

# **Perancangan Aplikasi Penggajian *Student Labor* Di Departemen Perpustakaan Universitas Advent Indonesia Menggunakan Rfid Berbasis Web**

**Raymond Maulany dan Dina Yosephine Simanjuntak  
Fakultas Teknologi Informasi, Universitas Advent Indonesia**

## **Abstrak**

Teknologi merupakan bagian dari kehidupan manusia saat ini dalam mendukung kegiatan sehari-hari dan sebagai alat yang digunakan untuk mempermudah pekerjaan. Salah satu contoh yang sedang mengalami permasalahan di bidang teknologi, yaitu Universitas Advent Indonesia mengenai penggajian *student labor* yang masih dilakukan secara manual. Perancangan yang telah penulis lakukan adalah membangun aplikasi menggunakan bahasa pemrograman PHP dan HTML dan basisdata MySQL 2015. Prosedur yang dirancang adalah mahasiswa mendekati KTM (Kartu Tanda Mahasiswa) ke RFID *Reader*, lalu RFID *Reader* akan membaca identitas kartu itu jika sudah terdaftar, dan memasukkannya ke basis data. Penelitian ini dilakukan agar *student labor* dapat mudah melakukan absensi dan penerimaan gaji setiap bulannya. Pada saat dijalankan, aplikasi ini dapat menyimpan data ke basis data MySQL 2015 yang telah dibuat. Penulis berharap bahwa di masa yang akan datang sistem ini dapat dikembangkan sehingga aplikasi ini dapat berfungsi lebih baik dan lebih lengkap lagi.

**Kata – kata Kunci:** Aplikasi Penggajian, RFID *Reader*, RFID *Card* (KTM), *student labor*.

## **Design of Web Based RFID Student Labor Payroll Application at Library Department Universitas Advent Indonesia**

### **Abstract**

*Technology is a part of human life today in support of daily activities and as a tool used to facilitate the work. One example that is experiencing problems in the field of technology, namely Advent Indonesia University on payroll student labor which is still done manually.*

*The design that I have done is build an application using PHP and HTML programming language and MySQL database 2015. The procedure is designed to bring student KTM (Student Identity Card) to RFID Reader, then RFID Reader will read the identity of the card if it is registered, and put it into database. This research is done so that student labor can be easy to do attendance and receipt of salary every month.*

*At run time, this application can save data to MySQL 2015 database that has been created. The author hopes that in the future this system can be developed so that this application can function better and more complete again.*

### **Pendahuluan**

Teknologi merupakan bagian dari kehidupan manusia saat ini dalam mendukung kegiatan sehari-hari, pendidikan, hiburan, bisnis, dan lain-lain. Teknologi ialah alat yang digunakan untuk mempermudah pekerjaan contoh permasalahan yang dihadapi oleh pihak Universitas Advent Indonesia dalam melakukan *input* gaji, di mana proses pengabsenan dan penggajian ditentukan dari kualitas cara kerja seorang *Student Labor*.

Sistem penggajian *Student Labor* masih dilakukan secara manual dengan melihat kertas kehadiran mahasiswa tersebut. Kemudian, kepala Departemen yang bersangkutan memberikan *grade* dan menyerahkan kertas tersebut kepada Biro Keuangan. Hal itu menyebabkan hambatan bagi Mahasiswa, Akademik, Keuangan, dan Kepala Departemen Perpustakaan dalam penyajian informasi berupa proses penggajian kuliah sambil kerja, di mana mahasiswa yang *Labor* harus sabar menunggu hasil data keuangan dari Kepala Departemen Keuangan karena sering sekali terlambat masuk ke *account* mahasiswa tersebut. Dengan adanya aplikasi penggajian *student labor* menggunakan teknologi RFID, sangat membantu pihak Departemen Perpustakaan dan Biro Keuangan dalam pemberian gaji *student labor*.

## **Landasan Teori Sistem**

Menurut Benner, dkk (2010) "Sistem merupakan suatu kumpulan yang kompleks dan juga saling berinteraksi satu sama lainnya apabila kumpulan tersebut digabungkan menjadi satu kesatuan yang utuh."

Menurut Romney dan Steinbart (2015:3) "Sistem adalah rangkaian dari dua atau lebih komponen-komponen yang saling berhubungan dan berinteraksi untuk mencapai suatu tujuan".

Menurut Mulyadi (2016:5) "Sistem adalah suatu jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan pokok perusahaan".

Menurut Azhar Susanto (2013:22) "Sistem adalah kumpulan atau grup dari sub sistem atau bagian atau komponen apapun baik fisik ataupun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu".

Jadi, dapat disimpulkan bahwa sistem merupakan suatu kumpulan yang kompleks yang saling berinteraksi satu sama lain secara terstruktur dalam mencapai tujuan tertentu.

## **Metode Penelitian**

### **SDLC (*Software Development Life Cycle*)**

Metode SDLC (*Software Development Life Cycle*). SDLC dimulai dari tahun 1960 untuk mengembangkan sistem skala usaha besar secara fungsional untuk para konglomerat pada zaman ini, sistem-sistem yang dibangun mengelola informasi kegiatan dan rutinitas dari perusahaan-perusahaan yang berpotensi memiliki data yang besar dalam perkembangannya.

