**BAB III**

**ANALISA DAN RANCANGAN SOFTWARE**

**3.1. Analisa Kebutuhan Software**

**3.1.1. Identifikasi Masalah**

Pada zaman yang moderen ini masih banyak perusahaan-perusahaan yang belum menggunakan teknologi terbaru, seperti pada perusahaan CV.Garuda Infinity Kreasindo dalam hal melakukan absensi karyawan. Di CV.Garuda Infinity Kreasindo ini dalam melakukan absensi masih menggunakan *google sheet* yang harus mengetik untuk melakukan absensi.

Berdasarkan latar belakang tersebut maka penulis ingin mengimplementasikan teknologi iBeacon tersebut dalam bentuk sistem absensi, agar bermanfaat dan membantu dalam meningkatkan efektifitas, efisiensi kerja serta menghasilkan kepuasan pengguna dalam menggunakan teknologi tersebut. Berdasarkan latar belakang diatas maka masalah yang teridentifikasi sebagai berikut :

1. Dalam pengolahan data absensi dan rekap absensi, admin masih menggunakan cara yang manual yaitu dengan menggunakan *Google Sheet.*
2. karyawan masih melakukan absesnsi secara manual dengan mengisi *Google Sheet.*
3. Data absensi karyawan belum *real time*.
   * 1. **Analisa Kebutuhan Aplikasi**

Hal pertama yang perlu dilakukan dalam analisis kebutuhan sistem adalah menentukan dan mengungkapkan kebutuhan sistem. Kebutuhan sistem terbagi menjadi dua yaitu: kebutuhan sistem fungsional dan kebutuhan sistem nonfungsional, yang diperlukan untuk mencapai tujuan yang ingin dicapai.

Sebuah sistem dengan sekumpulan elemen-elemen yang diorganisasikan untuk mencapai tujuan tertentu dengan cara mengolah infromasi yaitu perangkat lunak, perangkat keras, dokumentasi, manusia dan prosedur. Kerjasama dari semua elemen-elemen tersebut diharapkan mampu memberikan kemampuan terhadap system. Kemampuan-kemampuan tersebut antara lain :

1. Penggunamemporoleh informasi mengenai waktu kedatangan secara *real time* di kantor.
2. Penggunamemperoleh informasi mengenai waktu kepergian secara *real time* dari kantor.
3. Penggunamemperoleh informasi jam kerja setiap harinya.
4. Penggunamemperoleh informasi rekap absen per bulan.
5. *Admin* dapat merekap absensi bulanan karyawan CV. Garuda Infinity Kreasindo.
6. *Admin* dapat mereset *password account* pengguna*.*
7. *Admin* dapat mereset semua absensi per bulan.
8. *Admin* dapat menghapus akun pengguna*.*
9. Sistem yang dibangun dengan model pendeteksi *device* ini diharapkan akan memberikan kemudahan dalam mengakses informasi yang diinginkan oleh *user.*­­­­­­

Sehingga dengan memiliki kemampuan di atas maka *prototype* sistem ini dapat menjadi sebuah pengembangan yang nantinya siap untuk diimplementasikan dan *service* yang ada dapat di letakan dalam sebuah bentuk layanan *mobile* di jaringan nirkabel *(Wireless Network)*.

1. **Analisis Kebutuhan Fungsional**

Kebutuhan fungsional adalah kebutuhan-kebutuhan yang memiliki keterkaitan langsung dengan sistem.

Kebutuhan fungsional dari aplikasi ini meliputi:

1. Melihat rekap absensi karyawan per bulan di CV. Garuda Infinity Kreasindo.
2. Membuat ijin jika tidak dapat bekerja.
3. Melakukan absensi otomatis secra *real* *time.*
4. Mengganti *password user.*
5. Melakukan registrasi pengguna.
6. Melihat absensi karyawan
7. Menghapus *user.*
8. Mereset *password user.*
9. **Analisis kebutuhan *Non*-Fungsional**

Kebutuhan *non*-fungsional merupakan analisis terhadap sistem yang sudah ada serta kebutuhan terhadap sistem yang akan diusulkan. Adapun kebutuhan *non*-fungsional untuk menjalankan aplikasi absensi otomatis menggunakan iBeacon meliputi kebutuhan perangkat lunak, kebutuhan perangkat keras dan pengguna sistem yang akan menggunakan aplikasi.

* + - 1. **Analisis dan Kebutuhan Perangkat Keras**

Analisis perangkat keras *(hardware)* merupakan proses analisis yang lebih menekankan kepada aspek pemanfaatan perangkat keras yang diperlukan untuk menggunakan aplikasi absensi otomatis menggunakan iBeacon.

