

RANCANG BANGUN SISTEM KEHADIRAN KARYAWAN BERBASIS RADIO FREQUENCY IDENTIFICATION (RFID)

Muhamad Ibrohim¹, Maya Selvia Lauryn², Rama Dhanan Jaya³

Program Studi Teknik Informatika - Universitas Serang Raya

b41m.cyber@gmail.com¹, mysellyn@gmail.com², ramadhejhe@gmail.com³

Abstrak - Fakta bahwa masih banyak perusahaan-perusahaan yang menggunakan pencatatan kehadiran karyawan secara manual, yaitu dengan menggunakan buku pencatatan kehadiran pada saat masuk maupun selesai waktu kerja. Mengurangi efisiensi dan keakuratan perusahaan dalam mengoptimalkan produktivitas mereka. Oleh karena itu, sebuah *prototipe* sistem absensi *Radio Frequency Identification* (RFID) yang terintegrasi dengan *database* untuk mendukung program peningkatan sikap disiplin karyawan sebagai langkah awal dari peningkatan kinerja kerja perusahaan secara keseluruhan. Penelitian ini menggunakan mikrokontroler Arduino Nano dan terdapat input menggunakan RFID-RC522 yang berfungsi untuk mengidentifikasi kartu RFID. Penelitian ini menggunakan metode *waterfall* dan dirancang menggunakan *Unified Modeling Language* (UML) dan dibangun menggunakan bahasa pemrograman Arduino Ide, Borland Delphi 7, serta MySQL sebagai *database*. Dari hasil yang penelitian ini terciptanya sebuah alat dan aplikasi absensi kehadiran karyawan. Berdasarkan pemanfaatan RFID sangat membantu pada proses indentifikasi absensi kehadiran karyawan dengan ID kartu yang berbeda-beda, sehingga tidak akan ada lagi mengenai kesalahan absensi serta dengan penyimpanan *database* secara otomatis dan dengan waktu yang sangat cepat.

Kata Kunci: Absensi Kehadiran Karyawan, Mikrokontroler Arduino Nano, RFID-RC522

I. PENDAHULUAN

Pencatatan absensi karyawan merupakan salah satu faktor penting dalam pengelolaan sumberdaya manusia (*human resource management*). Informasi yang mendalam dan terperinci mengenai kehadiran seorang karyawan dapat menentukan prestasi kerja, gaji, produktivitas atau kemajuan instansi secara umum. *Radio Frequency Identification* (RFID) mulai dikembangkan sebagai salah satu teknologi baru yang akan memudahkan manusia untuk melakukan identifikasi berbagai hal, terdiri dari *tag* berupa *chip* khusus yang mempunyai kode-kode informasi yang unik dan suatu *reader* yang berfungsi untuk membaca kode-kode pada *tag* tersebut.

Proses pencatatan, pelaporan dan kehadiran karyawan merupakan proses yang *repetitive*. Pada umumnya, perusahaan yang belum menggunakan sistem absensi secara digital atau terkomputerisasi, karyawan datang pada waktu yang sudah ditentukan dan tanda tangan pada *form* absensi, jika semua karyawan sudah absen pada *form* absensi maka admin akan memasukkan data absensi ke dalam komputer setiap harinya terkadang dalam proses penginputan sering terjadi kesalahan jika tidak teliti dalam penginputan data setelah data disimpan maka *form* absensi akan disimpan pada setiap periode.

Mengacu pada permasalahan, tujuan yang hendak dicapai dalam penelitian ini adalah merancang suatu sistem absensi dengan menggunakan teknologi RFID yang dapat membantu instansi atau perusahaan untuk meningkatkan efektifitas dalam melakukan pengolahan data absensi pegawai dengan memperkecil kemungkinan-kemungkinan kesalahan yang akan terjadi dan agar keamanan informasi dapat terjamin.

II. KAJIAN PUSTAKA

Absensi

Absensi atau kartu jam hadir ialah dokumen yang mencatat jam hadir setiap pegawai. Catatan jam hadir pegawai tersebut dapat berupa daftar hadir biasa, dapat juga pula berbentuk kartu hadir yang diisi dengan mesin pencatat waktu. Pekerjaan mencatat waktu pada dasarnya dapat dipisahkan menjadi dua bagian yakni pencatatan waktu hadir (*attendance time keeping*) dan juga pencatatan waktu kerja (*shop time keeping*). Pencatatan jam hadir pada hadir pada kartu jam hadir yang dilakukan oleh pada setiap pegawai bisa mempengaruhi gaji bersih atau *take home pay* yang akan diterima oleh pegawai setiap bulannya.

Alat Absensi

Alat absensi merupakan suatu alat yang dapat digunakan oleh manusia untuk membantu dalam proses penentuan daftar kehadiran. Terdapat berbagai macam alat absensi yang telah ada saat ini. Salah satu alat absensi tersebut ialah alat absensi untuk mengukur manual. Kebanyakan alat absensi yang digunakan saat ini ialah alat absensi yang kurang efektif dan efisien.



Gambar 1. Absensi Manual

Mikrokontroler

Menurut Cristianto Tjahyadi (2006), mikrokontroler sering disebut *computer on a chip* atau *single chip computer*, karena sering digunakan sebagai otak elektronik guna mengendalikan peralatan, mainan atau mesin. Secara fisik, mikrokontroler adalah sebuah rangkaian terpadu (*chip*) yang berisi memori (untuk menyimpan program), prosesor (untuk mengolah program) dan pin input/output (dihubungkan dengan sensor dan aktuator).

Menurut Mohamad Eko Ari Bowo (2009), mikrokontroler adalah sebuah sistem mikroprosesor lengkap yang terkandung di dalam sebuah *chip* yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara yang khusus.

