**RANCANG BANGUN APLIKASI PENGELOLAAN SURAT**

**DI INTERNAL POLINDRA BERBASIS WEB**

**Tugas Akhir**

Diajukan untuk memenuhi sebagian persyaratan mencapai

gelar Ahli Madya pada jenjang Diploma iii

Program Studi Teknik Informatika



Oleh :

**MIA SILVIA**

**NIM. 1503071**

**JURUSAN TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI INDRAMAYU**

**2018**

**HALAMAN PENGESAHAN**

Tugas Akhir ini diajukan oleh:

Nama : Mia Silvia

NIM : 1503071

Program Studi : Teknik Informatika

Judul :Rancang Bangun Aplikasi Pengelolaan Surat di Internal

POLINDRA berbasis web

Pembimbing : 1. Munengsih Sari Bunga, S.Kom, M.Eng ........................

2. Eka Ismantohadi, S.Kom, M.Eng ........................

Telah berhasil dipertahankan dihadapan dewan penguji pada tanggal ... Agustus 2017 dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelas Ahli Madya Program Studi Teknik Informatika, Jurusan Teknik Informatika, Politeknik Negeri Indramayu

|  |  |  |  |
| --- | --- | --- | --- |
| DEWAN PENGUJI | | | |
| Nama | Jabatan | Tandatangan | Tanggal |
| 1. ...................... | Ketua Penguji | ............................. | ............................. |
| 1. ..................... | Sekretaris penguji | ............................. | ............................. |
| 1. ..................... | Anggota | ............................. | ............................. |

|  |
| --- |
| Indramayu, ............. Agustus 2017  Ketua Jurusan Teknik Informatika |
| **A. Sumarudin, S.Pd.,MT.,M.Sc**  **NIK. 09098630** |

**PERNYATAAN KEASLIAN**

Saya menyatakan dengan sesungguhnya bahwa Tugas Akhir ini adalah hasil karya saya sendiri, dan sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang pernah ditulis atau dipublikasikan oleh orang lain, kecuali yang secara tertulis disebutkan sumbernya dalam naskah dan dalam daftar pustaka.

|  |  |
| --- | --- |
|  | Indramayu, ............. Agustus 2018  Yang menyatakan,  **Mia Silvia**  NIM. 1503071 |

# 

**ABSTRAK**

Katakunci*:*

**ABSTRACK**

Keywords:

**MOTTO**

**KATA PENGANTAR**

Puji syukur penulis panjatkan kehadirat Allah SWT karena atas rahmat dan karunia-Nya, penulis dapat menyelesaikan Tugas Akhir ini. Shalawat dan salam senantiasa terlimpahkan atas nabi Muhammad Shalallahu ‘alaihi wa sallam.

Tugas Akhir ini merupakan salah satu persyaratan untuk memperoleh gelar Ahli Madya pada Program Studi Teknik Informatika, Jurusan Teknik Informatika Politeknik Negeri Indramayu.

Dalam penyusunan laporan Tugas Akhir ini penulis mengucapkan banyak terimakasih atas kerjasama dan dukungan dari semua pihak yang telah ikut serta membantu dalam penyusunan laporan Tugas Akhir (TA), untuk itu penulis mengucapkan Terimakasih kepada;

1. Kedua Orang tua dan Keluarga yang telah mendukung selama ini,.
2. Bapak Casiman Sukardi ST., MT selaku direktur Politeknik Negeri Indramayu,
3. Bapak A. Sumarudin S.Pd.,MT.,MSc selaku Ketua Jurusan Teknik Informatika,
4. Ibu Munengsih Sari Bunga, S.Kom., M.Eng selaku pembimbing utama,
5. Eka Ismantohadi, S.Kom., M.Eng selaku pembimbing pendamping
6. Rekan-rekan mahasiswa angkatan 2015 yang telah banyak membantu dalam pengerjaan tugas akhir.

Penulis berharap Tugas Akhir ini dapat bermanfaat bagi perkembangan ilmu pengetahuan menyadari bahwa pembuatan laporan ini sangatlah jauh dari sempurna, oleh karena itu penulis mengharapkan kritik dan saran yang membangun. Akhir kata dari penulis berharap semoga Laporan ini dapat memberikan manfaat bagi para pembacanya.