SDLC adalah proses pengembangan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (Rosa A.S dan M.Shalahudin, 2013).

Menurut Nugroho (2010:2) "Pengembangan/rekayasa sistem informasi (*system development*) dan atau perangkat lunak (*software engineering*) dapat berarti menyusun sistem/perangkat lunak yang benar-benar baru atau yang lebih sering terjadi menyempurnakan yang sebelumnya".

1. Perencanaan (*Planning*), adalah menyangkut studi tentang kebutuhan pengguna, kelayakan baik secara teknik maupun secara teknologi serta penjadwalan suatu proyek sistem informasi atau perangkat lunak.
2. Analisa (*Analysis*), adalah tahap dimana kita berusaha mengenai segenap permasalahan yang muncul pada pengguna dengan mendekomposisi dan merealisasikan *use case diagram* lebih lanjut, mengenai komponen-komponen sistem atau perangkat lunak, objek-objek, hubungan antara objek dan sebagainya.
3. Perencanaan (*Design*), adalah tahap dalam mencoba mencari solusi dari permasalahan yang didapat dari tahap analisis.
4. Implementasi, adalah tahap dalam mengimplementasikan perencanaan sistem ke situasi nyata yaitu dengan pemilihan perangkat keras dan penyusunan perangkat lunak aplikasi (*Coding*).
5. Pengujian (*Testing*), adalah untuk menentukan apakah sistem atau perangkat lunak yang dibuat sudah sesuai dengan kebutuhan pengguna atau belum. Jika belum, proses selanjutnya adalah bersifat iteratif, yaitu kembali ke tahap sebelumnya.
6. Pemeliharaan (*Maintenance*), adalah proses pengoperasian sistem dan jika diperlukan melakukan perbaikan kecil.

### **Metode *Waterfall***

Menurut Pressman (2015:42) "*Waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software*".

Fase-fase dalam metode *Waterfall*, antara lain:

1. *Communication (Project Initiation & Requirements Gathering)*, yaitu sebelum memulai pekerjaan yang bersifat teknis, sangat diperlukan adanya komunikasi dengan *customer* demi memahami dan mencapai tujuan yang ingin dicapai. Pengumpulan data-data tambahan bisa diambil dari jurnal, artikel, dan internet.
2. *Planning (Estimating, Scheduling, Tracking)*, yaitu menjelaskan tentang estimasi tugas-tugas teknis yang akan dilakukan, risiko-risiko yang dapat terjadi, sumber daya yang diperlukan dalam membuat

sistem, produk kerja yang ingin dihasilkan, penjadwalan kerja yang akan dilaksanakan, dan *tracking* proses pengerjaan sistem.

3. Modeling (*Analysis & Design*), yaitu tahap perancangan dan permodelan arsitektur sistem yang berfokus pada perancangan struktur data, arsitektur *software*, tampilan *interface*, dan algoritma program.
4. Construction (*Code & Test*), yaitu proses penerjemahan bentuk desain menjadi kode atau bentuk/bahasa yang dapat dibaca oleh mesin dan untuk menemukan kesalahan yang mungkin terjadi untuk nantinya diperbaiki.
5. Deployment (*Delivery, Support, Feedback*), yaitu tahap implementasi *software* ke *customer*, pemeliharaan *software* secara berkala, perbaikan *software*, evaluasi *software*, dan pengembangan *software* berdasarkan umpan balik yang diberikan agar sistem dapat tetap berjalan dan berkembang sesuai dengan fungsinya. (Pressman, 2015:17).

## Perancangan Sistem

### Use Case Diagram

Menurut Sukanto dan Shalahuddin (2013:155) "Use Case merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat".

Jadi, dapat disimpulkan bahwa *use case* merupakan suatu bentuk yang dapat menggambarkan proses kerja dari suatu sistem yang berhubungan dengan aktor-aktor yang terkait.

### Activity Diagram

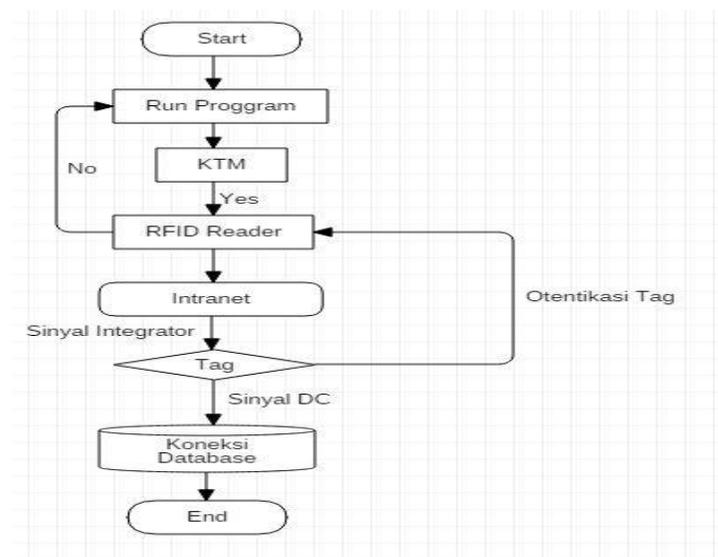
Menurut Sukanto dan Shalahuddin (2013:161) "Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak".