Menurut Esti Yuliana (2011), mikrokontroler adalah suatu alat elektronik digital yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus, cara kerja mikrokontroler sebenarnya membaca dan menulis data.

Berdasarkan uraian di atas, pengertian mikrokontroler bisa disimpulkan bahwa mikrokontroler adalah *single chip computer* yang digunakan sebagai pengontrol aplikasi secara otomatis. Mikrokontroler mempunyai memori yang dapat digunakan untuk menyimpan data, dimana data tersebut bisa ditulis dan dihapus dengan bantuan alat khusus. Mikrokontroler mempunyai standar minimum, berikut ini adalah standar minimum suatu mikrokontroler:

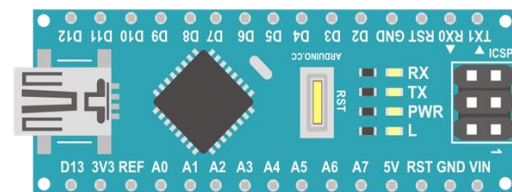
1. CPU (*Central Processing Unit*) untuk mengolah program.
2. RAM (*Random Access Memory*) untuk menyimpan program.
3. Jenis penyimpanan data ROM/ PROM/ EPROM/ EEPROM.
4. I/O Serial dan paralel yang dihubungkan dengan sensor.
5. Timer
6. Interrupt controller

Rata-rata mikrokontroler memiliki instruksi manipulasi bit, akses ke I/O secara langsung dan mudah, dan proses *interrupt* yang cepat dan efisien. Penggunaan pada mikrokontroler didesain tidak terlalu membutuhkan kecepatan pemrosesan yang sangat tinggi, mempunyai ukuran yang sangat kecil dengan konsumsi daya listrik yang sangat hemat serta fleksibel dalam penggunaannya. Pada saat ini mikrokontroler digunakan untuk mengontrol suatu proses, aspek-aspek dari lingkungan atau membuat segala sesuatu menjadi lebih mudah dan nyaman secara otomatis. Contoh aplikasi yang paling sederhana adalah lampu jalan raya yang bisa menyala secara otomatis pada waktu menjelang sore hari dan kemudian mati secara otomatis ketika keadaan mulai terang atau pagi. Mikrokontroler sendiri bekerja berdasarkan program yang diinputkan ke dalam

memori, program dikompilasi dan hasil kompilasi dimasukkan ke dalam mikrokontroler.

Arduino Nano

Arduino adalah pengendali *micro single-board* yang bersifat *open-source*, diturunkan dari *wiring platform*, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. *Hardware*-nya memiliki prosesor *Atmel AVR* dan *software*-nya memiliki bahasa pemrograman sendiri, Arduino ini berfungsi sebagai *board* yang di dalamnya sudah tertanam mikrokontroler. Berikut bentuk dari arduino nano.

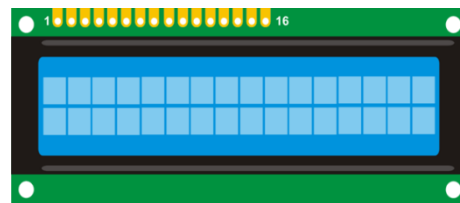


Gambar 2. Arduino Nano

Board ini memiliki keunggulan tambahan diantaranya: Ukuran *boot loader* hanya 1/4 *boot loader* sebelumnya sehingga lebih banyak ruang untuk program. Menggunakan ATmega 328 menggantikan FTDI *chip*, sehingga proses *upload* dan komunikasi serial menjadi lebih cepat, tidak perlu *driver* USB pada Linux dan Mac pada Windows hanya membutuhkan *file inf* dan *chip* ini bisa diprogram sehingga Arduino Nano dapat dikenali sebagai *keyboard*, *mouse*, *joystick* dan sebagainya.

Liquid Cristal Display (LCD) 16X2

Dalam modul LCD (*Liquid Cristal Display*) terdapat mikrokontroler yang berfungsi sebagai pengendali tampilan karakter LCD. Mikrokontroler pada suatu LCD dilengkapi dengan memori dan register, berikut adalah model LCD 16X2.



Gambar 3. LCD 16X2

Memori yang digunakan mikrokontroler internal LCD adalah:

1. DDRAM (*Display Data Random Access Memory*) merupakan memori tempat karakter yang akan ditampilkan.
2. CGRAM (*Character Generator Random Access Memory*) merupakan memori untuk menggambarkan pola sebuah karakter yang bisa diubah-ubah sesuai dengan keinginan.
3. CGROM (*Character Generator Read Only Memory*) merupakan memori untuk

menggambarkan pola sebuah karakter dimana pola tersebut merupakan karakter dasar yang sudah ditentukan secara permanen sehingga pengguna tinggal mengambilnya sesuai alamat memorinya.

Sensor RFID

Sensor RFID adalah teknologi *wireless*, RFID menggunakan sarana yang disebut dengan label RFID atau *transponder* untuk menyimpan atau mengambil data dari jarak tertentu. Label RFID terdiri atas *microchip* silikon dan antena. Label yang aktif pada RFID adalah yang membutuhkan sumber tenaga, sedangkan label yang pasif tidak memerlukan sumber tenaga.

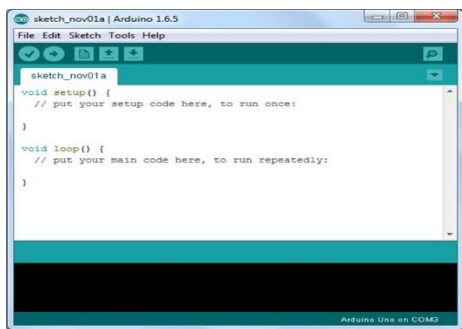


Gambar 5. Sensor RFID

Sensor RFID dibentuk dari dua buah unit, yaitu yang pertama adalah unit penerima dan yang kedua adalah unit pemancar. Kedua unit dalam sensor RFID ini memiliki struktur yang sangatlah sederhana, yaitu suatu *RFID Card*, yang didekatkan pada penerima akan mendeteksi sinyal elektromagnetik, disambungkan hanya dengan sebuah diafragma gelombang elektromagnetik. Kemudian kepada penerima diberikan tegangan 5 V.