Indramayu, 28 Agustus 2018

Penulis

**DAFTAR ISI**

**HALAMAN PENGESAHAN** ii

**PERNYATAAN KEASLIAN** iii

**ABSTRAK** iv

**MOTTO** vi

**KATA** **PENGANTAR** vii

**DAFTAR ISI** viii

**DAFTAR TABEL** xi

**DAFTAR GAMBAR** xii

**DAFTAR LAMPIRAN** xiv

**BAB 1 PENDAHULUAN** 1

1.1 Latar Belakang 2

1.2 Rumusan Masalah 2

1.3 Batasan Masalah 2

1.4 Tujuan Penelitian 2

1.5 Manfaat Penelitian 2

1.6 Sistematika Penulisan 3

**BAB II LANDASAN TEORI** 4

2.1 Pengertian Sistem 4

2.1.1 Karakteristik Sistem 5

2.1.2 Pengertian Sistem Informasi 6

2.1.3 Komponen sistem Informasi 6

2.1.4 Tipe-tipe Sistem Informasi 7

2.2 Pengerian Sistem Pendukung Keputusan 7

2.3 Hasil Penelitian Terdahulu 10

2.4 Metode Simple Additive Weighting (Saw) 12

2.5 Website 17

2.6 Tools Pemrograman 18

2.6.1 Sublime Text 18

2.6.2 Definisi Html 19

2.6.3 Definisi Php 19

2.6.4 Css 20

2.6.5 Javascript 21

2.6.6 Xampp 21

2.6.7 Mysql 22

2.6.8 Bootstrap 23

2.6.9 Balsamiq Mockup 23

2.7 *Use C*ase Diagram 24

2.8 *Activity* Diagram 25

2.9 *Sequence* Diagram ` 26

2.10 *Flowchart* 27

2.11 Teori Pengujian 29

**BAB III Metodologi Penelitian** 30

3.1 Metode Penelitian 30

3.1.1 Desain Penelitian 30

3.2 Metode Pengumpulan Data 32

3.2.1 Studi Pustaka 32

3.3 Kebutuhan Sistem 32

3.3.1 Kebutuhan *Hardware* 32

3.3.2 Kebutuhan *Software* 33

3.4 Metode Saw 33

3.5 Perancangan Sistem 41

3.5.1 *Use* *Case* Diagram 41

3.5.2 *Activity* Diagram 42

3.5.3. *Sequence* Diagram 44

3.5.4 *Class* Diagram 40

3.5.5 *Flowcahart* Sistem 45

3.5.6 *Flowchart* Program 45

3.6 Perancangan *Database* 46

3.6.1 Tabel *User* 47

3.6.2 Tabel Kriteria 47

3.6.3 Tabel Himpunan 47

3.6.4 Tabel Caldos 48

3.6.5 Tabel Proses 48

3.6.6 Relasi *Database* 48

3.7 Rancangan Antar Muka 49

3.7.1 Rancangan Halaman *Login* 49

3.7.2 Rancangan Halaman *Home* 50

3.7.3 Rancangan Halaman Kriteria 50

3.7.4 Rancangan Halaman Himpunan Kriteria 51

3.7.5 Rancangan Halaman Calon Dosen 52

3.7.6 Rancangan Halaman Hasil Rating 52

**BAB IV Hasil dan Pembahasan** 53

4.1 Hasil 53

4.2 Pembahasan 53

4.2.1 Implementasi Basis Data 53

4.2.2 Implementasi Desain Interface 56

4.2.3 Pengujian Aplikasi 66

4.2.4 Kesimpulan Pengujian 69

**Bab V Penutup**  70

4.1 Kesimpulan 70

4.2 Saran 70

**Daftar Pustaka** 71

**DAFTAR TABEL**

Tabel 2.1 Contoh Kriteria Dan Bobot Kepemimpinan (C1) 14

Tabel 2.2 Contoh Kriteria Dan Bobot Penguasaan Barang Dagangan (C2) 14

Tabel 2.3 Contoh Kriteria Dan Bobot Kepribadian (C3) 14

Tabel 2.4 Contoh Kriteria Dan Bobot Umum Dan Lain-Lain (C4) 15

Tabel 2.5 Contoh Data Evaluasi/Penilaian Staff 15

Tabel 2.6 Contoh Rating Kecocokan 15

Tabel 2.7 Simbol *Use Case* Diagram 24

Tabel 2.8 Simbol *Use Case* Diagram (Lanjutan) 25

Tabel 2.9 Simbol *Activity* Diagram 25

Tabel 2.10 Simbol *Activity* Diagram (Lanjutan) 26

Tabel 2.11 Simbol *Sequence* Diagram 26

Tabel 2.12 Simbol *Sequence* Diagram (Lanjutan) 27

Tabel 2.13 Simbol-Simbol *Flowchart* 28

Tabel 2.14 Simbol-Simbol *Flowchart* (Lanjutan) 29

Tabel 3.1 Kebutuhan *Hardware* 32

Tabel 3.2 Kebutuhan *Software* 33

