BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1. Imlementasi

Implementasi adalah suatu kelanjutan dari Aplikasi Pembantu dan Pendukung Kinerja Karyawan di Mitra Dinamika Konsultan yang dilakukan dengan menggunakan bahasa pemograman Java dan basis data yang digunakan adalah MySQL. Aplikasi ini dapat dijalankan pada sistem operasi Windows dan Linux, implementasi dan pengujian sepenuhnya dilakukan di perangkat keras komputer dengan sistem operasi Microsoft Windows 8.

5.1.1. Batasan Implementasi

Dalam pengimplementasian aplikasi ini ada beberapa hal yang menjadi batasan dari implementasi, hal ini bertujuan untuk membatasi ruang lingkup pemahaman. Adapun batasan-batasan dari implementasi tersebut adalah sebagai berikut :

1. Manajemen hak akses sesuai dengan deskripsi pekerjaan setiap bagian.
2. Basis data yang digunakan dalam pengimplementasian ini adalah MySQL dan basis data tersebut digunakan secara terpusat.
3. Antar muka untuk pengguna menggunakan bahasa Indonesia dan bahasa inggris.

5.1.2. Implementasi Perangkat Lunak

Adapun aplikasi, database dan sistem operasi yang diperlukan dalam implementasi perangkat lunak adalah sebagai berikut :

1. Java Netbeans 7.3

Untuk membuat Aplikasi Pembantu dan Pendukung Kinerja Karyawan di Mitra Dinamika Konsultan ini digunakan bahasa pemrograman java Netbenas 7.3

2. Database

Database yang digunakan adalah MySQL (5.0.67) untuk menyimpan sumber data Aplikasi.

3. Windows 8

Windows 8 sebagai Sistem Operasi untuk menginstall aplikasi ini.

4. JasperReport iReport 5.1.0

JasperReport iReport 5.1.0 digunakan untuk membuat output laporan

5. *Server Glassfish*

Server ini yang akan diperlukan untuk menjalankan program aplikasi ini.

6. Adobe *Reader* 9.0

Digunakan untuk membuka unduhan laporan yang degenerate oleh Program Aplikasi.

5.1.3. Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan *minimum requirement* yang harus dipenuhi dalam penerapannya adalah sebagai berikut :

1. Minimum requirement *hardware* komputer *server* :
   a. *Processor* : Intel Pentium 4 (2,6 Ghz) atau yang sekelasnya
   b. *Memory* : 1 GB
   c. VGA : 256 MB

d. *Hardisk* : 120 GB

e. CD-ROM *Drive*

f. *Monitor* & *Printer* standard

g. *Mouse* & *Keyboard* standard

2. *Minimum requirement* hardware komputer *client* :

a. *Processor* 1,5 GHz

b. 512 MB RAM

c. VGA 256 MB

d. *Harddisk* 40 GB

e. *Monitor*

f. *Keyboard*

g. *Mouse*

5.1.4. Implementasi Basis Data

Implementasi basis data dilakukan dengan menggunakan bahasa SQL, dimana aplikasi pemrograman yang digunakan adalah MySQL, implementasi basis datanya dalam bahasa SQL adalah sebagai berikut :

1. Tabel Pengguna

```
CREATE TABLE usr_tp (
  `user_id` int(10) NOT NULL AUTO_INCREMENT,
  `username` varchar(10) DEFAULT ' ' NOT NULL,
  `first_name` varchar(20) DEFAULT ' ' NOT NULL,
  `middle_name` varchar(20) DEFAULT ' ' NOT NULL,
  `last_name` varchar(20) DEFAULT ' ' NOT NULL,
```

```
  `password` char(32) DEFAULT ' ' NOT NULL,
  `user_employee_tc_id` int(10) DEFAULT 0 NOT NULL,
  PRIMARY KEY (`user_id`)
);
```

2. Tabel Grup Karyawan

```
CREATE TABLE employee_tp (
  `employee_tp_id` int(10) NOT NULL AUTO_INCREMENT,
  `employee_tp_key` varchar(5) DEFAULT ' ' NOT NULL,
  `employee_tp_name` varchar(50) DEFAULT ' ' NOT NULL,
  `employee_tp_desc` varchar(100) DEFAULT ' ' NOT NULL,
  PRIMARY KEY (`employee_tp_id`)
);
```

3. Tabel Karyawan

```
CREATE TABLE employee_tc (
  `employee_tc_id` int(10) NOT NULL AUTO_INCREMENT,
  `employee_tc_key` varchar(10) DEFAULT ' ' NOT NULL,
  `employee_tc_first_name` varchar(20) DEFAULT ' ' NOT NULL,
  `employee_tc_middle_name` varchar(20) DEFAULT ' ' NOT NULL,
  `employee_tc_last_name` varchar(20) DEFAULT ' ' NOT NULL,
  `employee_tc_address` varchar(100) DEFAULT ' ' NOT NULL,
  `employee_tc_phone` varchar(15) DEFAULT ' ' NOT NULL,
  `employee_tc_mobile` varchar(15) DEFAULT ' ' NOT NULL,
  `employee_tc_email` varchar(40) DEFAULT ' ' NOT NULL,
```

```
 `employee_tc_dob` varchar(20) DEFAULT ' ' NOT NULL,
 `employee_tc_id_no` varchar(10) DEFAULT ' ' NOT NULL,
 `employee_tc_employee_tp_id` int(10)  NOT NULL,
 `employee_tc_user_id` int(10)  NOT NULL,
 PRIMARY KEY (`employee_tc_id`)
);
```

4. Tabel Klien

```
CREATE TABLE client_tp (
 `client_tp_id` int(10) NOT NULL AUTO_INCREMENT,
 `client_tp_key` varchar(10) DEFAULT ' ' NOT NULL,
 `client_tp_first_name` varchar(20) DEFAULT ' ' NOT NULL,
 `client_tp_middle_name` varchar(20) DEFAULT ' ' NOT NULL,
 `client_tp_last_name` varchar(20) DEFAULT ' ' NOT NULL,
 `client_tp_address` varchar(100) DEFAULT ' ' NOT NULL,
 `client_tp_phone` varchar(15) DEFAULT ' ' NOT NULL,
 `client_tp_mobile` varchar(15) DEFAULT ' ' NOT NULL,
 `client_tp_email` varchar(40) DEFAULT ' ' NOT NULL,
 PRIMARY KEY (`client_tp_id`)
);
```