Jadi, dapat disimpulkan bahwa *activity diagram* merupakan gambaran aliran kerja dari sebuah proses bisnis.

### Sequence Diagram

Menurut Sukanto dan Shalahuddin (2013:165) "Sequence Diagram menggambarkan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek".

Jadi, dapat disimpulkan bahwa *sequence diagram* merupakan gambaran objek yang mendeskripsikan proses kerja yang terjadi pada sistem.



Gambar 1 Cara Kerja RFID

KTM digunakan sebagai *RFID Tag* yang berisi sebuah *Tag* yang unik dan berbeda satu dengan yang lainnya. Jadi, informasi mengenai objek yang terhubung ke *tag* ini hanya terdapat pada sistem atau *database* yang terhubung pada *RFID Reader*. Dan *RFID Reader* yang digunakan adalah *RFID* aktif karena memiliki sistem pembaca aktif yang memancarkan sinyal interogator ke *Tag* dan menerima balasan otentikasi dari *Tag*. Sinyal interogator ini juga menginduksi *Tag* dan akhirnya menjadi sinyal DC yang menjadi sumber daya *Tag* Pasif.

## Analisa Dan Perancangan Sistem

### Analisa Sistem Lama

Setelah penulis melakukan proses pengamatan melalui wawancara lapangan mengenai proses bisnis yang terjadi di perpustakaan Universitas Advent Indonesia, maka terdapat beberapa permasalahan yang timbul dalam sistem yang sedang berjalan saat ini.

Permasalahan tersebut, antara lain:

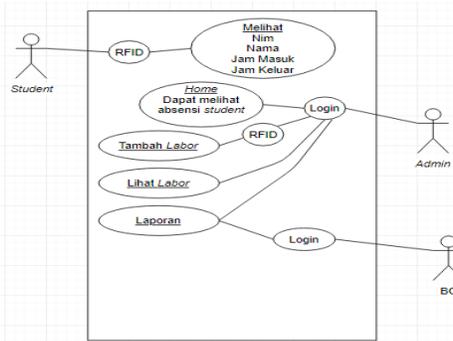
1. Sistem yang digunakan masih manual.
2. Media yang digunakan dalam penggajian masih sederhana.
3. Keterlambatan dalam pemberian gaji kepada *student labor*.
4. Terkadang terjadi salah penghitungan jam kerja *student labor*.
5. Terkadang *student labor* lupa dalam mengisi kertas kehadiran.
6. Masih menggunakan microsoft excel dalam memasukkan gaji *student labor*.

### Analisa Kebutuhan Fungsi

Berdasarkan penelitian yang telah dilakukan untuk sistem yang dibuat, maka diperlukan fungsi-fungsi yang mendukung kinerja dari sistem tersebut. Fungsi yang dibutuhkan oleh pengguna, antara lain:

1. Menghemat waktu dalam melaporkan total keseluruhan jam kerja setiap bulan per tanggal 21.
2. Untuk mencatat kehadiran *student*.
3. Untuk melihat *grade*.
4. Untuk menyamaratakan tanggal penerimaan gaji *student labor* pada setiap departemen

### Use Case Diagram



Gambar 2 Use Case Diagram

Gambar 3.2 dapat dilihat bahwa *student labor* hanya dapat melihat tampilan jam kehadiran (hadir terlambat maupun hadir tepat waktu), nim, nama, jam masuk, dan jam keluar dengan cara melakukan TAP KTM pada alat *RFID Reader*.

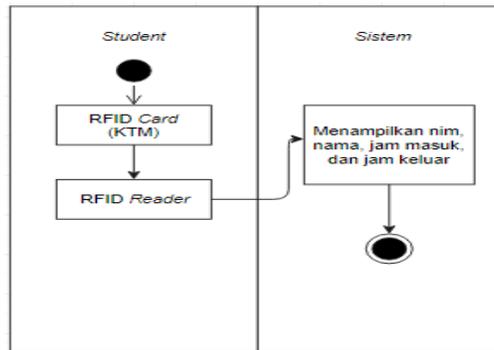
*Admin*, setelah melakukan *login* maka *admin* dapat melihat absen dari *student labor*. Jika *student labor* tersebut belum terdaftar, maka *admin* terlebih dahulu harus mendaftarkan *student* pada form tambah labor dengan melakukan TAP KTM agar dapat memunculkan beberapa form yang harus diisi, seperti nim, nama, dan nomor telepon. Lalu dapat melihat data yang telah dimasukkan pada form lihat *labor*. Apabila telah terdaftar, baru *student labor* dapat melakukan pengabsensian di form *home* dan laporannya dapat dilihat pada form laporan untuk mengetahui *student labor* tersebut telah bekerja berapa jam, *grade*, dan mendapat gaji berapa.

Dan *Business Office* (BO) setelah *login* hanya dapat melihat laporan dari berapa jam *student labor* tersebut bekerja, *grade* yang didapat, dan berapa gaji yang didapat.

### Activity Diagram

#### Activity Diagram Student

Di dalam *activity diagram student* tersebut, maka dapat dilihat bagaimana cara kinerja dari Kartu Tanda Mahasiswa (KTM) terhadap teknologi RFID sehingga *student labor* dapat melihat tampilan dari nim, nama, jam masuk, dan jam keluar.

