkinkan tingkat yang lebih tinggi mengoperasikan bel mikrokontroler.

2.2 *Internet of Things* (IoT)

Perkembangan ilmu pengetahuan dan teknologi saat ini semakin memberikan kemudahan dalam kehidupan sehari-hari. *Internet of Things* (IoT) dapat dianggap sebagai layanan yang dapat mencakup berbagai jenis permintaan. berdasarkan penggunaan beberapa kemampuannya [11]. Dimana semua hal banyak diaplikasikan ilmu pengetahuan dan teknologi dengan mesin-mesin atau elektronik, sehingga pekerjaan manusia dapat dilakukan dengan mudah tanpa harus membuang tenaga dan dapat mempersingkat waktu[12]. *Internet of Things* adalah penerapan teknologi elektronika modern yang memungkinkan kontrol, komunikasi, dan kerjasama perangkat keras melalui internet. Istilah *Internet of things* ini umumnya mengacu pada skenario dimana konektivitas jaringan dan kemampuan komputasi meluas ke objek, sensor, dan barang sehari-hari yang biasanya tidak dianggap sebagai komputer, memungkinkan perangkat ini untuk menghasilkan, bertukar, dan mengkonsumsi data dengan intervensi manusia yang minimal [13].Teknologi IoT ini bisa mengendalikan dan memantau perangkat konvensional dan tentunya mempermudah pekerjaan manusia. *Smart Things make our world smarter*[14]. Kemudahan dan keefektivitas pengoperasian tersebut menjadi penentu kemajuan sistem, termasuk sistem bel mikrokontroler. Sistem pengendalian bel mengunakan perangkat IoT itu penting dalam

kegiatan di Sekolahan. IoT bisa digunakan untuk mengendalikan peralatan elektronik, termasuk bel.

Penelitian sebelumnya oleh Wahyudi dan koleganya (2022) mengembangkan sistem bel mikrokontroler berbasis Android yang bertujuan untuk meningkatkan efisiensi pengelolaan waktu di sekolah. Penelitian ini mengimplementasikan teknologi IoT untuk memungkinkan pengaturan jadwal bel secara otomatis melalui aplikasi Android. Metode yang digunakan meliputi studi literatur, desain sistem, implementasi, dan pengujian lapangan. Data dikumpulkan melalui wawancara dengan staf sekolah dan observasi proses operasional. Hasil penelitian menunjukkan bahwa sistem ini dapat mengurangi kesalahan manual dan memberikan notifikasi kepada pengguna, serta memungkinkan kontrol bel dari jarak jauh. Tantangan yang dihadapi termasuk ketergantungan pada koneksi internet yang stabil dan biaya pengembangan serta pemeliharaan yang tinggi. Meskipun demikian, sistem ini terbukti meningkatkan efisiensi dan fleksibilitas dalam pengelolaan waktu sekolah, serta menawarkan solusi yang lebih modern dan andal dibandingkan dengan sistem bel konvensional.

Secara keseluruhan, penelitian tersebut menghasilkan sebuah sistem bel mikrokontroler berbasis Android yang menggunakan teknologi *Internet of Things* untuk meningkatkan efisiensi pengelolaan waktu di lingkungan sekolah. Melalui pendekatan studi literatur, desain sistem, implementasi, dan pengujian lapangan, penelitian ini berhasil mengembangkan aplikasi Android yang memungkinkan pengaturan jadwal bel secara otomatis dan notifikasi kepada pengguna. Data yang dikumpulkan melalui wawancara dan observasi memberikan pemahaman yang mendalam tentang kebutuhan dan tantangan dalam pengelolaan waktu di sekolah. Meskipun menghadapi beberapa tantangan seperti ketergantungan pada koneksi internet dan biaya pengembangan yang tinggi, sistem ini terbukti memberikan solusi yang modern, andal, dan efisien bagi sekolah dalam pengaturan jadwal bel dan meningkatkan responsivitas terhadap perubahan jadwal dengan cepat.

2.3 Teknologi Bel

Bel merupakan suatu alat yang dapat mengeluarkan bunyi dan mempunyai fungsi sebagai kode, alat pengingat dan alat komunikasi [4]. Teknologi bel merupakan salah satu inovasi penting dalam sistem pengaturan waktu dan pemberian isyarat suara di

berbagai lingkungan, seperti sekolah, kantor, dan pabrik. Lonceng, Genta atau Bel merupakan suatu peralatan sederhana yang digunakan untuk menciptakan bunyi. Bentuk sebuah tabung dengan salah satu sisi yang terbuka dan bergema saat dipukul[15]. Seiring dengan perkembangan zaman, bel yang semula masih sederhana dengan cara di pukul sampai diubah menjadi bel listrik manual dan sekarang ada yang menerapkan bel dengan bunyi terkoneksi memanfaatkan internet dengan zona waktu otomatis[16]. Otomatisasi sudah menjadi sebuah keharusan karena dengan serba otomatis itu banyak sekali yang bisa dicapai, diantaranya adalah efisiensi dan penghematan[17]. Selama ini cara yang digunakan untuk membunyikan bel listrik masih menggunakan tenaga manusia, cara ini masih dirasa kurang efisien dan kurang efektif[18]. Sistem bel modern sering kali menggunakan mikroprosesor atau mikrokontroler untuk mengotomatiskan jadwal bel, sehingga mengurangi kebutuhan akan intervensi manual dan meningkatkan ketepatan waktu. *Microcontroller* dapat digunakan untuk mengendalikan sebuah proses tunggal dan menjalankan instruksi yang sederhana[19].

Dalam penerapan teknologi bel berbasis *Internet of Things*, sistem ini tidak hanya mengandalkan *timer* sederhana tetapi juga dapat dikendalikan dan dipantau dari jarak jauh menggunakan perangkat seperti *smartphone* atau komputer. Contoh teknologi ini adalah sistem bel yang terintegrasi dengan aplikasi Android, yang memungkinkan pengaturan jadwal dan pengendalian bel dari mana saja, kapan saja. Dengan menggunakan IoT, teknologi bel dapat dihubungkan dengan berbagai sensor dan sistem lain, seperti sensor kehadiran atau sistem manajemen gedung, untuk memberikan kontrol yang lebih terintegrasi dan cerdas. Ini memungkinkan pengguna untuk tidak hanya mengatur waktu bel tetapi juga menyesuaikan volume, durasi, dan frekuensi sesuai kebutuhan operasional.

Berbagai penelitian telah mengeksplorasi pengembangan dan implementasi teknologi bel modern ini. Misalnya, studi tentang "Perancangan Sistem Bel Sekolah Berbasis *IoT* Menggunakan Aplikasi Android" membahas pengembangan sistem bel yang dapat dikendalikan melalui aplikasi mobile untuk meningkatkan efisiensi dan kemudahan dalam mengelola waktu belajar mengajar di sekolah. Penelitian lain, seperti "Implementasi Sistem Pengaturan Bel Sekolah Otomatis Berbasis Android," mengatasi masalah pengaturan bel yang masih dilakukan secara manual dengan menciptakan

sistem otomatis yang lebih efisien. Teknologi bel cerdas ini tidak hanya menawarkan solusi otomatisasi tetapi juga integrasi dengan berbagai teknologi modern lainnya, seperti sensor dan kontrol jarak jauh, untuk mendukung pengelolaan waktu yang lebih baik di berbagai lingkungan.

Secara keseluruhan, teknologi bel modern menawarkan banyak keunggulan, termasuk otomatisasi yang dapat mengurangi beban kerja manual dan meningkatkan ketepatan waktu. Namun, teknologi ini juga menghadapi beberapa tantangan, seperti ketergantungan pada konektivitas internet dan kebutuhan akan pemeliharaan yang berkelanjutan untuk memastikan sistem beroperasi dengan baik. Selain itu, biaya implementasi awal dan kompleksitas penggunaan juga menjadi pertimbangan penting dalam penerapan teknologi bel berbasis *IoT*. Meski demikian, manfaat yang diberikan oleh teknologi bel cerdas ini jauh lebih besar, terutama dalam hal efisiensi operasional dan fleksibilitas pengelolaan waktu.

2.4 Aplikasi Berbasis Android

Android merupakan sistem operasi mobile. Android tidak membedakan antar aplikasi inti dengan aplikasi pihak ketiga[20].Aplikasi Mobile Berbasis Android merujuk pada aplikasi yang secara spesifik dirancang untuk dijalankan pada sistem operasi Android. Aplikasi Mobile adalah perangkat lunak yang berjalan pada perangkat mobile seperti smartphone atau tablet PC[3]. Aplikasi ini memiliki beragam kegunaan yang mencakup berbagai keperluan,seperti pengendalian dan pengawasan peternakan ayam broiler, pembelajaran pemrograman pada mahasiswa Teknik Informatika, pendeteksi kebakaran Gedung, pengembangan bahan ajar teori ekonomi mikro, pembelajaran matematika pada materi persamaan linear satu variable, dan aplikasi mobile *driver online*. Selain itu juga, penerapan sistem operasi android dalam perangkat mobile membawa kita pada kenyaataan bahwa sistem operasi ini dapat dengan tepat mampu merealisasikan penyebaran dan kebutuhan asupan akan informasi secara *realtime*, serta mampu memberikan sebuah sistem informasi yang praktis, ringkas, interaktif dan mudah diakses oleh semua orang ke dalam sebuah perangkat mobile[21]. Pembuatan aplikasi ini menggunakan *software* Android Studio. Android Studio merupakan sebuah IDE (*Integrated Development Environment*) untuk pengembangan aplikasi android.[22].