Perangkat keras yang saat ini tersedia di CV. Garuda Infinity Kreasindo, telah memenuhi kebutuhan minimum untuk mengoprasikan aplikasi absensi otomatis menggunakan iBeacon yang dibangun, berikut adalah minimal spesifikasi perangkat keras, device *client*, server :

|  |  |
| --- | --- |
| **Perangkat keras** | ***Server*** |
| *Processor* | Intel core i3 |
| *Storage* | 160 GB |
| RAM | 2GB |
| VGA *Card* | Onboard |
| Monitor | 1,136 x 640 px |

**Tabel III.1 Spesifikasi *Server***

|  |  |
| --- | --- |
| **Perangkat Keras** | ***Device client*** |
| *Processror* | 1.3 GHz dual-core Apple-designed ARMv8-A 64-bitApple A7 (128 KBL1 + 1 MB L2 + 4 MB L3)with M7 motion coprocessor. |
| RAM | 2 GB |
| Memory Internal | Default |
| GPU | Default |
| Kamera | 8 MP |

**Tabel III.2 Spesifikasi *Devcie* *client***

Berikut ini adalah spesifikasi minimum perangkat keras yang digunakan dalam membangun dan mengimplementasikan apliksi absensi otomatis menggunakan iBeacon:

|  |  |
| --- | --- |
| **Perangkat Keras** | **Spesifikasi Perangkat Keras** |
| *Processror* | iOs 9 |
| RAM | Minimal 1GB |
| Memory Internal | 16 GB |
| GPU | Default |

**Tabel III.3 Spesifikasi Minimum Perangkat Keras**

* + - 1. **Analisis dan Kebutuhan Perangkat Lunak**

Analisis perangkat lunak *(software)* merupakan proses analisis yang lebih menekankan kepada aspek pemanfaatan perangkat lunak. Berdasarkan analisis, spesifikasi perangkat lunak untuk menjalankan aplikasi absensi otomatis menggunakan iBeacon adalah sebagai berikut :

|  |  |
| --- | --- |
| ***Software*** | ***Server*** |
| Sistem operasi | macOS Sierra |
| *Web Server* | Apache engine |
| *Database* | MySQL |

**Tabel III.4 Spesifikasi Perangkat Lunak**

|  |  |
| --- | --- |
| ***Software*** | ***Device Client*** |
| Sistem operasi | iOs 10 |
| Pemrograman | Swift 3.1 |
| Software Desain | xCode |
| Database | Core Data |

**Tabel III.5 Spesifikasi *Device client***

Berikut ini adalah spesifikasi perangkat lunak yang digunakan dalam pembuatan aplikasi absensi otomatis menggunakan iBeacon dengan basis mobile, yaitu sebagai berikut :

1. Sistem Operasi macOS Sierra.
2. Aplikasi xCode*.*
3. Platform iOS versi 10.

**3.2. Design**

**3.2.1. Rancangan Algoritma**

Dibutuhkan algoritma utnuk diimplementasikan ke dalam program aplikasi ini, yaitu langkah-langkah instruksi sehingga dicapai hasil yang diinginkan. Algoritma yang digunakan dalam aplikasi absensi otomatis menggunakan iBeacon adalah sebagai berikut:

1. *Admin* melakukan registrasi untuk pengguna.
2. Pengguna masuk menggunakan akun yang telah dibuat oleh admin dengan kata sandi yang telah ditetapkan, yaitu “a”.
3. Pengguna menekan tombol masuk.
4. Setelah berhasil melakukan proses login, pengguna akan diminta untuk mengubah kata sandi.
5. Setelah berhasil mengubah kata sandi, pengguna akan melakukan kembali proses Login dengan kata sandi yang baru.
6. Pengguna akan berada pada halaman beranda.
7. Pengguna memilih tombol rekap absensi untuk mendapatkan informasi absensi.
8. Pengguna memilih tombol izin untuk membuat laporan izin kepada pimpinan.
9. **Perancangan Sistem**

Sebelum membuat program aplikasi, terlebih dahulu dilakukan proses perancangan sistem menggunakan *tools* Power Designer15. Menurut Kendall (2003) hal ini digunakan untuk memodelkan perancangan yang telah ditetapkan berdasarkan analisis, sehingga menghasilkan informasi yang dibutuhkan.

Urutan perancangan sistem adalah sebagai berikut :

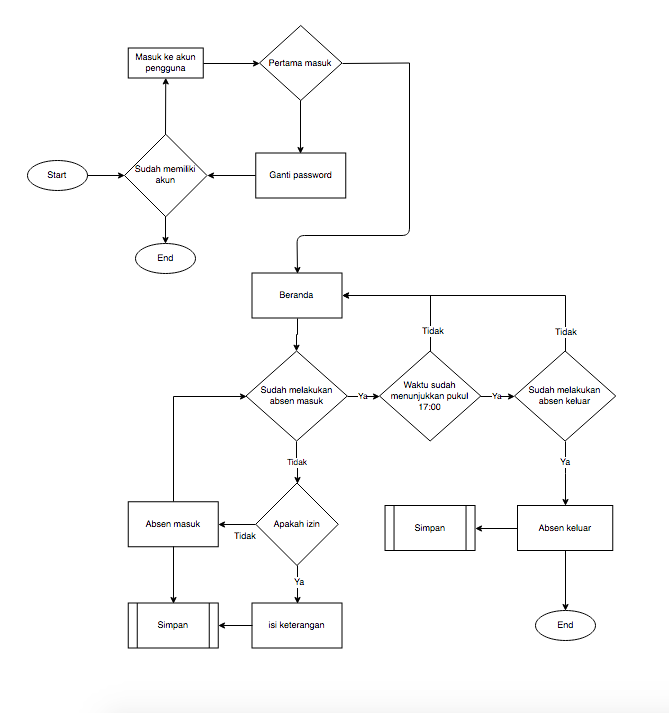
1. Sistem Flow
2. Data Flow Diagram (DFD)

Berikut langkah-langkah pemodelan sistem yang dapat memperjelas desain aplikasi yang akan dikembangkan.