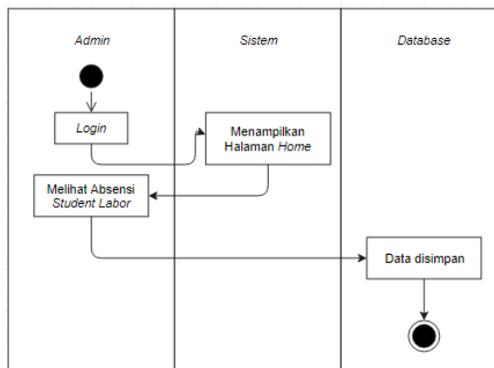


Gambar 3 Activity Diagram Student

Pada Gambar 3, *student labor* harus melakukan tap KTM pada teknologi RFID, lalu *student* dapat menampilkan pada sistem nim, nama, jam masuk, dan jam keluar jika dilakukan TAP KTM dua kali.

#### Activity Diagram Home

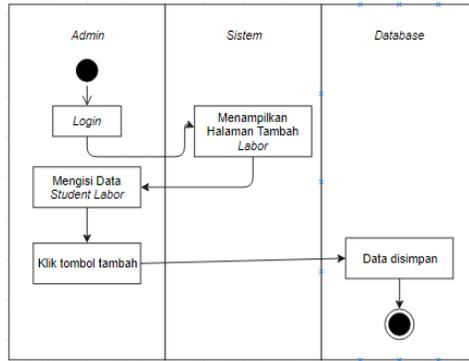
Di dalam *activity diagram home* tersebut, maka dapat dilihat bagaimana cara kinerja sistem *admin*. Pertama yang dilakukan oleh *admin*, ialah harus terlebih dahulu melakukan *login*. Setelah berhasil, maka pada sistem akan muncul halaman *home* yang berfungsi untuk melihat absensi dari *student labor* tersebut. Lalu, data tersebut dengan otomatis akan tersimpan ke dalam *database*.



Gambar 4 Activity Diagram Home

#### Activity Diagram Tambah Labor

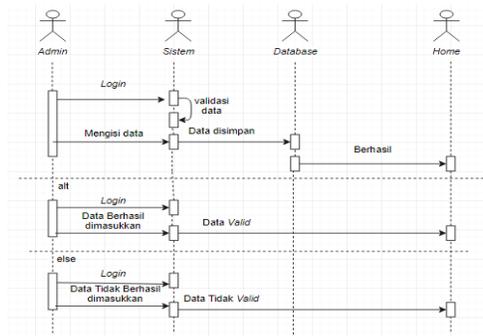
Di dalam *activity diagram tambah labor* tersebut, maka dapat dilihat bagaimana cara kinerja sistem *admin*. Setelah *login*, pihak *admin* dapat melihat halaman tambah *labor* pada sistem, lalu pihak *admin* mengisi data *student labor*, lalu *admin* mengklik tombol tambah dan sistem langsung menambahkan data ke dalam *database*.



Gambar 5 Activity Diagram Tambah Labor

**Sequence Diagram Tambah Labor**

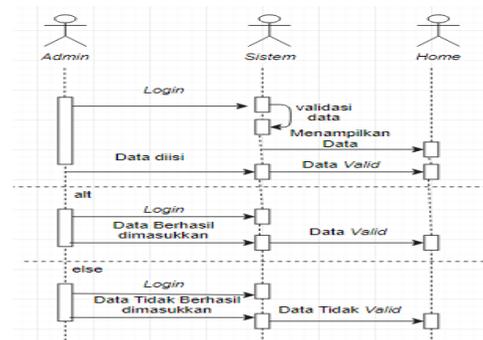
Di dalam *sequence diagram* tambah *labor* tersebut, maka dapat dilihat bagaimana cara kinerja sistem *admin*. Setelah *login*, pihak *admin* dapat melihat halaman tambah *labor* pada sistem, lalu pihak *admin* mengisi data *student labor*, lalu *admin* mengklik tombol tambah dan sistem langsung menambahkan data ke dalam *database*.



Gambar 6 Sequence Diagram Tambah Labor

**Sequence Diagram Lihat Labor**

Pada Gambar 7, *admin* dapat melihat apakah *student labor* tersebut sudah terdaftar atau belum, dan pada halaman itu juga terdapat tombol hapus yang berfungsi untuk menghapus *student labor* apabila sudah tidak bekerja di departemen tersebut. Gambar 3.11 merupakan tampilan dari halaman lihat *labor*.