Pengantar Software Arduino

Bahasa mesin adalah satu-satunya bahasa yang dimengerti oleh mikrokontroler. Bahasa ini tidak mudah untuk dimengerti oleh manusia, oleh karena itu diperlukan *compiler*. *Compiler* adalah penerjemah untuk bahasa pemrograman tingkat tinggi. Bekerja dengan cara menterjemahkan (misalnya pada PC) langsung ke bahasa mesin yang dimengerti oleh mikrokontroler. Salah satu *compiler* yang banyak digunakan pada mikrokontroler adalah bahasa "C".



Gambar 6. Program Arduino Ide

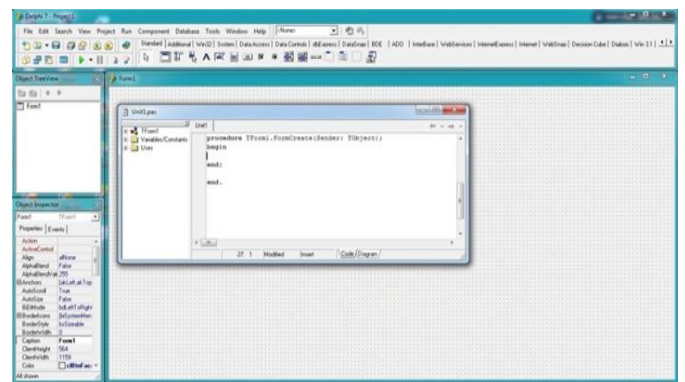
Program bahasa C adalah suatu program terdiri dari satu atau lebih fungsi-fungsi. Fungsi utama yang harus ada pada program C yang dibuat adalah fungsi *main()*. Fungsi *main()* ini adalah fungsi pertama yang akan diproses pada saat program di-*compile* dan dijalankan, sehingga bisa disebut sebagai fungsi yang mengontrol fungsi-fungsi lain. Karena struktur program bahasa C terdiri dari fungsi-fungsi lain sebagai program bagian (*subroutine*), maka bahasa C biasa disebut sebagai bahasa pemrograman terstruktur. Cara penulisan fungsi pada program bahasa C adalah dengan memberi nama fungsi dan kemudian dibuka dengan tanda kurung kurawal buka ({} dan ditutup dengan tanda kurung kurawal tutup (}).

Aturan pemrograman pada Bahasa C standar mencakup hal-hal sebagai berikut:

1. Sebagai konvensi, program ditulis dengan menggunakan huruf kecil.
2. Tanda titik-koma ";" digunakan untuk memisahkan dua pernyataan yang dapat dijalankan (*executable statements*).
3. Tanda kurung-kurawal "{" dan "}" digunakan untuk mengelompokkan unit-unit eksekusi dari fungsi-fungsi maupun pernyataan pengendali (*control statements*).
4. Fungsi dan variabel memerlukan deklarasi tipe data (*type declaration*).
5. Kata kunci (*reserved words*) tidak dapat digunakan sebagai pengenal (*identifiers*), seperti nama fungsi atau nama variabel.
6. Komentar yang terdiri dari satu baris atau lebih dituliskan diantara tanda "/*" dan "*/". Komentar yang hanya terdiri dari satu baris dapat dituliskan setelah tanda "/*".

Borland Delphi 7

Borland Delphi7 merupakan suatu bahasa pemrograman yang memberikan berbagai fasilitas pembuatan aplikasi untuk mengolah teks, grafik, angka, *database* dan aplikasi *web*. Program ini mempunyai kemampuan luas yang terletak pada produktifitas, kualitas, pengembangan perangkat lunak, kecepatan kompilasi, pola desain yang menarik serta bahasa pemrogramannya terstruktur.



Gambar 7. Delphi 7

Fasilitas pemrograman dibagi dalam dua kelompok yaitu *object* dan bahasa pemrograman. *Object* adalah suatu komponen yang mempunyai bentuk fisik dan biasanya dapat dilihat. *Object* biasanya dipakai untuk melakukan tugas tertentu dan mempunyai batasan-batasan tertentu. Sedangkan bahasa pemrograman dapat disebut sekumpulan teks yang mempunyai arti tertentu dan disusun dengan aturan tertentu untuk menjalankan tugas tertentu. Gabungan antara *object* dengan bahasa pemrograman sering disebut bahasa pemrograman berorientasi *object*.

MySQL

MySQL merupakan *database server open source* yang cukup populer keberadaannya. Dengan berbagai keunggulan yang dimiliki, membuat *software database* ini banyak digunakan karena praktis untuk membangun suatu *project*. Adanya fasilitas API (*Application Programming Interface*) yang dimiliki oleh MySQL, memungkinkan bermacam-macam aplikasi komputer yang ditulis dengan berbagai bahasa pemrograman dapat mengakses basis data MySQL. MySQL adalah program *database* yang mampu mengirim dan menerima data dengan sangat cepat dan *multi user*.

Kehandalan suatu sistem basisdata (DBMS) dapat diketahui dari cara kerja pengoptimasinya dalam melakukan proses perintah-perintah SQL yang dibuat oleh pengguna maupun program-program aplikasi yang memanfaatkannya. Sebagai peladen basis data, MySQL mendukung operasi basisdata transaksional maupun operasi basisdata non-transaksional.