5. Tabel Kehadiran

```
DROP TABLE IF EXISTS attendance_tp;
CREATE TABLE attendance_tp (
 `attendance_tp_id` int(10) NOT NULL AUTO_INCREMENT,
```

```
 `attendance_tp_employee_tc_id` int(10) NOT NULL,
 `attendance_tp_absent` varchar(10) DEFAULT ' ' NOT NULL,
 `attendance_tp_date` varchar(25) DEFAULT ' ' NOT NULL,
 `attendance_tp_desc` varchar(100) DEFAULT ' ' NOT NULL,
 PRIMARY KEY (`attendance_tp_id`)
);
```

6. Tabel Informasi Klien

```
CREATE TABLE client_care_tp (
 `client_care_tp_id` int(10) NOT NULL AUTO_INCREMENT,
 `client_care_tp_client_tp_id` int(10) NOT NULL,
 `client_care_tp_desc` varchar(250) DEFAULT ' ' NOT NULL,
 `client_care_tp_notes` varchar(250) DEFAULT ' ' NOT NULL,
 `client_care_tp_percent` int(20) DEFAULT 0 NOT NULL,
 `client_care_tp_date_begin` varchar(25) DEFAULT ' ' NOT NULL,
 `client_care_tp_date_end` varchar(25) DEFAULT ' ' NOT NULL,
 PRIMARY KEY (`client_care_tp_id`)
);
```

7. Tabel Pengolahan Modul

```
CREATE TABLE module_tp (
 `module_tp_id` int(10) NOT NULL AUTO_INCREMENT,
 `module_tp_employee_tc_id` int(10) NOT NULL,
 `module_tp_name` varchar(50) DEFAULT ' ' NOT NULL,
 `module_tp_date` varchar(25) DEFAULT ' ' NOT NULL,
```

```
  `module_tp_desc` varchar(100) DEFAULT ' ' NOT NULL,
  `module_tp_file_name` varchar(200) DEFAULT ' ' NOT NULL,
  PRIMARY KEY (`module_tp_id`)
);
```

8. Tabel Pengolahan Upah lembur

```
CREATE TABLE overtime_tp (
  `overtime_tp_id` int(10) NOT NULL AUTO_INCREMENT,
  `overtime_tp_employee_tc_id` int(10) NOT NULL,
  `overtime_tp_date` varchar(25) DEFAULT ' ' NOT NULL,
  `overtime_tp_criteria` varchar(50) DEFAULT ' ' NOT NULL,
  `overtime_tp_overtime_fee` int(10) NOT NULL DEFAULT 0 ,
  `overtime_tp_desc` varchar(100) DEFAULT ' ' NOT NULL,
  PRIMARY KEY (`overtime_tp_id`)
);
```

9. Tabel Hak Akses Pengguna

```
DROP TABLE IF EXISTS accessright_tp;
CREATE TABLE `accessright_tp` (
  `accs_tp_id` int(10) NOT NULL AUTO_INCREMENT,
  `accessright_key` varchar(10) NOT NULL DEFAULT ' ',
  `accessright_desc` varchar(40) NOT NULL DEFAULT ' ',
  PRIMARY KEY (`accs_tp_id`)
);
```

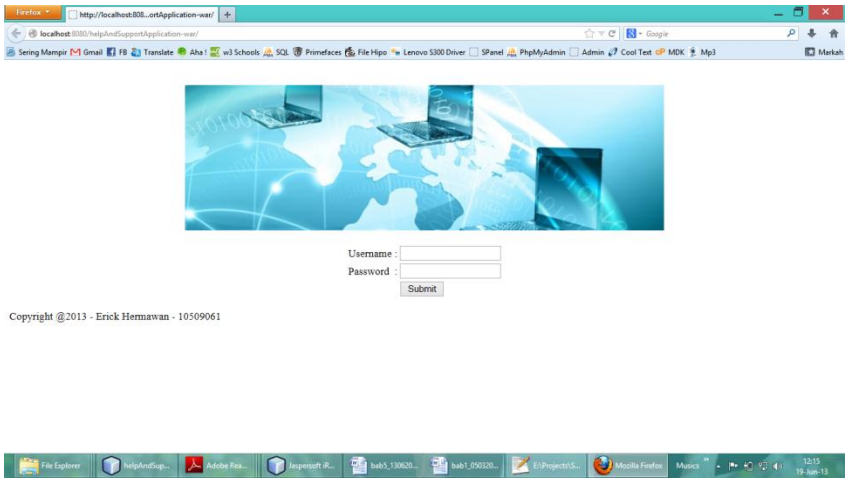
10. Tabel Join Hak Akses Pengguna dengan Pengguna

```
DROP TABLE IF EXISTS usr_accessright_tj;
CREATE TABLE usr_accessright_tj (
 `accs_id` int(10) NOT NULL AUTO_INCREMENT,
 `username` varchar(10) DEFAULT ' ' NOT NULL,
 `accessright_tj_id` varchar(10) DEFAULT ' ' NOT NULL,
 PRIMARY KEY (`accs_id`)
);
```

5.1.5. Implementasi Antar Muka

Implementasi dari proses perancangan antar muka, merupakan gambaran mengenai antarmuka perangkat lunak yang dibuat. Berikut ini merupakan implementasi antar muka dari perangkat lunak yang dirancang :

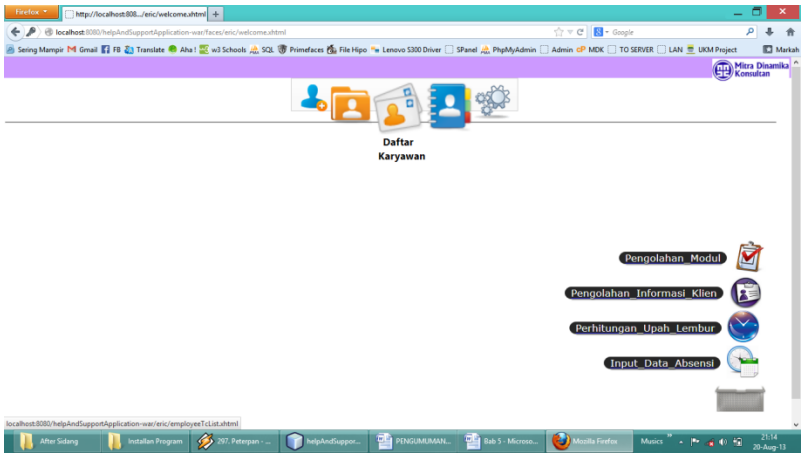
1. Tampilan Halaman Login



Gambar 5.1. Tampilan Halaman Login

Tampilan ini merupakan tampilan yang pertama kali keluar pada saat program dijalankan di *browser*. *User* dapat memasukkan *username* dan *password* untuk masuk kedalam Program Aplikasi ini.