Aplikasi Mobile Berbasis Android dapat membantu meningkatkan keterampilan dan efektivitas pembelajaran, seperti pada pengembangan aplikasi *Mobile Learning* berbasis Android untuk meningkatkan keterampilan pemrograman pada mahasiswa Teknik Informatika, dan pengembangan media pembelajaran Mobile Learning berbasis Android yang layak digunakan untuk menunjang pembelajaran matematika pada materi persamaan linear satu variable. Dalam pengembangan aplikasi Mobile Berbasis Android, digunakan berbagai metode dan teknik, seperti Model Siklus Hidup Perangkat Lunak, ADDIE (*analysis, design, development, implementation, dan evaluation*), dan teknik *prototype*.

Terdapat penelitian yang berjudul "Pelatihan Pembuatan Aplikasi Mobile Berbasis Android Pada SMK TKJ Yapim I Medan". Materi-materi pelatihan sudah disesuaikan dengan kebutuhan dunia kerja saat ini. Lalu ada jugxa penelitian yang berjudul "Pengembangan Bahan Ajar (Buku) Teori Ekonomi Mikro Berbasis Media Pembelajaran Aplikasi Android", yang membahas tentang meningkatkan kualitas pembelajaran pada mata kuliah teori ekonomi mikro di program studi Pendidikan Ekonomi, Fakultas Ekonomi, Unimed. Penelitian ini mengembangkan media pembelajaran teori ekonomi mikro berbasis Android yang efektif dan layak untuk digunakan dalam pembelajaran.Selain itu terdapat juga penelitian dengan judul "*Security Assessment* Pada Aplikasi Mobile Android XYZ Dengan Mengacu Pada Kerentanan OWASP Mobile Top Ten 2016". Kerentanan yang diidentifikasi mengacu pada kerentanan dari OWASP Mobile Top Ten 2016. Penelitian lain terkait dengan aplikasi mobile dengan judul "Efektifitas Penerapan *Flipped Learning* Model Berbasis Aplikasi Android Terhadap Hasil Belajar *Laboratory Skills* pada Mahasiswa Keperawatan di Kota Jambi". Penelitian ini bertujuan untuk mengidentifikasi perbedaan hasil belajar laboratory skills pada kelompok intervensi dan kontrol setelah dilakukan penerapan flipped learning model berbasis aplikasi Android pada mahasiswa/i keperawatan di Kota Jambi.

Dari penelitian-penelitian diatas, aplikasi mobile berbasis Android memiliki banyak potensi untuk digunakan dalam berbagai bidang, seperti pendidikan, keamanan, dan pelayanan publik. Aplikasi mobile berbasis Android juga dapat meningkatkan kualitas pembelajaran dan hasil belajar. Namun, perlu diperhatikan juga keamanan dari aplikasi tersebut agar tidak terjadi kerentanan yang dapat membahayakan pengguna.

Secara keseluruhan, dapat disimpulkan bahwa Pengembangan Smart Bell berbasis Android yang dirancang untuk SMP Setia Budi Sungailiat merupakan inovasi signifikan yang memanfaatkan teknologi IoT untuk meningkatkan manajemen waktu dan komunikasi di lingkungan sekolah. Teknologi ini bukan hanya menggantikan sistem bel tradisional dengan solusi yang lebih otomatis dan efisien tetapi juga turut andil dalam masa transisi alat tradisional ke teknologi.Sementara itu, penelitian yang kami buat memiliki fungsi yang menarik, yaitu bias dikontrol melalui aplikasi *Android*, bias *IOT*, dan tanpa penggunaan internet.

BAB III
METODE PELAKSANAAN

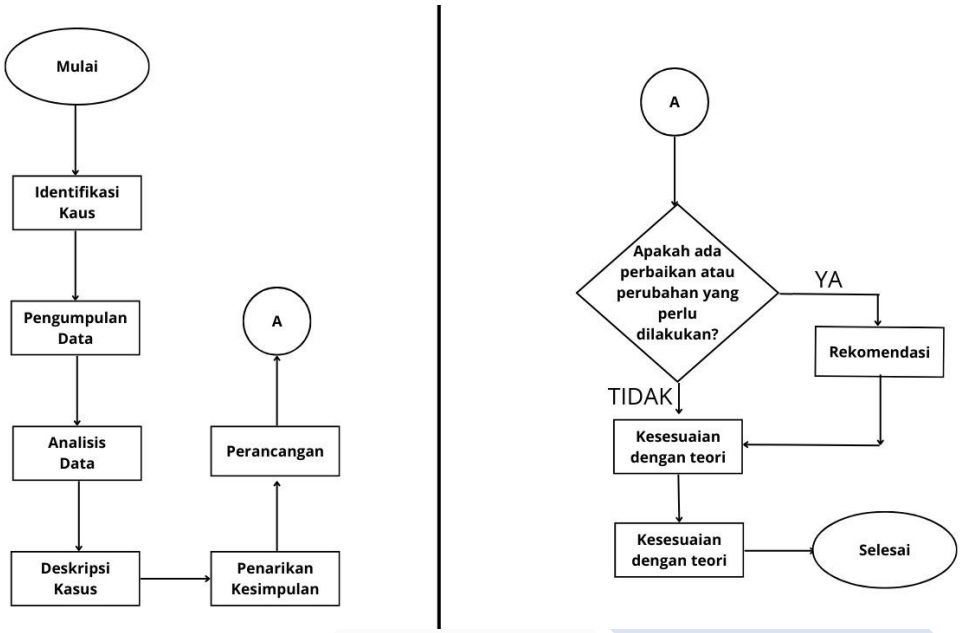
3.1 Model Rancangan

Berikut adalah beberapa model atau metode rancangan dari sistem bel mikrokontroler berbasis *android* :

3.1.1 Metode Pelaksanaan Studi Kasus

Metode pelaksanaan yang kami terapkan dalam pembuatan penelititian ini menggunakan metode studi kasus, Metode penelitian studi kasus adalah suatu pendekatan penelitian yang fokus pada pemahaman yang mendalam tentang suatu fenomena atau permasalahan tertentu di dalam konteks nyata. Penelitian studi kasus dilakukan dengan cara mendokumentasikan, menganalisis, dan menjelaskan kasus atau situasi yang terjadi dalam lingkungan yang sesungguhnya. Metode penelitian studi kasus adalah strategi yang tepat untuk digunakan dalam penelitian yang menggunakan pertanyaan penelitian utama “bagaimana” atau “mengapa”, diperlukan sedikit waktu untuk mengontrol peristiwa yang dipelajari, dan fokus penelitian adalah fenomena kontemporer[23]. Untuk tahapan-tahapan umum dalam proses pengerjaan dapat diilustrasikan melalui flowchart berikut.

Metode studi kasus dipilih karena memberikan pemahaman mendalam tentang fenomena atau permasalahan dalam konteks nyata, yang sangat relevan untuk menjawab pertanyaan penelitian "bagaimana" atau "mengapa." Melalui pendekatan ini, peneliti dapat mengeksplorasi fenomena kompleks yang tidak bisa direduksi menjadi variabel terisolasi, sambil menggunakan berbagai metode pengumpulan data seperti wawancara, observasi, dan dokumen. Studi kasus ideal untuk mempelajari fenomena kontemporer yang terjadi secara alami di lingkungan sebenarnya, terutama ketika peneliti memiliki sedikit kontrol atas variabel yang diteliti, sehingga memberikan fleksibilitas dan relevansi praktis dalam penelitian.



Gambar 3.1 *Flowchart* Studi Kasus

Flowchart diatas menggambarkan alur dari metode pelaksanaan studi kasus. Prosesnya dimulai dengan identifikasi kasus, dilanjutkan dengan pengumpulan data, analisis data, dan deskripsi kasus. Kemudian diambil kesimpulan berdasarkan penjelasan yang diberikan. Selanjutnya memasuki tahap evaluasi untuk mempertimbangkan apakah perlu adanya perbaikan atau perubahan. Jika perlu, rekomendasi dibuat untuk melaksanakan perbaikan. Setelah perbaikan selesai atau tidak diperlukan modifikasi, hasilnya dievaluasi untuk memastikan sesuai terhadap teori atau standar yang berlaku. Jika hasilnya benar maka proses selesai. Jika tidak, akan dilakukan desain ulang. Proses ini memastikan bahwa hasil akhir tidak hanya menyelesaikan masalah, namun juga memenuhi standar yang relevan.