Tabel 3.3 Data Kriteria 34

Tabel 3.4 Kriteria IPK 35

Tabel 3.4 Kriteria Tes Tulis 35

Tabel 3.6 Kriteria Wawancara 36

Tabel 3.7 Kriteria Microteaching 36

Tabel 3.8 Contoh Data Calon Dosen Yang Mendaftar 37

Tabel 3.9 Nilai Hasil Konversi Berdasarkan Nilai Himpunan Krirteria 37

Tabel 3.10 Tabel Hasil Normalisasi 39

Tabel 3.11 Hasil Proses Perangkingan 40

Tabel 3.12 Keterangan *Use Case* Diagram 42

Tabel 3.13 Tabel *User* 47

Tabel 3.14 Tabel Kriteria 47

Tabel 3.15 Tabel Himpunan 47

Tabel 3.16 Tabel Caldos 48

Tabel 3.17 Tabel Proses 48

Tabel 4.1 Tabel Hasil Pengujian Dengan *Black-Box Testing* 65

Tabel 4.2 Tabel Hasil Pengujian Dengan Black-Box Testing (Lanjutan) 66

**DAFTAR GAMBAR**

Gambar 2.1 Model Sistem 4

Gambar 2.2 Konsep Sistem Informasi 6

Gambar 2.3 Fase Proses Pengambilan Keputusan 9

Gambar 2.4 *Interface* Sublime Text 18

Gambar 2.5 Contoh *Script* Html 19

Gambar 2.6 Logo Php 20

Gambar 2.7 Struktur Css 20

Gambar 2.8 Logo Xampp 22

Gambar 2.9 Logo Mysql 22

Gambar 2.10 Logo Bootstrap 23

Gambar 2.11 Fitur Balsamiq Mockup 24

Gambar 3.1 Model *Waterfall* 31

Gambar 3.2 *Use Case* Diagram 42

Gambar 3.3 *Activity* Diagram 43

Gambar 3.4 *Sequence* Diagram 44

Gambar 3.5 *Class* Diagram 44

Gambar 3.6 *Flowchart* Sistem 45

Gambar 3.7 *Flowchart* Program 46

Gambar 3.8 Relasi Tabel 49

Gambar 3.9 Rancangan Halaman *Login* 50

Gambar 3.10 Rancangan Halaman *Home* 50

Gambar 3.11 Rancangan Halaman Kriteria 51

Gambar 3.12 Rancangan Halaman Himpunan Kriteria 51

Gambar 3.13 Rancangan Halaman Calon Dosen 52

Gambar 3.14 Rancangan Halaman Hasil Rating 52

Gambar 4.1 Tampilan Login 56

Gambar 4.2 Tampilan *Pop Up Login* Gagal 56

Gambar 4.3 Tampilan *Home* 57

Gambar 4.4 Tampilan Kriteria 57

Gambar 4.5 Tampilan *Pop Up* Data Berhasil Ditambahkan 58

Gambar 4.6 Tampilan Tabel Kriteria 58

Gambar 4.7 Tampilan Halaman Edit Kriteria 59

Gambar 4.8 Tampilan *Pop Up* Data Berhasit Di Perbaharui 59

Gambar 4.9 Tampilan *Pop Up* Data Berhasil Di Hapus 59

Gambar 4.10 Tampilan Form Tambah Himpunan Kriteria 60

Gambar 4.11 Tampilan Tabel Himpunan Kriteria 60

Gambar 4.12 Tampilan Form Data Calon Dosen 61

Gambar 4.13 Tampilan Tabel Calon Dosen 61

Gambar 4.14 Tampilan Nilai Bobot Alternatif 62

Gambar 4.15 Tampilan Modals Search Nama Calon Dosen 62

Gambar 4.16 Tampilan Form Inputan Nilai Himpunan Kriteria 63

Gambar 4.17 Tampilan Tabel Data Himpunan Kriteria 63

Gambar 4.18 Tampilan Hasil Rating 64

**Daftar lampiran**

Lampiran 1 Kuisoner Penguian Aplikasi

Lampiran 2 Source Code

**BAB I**

**PENDAHULUAN**

* 1. **Latar Belakang**

Menurut Sihotang (2018), surat adalah alat komunikasi tertulis yang berasal dari satu pihak dan ditujukan kepada pihak lain untuk menyampaikan berita atau informasi. Surat juga memiliki beberapa macam, salah satunya yaitu surat resmi. Surat resmi merupakan surat yang berisikan masalah kedinasan atau administrasi pemerintah. Surat resmi hanya dibuat oleh instansi pemerintah dan dapat dikirimkan kepada semua pihak yang memiliki hubungan dengan instansi tersebut.