1. **Sistem Flow**

Sistem *Flow* merupakan bagan alir sistem yang menghubungkan jalannya proses dari sistem. Dibawah ini merupakan sistem flow dari aplikasi absensi otomatis mengguanakan iBeacon:

* 1. **Sistem Flow Absensi Otomatis dengan menggunakan iBeacon**

****

**Gambar III.1 Sistem *Flow* Absensi Otomatis menggunakan iBeacon**

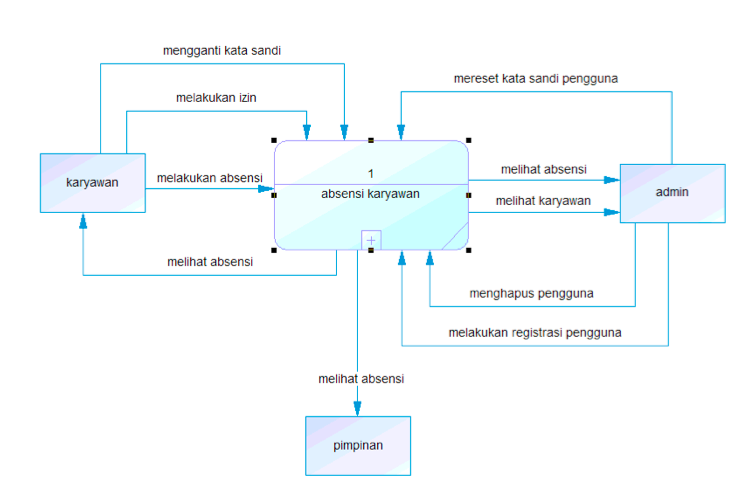
Gambar III.1 menunjukan sistem *flow* absensi otomatis menggunakan iBeacon. Pertama kali sistem akan membaca apakah pengguna sudah memiliki akun atau belum. Pengguna yang sudah memiliki akun akan melakukan proses *Login*. Setelah itu sistem akan membaca apakah pengguna ini baru pertama kali masuk atau tidak. Jika pengguna yang baru pertama kali masuk, maka sistem akan meminta pengguna untuk merubah/mengganti password terlebih dahulu. Untuk pengguna yang bukan pertama kali masuk maka sistem akan menampilkan halaman beranda. Kemudian, sistem akan membaca apakah pengguna sudah melakukan absen masuk. Sistem akan menyimpan data absen masuk pengguna tersebut. Selanjutnya pengguna melakukan absen keluar, maka sistem akan membaca apakah jam sudah menunjukan pukul 17.00 jika sudah maka sistem akan menyimpan data absen keluar pengguna tersebut.

1. ***Data Flow Diagram* (DFD)**

Data Flow Diagram (DFD) adalah Tabel aliran informasi yang terlibat dalam suatu prosedur yang terdpat dalam suatu sistem (Jogiyanto, 1998). Data flow suatu sistem dapat diawali dengan Context Diagram yang menjelaskan hubungan atau interaksi sistem dengan entitas-entitas yang mempunyai keterkaitan dengan sistem.

1. ***Context Diagram* (CD)**

*Context diagram* merupakan diagram pertama dalam rangkaian suatu DFD, yang menggambarkan entitas-entitas yang berhubungan dengan suatu sistem. Pada *context diagram* aplikasi absesnsi otomatis menggunakan iBeacon terdapat 2 eksternal *entity*, yaitu pelanggan dan admin. Masing-masing dari *entity* tersebut memberikan *input* dan oleh sistem akan diberikan keluaran atau *output* yang berupa laporan atau lainnya. Untuk lebih jelasnya context diagram dapat dilihat pada gambar III.2