Unified Modelling Language (UML)

Unified Modeling Language (UML) adalah sebuah bahasa untuk menentukan, visualisasi, konstruksi, dan mendokumentasikan *artifact* (bagian dari informasi yang digunakan atau dihasilkan dalam suatu proses pembuatan perangkat lunak). *Artifact* dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya. UML merupakan bahasa standar untuk penulisan *blueprint software* yang digunakan untuk visualisasi, spesifikasi, pembentukan dan pendokumentasian alat-alat dari sistem perangkat lunak.

Jenis - jenis Diagram UML:

1. Use Case Diagram

Use case adalah abstraksi dari interaksi antara *system* dan *actor*. *Use case* bekerja dengan cara mendeskripsikan tipe interaksi antara *user* sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai.

Biasanya suatu sistem atau perangkat lunak memiliki beberapa pengguna sekaligus, dimana masing-masing pengguna direpresentasikan sebagai sebuah *actor*. Dalam hal ini, *actor* berkaitan dengan peran yang dimainkan oleh seorang pengguna, tetapi bukan pengguna secara individu.

2. Activity Diagram

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* merupakan *state diagram* khusus, dimana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*).

Oleh karena itu *activity diagram* tidak menggambarkan *behaviour* internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Menggambarkan proses bisnis dan urutan aktivitas dalam sebuah proses.

3. Class Diagram

Tujuan utama dari *class diagram* adalah untuk menciptakan sebuah kosa kata yang digunakan oleh analis dan pengguna. Diagram kelas biasanya merupakan hal-hal, ide-ide atau konsep yang terkandung dalam aplikasi. Misalnya, jika anda sedang membangun sebuah aplikasi penggajian, diagram kelas mungkin akan berisi kelas yang mewakili hal-hal seperti karyawan, cek, dan pendaftaran gaji. Diagram kelas juga akan menggambarkan hubungan antara kelas.

4. Sequence Diagram

Sequence diagram yaitu salah satu jenis diagram pada UML yang menjelaskan interaksi objek yang berdasarkan urutan waktu, *sequence diagram* juga dapat menggambarkan urutan atau tahapan yang harus dilakukan untuk dapat menghasilkan output tertentu seperti pada *use case diagram*. *Sequence Diagram* diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan. Dimensi vertikal adalah sebuah poros waktu, dimana waktu berjalan ke arah bawah. Sedangkan dimensi horizontal merepresentasikan objek-objek individual. Tiap objek (termasuk *actor*) tersebut mempunyai waktu aktif yang direpresentasikan dengan menggunakan kolom vertikal yang disebut dengan *lifeline*. Diagram ini secara khusus berasosiasi dengan *use case diagram*. *Sequence diagram* juga memperlihatkan tahap demi tahap apa yang seharusnya terjadi.

III. METODE PENELITIAN

Tahap awal penelitian dengan melakukan interview atau wawancara dengan pegawai mengenai kesulitan proses absensi kemudian analisa terhadap proses rekapitulasi absensi bulanan. Tahap selanjutnya adalah perancangan sistem dengan model *Unified Modeling Language (UML)* sebagai salah satu model untuk merancang pengembangan sistem berbasis *object oriented*, kemudian membuat perancangan mikrokontroler arduino nano dengan model input

menggunakan RFID-RC522, dan terakhir adalah pembuatan dan pengujian sistem sebagai implementasi dari penelitian ini.

Teknik Pengumpulan Data

Pengumpulan data merupakan suatu kegiatan mencari data di lapangan yang akan digunakan untuk menjawab permasalahan penelitian. Oleh karena itu validitas instrumen pengumpulan data serta kualifikasi pengumpulan data sangat diperlukan untuk memperoleh data yang berkualitas. Semakin valid sebuah instrumen semakin valid juga data yang akan diperoleh. Dalam melakukan penelitian ini, peneliti mencari dan mengumpulkan data dan informasi yang berhubungan dengan penelitian ini melalui beberapa proses tahapan pengumpulan data, yaitu: Penelitian Lapangan (*Field Research*) dan Penelitian Kepustakaan (*Library Research*). Penelitian Kepustakaan (*Library Research*) merupakan cara untuk mencari referensi serta untuk mengetahui lebih dalam lagi untuk menganalisa sistem. Penelitian dilakukan dengan mengumpulkan bahan-bahan pustaka yang dilakukan dipergustakaan. Pencarian juga dilakukan melalui internet dan jurnal-jurnal yang ada.

Analisis Masalah

Berdasarkan masalah sering terjadinya kesalahan karena data form absensi diinput ulang atau dipindahkan datanya pada komputer dan proses penginputan data absensi pada komputer membutuhkan waktu yang cukup lama.

