




PENGEMBANGAN APLIKASI MONITORING PRESENSI KEHADIRAN DAN JURNAL KERJA KARYAWAN DI STIKES BANYUWANGI BERBASIS WEBSITE

BANNY FIRSHA BAHARI
1117101444

DOSEN PENGAMPU
Eko Heri Susanto, S.Kom, M.Kom

PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TINGGI ILMU KOMPUTER PGRI BANYUWANGI
2020



DAFTAR ISI

DAFTAR ISI.....	i
DAFTAR GAMBAR	iii
DAFTAR TABEL.....	iv
BAB I.....	1
IMPLEMENTASI	1
1.1. Lingkup Implementasi.....	1
1.2. Spesifikasi Kebutuhan Perangkat Keras	1
1.2.1. Laptop/PC.....	1
1.3. Spesifikasi Kebutuhan Perangkat Lunak.....	1
1.3.1. Laptop/PC.....	1
1.4. UML.....	2
1.4.1. Busines Proses	2
1.4.2. Use Case	4
1.4.3. Activity Diagram	5
BAB II.....	7
PENGERJAAN	7
2.1. Tampilan Antar Muka	7
2.1.1. Halaman Login	7
2.1.2. Halaman Beranda.....	8
2.1.3. Halaman Rekap Kehadiran	9
2.1.4. Halaman Jurnal	12
2.1.5. Halaman Karyawan.....	15
BAB III	17
PENUTUP.....	17
3.1. Kesimpulan	17
3.2. Saran	17
3.3. Daftar Pustaka.....	17

DAFTAR GAMBAR

(Gambar 1 - Busines Proses sebelum adanya pengembangan aplikasi)	2
(Gambar 2 - Busines proses setelah adanya pengembangan aplikasi)	3
(Gambar 3 - Use Case Diagram)	4
(Gambar 4 - Halaman Login)	7
(Gambar 5 - Proses Login).....	7
(Gambar 6 - Halaman Beranda).....	8
(Gambar 7 - Halaman Rekap Kehadiran).....	9
(Gambar 8 - Proses Ambil Data Presensi).....	10
(Gambar 9 - Prosedur Pengambilan Data).....	10
(Gambar 10 - Controller Ambil Data Presensi).....	11
(Gambar 11 - Penyajian Data Rekap Presensi).....	12
(Gambar 12 - Halaman Jurnal Kerja Karyawan)	12
(Gambar 13 - Controller Jurnal Kerja Karyawan).....	13
(Gambar 14 - Halaman View Jurnal Kerja Karyawan)	14
(Gambar 15 - Halaman Karyawan).....	15
(Gambar 16 - Controller Halaman Karyawan).....	16

DAFTAR TABEL

(Table 1 - Activity diagram melihat dan mengelolah data presensi)	5
(Table 2 - Activity diagram melihat data jurnal kerja).....	6
(Table 3 - Activity diagram melihat data karyawan).....	6

BAB I

IMPLEMENTASI

1.1. Lingkup Implementasi

Pada bab ini dijelaskan mengenai implementasi sistem yang telah dirancang ke dalam kode program dan tampilan antar muka. Dalam tahapan implementasi berikut ini terdapat beberapa spesifikasi hardware dan software yang dibutuhkan untuk mendukung jalannya Cloud Absen Stikes Banyuwangi (Admin Side).

1.2. Spesifikasi Kebutuhan Perangkat Keras

Adapun spesifikasi minimal hardware yang dianjurkan untuk sistem informasi ini sebagai berikut :

1.2.1. Laptop/PC

Processor : 2.2 Ghz
RAM : 2 GB
Hardisk : 1 GB

1.3. Spesifikasi Kebutuhan Perangkat Lunak

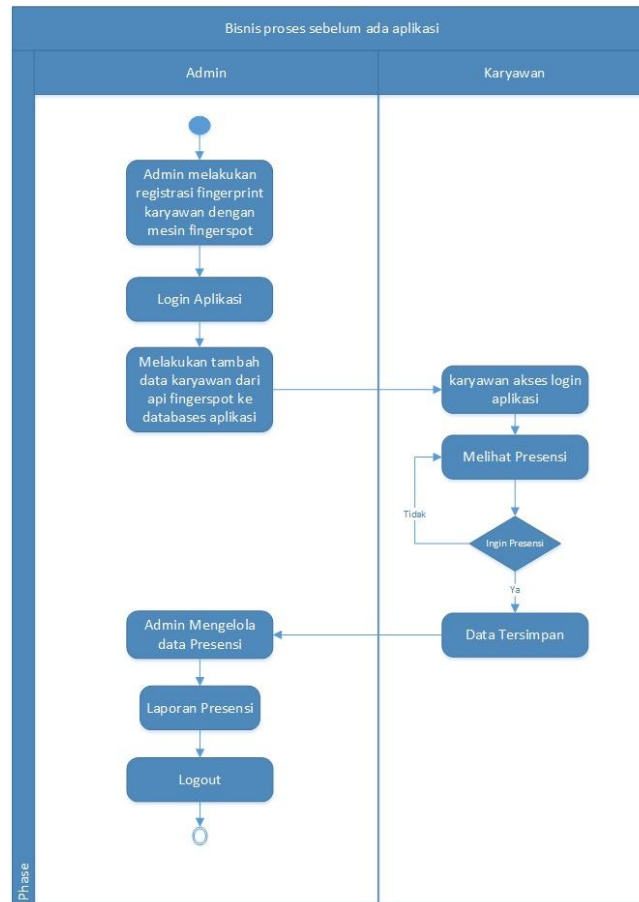
Adapun spesifikasi minimal software yang dianjurkan untuk sistem informasi ini sebagai berikut :