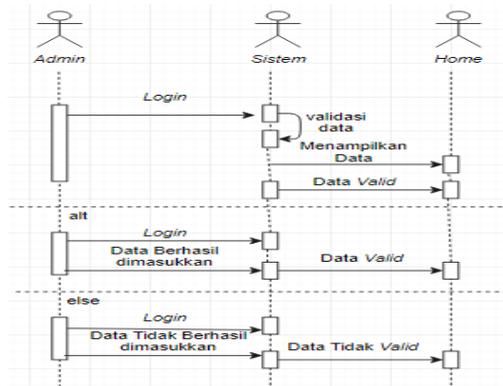
Gambar 7 Sequence Diagram Lihat Labor

**Sequence Diagram Laporan**

Pada Gambar 8 merupakan halaman laporan yang hanya dapat dilihat oleh *admin* dan *staff student account* (BO). Dalam halaman ini, *admin* maupun BO tidak perlu melakukan pengecekan ulang terhadap

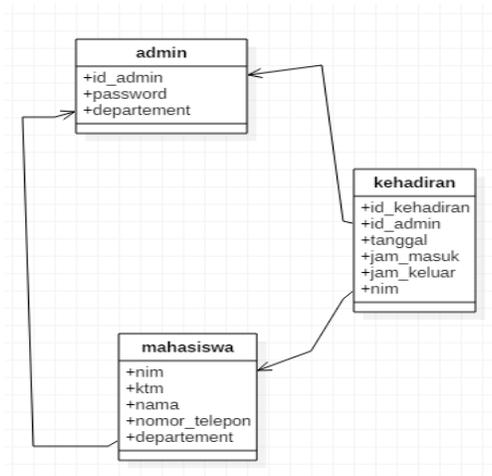
*student labor* karena sudah dipermudah dengan adanya aplikasi penggajian *student labor*. Jika admin maupun BO ingin mendapatkan data yang dibutuhkan, maka dapat memilih tahun berapa dan bulan apa data yang ingin dilihat, setelah itu masukkan nim yang akan dicari dan klik tombol cari.

Pada Gambar 8 ini juga gaji dapat langsung dilihat, jikalau jumlah jam kerja sudah memenuhi 1 jam. Jika belum tepat 1 jam, maka pada kolom jumlah jam akan menampilkan nilai 0. Pada kolom *grade* muncul secara otomatis saat *student* melakukan TAP KTM dua kali waktu melakukan absensi. Dalam kolom *grade* terbagi atas 3 kategori, yaitu (1) *Grade A* = Rp. 5.500,- per jam, (2) *Grade B* = Rp. 5.000,- per jam, dan (3) *Grade C* = Rp. 4.500,- per jam. Gambar 3.12 ini merupakan tampilan dari halaman laporan.



Gambar 8 Sequence Diagram Laporan

## Class Diagram



Gambar 9 Class Diagram

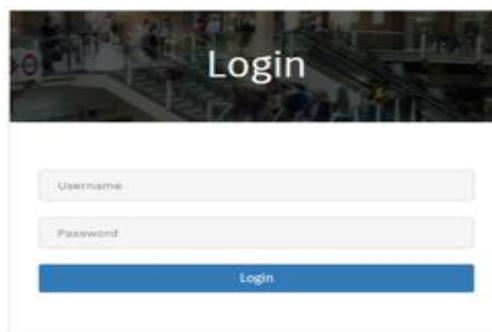
Pada Gambar 9, menjelaskan ada beberapa *class* yang saling terhubung. Di dalam masing-masing *class*, ada terdapat beberapa *field* yang mendukung *class* tersebut. *Class* mahasiswa yang didalamnya terdapat *field* nim yang terhubung dengan *class* kehadiran dan *class* admin dengan *field* departemen. *Class* kehadiran dapat terhubung juga dengan *class* admin melalui *field* id\_admin yang terdapat didalamnya.

## Hasil Dan Perancangan Sistem Perancangan Antar Muka

Tampilan terdiri dari tampilan antar muka admin, *student labor*, dan *student account*. Tampilan admin memiliki tampilan halaman *login* serta tampilan awal halaman meliputi *home*, tambah labor, lihat labor, dan laporan.

### Halaman *Login*

Halaman *login* merupakan batasan bagi mereka yang tidak memiliki hak akses untuk masuk atau mengakses suatu sistem. Untuk dapat mengakses sistem, *admin* dan *staff* BO harus mengetahui *username* dan *password*. Gambar 10 adalah tampilan halaman login.



**Gambar 10** Halaman *Login*

### Halaman *Home*

Jika *admin* telah memiliki hak akses untuk dapat masuk ke dalam sistem penggajian *student labor*, maka admin dapat melihat tampilan yang terdapat di dalam sistem tersebut, yaitu halaman *home*. Sistem ini memiliki cara kerja, yaitu ada 1 pc yang terhubung ke alat RFID, dan ketika *student labor* melakukan TAP pada alat RFID, maka sistem akan langsung menghubungkan absensi tersebut ke Biro Keuangan dan Kepala Departemen yang bersangkutan. Dan apabila *student labor* tersebut ingin mengganti jam kerja karena ada alasan tertentu, maka *student labor* itu harus segera melapor ke Kepala Departemen yang bersangkutan, setelah itu lakukan absensi seperti biasa. Tampilan ini berfungsi untuk melihat apakah *student labor* sudah terdaftar atau tidak di departemen tersebut. Jika belum terdaftar, maka akan muncul tulisan seperti pada Gambar 11.