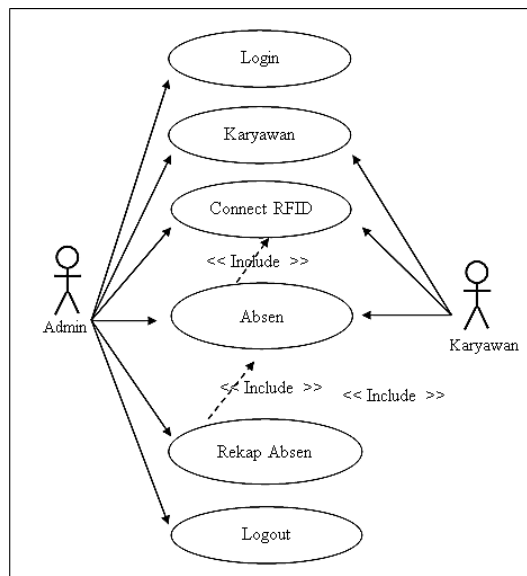
Analisis Data

Sistem pengolahan data absensi karyawan yang dilakukan masih belum optimal, karena diproses dengan menggunakan paket aplikasi *Microsoft Excel* dan sistem *cheklist* absensi karyawan. Hal ini mengakibatkan proses pengentrian data dan pembuatan laporan yang dibutuhkan agak lambat sehingga menimbulkan efek yang kurang kondusif dan waktu yang tidak efisien. Di sisi lain, keakuratan (*accurate*) dan keamanan (*safety*) data yang diproses tidak terjamin, karena data yang diolah dengan paket aplikasi ini harus dimasukkan ke dalam sel-sel yang tersedia, selanjutnya dirancang formula atau rumus untuk menghasilkan informasi seperti yang diharapkan. Maksudnya, setiap data-data yang akan diolah, maka operator harus melakukan *entry* data melalui paket aplikasi tersebut dan diproses dengan merancang formulanya. Kelemahan lain yang muncul akibat penggunaan paket aplikasi *Microsoft Excel* dalam pengolahan data absensi setiap karyawan adalah dalam hal pengelolaan arsip absensi yang rumit, karena membutuhkan waktu yang relatif lama apabila arsip absensi karyawan diperlukan atau dicek kembali.

Perancangan

Pada tahap ini, model perancangan sistem yang digunakan adalah *Unified Modeling Language*

(UML). Salah satu diagram UML yang digunakan adalah *Use Case* seperti gambar di bawah ini.



Gambar 8. Diagram Use Case

Use Case Diagram rancangan sistem aplikasi sistem informasi diatas menjelaskan interaksi antara *use case* dan aktor. Dimana aktor dapat berupa *user*. Aktor *user* memberikan interaksi ke beberapa *use case*.

1. *Use Case Description*

Setiap *Use Case* harus dijelaskan alur prosesnya melalui sebuah deskripsi *use case* (*use case description*). Berikut ini adalah *use case description* rancangan sistem pada aplikasi.

a. *Use Case Description Menu Utama.*

Use case description menu utama ini menjelaskan kegiatan yang dilakukan oleh Admin dalam menjalankan aplikasi absensi karyawan. Adapun penjelasannya dapat dilihat pada tabel 1.

Tabel 1. *Use Case Description* Menu Utama

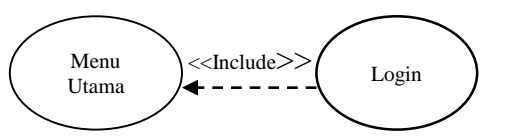
<i>Name Use Case</i>	Menu Utama.
<i>Actor</i>	Admin / Karyawan
Deskripsi singkat	<i>Use case</i> menggambarkan bagai mana Admin saat masuk ke halaman utama.
<i>Pre condition</i>	Admin membuka aplikasi.

<i>Flow of event</i>	<ol style="list-style-type: none"> 1. <i>Use case</i> dimulai ketika admin pertama kali membuka aplikasi. 2. Sistem akan menampilkan menu utama 3. <i>Use case</i> selesai.
<i>Post condition</i>	Sistem menampilkan menu utama aplikasi absensi karyawan
<i>Alternative flow</i>	-

b. *Use Case Description Login*

Use case description login ini menjelaskan kegiatan yang dilakukan oleh Admin untuk melakukan proses login kedalam aplikasi. Adapun penjelasannya dapat dilihat pada tabel 2.

Tabel 2. Use Case Description Login

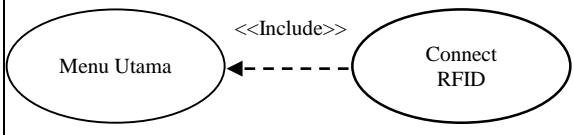


<i>Name Use Case</i>	<i>Login.</i>
<i>Actor</i>	Admin.
Deskripsi singkat	<i>Use Case</i> menggambarkan bagaimana admin melakukan <i>login</i> ke sistem.
<i>Pre Condition</i>	Admin sudah memiliki <i>account</i> .
<i>Flow of event</i>	<ol style="list-style-type: none"> 1. Admin menjalankan aplikasi. 2. Admin memilih menu <i>login</i>. 3. Sistem meminta Admin untuk memasukkan <i>username</i> dan <i>password</i>. 4. Sistem mengecek <i>username</i> dan <i>password</i> dan mencocokkan pada <i>database</i>. 5. <i>Use case</i> selesai.
<i>Post condition</i>	Sistem akan membuka data yang terkunci.
<i>Alternative flow</i>	Sistem akan menampilkan pesan " <i>Username</i> atau <i>password</i> anda salah".

c. *Use Case Description Connect*

Use case description connect ini menjelaskan kegiatan yang dilakukan oleh Admin untuk mengkoneksikan mikrokontroler dengan aplikasi. Adapun penjelasannya dapat dilihat pada tabel 3.

Tabel 3. Use Case Description Connect

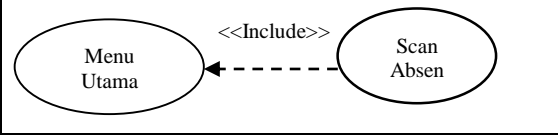


<i>Name Use Case</i>	<i>Connect.</i>
<i>Actor</i>	Admin.
Deskripsi singkat	<i>Use case</i> menggambarkan bagaimana Admin saat mengklik tombol <i>Connect</i> .
<i>Pre condition</i>	Admin berada pada menu utama aplikasi.
<i>Flow of event</i>	<ol style="list-style-type: none"> 1. <i>Use case</i> dimulai ketika admin membuka menu utama aplikasi. 2. Admin sudah <i>login</i>. 3. Admin memilih tombol <i>connect</i>. 4. Sistem akan mengkoneksikan aplikasi dengan mikrokontroler. 5. <i>Use case</i> selesai.
<i>Post condition</i>	Sistem menampilkan pesan "Sukses terhubung dengan <i>port</i> "
<i>Alternative flow</i>	Menampilkan pesan " <i>Port</i> tidak terhubung".

d. *Use Case Description Absensi*

Use case description absensi ini menjelaskan kegiatan yang dilakukan oleh Admin untuk men-*scan* kartu absensi karyawan dari aplikasi. Adapun penjelasannya dapat dilihat pada tabel 4.