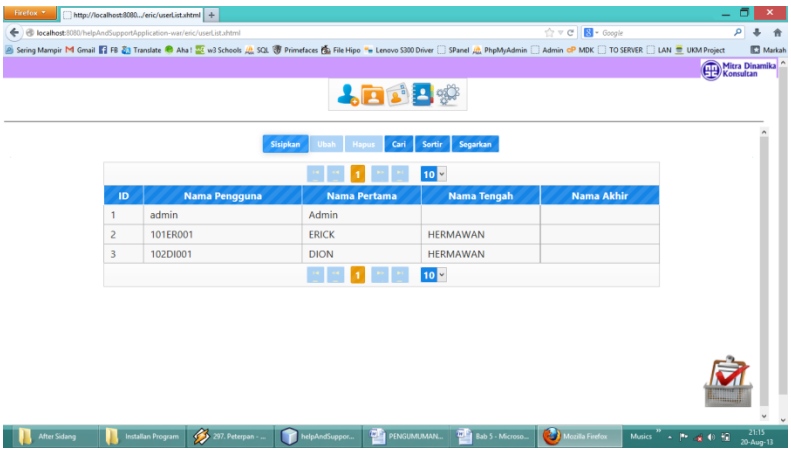
2. Tampilan Halaman Utama



Gambar 5.2. Tampilan Halaman Utama

Tampilan ini merupakan tampilan yang akan muncul ketika *user* berhasil memasukkan *username* dan passwordnya. Dimana di halaman utama ini, *user* dapat memilih menu yang diinginkan.

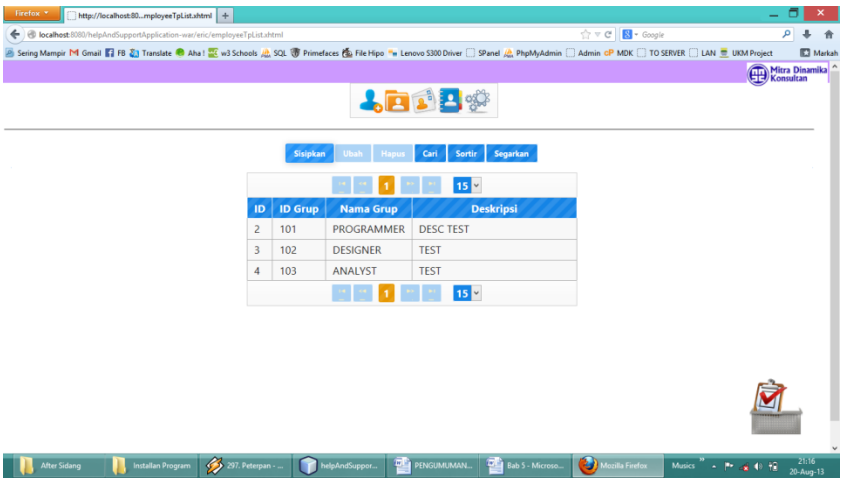
3. Tampilan Halaman Data Pengguna



Gambar 5.3. Tampilan Halaman Data Pengguna

Pada halaman ini, *user* dapat menyimpan, mengubah, menghapus, mencari dan menyortir data pengguna.

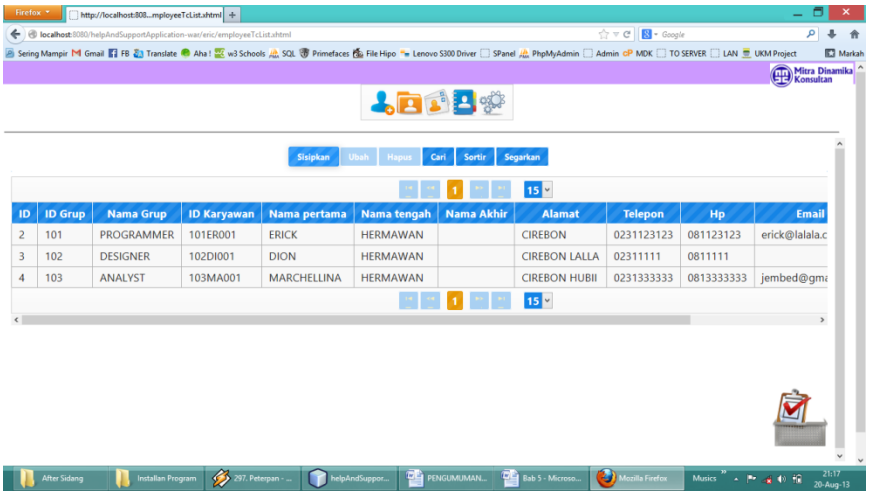
4. Tampilan Halaman Data Grup Karyawan



Gambar 5.4. Tampilan Halaman Data Grup Karyawan

Pada halaman ini, *user* dapat menyimpan, mengubah, menghapus, mencari dan menyortir data grup karyawan.

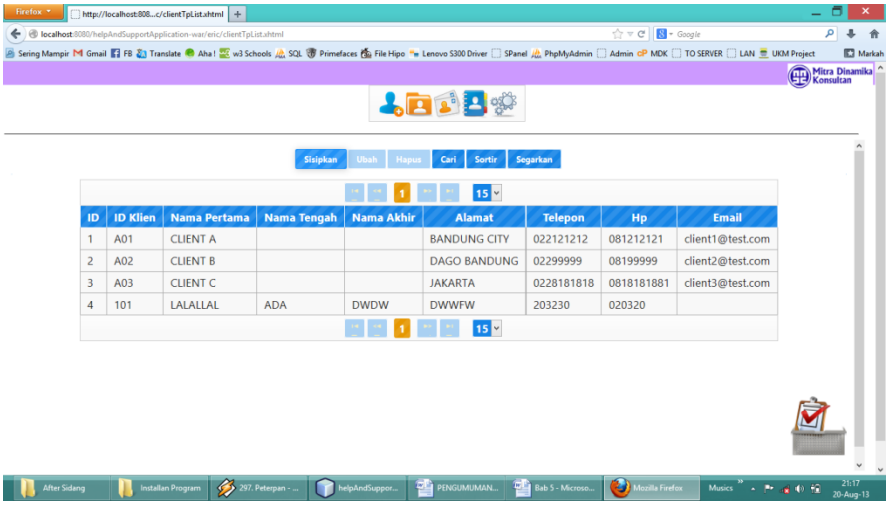
5. Tampilan Halaman Data Karyawan



Gambar 5.5. Tampilan Halaman Data Karyawan

Pada halaman ini, *user* dapat menyimpan, mengubah, menghapus, mencari dan menyortir data karyawan.