Surat resmi dalam suatu instansi terdapat surat masuk dan surat keluar. Surat masuk adalah surat yang masuk kedalam suatu instansi/perusahaan, baik yang berasal dari instansi/perusahaan lain atau bagian lain pada instansi/perusahaan yang sama. Sedangkan surat keluar adalah surat dikirim oleh suatu instansi/perusahaan, dan ditujukan kepada instansi/perusahaan lain atau yang sama (Ida, 2008).

Pengelolaan surat masuk dan keluar pada POLINDRA (Politeknik Negeri Indramayu) juga masih dilakukan secara manual oleh administrasi persuratan dengan mengelompokkan dokumen-dokumen yang sejenis. Hal tersebut memungkinkan inkonsistensi data terhadap surat masuk dan surat keluar, penumpukan catatan atau rekapan data, surat yang tercecer sehingga ketika data surat keluar dan surat masuk diperlukan maka harus dicari kembali, proses pencarian tidak bisa dilakukan dengan cepat dan tepat. Penomoran untuk surat keluar juga masih dilakukan secara manual yaitu dengan cara setiap bagian yang memerlukan nomor surat harus dilihat satu-persatu nomor surat untuk menghindari penomoran ganda. Salah satu proses penting dalam surat-menyurat pada POLINDRA adalah proses disposisi. Disposisi adalah proses persetujuan surat masuk yang dilakukan oleh Direktur yang terkait. Berdasarkan prosedur yang ada pada POLINDRA, seringkali terjadi kendala dalam ketepatan waktu disposisi yang dikarenakan ketidakberadaan pimpinan di kantor. Hal ini mengakibatkan lambatnya tanggapan yang harus diberikan kepada pihak yang membutuhkan konfirmasi dari surat yang dikirim. Dalam mengetahui status atau progress surat pun masih sulit, ketika pengirim surat ingin mengetahui sejauh mana surat yang dikirimkan sudah diproses, administrasi surat harus melihat data secara manual dan memakan waktu yang cukup lama.

Untuk meminimalisir hal-hal tersebut maka perlu dibuatlah sebuah aplikasi Rancang Bangun Aplikasi Pengelolaan Surat di Internal Polindra Berbasis Web. Aplikasi ini terdapat manajemen surat masuk dan surat keluar, disposisi surat, dan tracking progres surat. Aplikasi ini juga dapat menghasilkan laporan surat masuk dan surat keluar. Dalam menginputkan data surat keluar dapat mengurutkan secara otomatis nomor surat keluar sehingga tidak akan terjadi penomoran surat ganda. Aplikasi ini diharapkan dapat membantu proses pengelolaan dan penyimpanan surat agar lebih terorganisir dan mengurangi ketidaktepatan waktu dalam memberi tanggapan terhadap surat masuk, karena dapat diakses dimana saja dan kapan saja meskipun sedang tidak berada di kantor.

* 1. **Rumusan Masalah**

Dari latar belakang yang telah diuraikan dapat ditentukan rumusan masalah yaitu:

1. Bagaimana membuat aplikasi web yang dapat memudahkan proses disposisi surat masuk?
2. Bagaimana membuat aplikasi web yang dapat memudahkan dalam pemberian nomor pada surat keluar?
3. Bagaimana membuat aplikasi web yang dapat memudahkan dalam mengetahui sejauh mana surat sudah diproses?
4. Bagaimana cara membuat aplikasi web yang dapat menghasilkan laporan data surat masuk dan surat keluar?
   1. **Batasan Masalah**