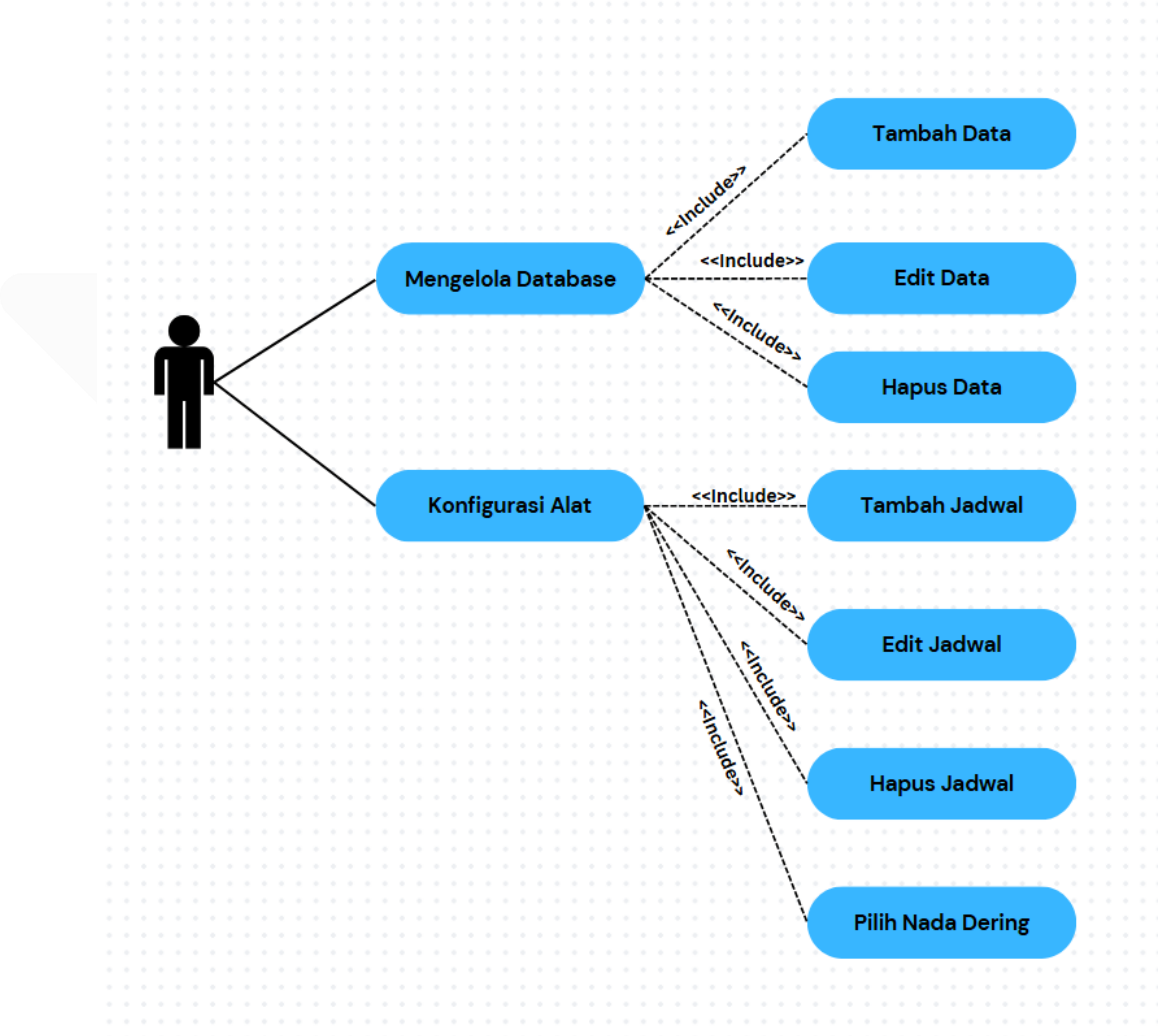
3.1.2 Metode *Prototype*

Model *Prototyping* merupakan model *Software Development Life Cycle* yang populer di mana prototipe dibuat, diuji, dan dikerjakan ulang sampai prototipe yang dapat diterima tercapai [24]. *Prototype* ini digunakan untuk menguji dan memvalidasi ide, konsep, atau fungsionalitas sebelum melanjutkan ke pengembangan penuh. *prototype* tidak menyajikan bentuk asli sistem secara lengkap akan tetapi metode *prototype* berperan penting dalam penelitian untuk memberikan gambaran aplikasi yang akurat terhadap client[25]. Metode *Prototype* ini juga memungkinkan pengembang dan

pengguna dapat lebih memahami kebutuhan dan mengidentifikasi serta memperbaiki masalah pada tahap awal proses pengembangan. Dalam metode *prototyping* dinilai mampu menyelesaikan masalah yang komplit dengan melibatkan user dalam perancangannya[26].

3.1.3 *Use Case Diagram*

Pemodelan diagram *use case* untuk operator pada aplikasi bel mikrokontroler untuk mengidentifikasi penggunaan dan system.



Gambar 3.2 Use Case Diagram

Pada gambar 3.2 admin dapat mengelola database dan konfigurasi alat seperti menambahkan, mengedit dan menghapus jadwal serta memilih suara atau nada dering yang diinginkan.

3.2 Analisis Kebutuhan

Terdapat beberapa *hardware* dan *software* yang digunakan untuk membuat aplikasi smart bell berbasis android pada SMP Setia budi Sungailiat, berikut uraian *hardware* dan *Software* nya.

3.2.1. Hardware

Aplikasi bel mikrokontroler berbasis android ini memerlukan beberapa komponen *hardware*, antara lain :

a) Mikrokontroler

Komponen ini berfungsi sebagai pusat kendali aplikasi bel mikrokontroler. Mikrokontroler akan memproses data aplikasi android dan mengirimkan sinyal untuk membunyikan bel.

b) Kabel *Jumper*

Komponen ini digunakan untuk menghubungkan komponen satu kekomponen lainnya.

c) RTC DS 3231

Komponen ini digunakan untuk melacak waktu dan tanggal secara presisi.

d) DF Player Mini

Komponen ini digunakan untuk memutar file mp3.

e) SD Card

Komponen ini digunakan untuk menyimpan file audi (mp3).

f) Speaker mini

Komponen ini digunakan untuk sebagai output suara.

3.2.2. *Software*

Selain kebutuhan *hardware*, aplikasi bel mikrokontroler berbasis android juga memerlukan beberapa komponen *software*, antara lain :

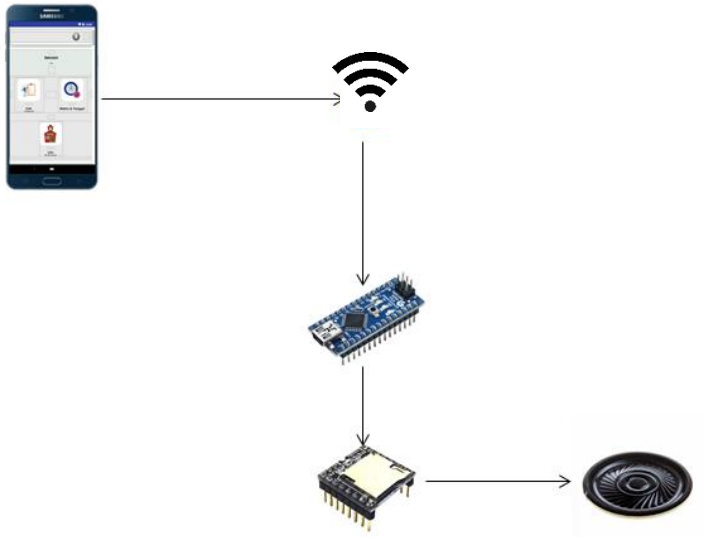
a) Sistem operasi Android

Aplikasi bel mikrokontroler berbasis android dirancang untuk berjalan pada operasi android.

b) Arduino IDE

Aplikasi Arduino berfungsi untuk memprogram dan mengendalikan mikrokontroler Esp32. Dengan menggunakan aplikasi ini, pengguna dapat mengembangkan berbagai proyek elektronik dan mekatronik.

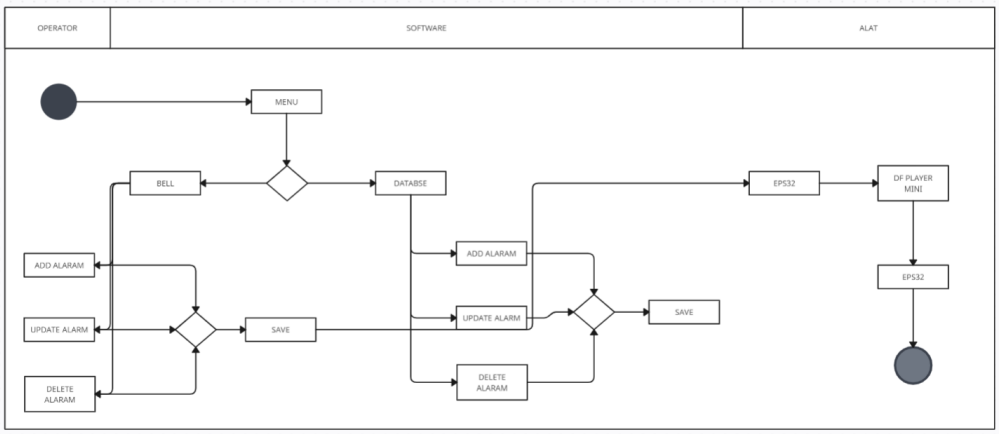
3.3 Perancangan Sistem



Gambar 3.3 Perancangan Sistem

Pada tahap ini berisi tentang rancangan sistem bel mikrokontroler berbasis android. Diawali dengan aplikasi android dimana operator akan membuka aplikasi login lalu mengklik menu edit jadwal untuk mengatur jadwal bel berbunyi serta memilih suara sesuai jadwal yang ada dan jika sudah selesai di set operator bias langsung mengirim data melalui Internet lalu di tangkap menggunakan arduino nano dan mengirimkan sinyal ke dfplayer sebelum bell berbunyi.

3.4 Activity Diagram



Gambar 3.4 Activity Diagram

Activity diagram menggambarkan alur kerja atau aktivitas yang dilakukan oleh sistem atau pengguna dalam mencapai suatu tujuan tertentu, dimulai dengan operator (guru atau staf) mengakses menu utama aplikasi untuk mengelola jadwal alarm. Operator dapat menambah, memperbarui, atau menghapus jadwal alarm sesuai kebutuhan, kemudian menyimpan perubahan tersebut. Data jadwal yang telah diatur akan disimpan dalam database aplikasi dan diteruskan ke perangkat keras melalui komunikasi dengan modul ESP32. ESP32 bertugas mengontrol DFPlayer Mini untuk memutar suara bell berupa file MP3 sesuai jadwal yang telah ditentukan. Proses ini memastikan bell berbunyi secara otomatis pada waktu yang diatur, tanpa perlu intervensi manual, sehingga mendukung efisiensi waktu dan pengelolaan yang lebih baik.