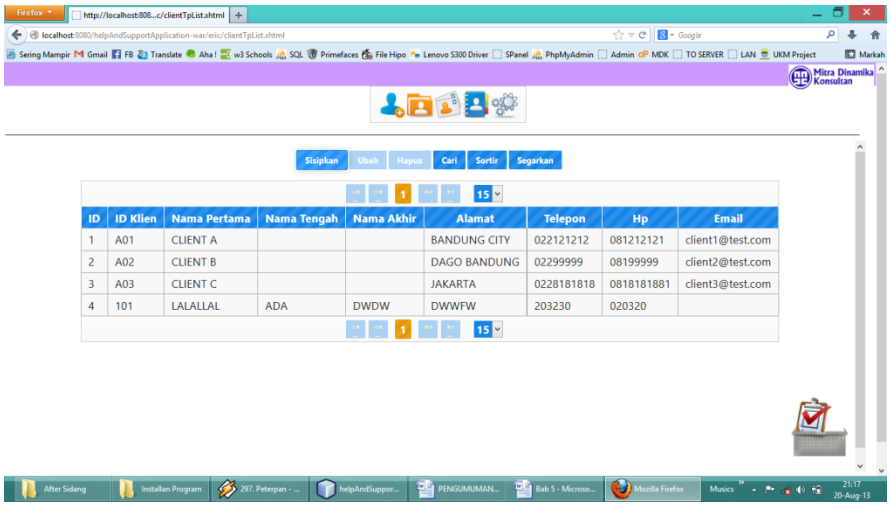
6. Tampilan Halaman Data Klien



Gambar 5.6. Tampilan Halaman Data Klien

Pada halaman ini, user dapat menyimpan, mengubah, menghapus, mencari dan menyortir data klien.

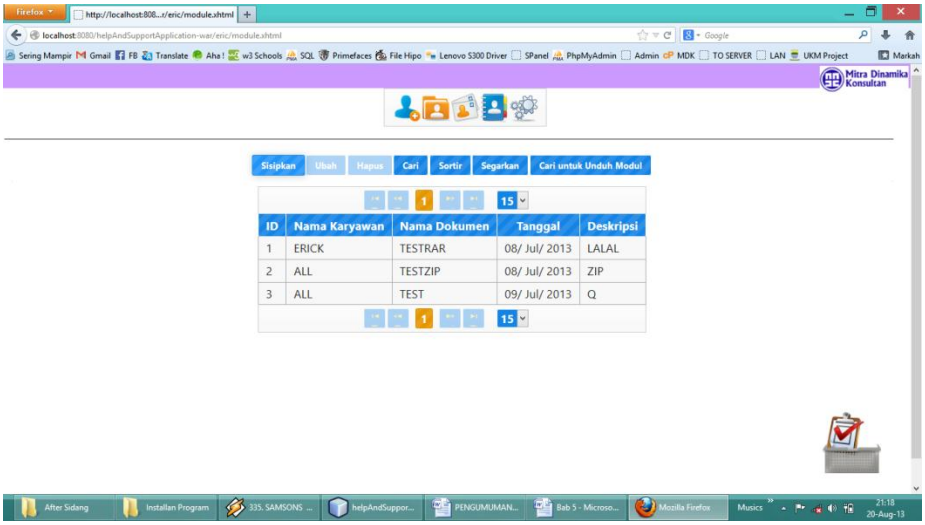
7. Tampilan Halaman Hak Akses Pengguna



Gambar 5.7. Tampilan Halaman Hak Akses Pengguna

Pada halaman ini, *user* dapat menyimpan, mengubah, menghapus dan mencari hak akses *user*.

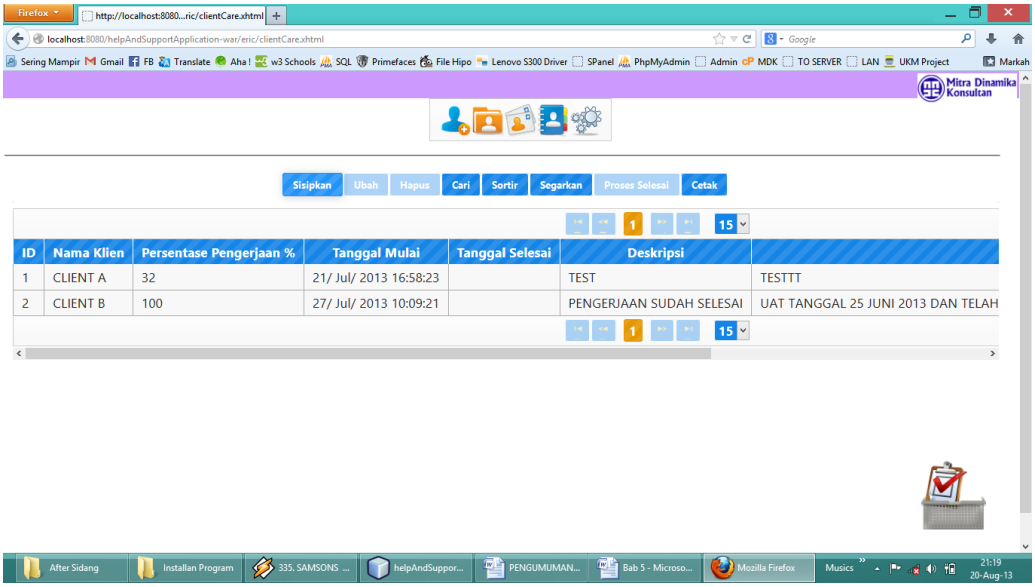
8. Tampilan Halaman Pengolahan Modul Karyawan



Gambar 5.8. Tampilan Halaman Pengolahan Modul Karyawan

Pada halaman ini, *user* (*Project Leader*) dapat menyimpan, mengubah, menghapus, mencari dan menyortir data module serta dapat menupload modul. Sedangkan *user* (karyawan) dapat mencari modul yang ditujukan kepadanya dan mendownloadnya.