Batasan masalah dari Rancang Bangun Aplikasi Pengelolaan Surat di Internal Polindra Berbasis Web ini adalah sebagai berikut:

1. Aplikasi ini hanya terdapat fitur manajemen surat masuk dan surat keluar, disposisi surat masuk, laporan surat masuk dan surat keluar, pembuatan surat keluar.
2. Pada disposisi surat masuk, hanya membuat disposisi surat masuk yang telah diinputkan dan mengisi disposisi yang diteruskan untuk ditindak lanjuti oleh unit kerja.
3. Pada tracking surat masuk, proses yang diketahui hanya dari menginputkan surat masuk, pembuatan disposisi, pengisian disposisi, sampai status SK (surat keputusan).
4. Aplikasi ini hanya menyediakan laporan berdasarkan surat masuk dan surat keluar berupa Microsoft Excel.
   1. **Tujuan Penelitian**

Berdasarkan perumusan masalah yang telah diuraikan maka tujuan yang ingin dicapai adalah :

1. Membuat aplikasi web yang dapat memudahkan proses disposisi surat masuk.
2. Membuat aplikasi web yang dapat memudahkan dalam pemberian nomor pada surat keluar.
3. Membuat aplikasi web yang dapat memudahkan dalam mengetahui sejauh mana surat sudah diproses.
4. Membuat aplikasi web yang dapat menghasilkan laporan data surat masuk dan surat keluar.
   1. **Manfaat Penelitian**

Manfaat dari Rancang Bangun Aplikasi Pengelolaan Surat di Internal Polindra Berbasis *Website* ini adalah :

1. Mempermudah bagian administrasi persuratan dalam proses disposisi surat masuk.
2. Mempermudah bagian administrasi persuratan dalam mengatur penomoran surat keluar.
3. Mempermudah bagian administrasi persuratan dalam membuat laporan surat masuk dan surat keluar.
4. Mempermudah bagian administrasi persuratan untuk mengetahui sejauh mana surat masuk yang sudah diproses.
   1. **Sistematika Penulisan**

Untuk mempermudah penulisan dan pemahaman yang lebih sistematis, penulisan laporan tugas akhir ini tersusun dalam lima bab, dengan penjelasan sebagai berikut :

Bab i pendahuluan

Bab ini menguraikan tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

Bab ii landasan teori

Bab ini berisi tentang uraian teori yang berkaitan dengan masalah yang ada dalam penelitian ini.

bab iii metode penelitian

Bab ini berisi tentang tahapan dan metode penelitian yang ditempuh untuk mencapai tujuan yang ditetapkan.

bab iv hasil dan pembahasan

Bab ini berisi tentang hasil-hasil dari tahapan penelitian, dari tahap analisis sistem, desain, implementasi desain, hasil pengujian dan implementasinya, berupa penjelasan teoritik, baik secara kualitatif, atau secara statistik.

bab v penutup

Bab ini berisi tentang kesimpulan dan saran dari hasil penelitian yang telah dilakukan.

**BAB II**

**LANDASAN TEORI**

* 1. **Surat**

Surat adalah sarana komunikasi untuk menyampaikan maksud secara tertulis. Dalam pengertian sehari-hari, surat umumnya hanya dikenal sebagai alat untuk menyampaikan berita secara tertulis. Pengertian tersebut merupakan pengertian dalam arti sempit, akibat dari anggapan bahwa surat hanya alat untuk menyampaikan berita, padahal surat mengandung aspek yang jauh lebih luas mencakup informasi tertulis. Adapun pengertian dengan informasi tertulis disini adalah informasi berupa kabar atau berita, misalnya penawaran, pesanan, panggilan dan permohonan. Surat juga bias sebagai informasi rekaman berita secara tertulis, misalnya surat tanda bukti, kartu identitas, akta dan kontrak. Maka dari uraian di atas dapat disimpulkan bahwa pada dasarnya surat adalah informasi tertulus yang dapat diperguanakan sebagai alat komunikasi tulisan yang dibuat dengan persyaratan tertentu yang khusus yang berlaku untuk surat menyurat. Penyampaian maksud melalaui surat dari suatu pihak ke pihak lain dapat diatasnamakan perseorangan (pribadi) dan dapat juga diatasnamakan jabatan dan juga suatu organisasi. Kegiatan saling berkirim oleh perseorangan (pribadi) atau oleh organisasi disebut surat menyurat korespondensi dan para pelakunya disebut koresponden. Berikut pengertian surat masuk dan surat keluar (Meliana, 2008):