3.5 Alur Penggunaan alat

Berikut ini adalah alur penggunaan dari alat Bel Mikrokontroler berbasis android :

1. Pengaturan awal
    - Pengguna mengakses aplikasi Android yang telah terhubung dengan sistem bel .
    - Pada aplikasi, pengguna memasukkan jadwal bel yang mencakup waktu dan pilihan suara yang akan digunakan.
2. Pengiriman Data
    - Data jadwal yang diatur oleh pengguna akan dikirimkan ke database secara *real-time*.

- Perangkat bel akan menerima data jadwal Esp32 untuk diimplementasikan.

3. Eksekusi jadwal bel
    - Saat waktu yang telah dijadwalkan tiba, perangkat akan mengaktifkan DFPlayer untuk membunyikan bel melalui speaker.
    - Suara bel sesuai dengan konfigurasi yang telah ditentukan oleh pengguna di aplikasi.
4. Pembaruan jadwal
    - Pengguna dapat memperbarui atau menghapus jadwal melalui aplikasi Android.
    - Data yang diperbarui akan langsung disinkronkan dengan database dan perangkat.
5. Monitoring dan pengelolaan
    - Pengguna dapat memantau status perangkat dan jadwal bel yang sedang aktif melalui aplikasi.
    - Informasi operasional alat dapat diperiksa untuk memastikan sistem berjalan dengan baik.

BAB IV
PEMBAHASAN

4.1 Perancangan Aplikasi dan Alat

Pada kali ini peneliti akan melakukan perancangan aplikasi dan alat dari Bel Otomatis, berikut perancangan menggunakan alur prototype:

4.1.1 Requirements

Dari tahap pengumpulan data yang telah dilakukan dengan menggunakan metode observasi didapatkan hasil kebutuhan sebagai berikut :

a) Kebutuhan fungsional

Kebutuhan fungsional dari rancangan ini adalah seperti aplikasi harus dapat secara otomatis membunyikan bel sesuai dengan jadwal yang telah ditentukan dan dapat menyesuaikan waktu berdasarkan perubahan jadwal. Bel juga harus mampu mengeluarkan suara yang berbeda yang mengandung informasi yang spesifik.

b) Kebutuhan non-fungsional

Kebutuhan non-fungsional dari rancangan ini adalah system harus dapat beroperasi tanpa sering mengalami gangguan dan juga harus mudah dimodifikasi untuk mengakomodasi perubahan

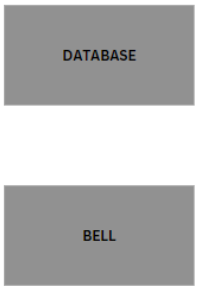
c) Kebutuhan pengguna

Kebutuhan pengguna dari rancangan ini adalah antarmuka aplikasi yang mudah digunakan dan fleksibilitas jadwal yang dapat dengan mudah mengatur dan mengubah jadwal bel sesuai kebutuhan

4.1.2 Quick Design

Pada tahap ini peneliti melakukan desain awal dari aplikasi *Bell* Otomatis

a) *HomePage*



Gambar 4.1 *Home Page*

Gambar 4.1 menampilkan tampilan dari *homepage* atau menu utama. Pada desain menu utama terdapat dua menu untuk mengatur jadwal pada alat dan database

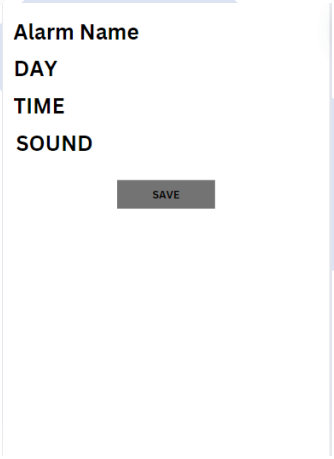
b) Menu Bell dan Add Alarm

Gambar 4.2 Menu Bell

Gambar 4.3 Menu Add Alarm

Pada tampilan gambar 4.2 merupakan tampilan awal menu bell untuk mengatur jadwal bell yang mana terdapat daftar jadwal yang sudah ditambahkan dan juga tombol untuk menambahkan jadwalnya. Lalu pada gambar 4.3 merupakan tampilan untuk penambahan jadwalnya.

c) Menu Database

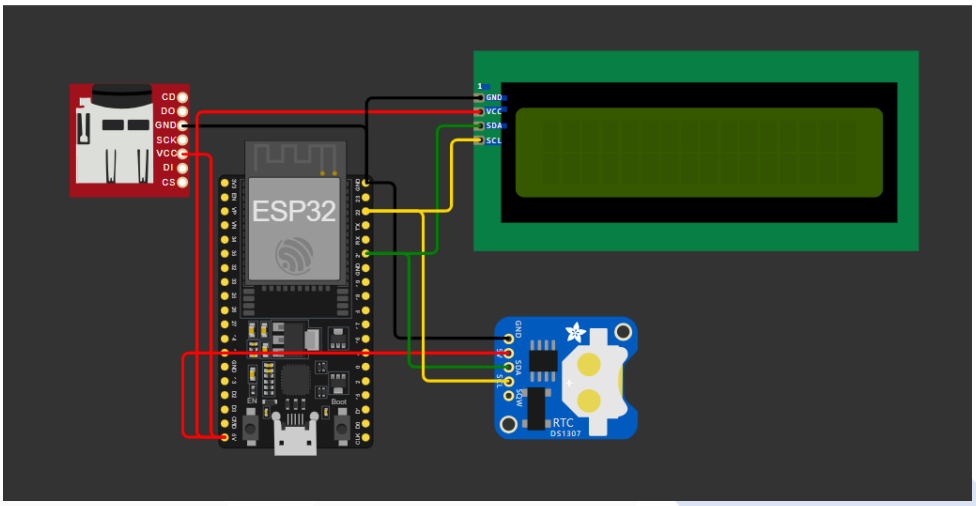


Gambar 4.4 Menu Database

Gambar 4.4 menampilkan menu Setting jadwal yang terhubung dengan database. Pada bagian ini terdapat juga menu untuk menambahkan waktu dan sound pada alat serta satu tombol untuk menyimpan jadwalnya.

4.1.3 Build Prototype

Pada tahap ini peneliti mulai merakit atau merancang prototype alat untuk Rancangan Smart Bell ini seperti merangkai komponen dan alat sesua skema pada board.



Gambar 4.5 Build Prototype

Gambar 4.5 menampilkan skema rangkaian dari *Smart Bell Berbasis Android*

4.1.4 User Evaluation

Pada tahap ini peneliti melakukan evaluasi user dengan metode observasi kepada pengguna langsung di SMP SetiaBudi Sungailiat. Proses ini berguna untuk menilai kebutuhan, pengalaman, dan kepuasan pengguna terhadap aplikasi yang dikembangkan. Hasil dari evaluasi ini akan digunakan untuk mengidentifikasi kelebihan dan kekurangan aplikasi, serta memastikan bahwa fitur-fitur yang tersedia sesuai dengan kebutuhan operasional sekolah. Peneliti juga mencatat umpan balik langsung dari pengguna, baik berupa saran maupun kritik, untuk kemudian dijadikan dasar dalam melakukan perbaikan atau pengembangan lebih lanjut. Dengan demikian, aplikasi diharapkan dapat memberikan manfaat maksimal dan meningkatkan efisiensi pengelolaan jadwal bel otomatis di SMP Setia Budi Sungailiat.

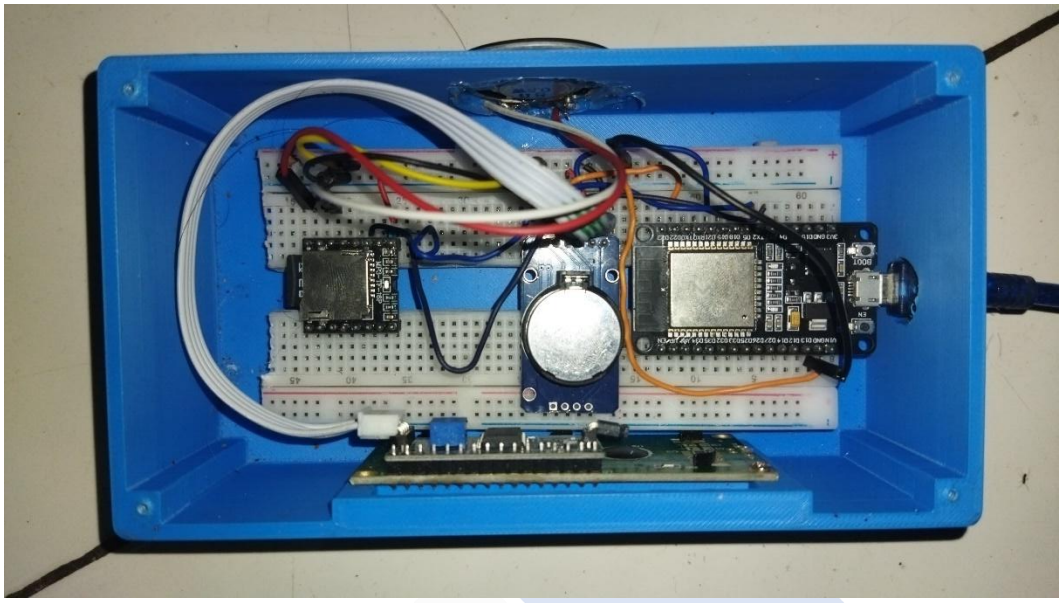
Tabel 4.1 User Evaluation

No	Pertanyaan
1	Apakah tampilan aplikasi bel mikrokontroler mudah dipahami?
2	Apakah fitur aplikasi bel mikrokontroler mudah digunakan?
3	Apakah aplikasi bel mikrokontroler menyelesaikan permasalahan yang ada?
4	Apakah perancangan bel mikrokontroler dapat meningkatkan ke-efektifan dan fleksibelitas dalam penggunaan bel sekolah?
5	Apakah aplikasi bel mikrokontroler membantu kegiatan yang ada disekolah?
6	Apakah aplikasi bel mikrokontroler mudah digunakan?