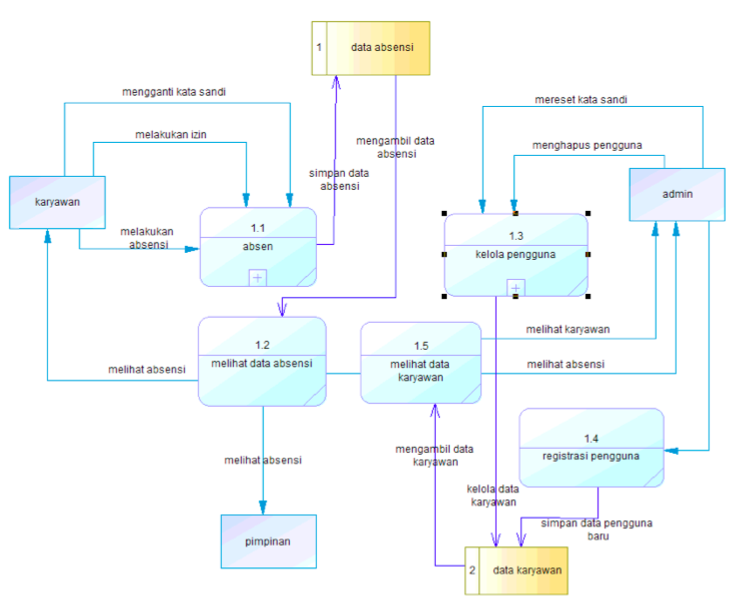


**Gambar III.2 *Context Diagram***

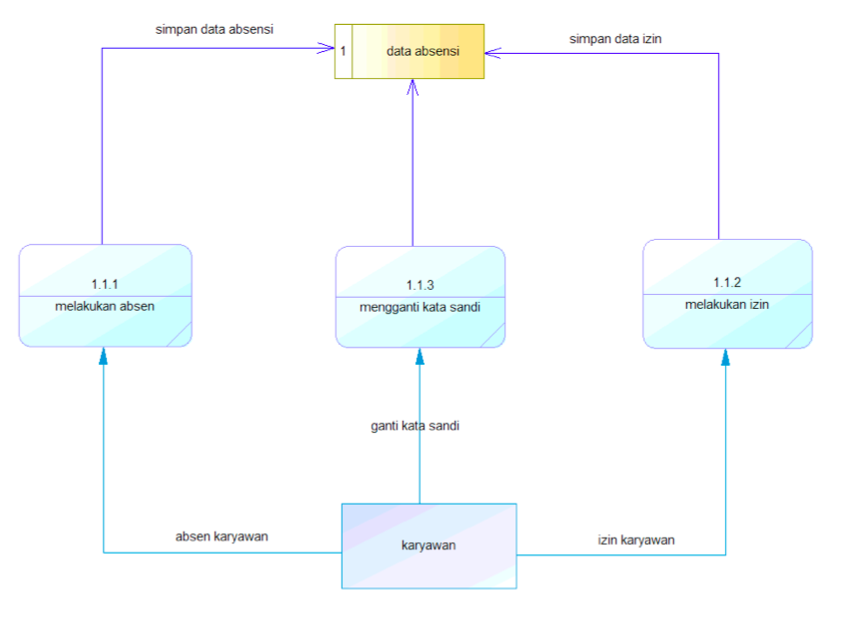
1. ***Data Flow Diagram***

*Data Flow Diagram* (DFD) dibuat untuk menggambarkan arus data yang terstruktur dan jelas dari mulai pengisian data sampai dengan keluarannya. Arus data pada *Data Flow Diagram* (DFD) ini berupa masukan untuk sistem oleh entitas eksternal atau keluaran dari sistem. Adapun gambar *Data Flow Diagram* (DFD) ini dapat dilihat pada gambar dibawah ini :

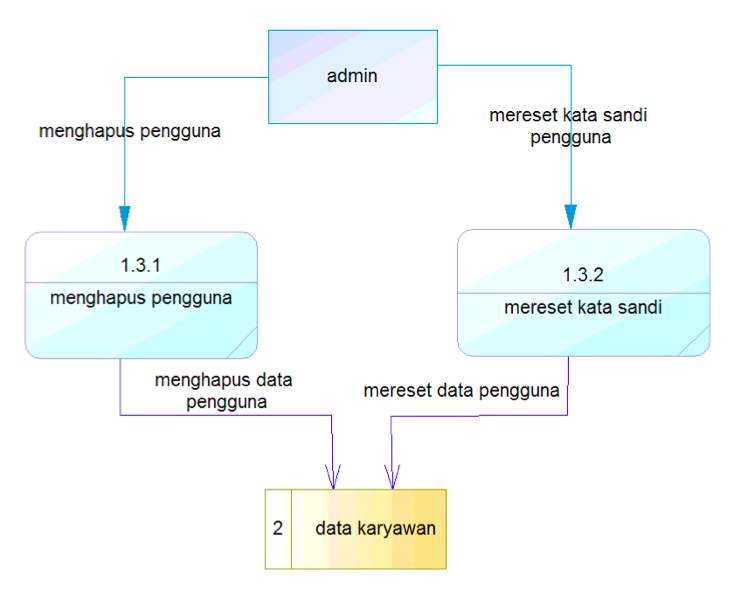
1. ***Data Flow Diagram* (DFD) level 0**



**Gambar III.3 *Data Flow Diagram*  level 0**

1.  ***Data Flow Diagram* (DFD) level 1**

**Gambar III.4 *Data Flow Diagram* level 1 Absensi Karyawan**

****

**Gambar III.5 *Data Flow Diagram* level 1 Data Karyawan**

* + 1. ***Database***

Pada tahap ini penulis melakukan perancangan *database* yang terdiri dari kamus data, normalisasi, desain table dan relasi antar table aplikasi absensi otomatis menggunakan iBeacon adalah sebagai berikut :

1. **Kamus Data**

Kamus data digunakan sebagai acuan dalam pembangunan suatu *database* dan sebagai panduan untuk keperluan pengembangan sistem *database* adapun table atau entitas yang dibentuk adalah seperti berikut ini :