Tabel 4. Use Case Description Absensi



<i>Name Use Case</i>	Absensi.
<i>Actor</i>	Karyawan.
Deskripsi singkat	<i>Use Case</i> menggambarkan bagaimana karyawan melakukan absen kartu ke sistem.
<i>Pre Condition</i>	Kartu sudah terdaftar pada aplikasi.
<i>Flow of event</i>	<ol style="list-style-type: none"> 1. Admin sudah <i>login</i>. 2. Kartu sudah terdaftar. 3. Pencet tombol pada alat dan pilih status yang akan digunakan. 4. Tempelkan kartu pada alat. 5. <i>Use case</i> selesai.
<i>Post condition</i>	Sistem akan menyimpan data menampilkan "sukses scan".
<i>Alternative flow</i>	"Anda tidak terdaftar"

e. *Use Case Description* tambah data karyawan
Use case description registrasi ini menjelaskan kegiatan yang dilakukan oleh Admin untuk menginputkan data karyawan pada aplikasi. Adapun penjelasannya dapat dilihat pada tabel 5.

Tabel 5. Use Case Description Tambah Data Karyawan

<pre> graph LR TambahData((Tambah Data)) -.-> <<Include>> MenuUtama((Menu Utama)) </pre>	
<i>Name Use Case</i>	Tambah data.
<i>Actor</i>	Admin.
<i>Deskripsi singkat</i>	<i>Usecase</i> menggambarkan bagaimana Admin saat menambah data.
<i>Pre condition</i>	Admin sudah memiliki akun.
<i>Flow of event</i>	<ol style="list-style-type: none"> 1. <i>Use case</i> dimulai ketika admin membuka aplikasi. 2. Admin sudah Login. 3. Admin mengklik menu tambah data. 4. Admin menginputkan data. 5. <i>Use Case</i> Selesai.
<i>Post condition</i>	Sistem akan menyimpan data absensi karyawan.
<i>Alternative flow</i>	Sistem akan menampilkan pesan jika karyawan sudah absen sebelumnya”.

f. *Use Case Description* Laporan
Use case description laporan ini menjelaskan kegiatan yang dilakukan oleh Admin untuk mencetak data absensi karyawan. Adapun penjelasannya dapat dilihat pada tabel 6.

Tabel 6. Use Case Description Laporan

<pre> graph LR Report((Report)) -.-> <<Include>> MenuUtama((Menu Utama)) </pre>	
<i>Name Use Case</i>	Laporan.
<i>Actor</i>	Admin
<i>Deskripsi singkat</i>	<i>Usecase</i> menggambarkan bagai mana Admin membuka menu <i>report</i> .
<i>Pre condition</i>	Admin sudah memiliki akun.

<i>Flow of event</i>	<ol style="list-style-type: none"> 1. <i>Use case</i> dimulai ketika admin membuka menu utama. 2. Admin sudah <i>login</i>. 3. Admin membuka menu laporan. 4. Admin menginputkan nama karyawan, bulan dan tahun. 5. <i>Use Case</i> Selesai
<i>Post condition</i>	Sistem akan menampilkan laporan data yang sudah diinputkan sebelumnya dan siap untuk di cetak sebagai laporan.
<i>Alternative flow</i>	Menampilkan pesan jika data tidak ada.

g. *Use Case Description* Keluar
Use case description keluar ini menjelaskan kegiatan yang dilakukan oleh Admin untuk keluar dari aplikasi. Adapun penjelasannya dapat dilihat pada tabel 7.

Tabel 7. Use Case Description keluar

<pre> graph LR Keluar((Keluar)) -.-> <<Include>> MenuUtama((Menu Utama)) </pre>	
<i>Name Use Case</i>	Keluar.
<i>Actor</i>	Admin.
<i>Deskripsi singkat</i>	<i>Use Case</i> menggambarkan bagaimana admin keluar dari sistem.
<i>Pre Condition</i>	Admin sudah membuka menu utama.
<i>Flow of event</i>	<ol style="list-style-type: none"> 1. <i>Usecase</i> dimulai ketika Admin membuka menu utama aplikasi. 2. Sistem akan menampilkan pesan pilihan keluar dari aplikasi. 3. Sistem akan keluar. 4. <i>Use case</i> selesai.
<i>Post condistion</i>	Aplikasi akan keluar.
<i>Alternative flow</i>	-

2. Struktur Rancangan Mikrokontroler

Pada perancangan dan implementasi sistem akan dijelaskan tentang cara kerja sistem yang terdapat dalam garis besar perancangan sistem dan diikuti dengan penjelasan tentang perancangan dan pembuatan perangkat keras (*hardware*) yang terdiri dari rangkaian mekanik dan rangkaian elektrik yang digunakan pada *prototype* Sistem informasi absensi karyawan dengan menggunakan Arduino Uno. Kemudian diikuti dengan perancangan dan pembuatan perangkat lunak (*software*) yang akan berfungsi untuk mengatur atau mengendalikan pergerakan sebuah sistem.