Pada Tabel 4.1 merupakan hasil evaluasi dari penggunaan bell mikrokontroler berbasis android.

4.1.5 Refining Prototype

Pada tahap ini peneliti masuk ke tahap penyempurnaan prototype dari bel mikrokontroler. Proses ini melibatkan literasi pada desain awal berdasarkan *Feedback* dan pengujian fungsional dan kegunaan



Gambar 4.7 Refining Prototype

4.1.6 Implement and Maintain

Pada tahap ini peneliti harus memastikan keberhasilan jangka panjang dari alat dan aplikasi ini. Pada tahap implementasi peneliti melakukan perencanaan, pengembangan dan pengujian pada alat dan aplikasi serta pada tahap pemeliharaan peneliti melakukan pemantauan serta perbaikan dan evaluasi berkala.

Gambar 4.8 Implement and Maintain

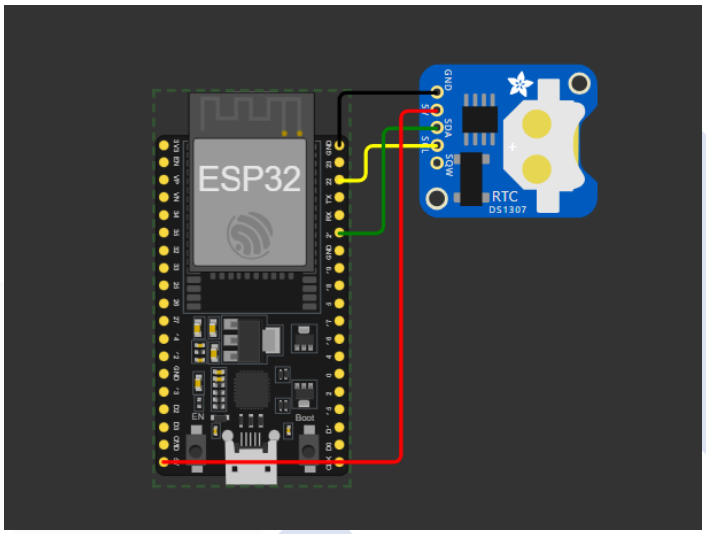
Pada gambar 4.8 kami melakukan implementasi penggunaan bel mikrokontroler berbasis android pada SMP SetiaBudi Sungailiat.

4.2 Pembuatan alat dan pemrograman sistem

Pada tahap ini peneliti akan melakukan pembuatan dan pemrograman alat

4.2.1 Rangkaian modul RTC DS3231

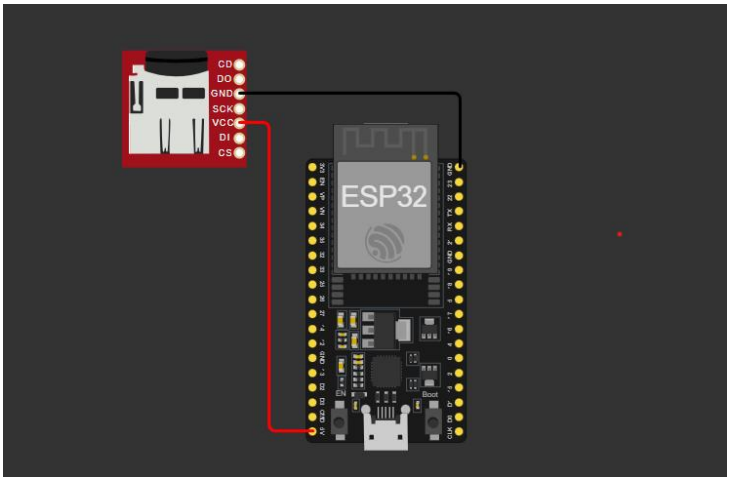
Rangkaian modul RTC DS3231 ini berfungsi untuk mengambil data waktu *Real Time* dari modul RTC ke sistem. Untuk menghubungkan modul RTC ke Arduino nano hubungkan pin SQW , SCL dan SDA seperti di bawah ini:



Gambar 4.9 Rangkaian Modul RTC DS3231

4.2.2 Rangkaian Modul DF Player

Modul DF Player ini berfungsi sebagai pemutar audio dan mengeluarkan output ke speaker dari rancangan bel mikrokontroler ini . untuk menghubungkan modul ini cukup menghubungkan pin RX dan TX nya ke pin Arduino serta menghubungkan pin ke speaker



Gambar 4.10 Rangkaian Modul DfPlayer

4.2.3 Pemrograman alat

```
#include <WiFi.h>
#include <WebServer.h>
#include <ArduinoJson.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <DFPlayerMini_Fast.h>
#include <RTClib.h>

// Pengaturan WiFi
const char* ssid = "Red";
const char* password = "12093487";

// Tentukan pin SDA dan SCL khusus untuk ESP32
#define SDA_PIN 21
#define SCL_PIN 22

// Tentukan pin untuk DFPlayer Mini
#define DFPLAYER_RX 16  // Pin RX ESP32 terhubung ke TX DFPlayer
#define DFPLAYER_TX 17  // Pin TX ESP32 terhubung ke RX DFPlayer
```

Gambar 4.11 Library Mikrokontroler

Program pada gambar 4.11 adalah program untuk pendefinisian dan memasukkan *library* . *Library* ini berfungsi untuk komunikasi alat seperti pada RTC,LCD,dan DFPlayer serta untuk menghubungkan wifi dan webserver. Pendefinisian juga dilakukan pada pin alat dan pengaturan wifi.

```
// Inisialisasi Komponen
LiquidCrystal_I2C lcd(0x27, 16, 2);
DFPlayerMini_Fast myDFPlayer;
RTC_DS3231 rtc;
HardwareSerial mySerial(1);  // Menggunakan Serial1 untuk DFPlayer Mini

// Deklarasi Server
WebServer server(80);

// Struktur Jadwal Bel
struct Schedule {
  String time;  // Waktu dalam format HH:MM
  int track;    // Nomor track yang akan dimainkan
  String day;   // Hari yang ditentukan untuk jadwal (misalnya: "Senin", "Selasa", dll.)
};

Schedule scheduleList[10];  // Maksimal 10 jadwal
int scheduleCount = 0;
```

Gambar 4.12 Inisialisasi Objek

Program pada gambar 4.12 merupakan program untuk inisialisasi objek dan juga pin RX dan TX dari modul Bluetooth dan DFPlayer yang sudah dihubungkan. Gambar diatas juga menunjukkan program untuk struktur waktu untuk jadwal bel.

```
void handleGetRequest() {
  DynamicJsonDocument doc(1024);
  JsonArray array = doc.to<JsonArray>();

  for (int i = 0; i < scheduleCount; i++) {
    JsonObject obj = array.createNestedObject();
    obj["id"] = i;
    obj["time"] = scheduleList[i].time;
    obj["track"] = scheduleList[i].track;
    obj["day"] = scheduleList[i].day;  // Tambahkan parameter hari
  }

  String output;
  serializeJson(doc, output);
  server.send(200, "application/json", output);
}


bool addScheduleFromJson(const String& json) {
  DynamicJsonDocument doc(1024);
  DeserializationError error = deserializeJson(doc, json);
  if (error) {
    Serial.println("Deserialization failed: " + String(error.c_str()));
    return false;
  }

  if (scheduleCount < 10) {
    scheduleList[scheduleCount].time = doc["time"].as<String>();
    scheduleList[scheduleCount].track = doc["track"];
    scheduleList[scheduleCount].day = doc["day"].as<String>();  // Tambahkan parameter h
```

Gambar 4.13 Komunikasi JSON

Program pada gambar 4.13 adalah Program untuk menyiapkan dan komunikasi seperti menambahkan, mengedit, serta menghapus data jadwal dari JSON yang akan di gunakan pada alat.

```
void setup() {
  Serial.begin(115200);
  Wire.begin(SDA_PIN, SCL_PIN);
  lcd.begin(16, 2);
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("Bel Sekolah");
  lcd.setCursor(0, 1);
  lcd.print("SETIA BUDI");
  delay(2000);

  if (!rtc.begin()) {
    lcd.clear();
    lcd.print("RTC Error");
    while (1);
  }

  if (rtc.lostPower()) {
    rtc.adjust(DateTime(F(_DATE), F(TIME_)));
  }

  // Mengatur komunikasi serial untuk DFPlayer Mini
  mySerial.begin(9600, SERIAL_8N1, DFPLAYER_RX, DFPLAYER_TX);
  if (!myDFPlayer.begin(mySerial)) {
```

Gambar 4.14 Setup

Program pada gambar 4.14 adalah program untuk *setup* atau Inisialisasi komunikasi serial, modul DFPlayer, RTC, dan LCD. Pada setup juga dilakukan proses penghubungan kepada webserver.