**Gambar 11** Tampilan *Home*

### Halaman *Tambah Labor*

Halaman *tambah labor* merupakan tempat admin untuk dapat memasukkan atau menambah data *student labor*. Namun, terlebih dahulu admin harus melakukan TAP KTM di RFID *Reader* agar muncul form yang harus diisi seperti nim, nama, dan nomor telepon. Lalu, jika data yang dimasukkan sudah benar, maka admin harus mengklik tombol *tambah* agar data tersebut dapat tersimpan di dalam *database* dan dapat dilihat di dalam halaman *lihat labor*. Gambar 12 merupakan tampilan dari halaman *tambah labor*.

**Gambar 12** Halaman Tambah *Labor*

## Pengujian

Pengujian adalah hal yang perlu dilakukan untuk mengetahui kekurangan atau kesalahan pada perangkat lunak yang diuji. Tujuan pengujian adalah untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut.

### Pengujian *Alpha*

Pengujian *alpha* dilakukan pada sebuah lingkungan yang terkendali, dilakukan pada sisi pengembang yang dianggap pemakai, dan merekam semua kesalahan, serta masalah pemakaian. Pendekatan perancangan yang digunakan untuk uji kasus terhadap aplikasi penggajian *student labor* yaitu menggunakan *Black Box Testing*. *Black Box Testing* merupakan metode pengujian yang berfokus pada persyaratan fungsional perangkat lunak.

**Tabel 1** Rencana pengujian

No.	Komponen Yang Diuji	Detail Pengujian	Jenis Pengujian
1	Login	Verifikasi Login	<i>Black Box</i>
2	Home	TAP IN TAP OUT	<i>Black Box</i> <i>Black Box</i>
3	Tambah Labor	TAP kartu Isi Data <i>Student Labor</i> Tambah Data <i>Student Labor</i>	<i>Black Box</i> <i>Black Box</i> <i>Black Box</i>
4	Lihat Labor	Lihat Data <i>Student Labor</i> Hapus Data <i>Student Labor</i>	<i>Black Box</i> <i>Black Box</i>
5	Laporan	Lihat Data <i>Student Labor</i> Masukkan NIM Mencari Data <i>Student Labor</i>	<i>Black Box</i> <i>Black Box</i> <i>Black Box</i>

### Pengujian *Black Box*

*Black box testing* adalah metode yang digunakan untuk menemukan kesalahan dan mendemonstrasikan fungsional aplikasi saat dioperasikan, apakah *input* diterima dengan benar dan *output* yang dihasilkan telah sesuai dengan yang diharapkan.

Tujuan metode ini adalah mencari kesalahan pada:

1. Fungsi yang salah atau hilang
2. Kesalahan pada struktur data atau akses *database*
3. Kesalahan pada *interface*

4. Kesalahan inisialisasi dan tujuan akhir
5. Kesalahan performansi

Ini merupakan kasus untuk menguji perangkat lunak yang sudah dibangun menggunakan metode *Black Box* berdasarkan Tabel 2 Rencana pengujian yang telah dibuat sebelumnya.

**Tabel 2** Pengujian *Black Box* Aplikasi Penggajian *Student Labor*

Form	Skenario Dan Hasil Uji		
	Skenario Uji	Hasil Yang Diharapkan	Kesimpulan
<i>Login</i>	Melakukan verifikasi	Masuk ke dalam aplikasi dengan hak akses dari yang berhak melakukan <i>login</i>	[ ✓ ] Berhasil [ ] Tidak Berhasil
<i>Login</i>	Form kosong atau tidak diisi semua	Jika form kosong, menampilkan perintah "Harap isi" form. Jika form tidak diisi, menampilkan pesan "Username atau Password Salah"	[ ✓ ] Berhasil [ ] Tidak Berhasil
	Username atau Password salah	Menampilkan pesan "Username atau Password Salah"	[ ✓ ] Berhasil [ ] Tidak Berhasil
<i>Home</i>	Melakukan absensi dengan cara TAP <i>IN</i> dan TAP <i>OUT</i> kartu ke alat RFID <i>Reader</i>	Menampilkan nim, nama, jam masuk, serta jam keluar, jika sudah terdaftar sebagai <i>student labor</i>	[ ✓ ] Berhasil [ ] Tidak Berhasil
<i>Home</i>	Melakukan absensi dengan cara TAP <i>IN</i> dan TAP <i>OUT</i> kartu ke alat RFID <i>Reader</i>	Tidak dapat menampilkan data nim, nama, jam masuk, serta jam keluar, jika belum terdaftar sebagai <i>student labor</i>	[ ✓ ] Berhasil [ ] Tidak Berhasil
Tambah Labor	Melakukan TAP kartu terlebih dahulu, kemudian mengisi form yang telah tersedia untuk dapat menambah data <i>student labor</i>	Menambah data <i>student labor</i> ke dalam <i>database</i> dan menampilkan pesan "Inputan Anda Berhasil"	[ ✓ ] Berhasil [ ] Tidak Berhasil
Tambah Labor	Melakukan TAP kartu, kemudian langsung mengklik tombol tambah tanpa mengisi form terlebih dahulu	Tidak dapat menambah data <i>student labor</i>	[ ✓ ] Berhasil [ ] Tidak Berhasil
Lihat Labor	Menampilkan data (nim, nama, dan <i>contact</i> ) <i>student labor</i>	Melihat seluruh data <i>student labor</i> yang telah di <i>input</i> dan terdapat tombol hapus yang berfungsi untuk menghapus data yang ada	[ ✓ ] Berhasil [ ] Tidak Berhasil
Laporan	Menampilkan data (nim, nama, jumlah jam, grade, dan gaji) <i>student labor</i>	Melihat seluruh data di form laporan dan dapat melakukan pencarian dengan mengklik tombol "Cari" berdasarkan data	[ ✓ ] Berhasil [ ] Tidak Berhasil