1.3.1. Laptop/PC

Linux, windows (7, 8, 10), Mac + Browser (Google Chrome)

1.4. UML

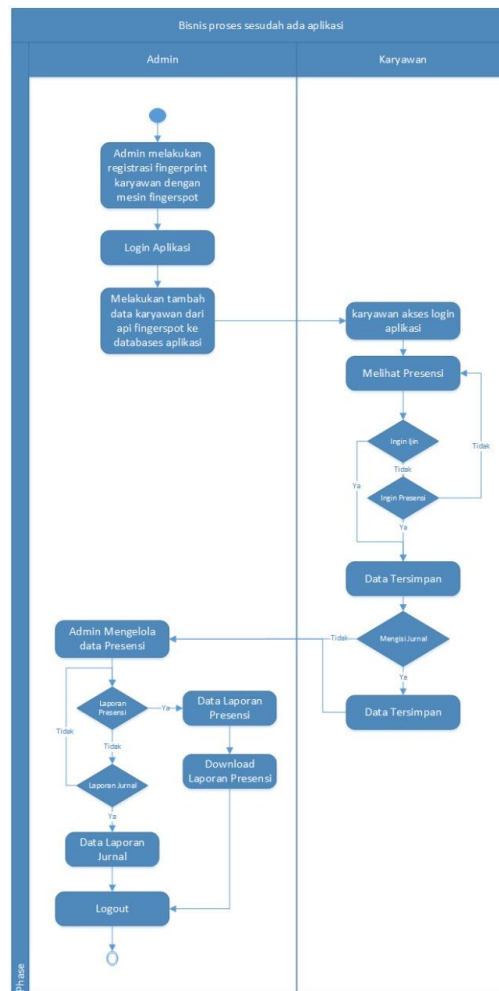
1.4.1. Business Proses



(Gambar 1 - Busines Proses sebelum adanya pengembangan aplikasi)

Gambar diatas adalah busines proses pada saat sebelum adanya pengembangan aplikasi. Penjelasan pada gambar diatas dapat dijelaskan sebagai berikut :

1. Karyawan hanya bisa melakukan presensi harian.
2. Admin hanya dapat melihat data rekapitulasi presensi karyawan.
3. Laporan presensi hanya berupa tabel, dan masih tidak adanya fitur download laporan presensi.

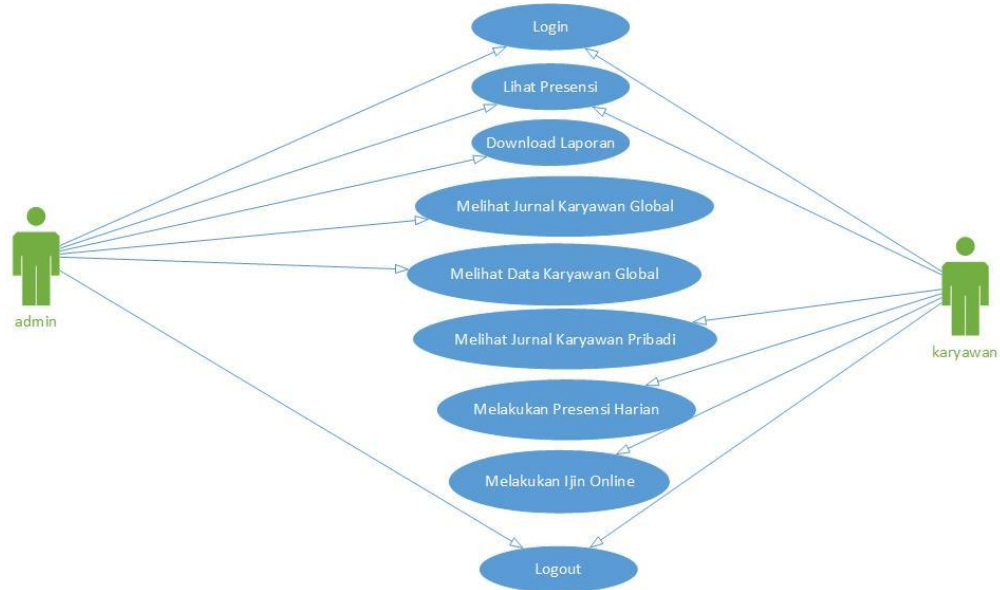


(Gambar 2 - Busines proses setelah adanya pengembangan aplikasi)

Gambar diatas adalah busines proses pada saat sebelum adanya pengembangan aplikasi. Penjelasan pada gambar diatas dapat dijelaskan sebagai berikut :

1. Karyawan bisa melakukan presensi harian dan melakukan pengajuan izin kehadiran secara online melalui aplikasi.
2. Selain melakukan pengajuan izin online, karyawan juga dapat melakukan pengisian jurnal kerja harian pada aplikasi.
3. Tersedianya fitur download rekap presensi kehadiran. Sehingga, admin dapat melakukan pengolahan data presensi karyawan lebih lanjut.
4. Laporan jurnal kerja karyawan membantu admin dalam mengetahui tingkat produktivitas karyawan yang sedang bekerja.