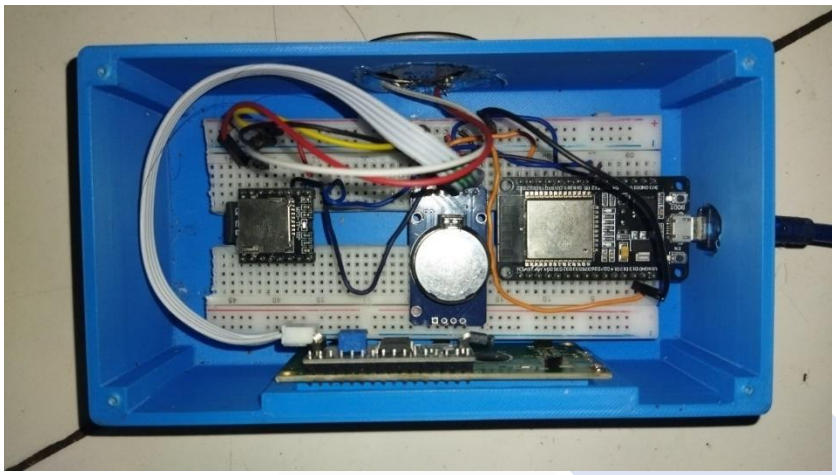
```
void loop() {
    DateTime now = rtc.now();
    lcd.setCursor(0, 0);
    lcd.print(now.timestamp());

    String currentDay = getDayOfWeek(now.dayOfTheWeek());  // Mendapatkan hari ini dalam format String

    // Cek jadwal dan mainkan track jika waktu dan hari sesuai
    for (int i = 0; i < scheduleCount; i++) {
        if (scheduleList[i].day == currentDay && now.timestamp().substring(11, 16) == scheduleList[i].time) {
            myDFPlayer.play(scheduleList[i].track);  // Memainkan track sesuai jadwal

            lcd.setCursor(0, 1);
            lcd.print("Bel: ");
            lcd.print(scheduleList[i].track);

            delay(60000);  // Mencegah bel berulang dalam 1 menit
        }
    }

    server.handleClient();  // Menangani request HTTP
    delay(1000);
}
```

Gambar 4.15 Fungsi *Looping*

Program pada gambar 4.15 digunakan untuk fungsi *looping* yang mana dilakukan proses eksekusi dari bel berdasarkan jadwal yang telah disiapkan

4.3 Perangkat Keras (*Hardware)*

Perangkat yang telah dirancang dalam penelitian ini adalah *Smart Bell Berbasis Android* . perangkat yang dapat terhubung ke aplikasi android melalui bluetooth.



Gambar 4.16 *Hardware*

Gambar 4.16 menampilkan rangkaian dan komponen dari *Smart Bell.* Komponen yang terdapat pada perangkat ini terdiri dari Arduino nano, Modul RTC, Modul Bluetooth, Modul DFPlayer, LCD 16x2, dan Speaker Mini.

4.4 Tampilan Aplikasi

Pada tahap kali ini peneliti membuat tampilan aplikasi *smart bell* dimana peneliti kali ini membuat aplikasi ini menggunakan website kodular.

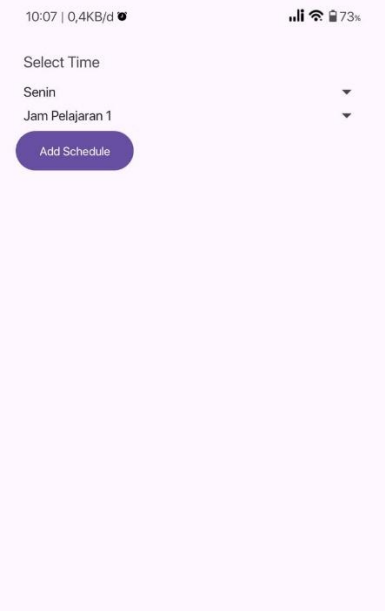
4.4.1. Tampilan *Home*

Berikut tampilan *home* untuk aplikasi *smart bell* sebagai tampilan seluruh menu dari aplikasi *smart bell* untuk penelitian kami, *home* adalah menu utama pada aplikasi kami. Pada bagian ini tersedia dua menu untuk mengatur jadwal ke database maupun alat bellnya.

Gambar 4.17 Tampilan Home

4.4.2. Tampilan Menu Tambah Jadwal

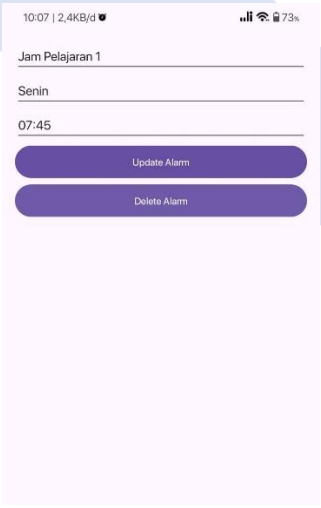
Berikut tampilan pada menu tambah jadwal pada aplikasi *smart bell*, menu ini digunakan untuk menambahkan jadwal dari bell sesuai dengan yang diinginkan. Pengguna dapat menambahkan waktu serta suara yang ingin diputar nantinya.



Gambar 4.18 Tampilan Tambah Jadwal

4.4.3. Tampilan Edit

Berikut tampilan dari menu edit jadwal, pada bagian ini berfungsi untuk mengatur kembali jadwal yang telah ditambahkan tadi. Jadi Ketika telah menambahkan jadwal namun terjadi kesalahan Ketika memasukkan datanya pengguna dapat mengedit Kembali jadwalnya atau bisa memilih menghapusnya



Gambar 4.19 Tampilan Edit

4.5 Pengujian

Pada tahap ini peneliti melakukan pengujian pada alat yang diteliti guna memastikan kinerjanya optimal dan dengan standar yang ditetapkan.

4.5.1. Pengujian Delay

Tabel 4.2 Pengujian Delay

Jadwal	Waktu Pengujian	Waktu Berbunyi	Input Jam	Delay (Detik)
1	10.12	10.12	Sukses	0
2	11.44	11.44	Sukses	0
3	14.51	14.51	Sukses	0
4	15.00	15.00	Sukses	0
5	15.30	15.30	Sukses	0
6	16.25	16.25	Sukses	0
7	16.30	16.30	Sukses	0
8	17.10	17.10	Sukses	0
9	17.15	17.15	Sukses	0
10	17.30	17.30	Sukses	0
11	17.45	17.45	Sukses	0
12	18.05	18.05	Sukses	0
13	18.15	18.15	Sukses	0
14	19.00	19.00	Sukses	0
15	19.10	19.10	Sukses	0

Dari tabel 4.2 dapat dilihat data pengujian delay pada alat *smart bell* dari pengujian yang telah dilakukan oleh peneliti hasil dari setiap jadwal yang ada bahwa secara keseluruhan ketika waktu / jam di atur semuanya sukses di input dan bell berbunyi tepat waktu, sesuai denga napa yang di atur pada aplikasi.

4.5.2. Hasil Pengujian Kuesioner

Pengujian kuesioner telah dilakukan dengan memberikan evaluasi terhadap pertanyaan yang disajikan pada table 4.3, yang diberikan kepada seluruh guru & staf di sekolah dengan total responden yaitu 9 orang.

Tabel 4.3 Pertanyaan Kuesioner

No	Pertanyaan
1	Apakah tampilan aplikasi bel mikrokontroler mudah dipahami?
2	Apakah fitur aplikasi bel mikrokontroler mudah digunakan?
3	Apakah aplikasi bel mikrokontroler menyelesaikan permasalahan yang ada?
4	Apakah perancangan bel mikrokontroler dapat meningkatkan ke-efektifan dan fleksibelitas dalam penggunaan bel sekolah?
5	Apakah aplikasi bel mikrokontroler membantu kegiatan yang ada disekolah?
6	Apakah aplikasi bel mikrokontroler mudah digunakan?

1) Sangat tidak setuju
2) Tidak setuju
3) Netral
4) Setuju
5) Sangat Setuju

Dari pertanyaan kuesioner diatas, diperoleh beberapa jawaban yang dapat dilihat pada Table 4.4.

Tabel 4.4 Hasil Kuesioner

No	Nama	Hasil Jawaban					
		1	2	3	4	5	6
1	Responden 1	5	4	4	5	4	5
2	Responden 2	5	5	4	5	4	5
3	Responden 3	5	5	5	4	4	5
4	Responden 4	5	4	4	4	4	5
5	Responden 5	5	3	4	5	4	5
6	Responden 6	4	4	4	5	5	4
7	Responden 7	5	5	4	3	5	4
8	Responden 8	3	4	4	4	5	5
9	Responden 9	5	4	5	5	4	5

Hasil perhitungan presentase hasil jawaban :

Total : 240

Nilai Presentase : 240 / 300 * 100%

: 80%

4.5.3 Pengujian Jarak

Tabel 4.5 Hasil Pengujian Jarak

Jarak	Respon Alat
1 Meter	Valid
2 Meter	Valid
3 Meter	Valid
4 Meter	Valid
5 Meter	Valid
6 Meter	Valid
7 Meter	Tidak Valid

Tabel 4.5 menunjukkan bahwa alat berfungsi dengan baik hingga jarak 6 meter, di mana semua respon masih valid. Namun, pada jarak 7 meter, alat tidak lagi memberikan respon yang valid. Hal ini menunjukkan bahwa jangkauan efektif alat dalam mendeteksi atau menerima sinyal maksimal berada di bawah 7 meter, sehingga mungkin perlu dilakukan penyesuaian atau peningkatan untuk jangkauan yang lebih luas jika diperlukan.