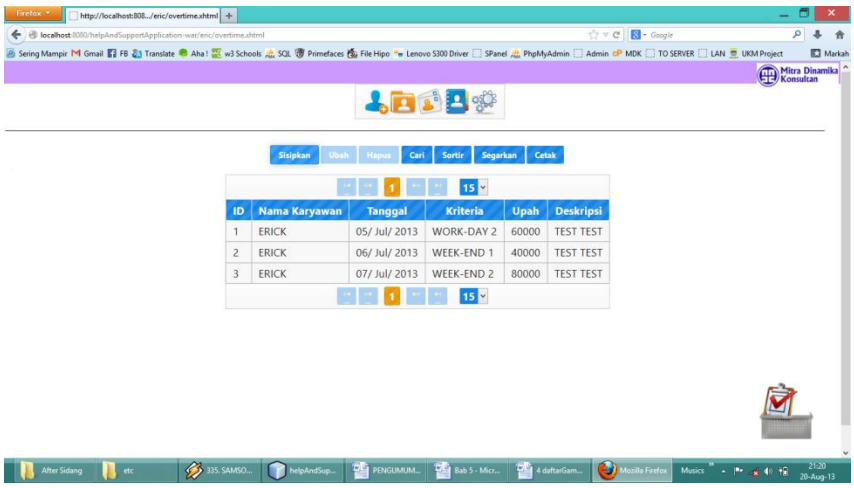
9. Tampilan Halaman Pengolahan Informasi Klien



Gambar 5.9. Tampilan Halaman Pengolahan Informasi Klien

Pada halaman ini, *user* dapat menyimpan, mengubah, menghapus, mencari dan menyortir data informasi klien serta dapat mendownload laporan yang dihasikan dari proses ini.

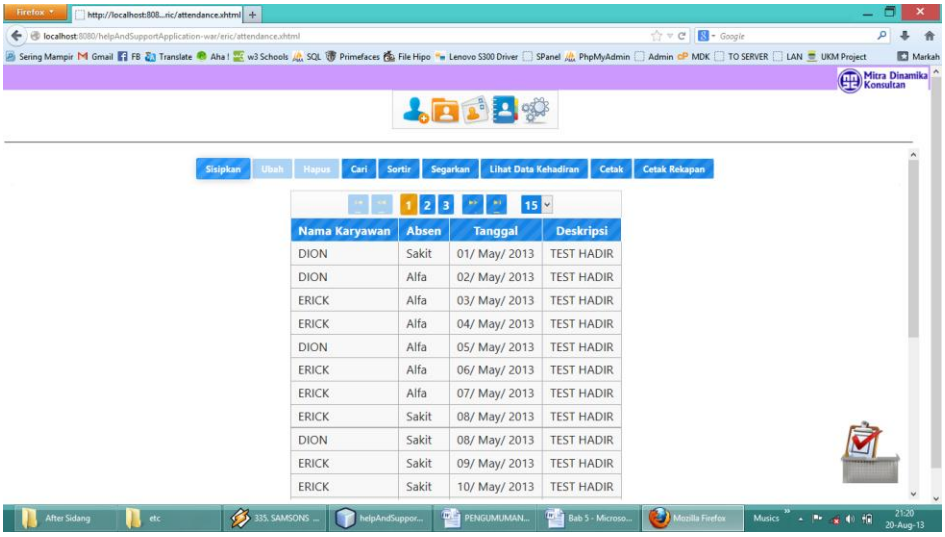
10. Tampilan Halaman Perhitungan Upah Lembur



Gambar 5.10. Tampilan Halaman Perhitungan Upah Lembur

Pada halaman ini, *user* dapat menyimpan, mengubah, menghapus, mencari dan menyortir data perhitungan upah lembur serta dapat mendownload laporan yang dihasikan dari proses ini.

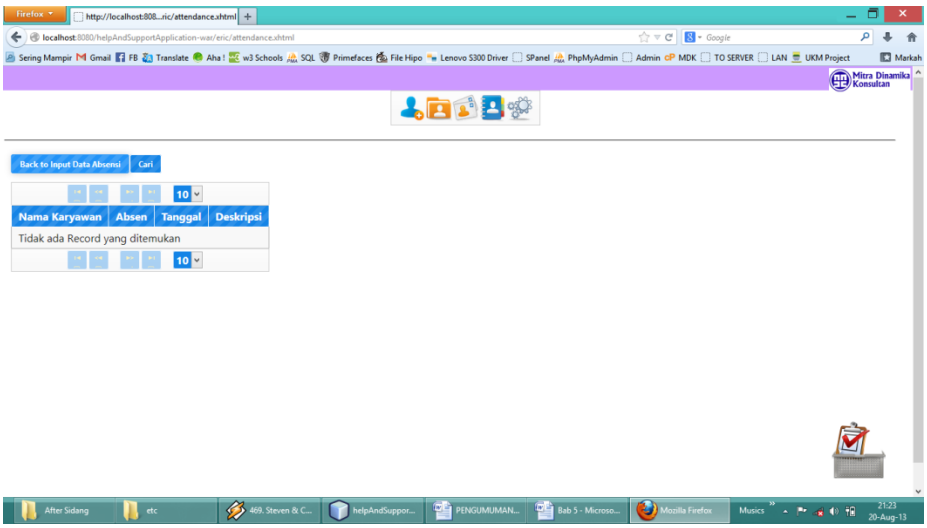
11. Tampilan Halaman Pengolahan Absensi Karyawan



Gambar 5.11. Tampilan Halaman Pengolahan Absensi Karyawan

Pada halaman ini, *user* dapat menyimpan, mengubah, menghapus, mencari dan menyortir data pengolahan absensi karyawan serta dapat mendownload laporan yang dihasikan dari proses ini. Dan *user* dapat langsung melihat rekapan absensinya dengan mengklik tombol *Attendance View*. Maka aplikasi tersebut akan pindah halaman ke halaman rekapan absensi karyawan

12. Tampilan Halaman Rekapan Absensi Karyawan

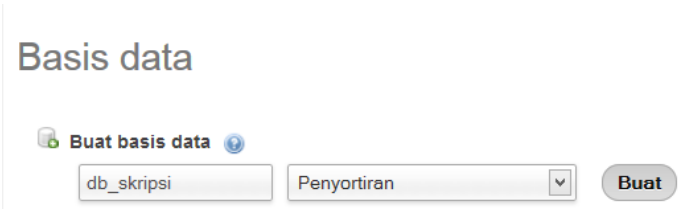


Gambar 5.12. Tampilan Halaman Rekapan Absensi Karyawan

Pada halaman ini, *user* dapat mencari rekapan absensinya, dengan mengklik tombol *search*.