1.4.2. Use Case



(Gambar 3 - Use Case Diagram)

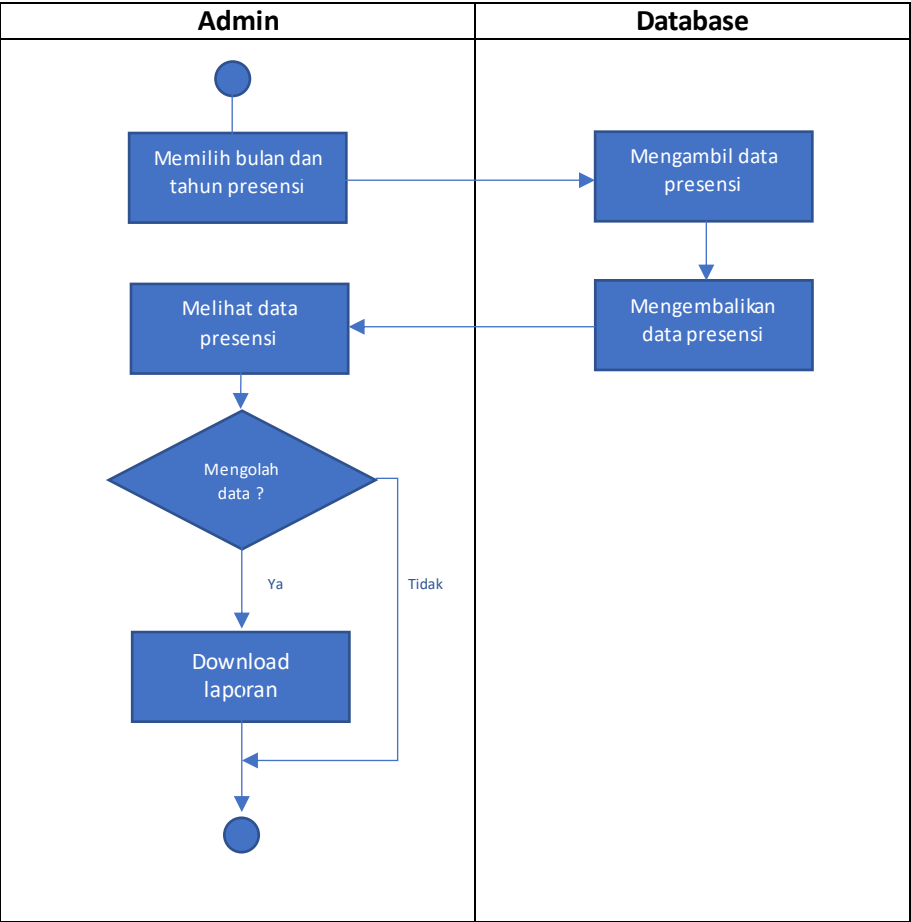
Dijelaskan pada gambar diatas, terdapat 2 aktor yang berperan dalam aplikasi ini. Setelah karyawan melakukan presensi kehadiran, pengajuan izin, dan menambahkan jurnal kerja, terdapat beberapa transaksi yang dapat dilakukan oleh admin setelah login sebagai berikut :

1. Admin dapat melihat data presensi.
2. Admin dapat mendownload laporan data presensi.
3. Admin dapat melihat jurnal kerja karyawan.
4. Admin dapat melihat data karyawan.

1.4.3. Activity Diagram

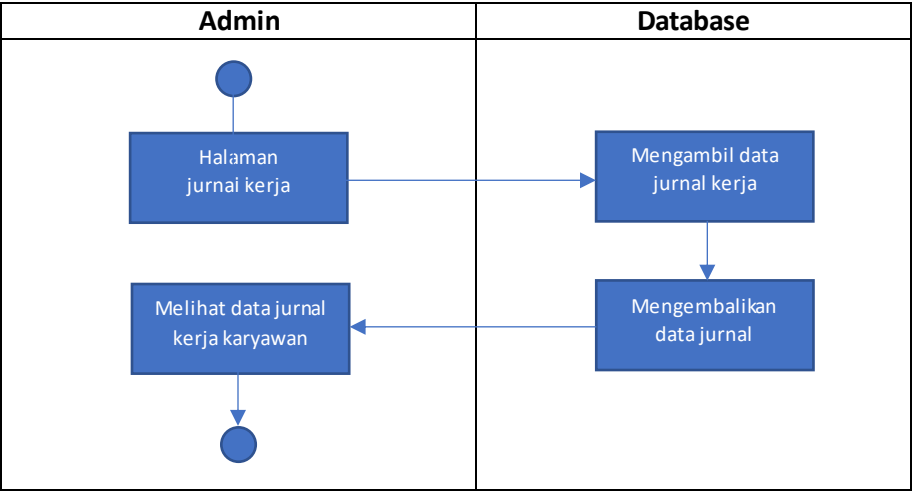
Dari beberapa point yang terdapat pada use case, terdapat beberapa activity diagram pada admin sebagai berikut :