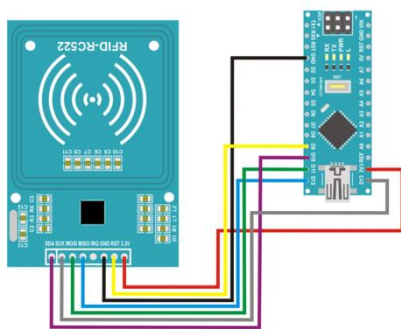


Gambar 8. Rancangan Mikrokontroler

Gambar 8 merupakan komponen yang akan dirancang menjadi satu sebelum merancang pada penelitian ini alat dicoba satu persatu untuk memastikan tidak ada kerusakan pada salah satu alat yang akan digunakan agar penelitian ini berjalan dengan baik.

3. Perancangan RFID

RFID Mifare RC522 RFID Reader Module adalah sebuah modul berbasis IC Philips MFRC522 yang dapat membaca RFID dengan penggunaan yang mudah dan harga yang murah, karena modul ini sudah berisi komponen-komponen yang diperlukan oleh MFRC522 untuk dapat bekerja.

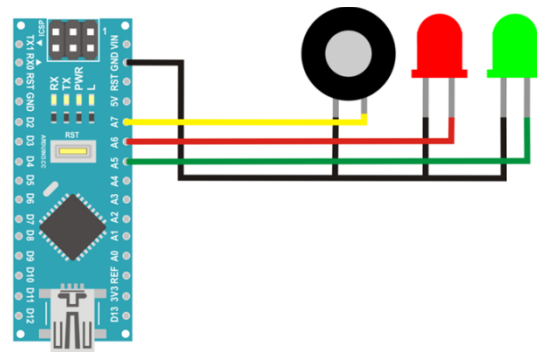


Gambar 9. RFID RC-522

Gambar 9 menunjukkan bentuk perancangan RFID RC-522 dengan arduino tampak depan dan tampak bawah. Setelah semua dihubungkan, bisa mencoba dengan meng-upload program untuk RFID RC-522 dan bisa dilihat langsung di serial monitor sebelum menggunakan LCD maka data bisa dilihat dan data berhasil ditampilkan data ini akan diteruskan ke aplikasi yang akan dibuat.

4. Perancangan Lampu LED dan Buzzer

Lampu LED adalah produk diode pancaran cahaya (LED) yang disusun menjadi sebuah lampu sedangkan buzzer listrik adalah sebuah komponen elektronika yang dapat mengubah sinyal listrik menjadi getaran suara. Lampu LED dan Buzzer digunakan untuk tanda pemberitahuan saat men-scan kartu RFID. berikut perancangan pada Lampu LED dan Buzzer dapat dilihat pada gambar 10.

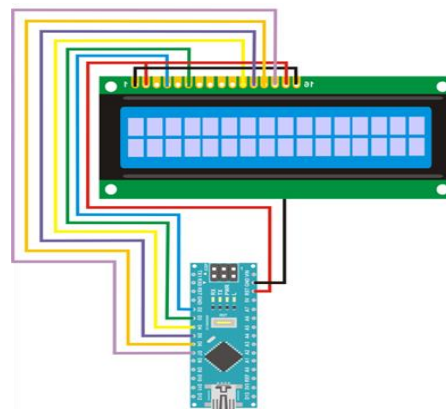


Gambar 10. Perancangan Lampu LED dan Buzzer

Pada gambar 10 sudah dijelaskan untuk perancangan pada mikrokontroler, Lampu LED dan Buzzer dapat diuji coba dengan memasukan program *high low* pada arduino ide dan kemudian *upload* program pada mikrokontroler dan hasilnya dapat dilihat pula pada nyalanya lampu dan suara.

5. Perancangan LCD 16x2

Pada penelitian ini penulis menggunakan LCD 16X2 *Liquid Cristal* sebagai output dari mikrokontroler yang dapat digunakan untuk menampilkan nomor RFID pada *display* LCD, berikut perancangan LCD pada arduino dapat dijelaskan gambar 11.

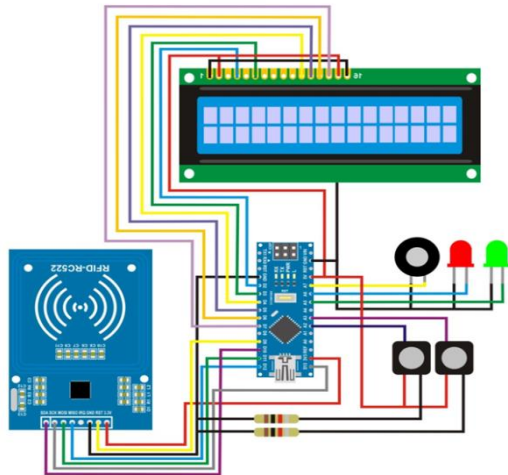


Gambar 11. Perancangan LCD 16 x 2

Pada gambar 11 sudah dijelaskan secara detail dari pemasangan kabel dari arduino ke LCD, setelah proses perancangan selesai LCD dapat dicoba dengan memasukkan program sederhana yang ada di *example* pada aplikasi arduino dengan cara, klik *file example liquid cristal hello word* dan ganti no *pin* pada program agar program tidak *error* pada LCD, setelah mengganti no *pin*, *upload* program ke mikrokontroler dan hasil dapat dilihat di LCD dengan teks "Hello Word" setelah LCD dapat digunakan, pada penelitian ini LCD akan dimanfaatkan lebih tepatnya untuk menampilkan nomor kartu RFID dan pemberitahuan lainnya.

6. Perancangan Mikrokontroler

Setelah melakukan perancangan tahap demi tahap maka dalam penelitian ini semua alat mikrokontroler akan dijadikan satu dimana alat ini akan menjadi sistem informasi absensi karyawan berbasis mikrokontroler maka proses penggabungan alat dapat dilihat pada gambar 12.