		yang terisi di form yang telah disediakan	
RFID	TAP IN kartu	Sistem dapat membaca jam masuk <i>student labor</i>	[ <input checked="" type="checkbox"/> ] Berhasil [ <input type="checkbox"/> ] Tidak Berhasil
	TAP OUT kartu	Sistem dapat membaca jam keluar <i>student labor</i>	[ <input checked="" type="checkbox"/> ] Berhasil [ <input type="checkbox"/> ] Tidak Berhasil

### Kesimpulan Pengujian *Alpha*

Dari hasil pengujian yang dilakukan, maka disimpulkan bahwa sistem sudah berjalan dengan baik. Secara fungsional sistem sudah dapat menghasilkan keluaran yang diharapkan.

### Pengujian *Beta*

Pengujian *beta* merupakan pengujian yang dilakukan secara objektif dengan kata lain pengujian ini ialah pengujian secara langsung dilapangan atau tempat di mana aplikasi ini dibuat.

#### Wawancara

Wawancara dilakukan kepada *student labor* yang bekerja pada setiap departemen. Dari hasil wawancara didapatkan hasil pengoreksian terhadap pengabsenan yang berpengaruh ke penggajian yang telah dibangun, untuk mengetahui tanggapan *student labor* terhadap aplikasi penggajian yang telah dirancang. Setelah wawancara selesai, maka hasil wawancara yang diperoleh adalah sebagai berikut.

**Tabel 3** Skala *Likert*

No.	Keterangan
1	Sangat Setuju
2	Setuju
3	Kurang Setuju
4	Tidak Setuju

Responden yang digunakan sebagai sampel 5 orang. Persentase masing-masing jawaban dicari berdasarkan dari data hasil kuesioner dengan menggunakan rumus kuesioner:

Keterangan:

P : Banyaknya responden dari setiap soal

Q : Jumlah responden

Y : Nilai persentase

Hasil persentase setiap pertanyaan dari hasil kuesioner dapat dilihat sebagai berikut.

1. Apakah anda setuju bahwa aplikasi penggajian *student labor* ini dapat mempermudah cara melakukan absensi?

**Tabel 4** Skala *likert* pertanyaan 1

No.	Keterangan	Responden	Persentase
1	Sangat Setuju	5	100%
2	Setuju	-	-
3	Kurang Setuju	-	-
4	Tidak Setuju	-	-

Di dalam Tabel 4, dapat dilihat hasil dari setiap responden yang memiliki pernyataan sangat setuju, setuju, kurang setuju, dan tidak setuju. Dari pertanyaan nomor 1 dapat dilihat bahwa semua responden sangat setuju dengan adanya aplikasi penggajian *student labor* yang dapat mempermudah dalam melakukan absensi.

2. Apakah anda setuju bahwa tampilan *website* cukup menarik dan mudah dipahami *user friendly*?

**Tabel 5** Skala *likert* pertanyaan 2

No.	Keterangan	Responden	Persentase
1	Sangat Setuju	2	40%
2	Setuju	3	60%
3	Kurang Setuju	-	-
4	Tidak Setuju	-	-

Di dalam Tabel 5, dapat dilihat hasil dari setiap responden yang memiliki pernyataan sangat setuju, setuju, kurang setuju, dan tidak setuju. Dari pertanyaan nomor 2 dapat dilihat bahwa ada 2 responden sangat setuju dengan adanya aplikasi penggajian *student labor* dengan persentase hasil 40% dan ada 3 responden yang setuju dengan persentase hasil 60% bahwa aplikasi ini bersifat *userfriendly*.

3. Apakah anda setuju bahwa informasi mengenai penggajian *student labor* yang diberikan oleh *website* ini sesuai dengan informasi yang anda butuhkan?

**Tabel 6** Skala *likert* pertanyaan 3

No.	Keterangan	Responden	Persentase
1	Sangat Setuju	3	60%
2	Setuju	2	40%
3	Kurang Setuju	-	-
4	Tidak Setuju	-	-

Di dalam Tabel 6, dapat dilihat hasil dari setiap responden yang memiliki pernyataan sangat setuju, setuju, kurang setuju, dan tidak setuju. Dari pertanyaan nomor 3 dapat dilihat bahwa ada 3 responden sangat setuju dengan adanya aplikasi penggajian *student labor* dengan persentase hasil 60% dan ada 2 responden yang setuju dengan persentase hasil 40% bahwa aplikasi ini memberikan informasi yang sesuai dengan kebutuhan *user*.