1. **Tabel Data Absensi**
2. NIK
3. Username
4. Password
5. Nama\_Lengkap
6. No\_KTP
7. Tempat\_Tanggal\_Lahir
8. Alamat
9. Jenis\_Kelamin
10. Jabatan
11. Status\_Karyawan
12. Status
13. Change\_Password
14. **Data Karyawan**
15. Id\_absensi
16. NIK
17. tanggal
18. jam\_masuk
19. jam\_keluar
20. jam\_lembur
21. izin
22. keterangan
23. **Desain Tabel**

Adapun rancangan desain tabel *database* yang penulis gunakan dalam sistem aplikasi absensi otomatis menggunakan iBeacon di CV. Garuda Infinity Kreasindo adalah sebagai berikut :

1. **Tabel Data Absensi**

Tabel Data Absensi ini digunakan untuk akses login pelanggan dan admin ke sistem aplikasi absensi otomatis menggunakan iBeacon di CV. Garuda Infinity Kreasindo.

Nama *database* : Karyawan

Nama tabel : Data Absensi*.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Nama *Field*** | **Tipe Data** | **Panjang** | **Keterangan** |
| 1. | NIK | Integer | 11 | Primary Key |
| 2. | Username | varchar | 15 |  |
| 3. | Password | varchar | 200 |  |
| 4. | Nama\_Lengkap | Varchar | 60 |  |
| 5. | No\_KTP | varchar | 16 |  |
| 6. | Tempat\_Tanggal\_Lahir | varchar | 60 |  |
| 7. | Alamat | varchar | 60 |  |
| 8. | Jenis\_Kelamin | Varchar | 15 |  |
| 9. | Jabatan | Varchar | 15 |  |
| 10. | Status\_Karyawan | Varchar | 15 |  |
| 11. | Status | Varchar | 15 |  |
| 12. | Change\_Password | Varchar | 11 |  |

**Tabel III.6 Data Absensi**

1. **Tabel Data Karyawan**

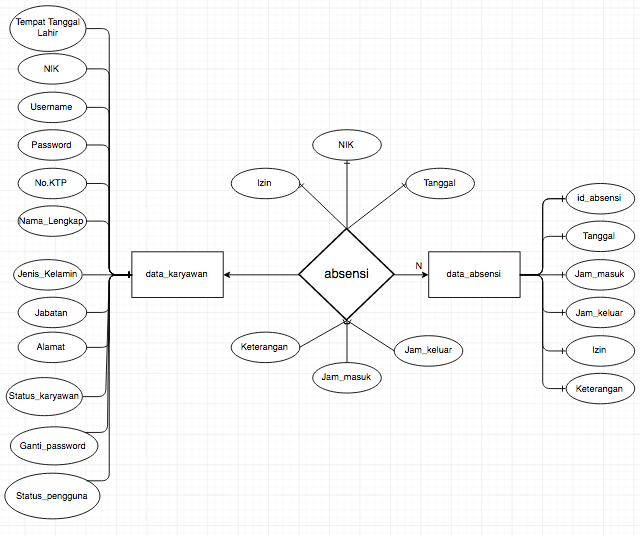
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Nama *Field*** | **Tipe Data** | **Panjang** | **Keterangan** |
| 1. | Id\_absensi | Integer | 11 | Primary Key |
| 2. | NIK | Integer | 11 |  |
| 3. | Tanggal | Date |  |  |
| 4. | jam\_masuk | Time |  |  |
| 5. | Jam\_keluar | Time |  |  |
| 6. | jam\_lembur | Integer | 11 |  |
| 7. | Izin | Integer | 11 |  |
| 8. | keterangan | Text |  |  |

**Tabel III.7 Data Karyawan**

1. ***Entity Relationship Diagram* (ERD)**

*Entity Relationship Diagram* atau ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi.

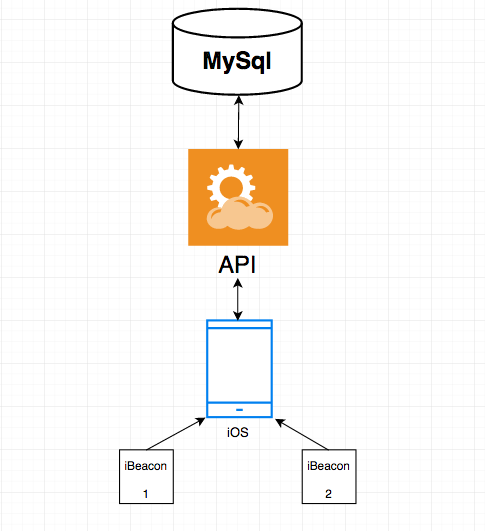
*Entity Relationship Diagram* (ERD) untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkannya digunakan beberapa notasi dan simbol. Gambar di bawah ini mengambarkan bagaimana relasi antar entitas yang saling berhubungan pada aplikasi absensi otomatis menggunakan iBeacon. Gambaran ERD pada aplikasi absensi otomatis dapat dilihat pada Gambar berikut