1.4.3.1. Melihat dan download data presensi



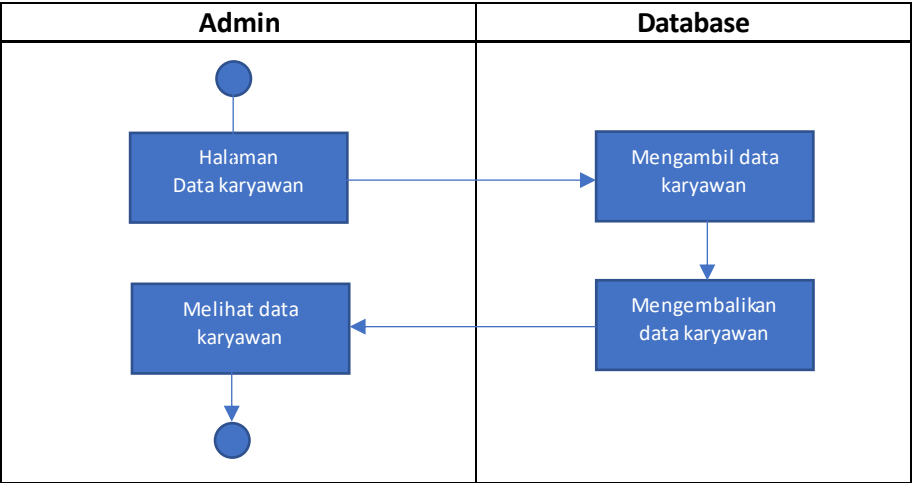
(Table 1 - Activity diagram melihat dan mengelolah data presensi)

1.4.3.2. Melihat data jurnal kerja



(Table 2 - Activity diagram melihat data jurnal kerja)

1.4.3.3. Melihat data karyawan



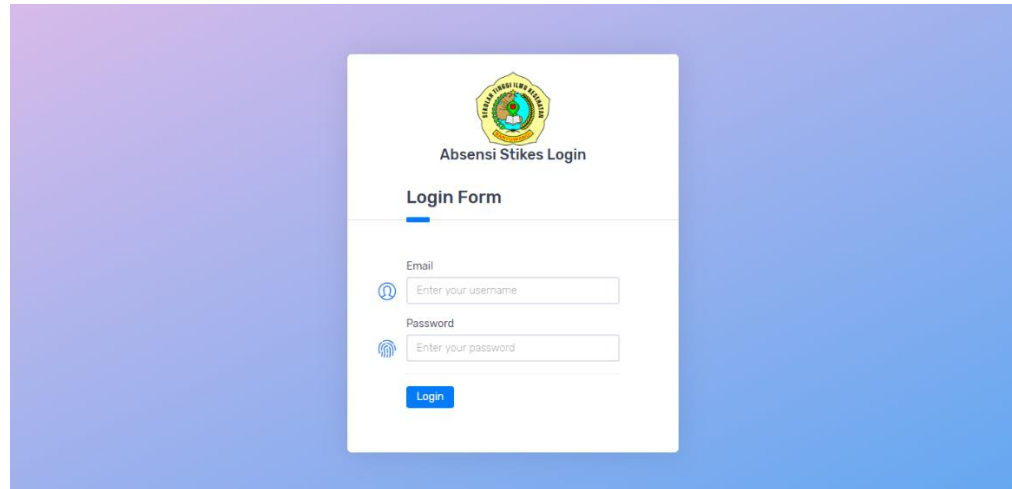
(Table 3 – Activity diagram melihat data karyawan)

BAB II

PENGERJAAN

2.1. Tampilan Antar Muka

2.1.1. Halaman Login



(Gambar 4 - Halaman Login)

Halaman tampilan awal menu login ini digunakan untuk mengidentifikasi pengguna aplikasi, karena aplikasi ini hanya untuk Admin. Admin diminta untuk memasukkan Username dan Password untuk dapat masuk ke dalam aplikasi.

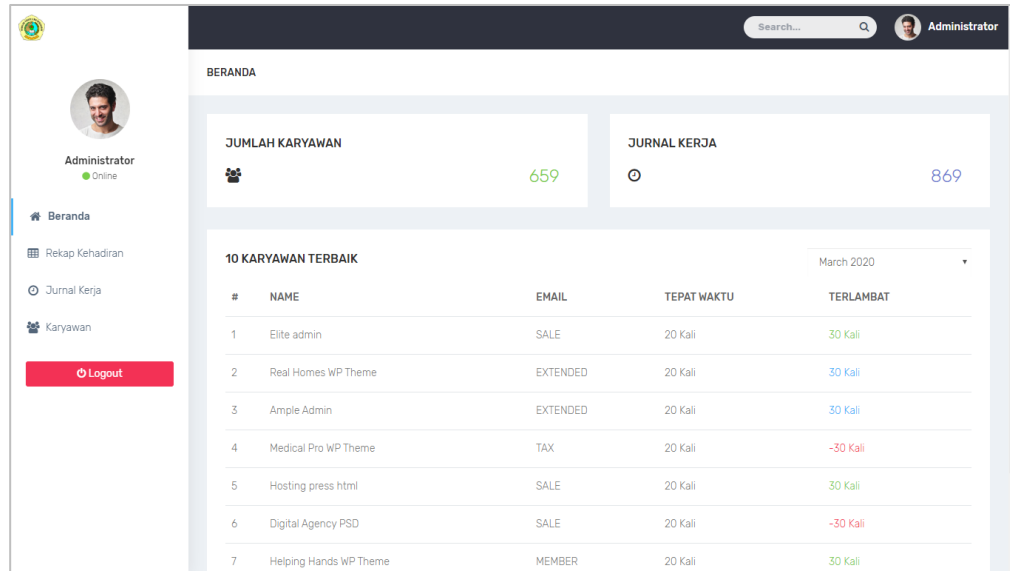
```
public function login($data){
    $uname = $data['email'];
    $password = password_hash($data['password'], PASSWORD_DEFAULT);
    $user = $this->db->get_where('users', ['email' => $uname])->row();
    if ($user) {
        if (password_verify($data['password'], $user->password)) {
            $this->session->set_userdata('login', $user->id);
            $this->addLog('berhasil login');
            $this->apps->_alert('success', 'Selamat Datang, ' . $user->first_name . $user->last_name . '!');
            redirect('');
        }else{
            $this->apps->_alert('danger','Password anda salah!');
            redirect('auth');
        }
    }else{
        $this->apps->_alert('danger','Email tidak ditemukan!');
        redirect('auth');
    }
}
```