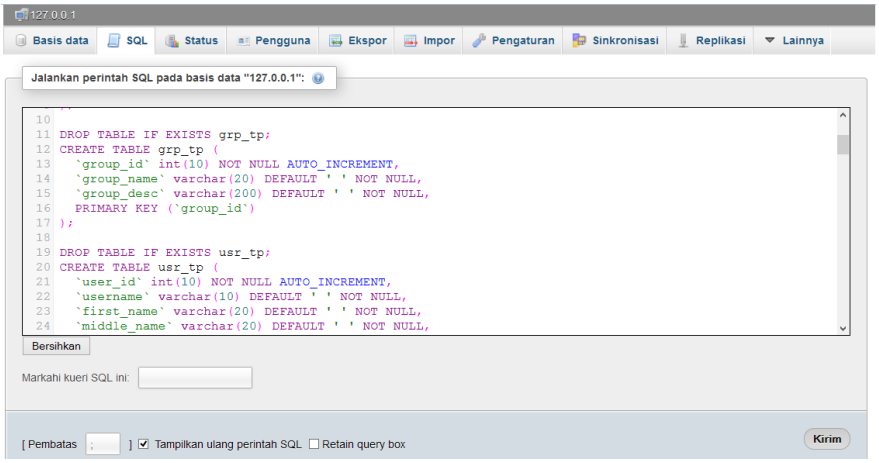
5.1.6. Implementasi Instalasi Program

1. Buat dahulu databasenya



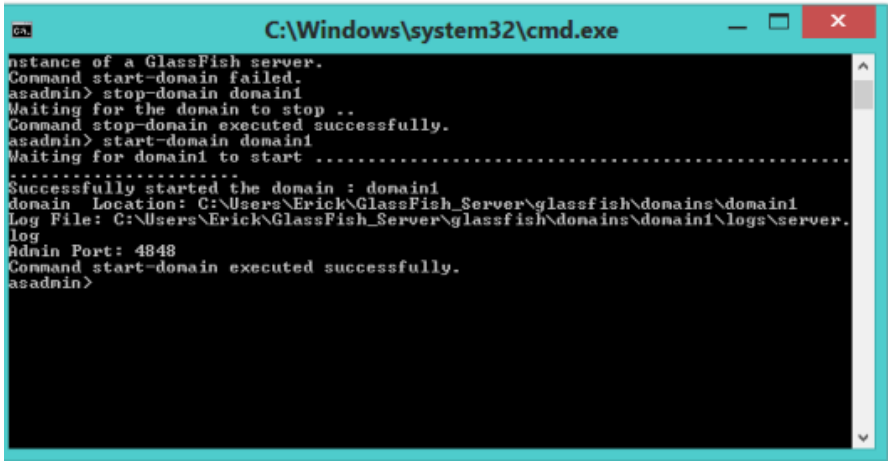
Gambar 5.13. Cara Membuat Database

2. Jalankan *script* sql yang ada di Implementasi Basis Data diatas



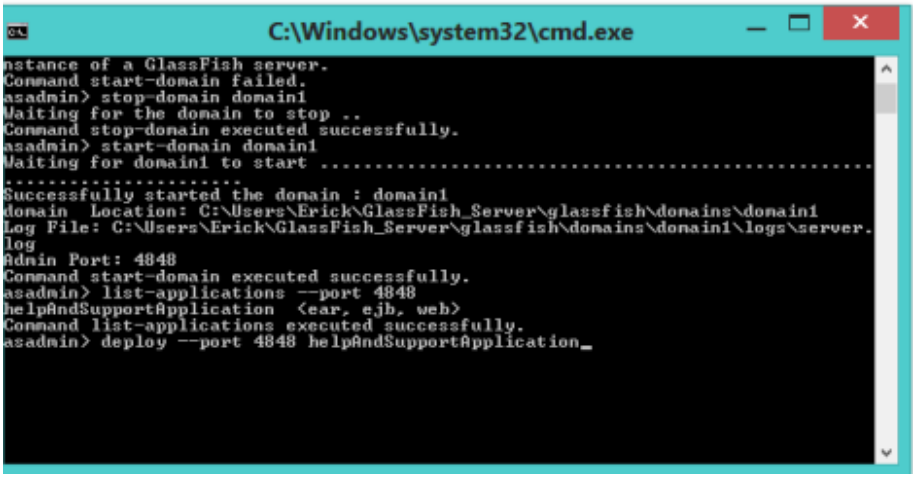
Gambar 5.14. Cara Menjalankan *Script* SQL

3. *Start domain* Glassfishnya dengan cara masukkan *script* seperti terlihat di gambar 5.16. dibawah ini.



Gambar 5.15. Cara *Start Server* Glassfish

4. *Deploy* aplikasinya, setelah sudah ada di list aplikasi yang tersedia, maka aplikasi tersebut sudah bisa dijalankan di browser.



Gambar 5.16. Cara *Deploy* Aplikasi

5. Aplikasi sudah dapat dijalankan, dengan *port deploy* yang tadi.



Gambar 5.17. Aplikasi Berjalan di Browser

5.2. Pengujian

Pengujian diperlukan sebagai salah satu tahapan untuk menguji tingkat minimal kesalahan dan keakuratan perangkat lunak yang dirancang. Pengujian

dilakukan dengan metode black box. Metode pengujian *black box* dipilih karena metode pengujian tersebut karena tidak memperhatikan struktur logika internal (*coding*) dalam perangkat lunak.

5.2.1. Rencana Pengujian

Dalam metode pengujian *black box*, terdapat faktor-faktor pengujian untuk mengetahui tingkat kelayakan penggunaan perangkat lunak. Rencana pengujian untuk perangkat lunak yang dirancang akan dilakukan dengan menggunakan lima faktor pengujian. Lima faktor pengujian tersebut adalah :

1. *File Integrity*

Yaitu pengujian menekankan pada data yang dimasukan melalui akan tidak bisa diubah. Prosedur yang akan memastikan bahwa *file* yang digunakan benar dan data dalam *file* tersebut akan disimpan sekuensial dan benar.

2. *Access control*

Yaitu pengujian menekankan pada sumberdaya sistem yang harus dilindungi dari kemungkinan modifikasi, pengrusakan, penyalahgunaan dan prosedur keamanan harus dijalankan secara penuh untuk menjamin integritas data dan program aplikasi.

3. *Authorization*

Yaitu pengujian yang dilakukan untuk menjamin data yang diproses sesuai dengan ketentuan manajemen. *Authorisasi* menyangkut proses transaksi secara umum dan khusus.