1. Pengertian Surat Masuk

Surat masuk yaitu semua surat yang dialamatkan, ditunjukan dan diterima oleh organisasi, perusahaan atau instansi, baik yang berasal dari perorangan maupun dari suatu organisasi perusahaan instansi dan lainya. Pengendalian dan pengurusan surat masuk dalam instansi dapat digolongkan menurut penggolongan jenis surat yaitu:

1. Surat penting

Semua surat yang mengemukakan semua masalah-masalah pokok yang memepengaruhi langsung ataupun tidak langsung, berhasil tidaknya pencapaian tujuan organisasi.

1. Surat Rutin atau Biasa

Surat yang tidak tergolong penting, dimana surat-surat tersebut langsung di tindak lanjuti, relatif singkat dan tidak disimpan terlalu lama.

1. Surat Rahasia

Surat yang harus disampaikan sesegera mingkin kepada pimpinan (orang yang bersangkutan) yang masih dalam keadaan tertutup, sehingga surat tersebut tidak boleh dibuka oleh penerima surat.

1. Surat Pribadi

Surat yang disimpulkannya tercantum nama pribadi orang yang bersangkutan, walaupun disertai jabatan formalnya.

1. Pengertian Surat Keluar

Surat yang dibuat oleh instansi atau perusahaan yang ditujukan kepada instansi atau perusahaan lain atau untuk kepentingan internal.

* 1. **Disposisi**

Disposisi merupakan petunjuk singkat tentang tindak lanjut (penyelesaian) terhadap suatu urusan atau surat masuk. Disposisi dibuat oleh pimpinan untuk staf atau bawahan sesuai dengan bidang keahlian atau kewenangannya. Nota pengantar disposisi adalah formulir yang digunakan oleh pengelola atau pimpinan (atasan) untuk memberikan intruksi tanggapan kepada bawahan atau ke bagian lain dari perusahaan atas kelanjutan satu surat (Triyono, 2013).

* 1. **Arsip**

Menurut Vironica (2014), arsip adalah setiap catatan yang tertulis, tercetak atau ketikan dalam bentuk huruf, angka atau gambar, yang mempunyai arti dan tujuan tertentu sebagai bahan komunikasi dan informasi yang terekam pada kertas (kartu,formulir), kertas film (slide, film‐strip, mikro film), media komputer (pita tape, piringan,rekaman, disket), kertas fotocopy dan lain-lain. Berdasarkan uraian diatas dapat didefinisikan bahwa arsip merupakan naskah-naskah atau dokumen‐dokumen sebagai pusat pengingat dari berbagai kegiatan atau organisasi dimana naskah‐naskah tersebut disimpan sebaik mungkin secara sistematis ditempat yang telah disediakan agar lebih mudah dicari apabila diperlukan kembali.

* 1. **Website**

Menurut Abbas (2013), website disebut juga site, situs, situs web atau portal. Merupakan kumpulan halaman web yang berhubungan antara satu dengan lainnya, halaman pertama sebuah website adalah home page, sedangkan halaman demi halamannya secara mandiri disebut web page, dengan kata lain website adalah situs yang dapat diakses dan dilihat oleh para pengguna internet diseluruh dunia. Website adalah situs yang dapat diakses dan dilihat oleh para pengguna Internet. Pengguna Internet semakin hari semakin bertambah banyak, sehingga hal ini adalah potensi pasar yang berkembang terus.