4. Apakah anda setuju dengan adanya *website* ini proses absensi dan penggajian menjadi lebih mudah?

**Tabel 7** Skala *likert* pertanyaan 4

No.	Keterangan	Responden	Persentase
1	Sangat Setuju	5	100%
2	Setuju	-	-
3	Kurang Setuju	-	-
4	Tidak Setuju	-	-

Di dalam Tabel 7, dapat dilihat hasil dari setiap responden yang memiliki pernyataan sangat setuju, setuju, kurang setuju, dan tidak setuju. Dari pertanyaan nomor 4 dapat dilihat bahwa semua responden

sangat setuju dengan adanya aplikasi penggajian *student labor* yang dapat mempermudah dalam melakukan absensi dan proses penggajian.

### **Kesimpulan Pengujian Beta**

Setelah melalui beberapa wawancara tahapan dalam pengujian *beta*, maka dapat disimpulkan bahwa:

1. Aplikasi penggajian *student labor* memudahkan *user* dalam mendapatkan informasi tentang gaji.
2. *User* dapat melakukan absensi dengan cepat tanpa harus menulis di kertas kehadiran.
3. Aplikasi ini juga dapat memudahkan kepala departemen dan biro keuangan dalam pengecekan data *student labor*.

## **Kesimpulan Dan Saran**

### **Kesimpulan**

Setelah melakukan penelitian, maka dapat diambil kesimpulan sebagai berikut:

1. Dengan adanya aplikasi penggajian ini, dapat memberikan kemudahan kepada pihak *Business Office* dalam memasukkan gaji ke *account* setiap *student labor*, dengan cara memasukkan nim *student labor*, maka data yang diinginkan akan muncul, lalu pihak BO melakukan pengecekan terhadap data, jika benar data langsung disimpan dan langsung masuk ke *account* masing-masing.
2. Aplikasi penggajian *student labor* ini, cukup memberikan kemudahan dalam penggunaannya saat melakukan absensi dengan cara *tap in* untuk mengetahui jam masuk dari *student labor* dan *tap out* untuk mengetahui jam berapa *student labor* tersebut selesai bekerja.

### **Saran**

Berdasarkan perancangan aplikasi penggajian *student labor*. Maka diberikan saran untuk pengembangan lebih lanjut mengenai aplikasi ini, yaitu:

1. Pengembangan terhadap aplikasi penggajian *student labor* agar dibuat lebih lengkap lagi supaya aplikasi dapat digunakan lebih efisien dan lebih efektif lagi, dengan menambah ide-ide guna untuk menyempurnakan aplikasi yang telah ada, agar dapat diimplementasikan untuk kemajuan Universitas Advent Indonesia dalam hal teknologi.

## **Referensi**

1. Agus Mulyanto. (2009). Sistem Informasi Konsep dan Aplikasi, 12. 18 September 2017
2. Candra, A. H., Town, U. C., & Timur, J. (n.d.). Rancang Bangun Human Resource Management Untuk Perusahaan Skala Besar, 90–98. 01 November 2017
3. Frequency, R., Rfid, I., & Terbuka, L. (n.d.). Efektifitas Radio Frequency Identification ( Rfid ) Di Kelompok Layanan Terbuka Perpustakaan Nasional Republik Indonesia Oleh : Muhammad Jevi Rian Aipasha \* Pembimbing : Dra . Tri Wahyu Hari Murtiningsih , M . Si Program Studi Ilmu Perpustakaan , Fakultas. 18 Desember 2017
4. Hidayat, R. (2010). Untuk Perpustakaan Polnes: Suatu Peluang Rahmad Hidayat. *Informatika Mulawarman*, 5(1), 42–49. Diambil dari <https://informatikamulawarman.files.wordpress.com/2010/02/07-jurnal-ilkom-unmul-v-5-1-0.pdf>. 24 Oktober 2017
5. Ii, B. A. B., & Pustaka, T. (2012). No Title, 6–18.(Julia & Sitorus, 2013) 06 Maret 2018
6. Julia, C., & Sitorus, P. M. (2013). Pengukuran Benefit Teknologi Investasi Teknologi Informasi Menggunakan Metode Information Economics (Studi Kasus di PT. Telekomunikasi Indonesia, Tbk.). *Jurnal Manajemen Indonesia*. 01 November 2017
7. Lpkia, S. (n.d.). Sistem absensi pegawai menggunakan teknologi rfid, 1–6. 01 November 2017
8. Rfid, M. (2016). No Title, 5, 126–134. 24 Oktober 2017
9. Rismawati, N. (2016). Sistem Absensi Dosen Menggunakan Near Field Communication ( Nfc ) Technology. *Faktor Exacta*, 9(2), 135–142. 18 September 2017
10. Saputra, F. H. (2008). Sistem absensi menggunakan teknologi rfid, 1–82.(Rismawati, 2016). 24 Oktober 2017
11. Soediono, B., Mustofa, A., Informatika, T., & Nuswantoro, U. D. (2014). Analisis Dan Perancangan Sistem Informasi E-Education Berbasis Web Di Sma Pembangunan Mranggen. *Journal of Chemical*

*Information and Modeling*, 53(2004), 1–6. <https://doi.org/10.1017/CBO9781107415324.004> . 24  
[Oktober 2017](#)

12. Yuliyanti, W. (2016). Perencanaan dan Pembuatan Aplikasi Absensi Dosen Menggunakan Radio Frequency Identification ( RFID ) 8[ Studi Kasus Politeknik Negeri Tanah Laut ]. 30 Oktober 2017