4. *Correctness*

Yaitu menjamin pada data yang dimasukan, proses dan *output* yang dihasilkan dari aplikasi harus akurat dan lengkap. Kelengkapan dan akurasi akan dicapai melalui kontrol transaksi dan elemen data.

5. *Ease of use*

Yaitu menekankan pada perluasan usaha yang diminta untuk belajar, mengoperasikan dan menyiapkan inputan, dan menginterprestasikan output dari sistem terhadap interaksi antara manusia dan sistem.

5.2.2. Kasus dan Hasil Pengujian

Setelah melaksanakan rencana pengujian maka dilanjutkan dengan kasus dan hasil pengujian. Berikut kasus dan hasil pengujian :

5.2.2.1. Pengujian *Login*

Berikut ini kasus dan hasil uji pengujian *login* :

Tabel 5.1.

Pengujian *Form Login*

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username = *admin* *Password* = 123	*Login* berhasil dan dapat masuk ke halaman utama	Dapat mengisi *login* sesuai yang diharapkan	[ ] Diterima [ ] Ditolak
Klik *Submit*	Dapat masuk ke halaman utama	Tombol *Submit* dapat berfungsi sesuai yang diharapkan	[ ] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Username = (tidak sesuai dengan *username user*) *Password* = (tidak sesuai dengan *password user*)	Tidak dapat *login*	*Login* gagal	[ ] Diterima [ ] Ditolak

5.2.2.2. Pengujian Pengisian *Data*

Pada pengujian ini, pengujian pengisian data dibagi menjadi 4, yaitu data *user*, data grup karyawan, data karyawan dan data klien.

1. Pengisian *Data User*

Berikut ini kasus dan hasil uji pengujian *data user* :

Tabel 5.2.

Pengujian pengisian *data user*

Kasus dan Hasil Uji			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol *Insert*	Memulai proses tambah data	Mengeluarkan form inputan	[ ] Diterima [ ] Ditolak
Klik tombol *Update*	Memulai proses ubah data	Mengeluarkan form *update* yang berupa data yang ingin diupdate	[ ] Diterima [ ] Ditolak
Klik tombol *Delete*	Memulai proses hapus data	Mengeluarkan konfirmasi untuk menghapus data	[ ] Diterima [ ] Ditolak
Klik tombol *Search*	Memulai proses mencari data	Mengeluarkan inputan data *username* yang ingin dicari	[ ] Diterima [ ] Ditolak
Klik tombol *Sort*	Memulai proses menyortir data	Mengeluarkan pilihan data yang ingin disortir	[ ] Diterima [ ] Ditolak
Klik tombol *Refresh*	Memulai proses segarkan data	Merefresh form	[ ] Diterima [ ] Ditolak

2. Pengisian *Data* Grup Karyawan

Berikut ini kasus dan hasil uji pengujian *data* grup karyawan :

Tabel 5.3.

Pengujian pengisian *data* grup karyawan

Kasus dan Hasil Uji			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol *Insert*	Memulai proses tambah data	Mengeluarkan form inputan	[ ] Diterima [ ] Ditolak
Klik tombol *Update*	Memulai proses ubah data	Mengeluarkan form *update* yang berupa data yang ingin diupdate	[ ] Diterima [ ] Ditolak
Klik tombol *Delete*	Memulai proses hapus data	Mengeluarkan konfirmasi untuk menghapus data	[ ] Diterima [ ] Ditolak
Klik tombol *Search*	Memulai proses mencari data	Mengeluarkan inputan data *group name* yang ingin dicari	[ ] Diterima [ ] Ditolak
Klik tombol *Sort*	Memulai proses menyortir data	Mengeluarkan pilihan data yang ingin disortir	[ ] Diterima [ ] Ditolak
Klik tombol *Refresh*	Memulai proses segarkan data	Merefresh form	[ ] Diterima [ ] Ditolak

3. Pengisian *Data* Karyawan

Berikut ini kasus dan hasil uji pengujian *data* karyawan :

Tabel 5.4.

Pengujian pengisian *data* karyawan

Kasus dan Hasil Uji			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol *Insert*	Memulai proses tambah data	Mengeluarkan form inputan	[ ] Diterima [ ] Ditolak
Klik tombol *Update*	Memulai proses ubah data	Mengeluarkan form *update* yang berupa data yang ingin diupdate	[ ] Diterima [ ] Ditolak
Klik tombol *Delete*	Memulai proses hapus data	Mengeluarkan konfirmasi untuk menghapus data	[ ] Diterima [ ] Ditolak
Klik tombol *Search*	Memulai proses mencari data	Mengeluarkan inputan data *first name* karyawan yang ingin dicari	[ ] Diterima [ ] Ditolak
Klik tombol *Sort*	Memulai proses menyortir data	Mengeluarkan pilihan data yang ingin disortir	[ ] Diterima [ ] Ditolak
Klik tombol *Refresh*	Memulai proses segarkan data	Merefresh form	[ ] Diterima [ ] Ditolak

4. Pengisian *Data* Klien

Berikut ini kasus dan hasil uji pengujian *data* klien :

Tabel 5.5.

Pengujian pengisian *data* klien

Kasus dan Hasil Uji			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol *Insert*	Memulai proses tambah data	Mengeluarkan form inputan	[ ] Diterima [ ] Ditolak
Klik tombol *Update*	Memulai proses ubah data	Mengeluarkan form *update* yang berupa data yang ingin diupdate	[ ] Diterima [ ] Ditolak
Klik tombol *Delete*	Memulai proses hapus data	Mengeluarkan konfirmasi untuk menghapus data	[ ] Diterima [ ] Ditolak
Klik tombol *Search*	Memulai proses mencari data	Mengeluarkan inputan data *first name* klien yang ingin dicari	[ ] Diterima [ ] Ditolak
Klik tombol *Sort*	Memulai proses menyortir data	Mengeluarkan pilihan data yang ingin disortir	[ ] Diterima [ ] Ditolak
Klik tombol *Refresh*	Memulai proses segarkan data	Merefresh form	[ ] Diterima [ ] Ditolak

5.2.2.3. Pengujian Proses

Pada pengujian ini, pengujian proses dibagi menjadi 4 proses , yaitu proses pengolahan modul karyawn, proses pengolahan informasi klien, proses perhitungan upah lembur dan proses absensi karyawan.

1. Proses Pengolahan Modul Karyawan

Berikut ini kasus dan hasil uji pengujian proses pengolahan modul karyawan :

Tabel 5.6.