BAB V
KESIMPULAN DAN SARAN

5.1. Kesimpulan

Proyek pengembangan aplikasi Smart Bell berbasis Android di SMP Setia Budi Sungailiat telah berhasil menciptakan sistem bel mikrokontroler yang efisien dan mudah digunakan. Aplikasi ini tidak hanya menyediakan akses yang mudah bagi pengguna, tetapi juga menghilangkan kebutuhan intervensi manual untuk membunyikan bel, menggantinya dengan solusi otomatis yang diatur oleh sistem.

Sistem Smart Bell memungkinkan integrasi dengan jadwal pelajaran dan kalender akademik sekolah, sehingga bel dapat berbunyi sesuai dengan perubahan yang terjadi tanpa memerlukan penyesuaian manual. Fitur notifikasi otomatis yang disematkan juga meningkatkan kehandalan sistem dengan memberi tahu pengguna jika terjadi gangguan teknis, sehingga masalah dapat diatasi dengan cepat.

Evaluasi sistem ini menunjukkan bahwa Smart Bell lebih efektif dan handal dibandingkan dengan bel listrik konvensional, terutama dalam hal fleksibilitas dan efisiensi penggunaan sehari-hari di lingkungan sekolah. Pengujian yang dilakukan menunjukkan bahwa Smart Bell mampu mengoptimalkan waktu dan penggunaan sumber daya sekolah, serta memberikan kontribusi positif terhadap pengelolaan waktu pembelajaran di SMP Setia Budi. Dengan demikian, Smart Bell menjadi solusi yang efisien dan modern dalam manajemen waktu sekolah, serta meningkatkan pengalaman belajar bagi siswa dan guru.

5.2. Saran

Berikut ini adalah beberapa saran untuk proyek ini antara lain:

a) Alat ini masih dapat dikembangkan dari sisi desain antarmuka Androidnya, agar lebih user-friendly dan intuitif, sehingga pengguna dengan berbagai tingkat keahlian teknologi dapat dengan mudah mengoperasikannya.
b) Pengembangan lebih lanjut dapat dilakukan dengan menambahkan fitur integrasi dengan sistem manajemen sekolah lainnya, seperti pengelolaan jadwal pelajaran, kalender akademik, dan pemberitahuan otomatis untuk acara-acara khusus.

c) Pertimbangkan untuk meningkatkan keamanan aplikasi dengan menambahkan fitur otentikasi pengguna, seperti login berbasis PIN atau sidik jari, untuk mencegah akses tidak sah ke pengaturan bel.
d) Pengembangan lebih lanjut dapat mencakup pembuatan versi aplikasi untuk platform selain Android, seperti iOS, agar lebih banyak pengguna dapat mengakses dan menggunakan aplikasi ini.

DAFTAR PUSTAKA

[1] A. Linarta, "I N F O R M A T I K A APLIKASI BEL SEKOLAH OTOMATIS BERBASIS ARDUINO DILENGKAPI DENGAN OUTPUT SUARA," *Jurnal Informatika, Manajemen dan Komputer*, vol. 10, no. 2, 2018.

[2] M. Pauzan, B. Sekolah Otomatis Berbasis, I. Yanti, K. Kunci-Bel Sekolah, P. Bel, and A. Berbasis Android, "Bel Sekolah Otomatis Berbasis Arduino yang Dikontrol Menggunakan Aplikasi Mobile (Automatic School Bell Based on Arduino Controlled By Mobile Application)," 2021.

[3] M. Irsan, T. Jl, H. Hadari, and N. Pontianak, "RANCANG BANGUN APLIKASI MOBILE NOTIFIKASI BERBASIS ANDROID UNTUK MENDUKUNG KINERJA DI INSTANSI PEMERINTAHAN."

[4] A. Lukman Affandy Sistem Komputer and S. Handayani Makassar, "Rancang Bangun Bel Sekolah Otomatis Berbasis Personal Computer (PC)," 2017.

[5] D. Susilo, R. Dwi Laksono, and Y. E. Ardiansyah, "Rancang Bangun Sistem Bel Sekolah Otomatis Berbasis Mikrokontroller Menggunakan ISD 4003," 2022.

[6] M. Aswin, D. Setiawan, S. Kom, M. Kom, and G. Syahputra, "PERANCANGAN JAM DIGITAL DAN SISTEM BEL OTOMATIS PADA SEKOLAH DENGAN TEKNIK COUNTER BERBASIS MIKROKONTROLER," 2020. [Online]. Available: https://ojs.trigunadharma.ac.id/

[7] T. Moch Cadafi Fahlefi Sani, "AUTOPOWER MENGGUNAKAN INTERFACE BERBASIS ANDROID," vol. 3, pp. 35–40.

[8] M. Subianto, "SISTEM BEL OTOMATIS TERPROGRAM BERBASIS RASPBERRY PI."

[9] H. Hamzah, A. Hasin, S. Mehora, Muh. R. Kadir, and M. N. Agriawan, "Prototype Bel Cerdas Cermat Berbasis Arduino Uno," *SAINTIFIK*, vol. 9, no. 1, pp. 1–6, Jan. 2023, doi: 10.31605/saintifik.v9i1.414.

[10] A. Linarta, "Sistem Informasi Penjadwalan Bel Sekolah Otomatis Berbasis Microcontroller," vol. 11, no. 2, 2018.

[11] S. Villamil, C. Hernández, and G. Tarazona, “An overview of internet of things,” *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 18, no. 5, pp. 2320–2327, Oct. 2020, doi: 10.12928/TELKOMNIKA.v18i5.15911.

[12] K. Abdi Azizi, A. Sembiring, M. Arfisyari, and S. Kaputama Binjai, “Design and Build an Automated School Bell System Using Voice-Based IoT,” 2022. [Online]. Available: https://ijhet.com/index.php/ijhess/

[13] K. Rose, S. Eldridge, and L. Chapin, “The Internet of Things: An Overview Understanding the Issues and Challenges of a More Connected World.”

[14] S. Madakam, “Internet of Things: Smart Things,” *International Journal of Future Computer and Communication*, vol. 4, no. 4, pp. 250–253, 2015, doi: 10.7763/IJFCC.2015.V4.395.

[15] M. Prodi, P. Teknik, E. F. Tarbiyah, and D. Keguruan, “PERANCANGAN PROTOTYPE BEL SEKOLAH OTOMATIS BERBASIS ARDUINO SKRIPSI Diajukan oleh : RIZKA ANGGIA DINDA NIM. 170211047.”

[16] N. Asyiah, “Spectrum: Multidisciplinary Journal Perancangan Sistem Bel Otomatis Dan Informasi Waktu Belajar Di Sekolah Berbasis Internet Of Things (Studi Kasus: SMK Bina Mandiri),” 2024.

[17] V. Yasin, M. Zarlis, E. Budhiarti Nababan, and P. Sihombing, “Jln. Salemba I No. 10 Jakarta Pusat 10430 Indonesia”.

[18] H. Jurnal and A. Nurtado, “JURNAL MANAJEMEN DAN TEKNIK INFORMATIKA SIMULASI BEL SEKOLAH OTOMATIS BERBASIS ARDUINO UNO,” *JUMANTAKA*, vol. 03, p. 1, 2019.

[19] A. Tahir *et al.*, “SISTEM KONTROL PINTU PUTAR OTOMATIS DI PERPUSTAKAAN,” 2016.

[20] N. Kadek *et al.*, “RANCANG BANGUN APLIKASI MOBILE SISKA BERBASIS ANDROID,” Online, 2018. [Online]. Available: http://jurnal.stiki-indonesia.ac.id/index.php/sintechjournal

[21] N. Noor, K. Sari, P. Bagus, A. A. Putra, and E. Christian, “RANCANG BANGUN APLIKASI MOBILE LEARNING TENSES BAHASA INGGRIS,” *Jurnal Teknologi Informasi*, vol. 13, no. 2.

[22] E. Maiyana, “Pemanfaatan Android Dalam Perancangan Aplikasi Kumpulan Doa,” *Jurnal Sains dan Informatika*, vol. 4, no. 1, pp. 54–65, Apr. 2018, doi: 10.22216/jsi.v4i1.3409.

[23] R. Dewi Nur'aini, "PENERAPAN METODE STUDI KASUS YIN DALAM PENELITIAN ARSITEKTUR DAN PERILAKU," 2020.

[24] U. S. Senarath, "Waterfall Methodology, Prototyping and Agile Development," 2021, doi: 10.13140/RG.2.2.17918.72001.

[25] A. Ichwani, N. Anwar, K. Karsono, and M. Alrifqi, "Sistem Informasi Penjualan Berbasis Website dengan Pendekatan Metode Prototype."

[26] F. K. Putra, "Penerapan Metode Prototyping Dalam Rancangan Sistem Informasi Absensi Berbasis Website," *Journal of Information System Research (JOSH)*, vol. 3, no. 4, pp. 431–436, Jul. 2022, doi: 10.47065/josh.v3i4.1835.