**Gambar III.6 *EntityRelationship Diagram***

* + 1. ***Software Architecture***

*Software Architecture* adalah tahapan untuk mendapatkan gambaran umum sistem yang akan dibangun. Baik itu gambaran sistem *backend* admin maupun gambaran sistem *frontend*  pengguna *mobile*. Gambaran arsitektur sistem dapat dilihat pada Gambar berikut

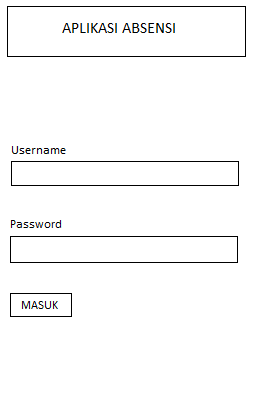


**Gambar III.7 *Software Architecture* Aplikasi Absensi Otomatis**

* iBeacon 1 akan *mentrigger client* yang melewat di area sekitar agar mengirimkan data absensi masuk ke API*.*
* iBeacon 2 akan *mentrigger client* yang melewat di area sekitar agar mengirimkan data absensi keluar ke API*.*
* iOS *device* akan menerima *trigger* dari iBeacon dan setelah itu mengambil data dari *device* tersebut lalu mengirimkannya keAPI.
* API akan menerima dan mengirim data dari iOS *device* yang berupa JSON dan kemudian memeriksa data yang di kirim, setelah itu API akan mengirimkan data tersebut ke *MySql.*
* *MySql* akan menerima data dari API dan akan menyimpan data tersebut.
  + 1. **User Interface**

1. Perancangan *Interface* *Login*

Perancangan *Login* digunakan untuk pengguna saat akan mengakses informasi absensi karyawan yang di dalamnya berupa *form Username* dan *Password.*

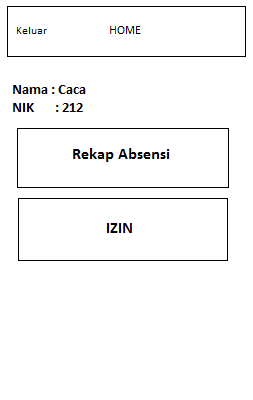
****

**Gambar III.8 *Interface Login***

|  |  |
| --- | --- |
| Informasi | Fungsi |
| Aplikasi Absensi | Menampilkan judul aplikasi. |
| *Username* | Menampilkan inputan username yang dilakukan oleh pengguna. |
| *Password* | Menampilkan inputan password yang dilakukan oleh pengguna. |
| Masuk | Meneruskan ke *home page*. |

**Tabel III.8 Keterangan *Interface Login***

1. Perancangan *Interface Home Page*

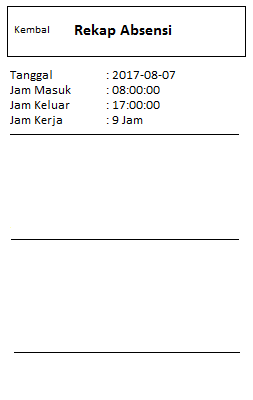


**Gambar III.9 *Interface Home Page***

|  |  |
| --- | --- |
| Informasi | Fungsi |
| *Home* | Menampilkan judul pada halaman. |
| Nama, NIK | Menampilkan Informasi pengguna. |
| Rekap Absensi | Meneruskan ke halaman Rekap Absensi. |
| Izin | Meneruskan ke halaman Izin. |
| Keluar | Meneruskan ke halaman *Login.* |

**Tabel III.9 Keterangan *Interface Home Page***

1. **Perancangan *Interface* Rekap Absensi**

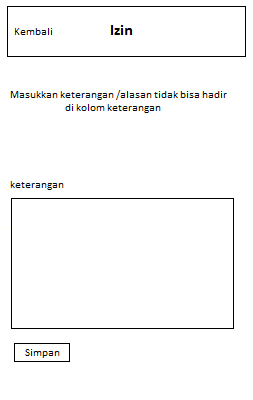
****

**Gambar III.10 *Interface* Rekap Absensi**

|  |  |
| --- | --- |
| Informasi | Fungsi |
| Rekap Absensi | Menampilkan judul pada halaman. |
| Tanggal, jam masuk, jam keluar, jam kerja | Menampilkan Informasi absensi pengguna |
| Kembali | Meneruskan ke *home page* |

**Tabel III.11 Keterangan *Interface* Rekap Absensi**

1. **Perancangan *Inteface* Izin**

****

**Gambar III.11 *Interface* Izin**

|  |  |
| --- | --- |
| Informasi | Fungsi |
| Izin | Menampilkan judul pada halaman. |
| Keterangan | Menampilkan keterangan izin yang dibuat oleh pengguna. |
| Simpan | Menyimpan data ke *database* dan meneruskan ke *home page.* |