Gambar 12. Perancangan Penggabungan Mikrokontroler

Gambar 12 menunjukkan keseluruhan elektronika yang digunakan, dalam penelitian ini dimana mikrokontroler ini akan membaca kartu RFID dan mengirimkan nomor RFID pada aplikasi untuk disimpan pada *database* secara otomatis.

IV. HASIL DAN PEMBAHASAN

Hasil dan pembahasan uji coba terhadap sistem yang dibangun dan dirancang yang akan dikembangkan menggunakan mikrokontroler Arduino Nano, Borland Delphi 7 dan *database* menggunakan MySQL. Masuk ke *menu Setting* (pengaturan), maka akan tampil seperti pada gambar 13.

1. Halaman *Login*

Halaman *login* adalah hal yang harus diisi ketika akan masuk ke aplikasi, di halaman *login* ini terdapat dua kolom yang harus diisi yaitu *username* dan *password*.



Gambar 13. Tampilan Login

2. Halaman Menu Utama

Pada menu utama akan tampil menu data Absensi Karyawan dan terdapat beberapa tombol yang nanti akan digunakan.



Gambar 14. Tampilan Menu Utama

3. Halaman Tambah Data

Pada halaman pendaftaran kartu Karyawan *user* memasukkan Nama Karyawan, No ID, alamat, No telepon, tanggal lahir dan No RFID data ini disimpan agar kartu terdaftar dan kartu dapat di-*scan*.



Gambar 15. Tambah Data

4. Halaman Absensi

Pada sistem *scan* kartu ini digunakan untuk men-*scan* kartu yang sudah didaftarkan sehingga Karyawan saat men-*scan* kartu maka data Karyawan tersebut akan masuk pada *database*.



Gambar 16. Tampilan Absensi

5. *Form* Laporan Absensi
Form Print digunakan sebagai laporan absensi data Karyawan yang hasilnya bisa dicetak.

No	Tanggal	Jam Masuk	Jam Pulang	Shift	Keterangan
1	27 April 2017	13:15	13:20	Shift 2	Masuk

Gambar 17. Tampilan Laporan Absensi Data Karyawan

V. KESIMPULAN

Berdasarkan hasil penelitian dan analisis yang telah dilakukan maka dapat diambil kesimpulan sebagai berikut:

1. Keuntungan yang diperoleh dari pemanfaatan RFID sangat membantu pada proses indentifikasi absensi kehadiran karyawan dengan ID kartu yang berbeda-beda, sehingga tidak ada lagi akan kesalahan absensi.
2. Dengan adanya aplikasi absensi kehadiran karyawan maka aplikasi dapat terkoneksi dengan alat sehingga data absensi karyawan dapat disimpan secara otomatis pada *database* dengan waktu yang sangat cepat.

DAFTAR PUSTAKA

- Dinata Marta Yuwono. (2015), Arduino itu Mudah. Jakarta: PT Elex Media Komputindo
<http://ojs.amikom.ac.id/index.php/semnasteknomedia/article/view/828/794> ISSN: 2302-3805
- Jurnal CoreIT, Vol.1, No.2, Desember 2015 ISSN: 2460-738X
[http://download.portalgaruda.org/article.php?article=387732&val=8534&title=Perancangan%20Sistem%20Absensi%20Kehadiran%20Perkuliahan%20dengan%20Menggunakan%20Radio%20Frequency%20Identification%20\(RFID\)](http://download.portalgaruda.org/article.php?article=387732&val=8534&title=Perancangan%20Sistem%20Absensi%20Kehadiran%20Perkuliahan%20dengan%20Menggunakan%20Radio%20Frequency%20Identification%20(RFID)) Issn. 1412-0100
 Vol 14, No 2, Oktober William2 Vincent Otniel Pangabea3 , Felix Pandi4
- [http://download.portalgaruda.org/article.php?article=120276&val=5499&title=SISTEM%20ABSENSI%20BERBASIS%20RADIO%20FREQUENCY%20IDENTIFICATION%20\(RFID\)%20PADA%20MIKROSKIL](http://download.portalgaruda.org/article.php?article=120276&val=5499&title=SISTEM%20ABSENSI%20BERBASIS%20RADIO%20FREQUENCY%20IDENTIFICATION%20(RFID)%20PADA%20MIKROSKIL) ISSN: 1410-2331 Vol. 19, No. 3, Oktober 2015: 211-216

<http://download.portalgaruda.org/article.php?article=372529&val=7211&title=PERANCANGAN%20DAN%20PENGUJIAN%20MINIATUR%20LIFT%20BERBASIS%20ARDUINO%20DENGAN%20MENGUNAKAN%20RFID%20SEBAGAI%20SISTEM%20IDENTIFIKASI%20LANTAI> Sistem Komputer Untan Volume 03, No. 1 (2015), hal 22-31 [1]

[http://download.portalgaruda.org/article.php?article=318724&val=5162&title=PROTOTYPE%20SISTEM%20KEAMANAN%20PINTU%20MENGUNAKAN%20RADIO%20EQUENCY%20IDENTIFICATION%20\(RFID\)%20DENGAN%20KATA%20SANDI%20BERBASIS%20MIKROKONTROLER](http://download.portalgaruda.org/article.php?article=318724&val=5162&title=PROTOTYPE%20SISTEM%20KEAMANAN%20PINTU%20MENGUNAKAN%20RADIO%20EQUENCY%20IDENTIFICATION%20(RFID)%20DENGAN%20KATA%20SANDI%20BERBASIS%20MIKROKONTROLER)

Nur Lutfian, Susilo Bangbang, Komar Nur. (2014). "Pencatat Digital Keluar Masuknya Beras dalam Gudang Berbasis RFID (Radio Frequency Identification) dengan Menggunakan Bahasa Pemrograman Delphi 7.0." *Jurnal Keteknikan Pertanian Tropis dan Biosistem*. Vol. 2. No. (1). 74-82.