(Gambar 5 - Proses Login)

Untuk fitur login saya menggunakan fungsi diatas. Di dalam fungsi diatas terdapat parameter *\$data* yang berfungsi untuk menampung informasi login yang dikirim dari form login. Setelah tertampung, *\$data* akan dipisahkan berdasarkan username dan password. Saya menggunakan fungsi PHP *password_hash()* yang berguna untuk menghasilkan hash informasi password pengguna. Setelah informasi username dan

password berhasil didapatkan. \$user bertugas mengambil data informasi pengguna berdasarkan username yang diterima pada \$data. Jika hasil pengambilan data mendapatkan hasil. Maka program akan melakukan verifikasi password berdasarkan hash password pada \$password. Jika informasi password benar, maka user akan mendapatkan session untuk bisa login ke dalam aplikasi.

2.1.2. Halaman Beranda



(Gambar 6 - Halaman Beranda)

Untuk halaman beranda, saya membaginya menjadi 3 bagian sebagai berikut :

1. Navbar

Navbar adalah bagian atas pada halaman website. Pada halaman beranda, saya memberikan informasi nama pengguna yang berhasil login dan sebuah inputan text untuk melakukan pencarian data karyawan.

2. Sidebar

Sidebar adalah bagian sebelah kiri atau kanan. Pada umumnya sidebar berisikan daftar menu yang tersedia pada sebuah website. Pada halaman beranda, saya memberikan 4 menu sebagai controller untuk berpindah halaman. Diantaranya sebagai berikut :

- a. Halaman Beranda
- b. Halaman Rekap kehadiran
- c. Halaman Jurnal Karyawan
- d. Halaman Karyawan

3. Isi

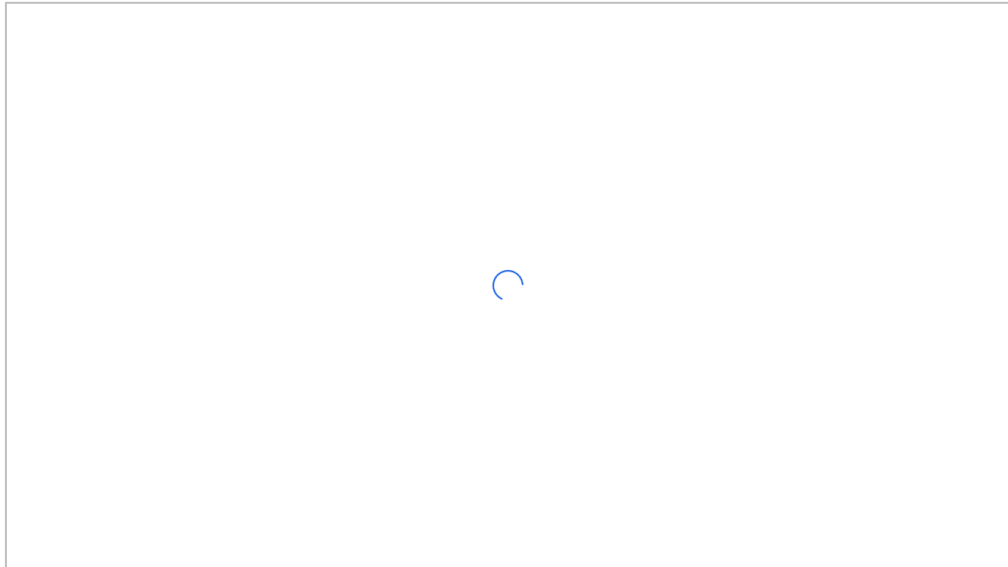
Isi halaman saya letakkan sejajar dengan sidebar dengan posisi yang berlawanan. Isi mengandung informasi dari setiap halaman yang saya buat. Seperti yang sudah saya jelaskan pada 4 menu pada point diatas.

Pada isi halaman beranda, saya memberikan informasi seputar karyawan dan jurnal. Informasi yang saya berikan hanya jumlah karyawan dan jumlah transaksi jurnal kegiatan yang sudah tersimpan. Dilanjutkan dengan table 10 karyawan terbaik sebagai penghibur.

2.1.3. Halaman Rekap Kehadiran

(Gambar 7 - Halaman Rekap Kehadiran)

Halaman rekap kehadiran bertugas untuk menyajikan informasi rekap kehadiran karyawan per masing – masing bulan. Pada halaman rekap kehadiran, terdapat 2 inputan yang harus diisi untuk menyajikan data rekapitulasi. Kondisi awal pada inputan bulan saya buat *disable* artinya, pengguna tidak bisa melakukan pengisian data pada inputan ini. Pengguna dapat mengisi inputan Bulan setelah berhasil mengisi inputan pada tahun.