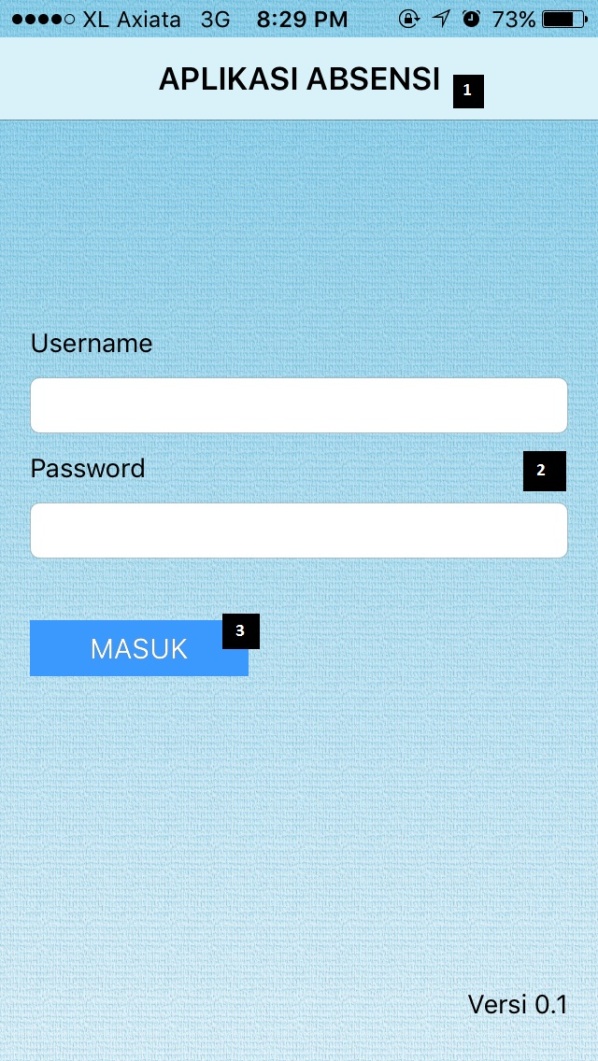
**Tabel III.11 Keterangan *Interface* Izin**

* 1. **Implementasi**

Pada tahap implementasi, rancangan desain yang telah dirancang diimplementasikan kedalam bahasa pemrograman yang dapat dieksekusi oleh komputer yang biasa disebut dengan istilah *coding*. Berdasarkan pada desain yang telah dirancang oleh peneliti, tahap implementasi terbagi dalam 2 lingkungan yaitu lingkungan implementasi *user interface* dan lingkungan implementasi sistem.

**3.3.1. Implementasi *User Interface***

Tampilan antarmuka pada aplikasi telah dirancang dengan sederhana yang bertujuan untuk memudahkan *user* dalam penggunaannya. Aplikasi ini diberi nama Aplikasi absensi otomatis menggunakan iBeacon.



**Gambar III.12 Implementasi *Login***

Keterangan Gambar III.10 :

1. Judul Halaman
2. Text area untuk *Username* dan *Password*
3. *Button* masuk ke *home page*.



**Gambar III.13 Implementasi *Home Page***

Keterangan Gambar III.11 :

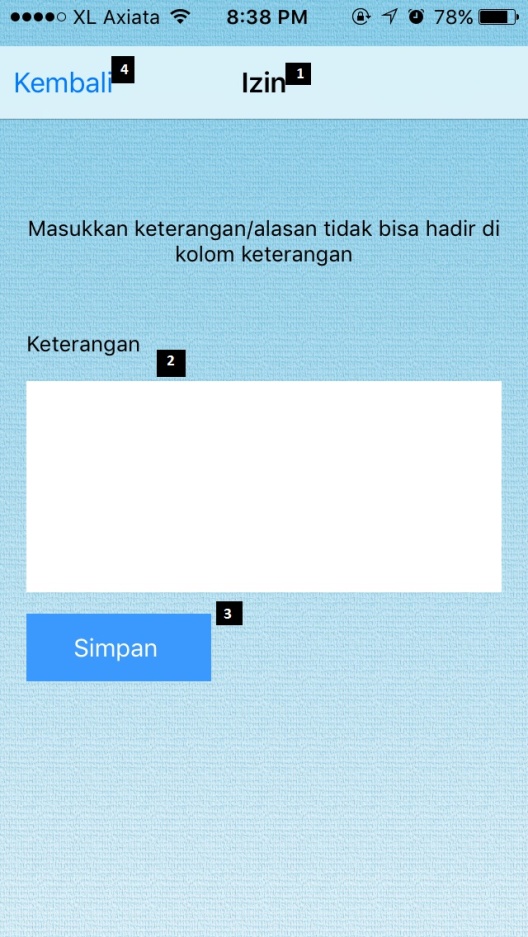
1. Judul Halaman.
2. Keterangan identitas pengguna.
3. *Button* untuk masuk ke halaman rekap absensi.
4. *Button* untukmasuk ke halaman izin.
5. Untuk keluar dari akun pengguna.