Pengujian pengisian proses pengolahan modul karyawan

Kasus dan Hasil Uji			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol *Insert*	Memulai proses tambah data	Mengeluarkan form inputan	[ ] Diterima [ ] Ditolak
Klik tombol *Update*	Memulai proses ubah data	Mengeluarkan form *update* yang berupa data yang ingin diupdate	[ ] Diterima [ ] Ditolak
Klik tombol *Delete*	Memulai proses hapus data	Mengeluarkan konfirmasi untuk menghapus data	[ ] Diterima [ ] Ditolak
Klik tombol *Search*	Memulai proses	Mengeluarkan inputan data *first name* karyawan	[ ] Diterima

	mencari data	yang ingin dicari modulnya	[ ] Ditolak
Klik tombol *Sort*	Memulai proses menyortir data	Mengeluarkan pilihan data yang ingin disortir	[ ] Diterima [ ] Ditolak
Klik tombol *Refresh*	Memulai proses segarkan data	Merefresh form	[ ] Diterima [ ] Ditolak
Klik tombol *Search for Download Modul*	Memulai proses pencarian spesifik modul karyawan tersebut	Mengelurkan inputan *username* & *password*.	[ ] Diterima [ ] Ditolak
Klik tombol *Download Modul*	Memulai proses *Download Module*	Mengelurkan pilihan apakah modul tersebut akan dibuka saja (*open with*) atau akan langsung disimpan	[ ] Diterima [ ] Ditolak

2. Proses Pengolahan Informasi Klien

Berikut ini kasus dan hasil uji pengujian proses pengolahan informasi klien:

Tabel 5.7.

Pengujian pengisian proses pengolahan informasi klien

Kasus dan Hasil Uji			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**

Klik tombol *Insert*	Memulai proses tambah data	Mengeluarkan form inputan	[ ] Diterima [ ] Ditolak
Klik tombol *Update*	Memulai proses ubah data	Mengeluarkan form *update* yang berupa data yang ingin diupdate	[ ] Diterima [ ] Ditolak
Klik tombol *Delete*	Memulai proses hapus data	Mengeluarkan konfirmasi untuk menghapus data	[ ] Diterima [ ] Ditolak
Klik tombol *Search*	Memulai proses mencari data	Mengeluarkan inputan data *first name* klien yang ingin dicari data prosesnya	[ ] Diterima [ ] Ditolak
Klik tombol *Sort*	Memulai proses menyortir data	Mengeluarkan pilihan data yang ingin disortir	[ ] Diterima [ ] Ditolak
Klik tombol *Refresh*	Memulai proses segarkan data	Merefresh form	[ ] Diterima [ ] Ditolak
Klik tombol *Finish Processing*	Memulai proses penyelesaian proses	Mengelurkan konfirmasi apakah proses tersebut sudah benar-benar selesai atau belum	[ ] Diterima [ ] Ditolak
Klik tombol *Print*	Memulai pencetakan data	Mengelurkan pilihan proses dari klien siapa yang ingin dicetak informasinya.	[ ] Diterima [ ] Ditolak

3. Proses Perhitungan Upah Lembur

Berikut ini kasus dan hasil uji pengujian proses perhitungan upah lembur karyawan :

Tabel 5.8.

Pengujian pengisian proses perhitungan upah lembur

Kasus dan Hasil Uji			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol *Insert*	Memulai proses tambah data	Mengeluarkan form inputan	[ ] Diterima [ ] Ditolak
Klik tombol *Update*	Memulai proses ubah data	Mengeluarkan form *update* yang berupa data yang ingin diupdate	[ ] Diterima [ ] Ditolak
Klik tombol *Delete*	Memulai proses hapus data	Mengeluarkan konfirmasi untuk menghapus data	[ ] Diterima [ ] Ditolak
Klik tombol *Search*	Memulai proses mencari data	Mengeluarkan inputan data *first name* karyawan yang ingin dicari data lemburnya.	[ ] Diterima [ ] Ditolak
Klik tombol *Sort*	Memulai proses menyortir data	Mengeluarkan pilihan data yang ingin disortir	[ ] Diterima [ ] Ditolak
Klik tombol *Refresh*	Memulai proses	Merefresh form	[ ] Diterima

	segarkan data		[ ] Ditolak
Klik tombol *Print*	Memulai pencetakan data	Mengeluarkan pilihan proses dari karyawan siapa yang ingin dicetak data upah lemburnya.	[ ] Diterima [ ] Ditolak

4. Proses Pengolahan Absensi Karyawan

Berikut ini kasus dan hasil uji pengujian proses pengolahan absensi karyawan :

Tabel 5.9.

Pengujian pengisian proses pengolahan absensi karyawan

Kasus dan Hasil Uji			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol *Insert*	Memulai proses tambah data	Mengeluarkan form inputan	[ ] Diterima [ ] Ditolak
Klik tombol *Update*	Memulai proses ubah data	Mengeluarkan form *update* yang berupa data yang ingin diupdate	[ ] Diterima [ ] Ditolak
Klik tombol *Delete*	Memulai proses hapus data	Mengeluarkan konfirmasi untuk menghapus data	[ ] Diterima [ ] Ditolak

Klik tombol *Search*	Memulai proses mencari data	Mengeluarkan inputan data *first name* karyawan yang ingin dicari data absensinya.	[ ] Diterima [ ] Ditolak
Klik tombol *Sort*	Memulai proses menyortir data	Mengeluarkan pilihan data yang ingin disortir	[ ] Diterima [ ] Ditolak
Klik tombol *Refresh*	Memulai proses segarkan data	Merefresh form	[ ] Diterima [ ] Ditolak
Klik tombol *Attendance View*	Memulai proses lihat absensi karyawan per-periode	Mengeluarkan pilihan tanggal berapa sampai tanggal berapa untuk mencari data absensi.	[ ] Diterima [ ] Ditolak
Klik tombol *Print*	Memulai pencetakan data	Mengeluarkan pilihan proses dari karyawan siapa yang ingin dicetak data absensinya.	[ ] Diterima [ ] Ditolak

5.2.3. Kesimpulan Hasil Pengujian

Berdasarkan dari hasil implementasi dan pengujian terhadap hasil dari perancangan Aplikasi pembantu dan pendukung kinerja karyawan di Mitra Dinamika Konsultan ini, dapat disimpulkan bahwa perangkat lunak yang dibangun telah sesuai dengan kasus - kasus yang diberikan terhadap pengujian yang dilakukan. Prosedur - prosedur perangkat lunak telah berjalan dengan baik

dengan tingkat pemahaman pengguna terhadap perangkat lunak yang dibangun dapat dipahami dengan mudah.