(Gambar 8 - Proses Ambil Data Presensi)

Ketika pengguna berhasil memilih bulan pada inputan Bulan, sistem akan mengirimkan data kedua inputan ini, dan menjalankan beberapa fungsi sebagai berikut :

2.1.3.1. Request Data

```
<script>
    $("#tahun").change(function(){
        $("#bulan").prop('disabled', false);
    });

    $("#bulan").change(function(){
        var bulan = $("#bulan").val();
        var tahun = $("#tahun").val();
        var key = "<?= $this->apps->_csrf()->hash; ?>";

        $("#content").hide();
        $(".preloader").show();
        $(this).prop('disabled', true);
        $("#tahun").prop('disabled', true);

        $.ajax({
            type: 'POST',
            url: "<?= base_url('presence/getReport'); ?>",
            data: 'bulan='+bulan+'&tahun='+tahun+'&csrf_test_name='+key,
            success: function(msg){
                $(".preloader").hide();
                $("#content").show();
                $("#content").html(msg);
                $(this).prop('disabled', false);
                $("#tahun").val('');
                $("#tahun").prop('disabled', false);
            }
        });
    });
</script>
```

(Gambar 9 - Prosedur Pengambilan Data)

Sebelum melakukan request data, saya harus menampung kedua inputan dari tahun dan bulan pada halaman rekapitulasi kehadiran. Dalam hal ini saya menggunakan AJAX. Saya menampung kedua inputan tersebut dalam *variable* bulan dan tahun. Dan sebuah *variable* csrf untuk Framework PHP Code Igniter.

Setelah data berhasil terkumpul, AJAX akan megeksekusi pengiriman data POST menuju url (*presence/getReport*) dengan membawa 3 buah data yang sudah saya siapkan sebelumnya.

```
public function getReport(){
    $bulan      = $this->input->post('bulan');
    $tahun      = $this->input->post('tahun');

    if ($bulan < 10) { $m = '0'.$bulan; }else{ $m = $bulan; }
    $report     = $this->Mdata->getAbsen($m, $tahun)->result();

    if ($report) {
        $data['kar']    = json_encode($this->Mdata->getKaryawan()->result());
        $data['abs']    = json_encode($this->Mdata->getAbsen($m, $tahun)->result());
        $data['bulan']  = $m; $data['tahun'] = $tahun;

        $this->load->view('home/report', $data);
    }else{
        echo "<div class='col-md-12 col-lg-12 col-sm-12'><div class='white-box'><p class='alert alert-danger'>";
        echo "Tidak ada data rekap kehadiran pada bulan ini di tahun ".$tahun."!";
        echo "</p></div></div>";
    }
}
```

(Gambar 10 - Controller Ambil Data Presensi)

Setelah data yang dikirim sampai, maka tugas akan diteruskan oleh fungsi diatas. Dengan penjelasan sebagai berikut :

Seperti biasa, saya menampung variable tahun dan bulan terlebih dahulu. Kemudian, saya membuat sebuah variable *\$report* yang bertugas untuk menampung data rekapitulasi. Jika *\$report* tidak menghasilkan sebuah data, maka AJAX akan mengembalikan pesan berupa peringatan bahwa tidak adanya data pada bulan dan tahun tersebut. Sebaliknya, jika *\$report* menghasilkan sebuah data, AJAX akan mengembalikan data pada halaman rekapitulasi kehadiran dalam bentuk JSON.

REKAP KEHADIRAN

Silahkan masukan informasi bulan dan tahun untuk menampilkan data.

Pilih Tahun: Pilih Bulan:

Download Excel

No.	Nama Karyawan	NIK	1		2		3		4		5		In
			In	Out	In	Out	In	Out	In	Out	In	Out	
1	ARIK MUJI WINARNI, SE	60020906	L	L	07:55	15:58	08:02	16:58	L	L	L	L	08:10
2	FEBRY YULIANTI, SE	60030906	L	L	08:22	15:17	08:24	15:58	L	L	L	L	07:42
3	NURHASANAH	60040906	L	L	08:20	15:23	08:23	15:24	L	L	L	L	10:00
4	SHOLIHIN, S.Kep. Ners	60050906	L	L	?	?	?	?	L	L	L	L	06:45

(Gambar 11 - Penyajian Data Rekap Presensi)

Jika proses pengambilan data berhasil, maka loading page yang tampil setelah petugas memilih bulan otomatis akan hilang. Dan data rekapitulasi dapat disajikan.