**Gambar III.14 Implementasi Rekap Absensi**

Keterangan Gambar III.12 :

1. Judul Halaman.
2. Informasi absensi karyawan.
3. *Button* untuk kembali ke *home page*.



**Gambar III.15 Implementasi izin**

Keterangan Gambar III.13 :

1. Judul Halaman.
2. Text area untuk pengguna menuliskan keterangan izin.
3. *Button* untuk menyimpan keterangan izin dan meneruskan ke *home page*.
4. Untuk kembali ke *home page.*
   1. **Pengujian (*Testing)***

**3.4.1. *Black Box Testing***

Aplikasi ini dibangun dengan dua proses ini yaitu menampilkan informasi rekap absensi dan menampilkan form izin pengguna. Untuk mengetahui bahwa sistem dapat berjalan sesuai dengan fungsionalitas yang diterapkan, maka perlu dilakukan pengujian. Berikut ini merupakan pengujian menu pada aplikasi:

1. Halaman *Log In*

|  |  |  |
| --- | --- | --- |
| Pengujian | Fungsi | Hasil Pengujian |
| halaman *Log In* | Untuk mengakses sistem | sesuai |
| Input *Username* | Untuk memasukan username pengguna | Sesuai |
| Input *Password* | Untuk memasukan password pengguna | Sesuai |
| Memilih *button Masuk* | Untuk mengakses *home page* | sesuai |

**Tabel III.12 Keterangan halaman *Log In***

1. Beranda (*home page*)

|  |  |  |
| --- | --- | --- |
| Pengujian | Fungsi | Hasil Pengujian |
| *Home page* | Untuk mengakses informasi absensi dan membuat izin. | sesuai |
| Keterangan identitas pengguna | Untuk mengetahui identitas pengguna | Sesuai |
| Memilih *button* Rekap Absensi | Untuk mengakses informasi rekap absesnsi. | Sesuai |
| Memilih *button* izin | Untuk mengakses pembuatan izin. | Sesuai |
| Memilih *button* keluar | Untuk keluar dari akun pengguna dan kembali ke halaman *Log In* | Sesuai |

**Tabel III.13 Keterangan halaman beranda karyawan**

1. Menu Rekap Absensi

|  |  |  |
| --- | --- | --- |
| Pengujian | Fungsi | Hasil Pengujian |
| Memilih *button* Rekap Absensi | Untuk mengakses informasi rekap absensi | sesuai |
| Informasi detail absensi | Untuk melihat informasi absensi | sesuai |
| Memilih *button* kembali | Untuk kembali ke *home page.* | Sesuai |

**Tabel III.14 Keterangan halaman rekap absensi karyawan**

1. Menu Izin

|  |  |  |
| --- | --- | --- |
| Pengujian | Fungsi | Hasil Pengujian |
| Memilih *button* Izin | Untuk mengakses halaman pembuatan izin karyawan | sesuai |
| Text area penulisan izin | Untuk pembuatan izin karyawan | sesuai |
| Memilih *button* simpan | Untuk menyimpan izin karyawan ke *database.* | Sesuai |
| Memilih *button* kembali | Untuk kembali ke *home page* | Sesuai |

**Tabel III.15 Keterangan halaman izin karyawan**

1. Beranda admin

|  |  |  |
| --- | --- | --- |
| Pengujian | Fungsi | Hasil Pengujian |
| Memilih *button* registrasi karyawan | Untuk mendaftarkan data karyawan baru | sesuai |
| Absensi *button* karyawan | Untuk melihat semua data absensi karyawan | sesuai |
| Memilih *button* keluar | Untuk kembali ke halaman *Log In* | Sesuai |

**Tabel III.16 Keterangan halaman beranda admin**

1. Registrasi karyawan

|  |  |  |
| --- | --- | --- |
| Pengujian | Fungsi | Hasil Pengujian |
| Memilih *button* registrasi karyawan | Untuk mendaftarkan data karyawan baru | sesuai |
| *Input textfield* NIK | Untuk memasukkan nomor induk karyawan | sesuai |
| *Input textfield username* | Untuk memasukkan *username* karyawan baru | Sesuai |
| *Input textfield* Nama Lengkap | Untuk memasukkan nama lengkap karyawan | sesuai |
| *Input textfield* Tempat tanggal lahir | Untuk memasukkan Tempat tanggal lahir karyawan baru | Sesuai |
| *Input textfield* No.KTP | Untuk memasukkan nomor kartu tanda penduduk karyawan | sesuai |
| *Input textfield* Alamat Lengkap | Untuk memasukkan alamat lengkap karyawan baru | Sesuai |
| *Input textfield* Jenis Kelamin | Untuk memasukkan Jenis kelamin karyawan baru | sesuai |
| *Input textfield* Jabatan | Untuk memasukkan Jabatan karyawan baru | Sesuai |
| *Input PickerView* Status karyawan | Untuk memasukkan Status karyawan baru | sesuai |
| *Input PickerView* Status pengguna | Untuk memasukkan Status pengguna karyawan baru | Sesuai |
| Memilih *button* kembali | Untuk kembali ke halaman beranda admin | Sesuai |

**Tabel III.17 Keterangan halaman registrasi karyawan**

1. Rekap absensi admin

|  |  |  |
| --- | --- | --- |
| Pengujian | Fungsi | Hasil Pengujian |
| Memilih *button* absensi karyawan | Untuk Melihat seluruh data karyawan | sesuai |
| *TableView select* nama karyawan | Untuk melihat data absensi karyawan terpilih | sesuai |
| Memilih *button* kembali | Untuk kembali ke halaman beranda admin | Sesuai |

**Tabel III.18 Keterangan halaman beranda admin**