LAMPIRAN 1

LAMPIRAN 1: DAFTAR RIWAYAT HIDUP PENULIS

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap	:	Rangga Febri Hadi Tanjung
Tempat & Tanggal Lahir	:	Surabaya, 23 Februari 2002
Alamat Rumah	:	Gg. Andalas 1, Air Ruay.
		Hp: 081367277231
		Email: ranggafebri0811@gmail.com
Jenis Kelamin	:	Laki-Laki
Agama	:	Islam

2. Riwayat Pendidikan

SD Muhammadiyah	:	2008 – 2014
SMPN 1 SUNGAILIAT	:	2014 – 2017
SMKN 1 SUNGAILIAT	:	2017 – 2020

Sungailiat, 24 July 2024

Rangga Febri Hadi Tanjung

DAFTAR RIWAYAT HIDUP

3. Data Pribadi

Nama Lengkap	:	Zacky Alfarizi
Tempat & Tanggal Lahir	:	Sungailiat, 16 November 2003
Alamat Rumah	:	Kelurahan Matras
		Hp: 081438082520
		Email: zackyalfarizi077@gmail.com
Jenis Kelamin	:	Laki-Laki
Agama	:	Islam

4. Riwayat Pendidikan

SDN 20 SUNGAILIAT	:	2009-2015
SMPN 1 SUNGAILIAT	:	2015-2018
SMAN 1 SUNGAILIAT	:	2018-2021

Sungailiat, 24 July 2024

Zacky Alfarizi

LAMPIRAN 2

Dokumentasi

LAMPIRAN 3

```
#include <WiFi.h>
#include <WebServer.h>
#include <ArduinoJson.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <DFPlayerMini_Fast.h>
#include <RTClib.h>

// Pengaturan WiFi
const char* ssid = "Red";
const char* password = "12093487";

// Tentukan pin SDA dan SCL khusus untuk ESP32
#define SDA_PIN 21
#define SCL_PIN 22

// Tentukan pin untuk DFPlayer Mini
#define DFPLAYER_RX 16  // Pin RX ESP32 terhubung ke TX DFPlayer
#define DFPLAYER_TX 17  // Pin TX ESP32 terhubung ke RX DFPlayer

unsigned long previousMillis = 0;
const long interval = 1000;  // 1 detik

// Inisialisasi Komponen
LiquidCrystal_I2C lcd(0x27, 16, 2);
DFPlayerMini_Fast myDFPlayer;
RTC_DS3231 rtc;
HardwareSerial mySerial(1);  // Menggunakan Serial1 untuk DFPlayer Mini

// Deklarasi Server
WebServer server(80);
```

```
// Struktur Jadwal Bel
struct Schedule {
  String time;  // Waktu dalam format HH:MM
  int track;    // Nomor track yang akan dimainkan
  String day;   // Hari yang ditentukan untuk jadwal (misalnya: "Senin", "Selasa", dll.)
};

Schedule scheduleList[10];  // Maksimal 10 jadwal
int scheduleCount = 0;

void connectToWiFi() {
  WiFi.begin(ssid, password);
  Serial.print("Menghubungkan ke WiFi...");

  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.print(".");
  }

  Serial.println("Terhubung ke WiFi!");
  Serial.print("IP Address: ");
  Serial.println(WiFi.localIP());
}

void handleGetRequest() {
  DynamicJsonDocument doc(1024);
  JsonArray array = doc.to<JsonArray>();
for (int i = 0; i < scheduleCount; i++) {
  JsonObject obj = array.createNestedObject();
  obj["id"] = i;
  obj["time"] = scheduleList[i].time;
  obj["track"] = scheduleList[i].track;
  obj["day"] = scheduleList[i].day;  // Tambahkan parameter hari
}

String output;
serializeJson(doc, output);
server.send(200, "application/json", output);
}

bool addScheduleFromJson(const String& json) {
  DynamicJsonDocument doc(1024);
  DeserializationError error = deserializeJson(doc, json);
  if (error) {
    Serial.println("Deserialization failed: " + String(error.c_str()));
    return false;
  }

  if (scheduleCount < 10) {
    scheduleList[scheduleCount].time = doc["time"].as<String>();
    scheduleList[scheduleCount].track = doc["track"];
    scheduleList[scheduleCount].day = doc["day"].as<String>();  // Tambahkan parameter hari
    scheduleCount++;
    Serial.println("Schedule Added: Time=" + scheduleList[scheduleCount-1].time + ", Track=" + String(scheduleList[scheduleCount-1].track) + ", Day=" + scheduleList[scheduleCount-1].day);
    return true;
  } else {
    Serial.println("Schedule List Full");
    return false;
  }
}
void handlePostRequest() {
  if (server.hasArg("plain")) {
    String body = server.arg("plain");
    Serial.println("Received Body: " + body);

    if (addScheduleFromJson(body)) {
      server.send(200, "application/json", "{\"status\":\"success\", \"message\":\"Schedule added\"}");
    } else {
      server.send(400, "application/json", "{\"status\":\"error\", \"message\":\"Failed to add schedule\"}");
    }
  } else {
    Serial.println("No body received");
    server.send(400, "application/json", "{\"status\":\"error\", \"message\":\"No body received\"}");
  }
}

bool updateScheduleFromJson(const String& json) {
  DynamicJsonDocument doc(1024);
  DeserializationError error = deserializeJson(doc, json);
  if (error) {
    Serial.println("Deserialization failed: " + String(error.c_str()));
    return false;
  }
```

```
  int id = doc["id"];
  if (id >= 0 && id < scheduleCount) {
    scheduleList[id].time = doc["time"].as<String>();
    scheduleList[id].track = doc["track"];
    scheduleList[id].day = doc["day"].as<String>();  // Tambahkan parameter hari
    Serial.println("Schedule Updated: ID=" + String(id) + ", Time=" + scheduleList[id].time + ", Track=" + String(scheduleList[id].track) + ", Day=" + scheduleList[id].day);
    return true;
  } else {
    Serial.println("Schedule Update Failed: Invalid ID");
    return false;
  }
}


void handlePutRequest() {
  if (server.hasArg("plain")) {
    String body = server.arg("plain");
    Serial.println("Received Body: " + body);

    if (updateScheduleFromJson(body)) {
      server.send(200, "application/json", "{\"status\":\"success\", \"message\":\"Schedule updated\"}");
    } else {
      server.send(400, "application/json", "{\"status\":\"error\", \"message\":\"Failed to update schedule\"}");
    }
  } else {
    Serial.println("No body received");
    server.send(400, "application/json", "{\"status\":\"error\", \"message\":\"No body received\"}");
  }
}
bool deleteSchedule(int id) {
  if (id >= 0 && id < scheduleCount) {
    Serial.println("Deleting Schedule: ID=" + String(id) + ", Time=" + scheduleList[id].time + ", Track=" + String(scheduleList[id].track));
    for (int i = id; i < scheduleCount - 1; i++) {
      scheduleList[i] = scheduleList[i + 1];
    }
    scheduleCount--;
    return true;
  } else {
    Serial.println("Schedule Deletion Failed: Invalid ID");
    return false;
  }
}

void handleDeleteRequest() {
  if (server.hasArg("id")) {
    int id = server.arg("id").toInt();
    Serial.println("DELETE Request Received with ID: " + String(id));
    if (deleteSchedule(id)) {
      server.send(200, "application/json", "{\"status\":\"success\", \"message\":\"Schedule deleted\"}");
    } else {
      server.send(400, "application/json", "{\"status\":\"error\", \"message\":\"Failed to delete schedule\"}");
    }
  } else {
    Serial.println("ID not provided");
    server.send(400, "application/json", "{\"status\":\"error\", \"message\":\"ID not provided\"}");
  }
}
```

```
void setup() {
  Serial.begin(115200);
  Wire.begin(SDA_PIN, SCL_PIN);
  lcd.begin(16, 2);
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("Bel Sekolah");
  lcd.setCursor(0, 1);
  lcd.print("SETIA BUDI");
  delay(2000);

  if (!rtc.begin()) {
    lcd.clear();
    lcd.print("RTC Error");
    while (1);
  }

  if (rtc.lostPower()) {
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
  }

  // Mengatur komunikasi serial untuk DFPlayer Mini
  mySerial.begin(9600, SERIAL_8N1, DFPLAYER_RX, DFPLAYER_TX);
  if (!myDFPlayer.begin(mySerial)) {
    lcd.clear();
    lcd.print("DFPlayer Error");
    while (1);
  }

  myDFPlayer.volume(30);

  connectToWiFi();

 // Rute API
 server.on("/schedule", HTTP_GET, handleGetRequest);
 server.on("/schedule", HTTP_POST, handlePostRequest);
 server.on("/schedule", HTTP_PUT, handlePutRequest);
 server.on("/schedule", HTTP_DELETE, handleDeleteRequest);

 server.begin();
 lcd.clear();
 lcd.print("Setup Selesai");
 delay(2000);
}

void loop() {
    DateTime now = rtc.now();
    lcd.setCursor(0, 0);
    lcd.print(now.timestamp());

    String currentDay = getDayOfWeek(now.dayOfTheWeek());  // Mendapatkan hari ini dalam format String

    // Cek jadwal dan mainkan track jika waktu dan hari sesuai
    for (int i = 0; i < scheduleCount; i++) {
        if (scheduleList[i].day == currentDay && now.timestamp().substring(11, 16) == scheduleList[i].time) {
            myDFPlayer.play(scheduleList[i].track);  // Memainkan track sesuai jadwal

            lcd.setCursor(0, 1);
            lcd.print("Bel: ");
            lcd.print(scheduleList[i].track);

            delay(60000);  // Mencegah bel berulang dalam 1 menit
        }
    }
```

```
    server.handleClient();  // Menangani request HTTP
    delay(1000);
}

// Fungsi untuk mendapatkan nama hari dari indeks
String getDayOfWeek(int dayIndex) {
    switch (dayIndex) {
        case 1: return "Senin";
        case 2: return "Selasa";
        case 3: return "Rabu";
        case 4: return "Kamis";
        case 5: return "Jumat";
        case 6: return "Sabtu";
        case 0: return "Minggu";
        default: return "";
    }
}
```