2.1.4. Halaman Jurnal

JURNAL KERJA

Admin

ACHMAD EFENDI, S.Kep. Ners

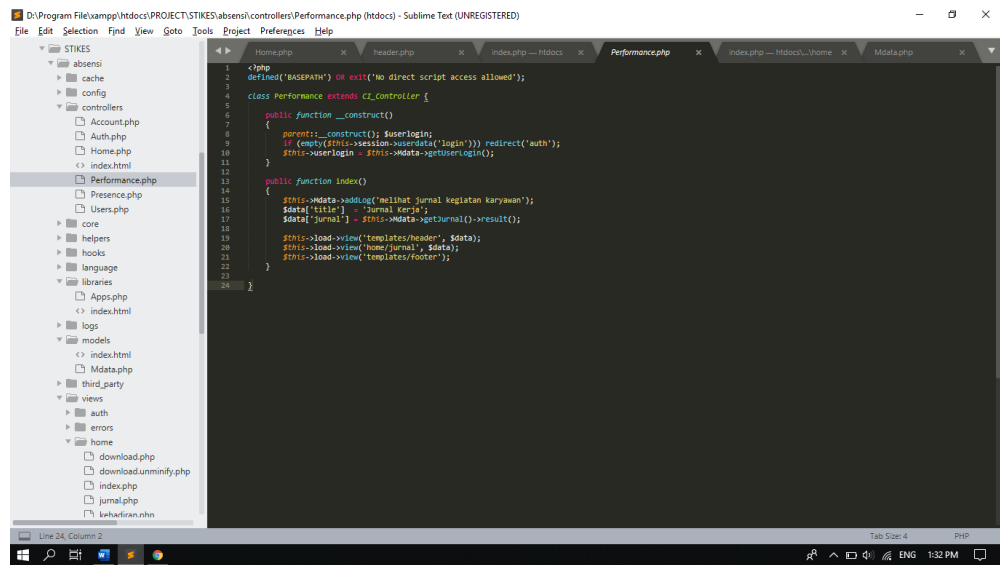
Mempelajari Flutter

Mempelajari flutter mulai dari materi dasar sampai dengan materi lanjutan.

04 July 2020 00:00:00

(Gambar 12 - Halaman Jurnal Kerja Karyawan)

Halaman ini bertugas menyajikan jurnal kerja karyawan yang dilaporkan setiap harinya. Karyawan dapat melaporkan pada **aplikasi android jurnal** yang sudah terinstall pada gadget masing-masing karyawan. Ketika data berhasil masuk, record jurnal karyawan akan tersaji secara otomatis pada halaman jurnal karyawan. Sehingga, admin akan mengetahui tingkat produktifitas pada masing-masing karyawan STIKES Banyuwangi.

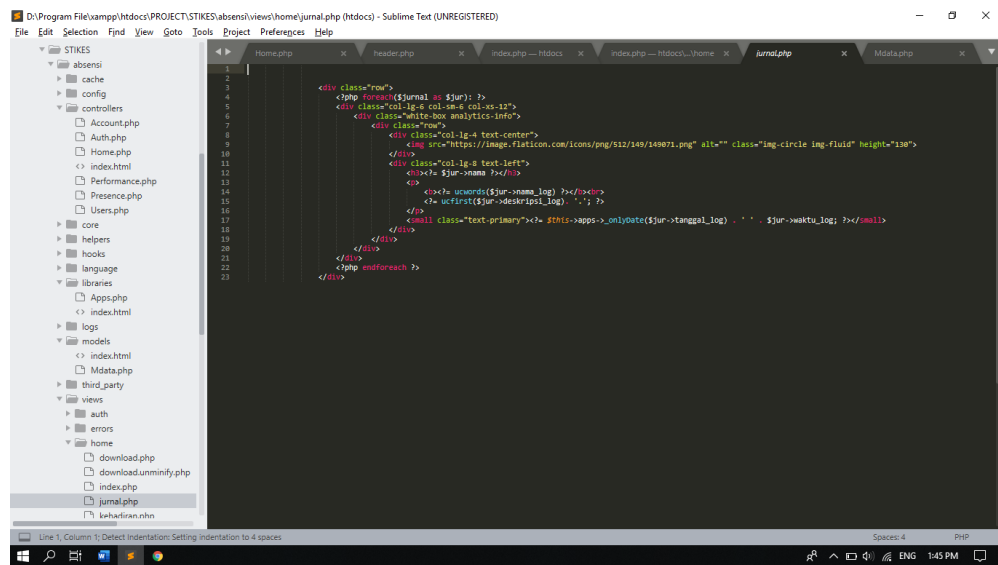


(Gambar 13 - Controller Jurnal Kerja Karyawan)

Untuk memproses halaman jurnal karyawan, sistem akan mengambil beberapa data untuk selanjutnya disajikan pada halaman index dengan mengambil data sebagai berikut :

1. Pada baris ke 15, sistem akan menjalankan perintah simpan log aktivitas.
2. Baris ke 16, sistem menyimpan data judul halaman yang tersimpan pada variable \$data dengan key *title*.
3. Baris ke 17 bertugas menyimpan data jurnal kegiatan karyawan pada variable \$data dengan key *berama jurnal*.
4. Setelah variable \$data berhasil disimpan, variable \$data akan dikirim pada view content pada baris ke 20.

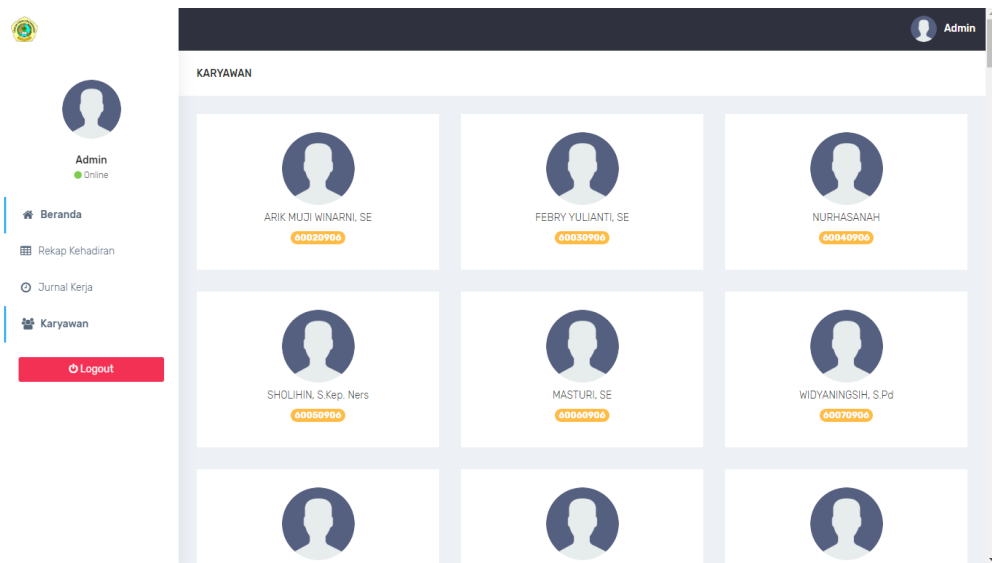
Pada halaman view content, variable \$data akan diberlakukan perulangan sebanyak data yang tersimpan pada \$data['jurnal'].



(Gambar 14 - Halaman View Jurnal Kerja Karyawan)

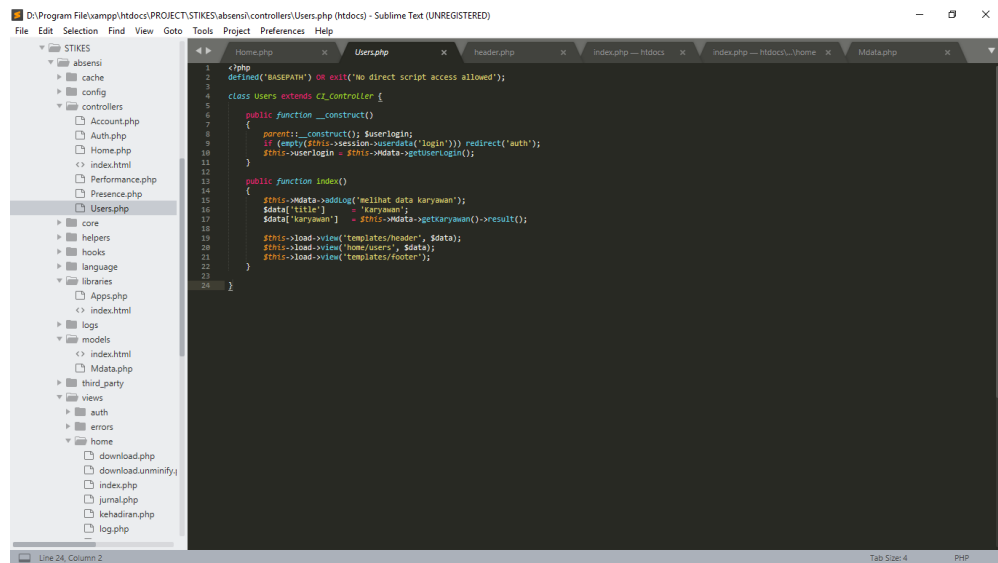
Setelah variable \$data dikirimkan pada view content, saya melakukan perulangan data yang tersimpan pada \$data[jurnal]. Pada setiap perulangan, saya menyajikan nama karyawan, judul kegiatan, deskripsi, dan waktu pengisian jurnal kegiatan karyawan.

2.1.5. Halaman Karyawan



(Gambar 15 - Halaman Karyawan)

Halaman karyawan digunakan untuk menyajikan data karyawan SIKES Banyuwangi. Pada halaman ini, admin dapat mengetahui informasi karyawan berupa nama dan id pengguna.



(Gambar 16 - Controller Halaman Karyawan)

Untuk memproses halaman karyawan, ada beberapa variable yang sudah saya siapkan. Penjelasannya sebagai berikut :

1. Controller index akan dimuat untuk memproses halaman karyawan.
2. Variable \$data digunakan untuk menampung beberapa data yang akan diteruskan pada view halaman karyawan.
3. \$data['title'] bertugas untuk menampung judul halaman yang akan ditampilkan pada browser.
4. \$data['karyawan'] bertugas menampung data karyawan STIKES Banyuwangi yang diambil dari model *Mdata*.
5. Setelah \$data berhasil disiapkan, variable \$data akan diteruskan pada view halaman karyawan.

BAB III

PENUTUP

3.1. Kesimpulan

Dari implementasi pada BAB II, penulis menarik beberapa kesimpulan sebagai berikut :

1. Implementasi tata letak yang tepat, membuat admin dapat berlama-lama menggunakan aplikasi.
2. Pengambilan data menggunakan AJAX membuat proses lebih efisien.
3. Pada fitur jurnal kerja karyawan sangat membantu pihak STIKES Banyuwangi dalam meninjau tingkat produktifitas setiap karyawannya.

3.2. Saran

Adapun saran-saran yang diberikan untuk penyempurnaan dan pengembangan aplikasi ini adalah sebagai berikut :

1. Sistem aplikasi ini masih menggunakan PHP sebagai jembatan penghubung antara aplikasi dengan sisi server. Pengembang dapat menggunakan Node.js untuk lebih mengoptimalkan proses.
2. Aplikasi ini masih tergolong sederhana dan masih banyak kekurangan. Sehingga aplikasi ini masih belum bisa diwujudkan seperti yang diharapkan. Pengembangan aplikasi ini ke depan diharapkan dapat menjadikan program aplikasi yang lebih baik lagi.

3.3. Daftar Pustaka

- [1] A.S., Rosa dan M.Shalahuddin., 2013. Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek). Bandung
- [2] Basuki, Awan Pribadi., 2007. Membangun Web berbasis PHP dengan Framework Codeigniter. Yogyakarta
- [3] Whitten, Jeffrey L. and Bentley Lonie D., 2004. System Analysis & Design Methods (seventh Edition). New York: McGraw-Hill.