* 1. **Proses Bisnis Pengelolaan Surat di POLINDRA**

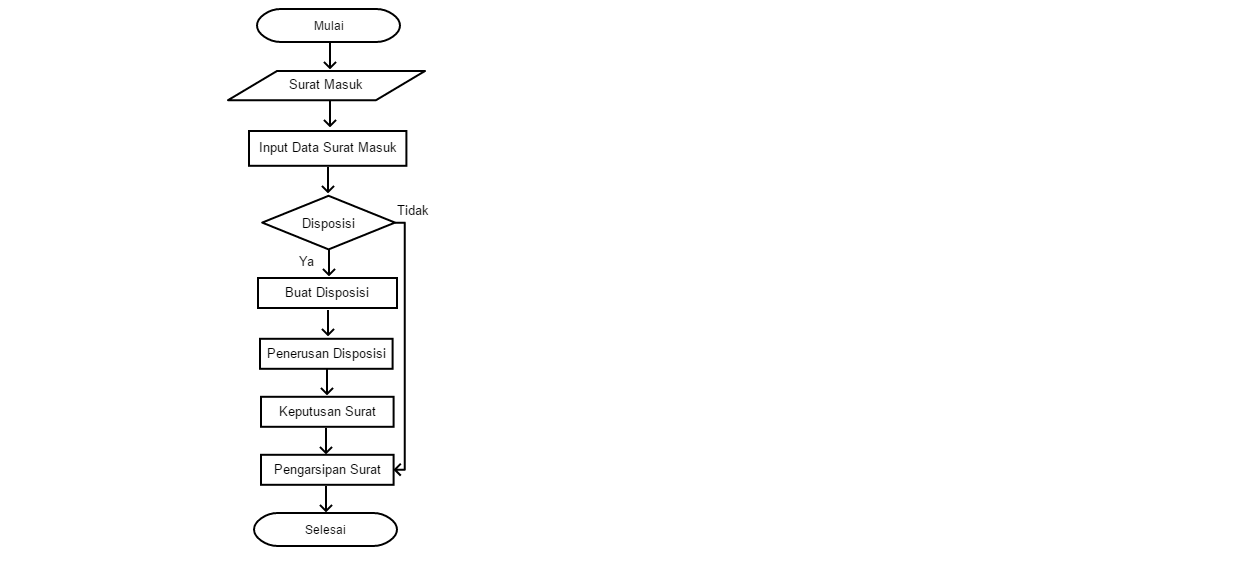
Terdapat Proses bisnis dalam pengelolaan surat di Politeknik Negeri Indramayu (POLINDRA), yaitu surat masuk dan surat keluar.

Pada surat masuk, proses awal yang dilakukan oleh admin adalah menginputkan data surat masuk yang diterima. Setelah diinputkan, Jika surat masuk tidak membutuhkan disposisi, maka surat masuk langsung diarsipkan. Jika surat membutuhkan disposisi dibuatkan disposisi surat. Setelah dibuat disposisi suratnya, surat masuk dan disposisi surat diteruskan oleh unit kerja secara bertahap. Disposisi akan diisi oleh unit kerja yang diteruskan. Kemudian jika sudah selesai dibuat surat keputusannya. Langkah terakhir admin mengarsipkan surat yang telah selesai diproses.

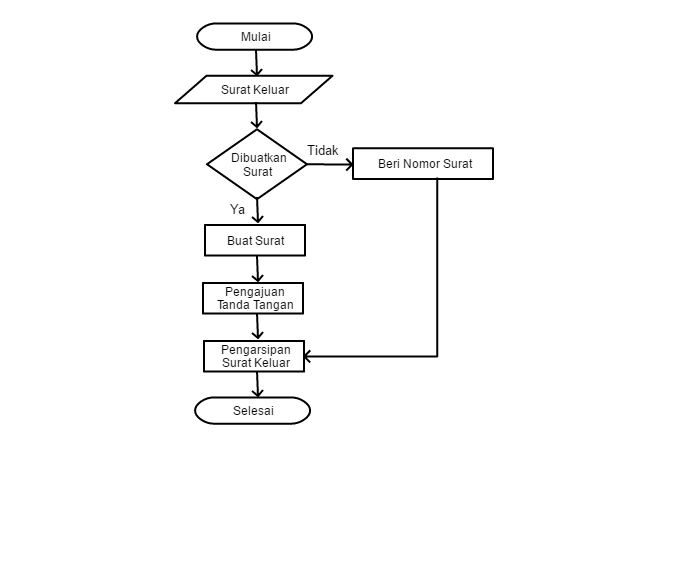
Pada surat keluar, proses awal yang dilakukan oleh admin adalah pembuatan surat keluar, jika surat tidak perlu dibuat oleh admin, admin hanya memberi nomor surat keluar saja, lalu jika surat keluar sudah diberikan ke admin maka langsung diarsipkan. Jika perlu dibuatkan surat oleh admin, admin akan membuat surat.

Setelah itu surat keluar akan diajukan untuk ditandatangani. Setelah ditandatangani, surat keluar diberikan pada tujuan suratnya. Langkah terakhir adalah admin mengarsipkan surat keluar.

Berikut adalah *flowchart* dari proses bisnis surat masuk dan surat keluar:

****

Gambar 2.1 Proses Bisnis Surat Masuk



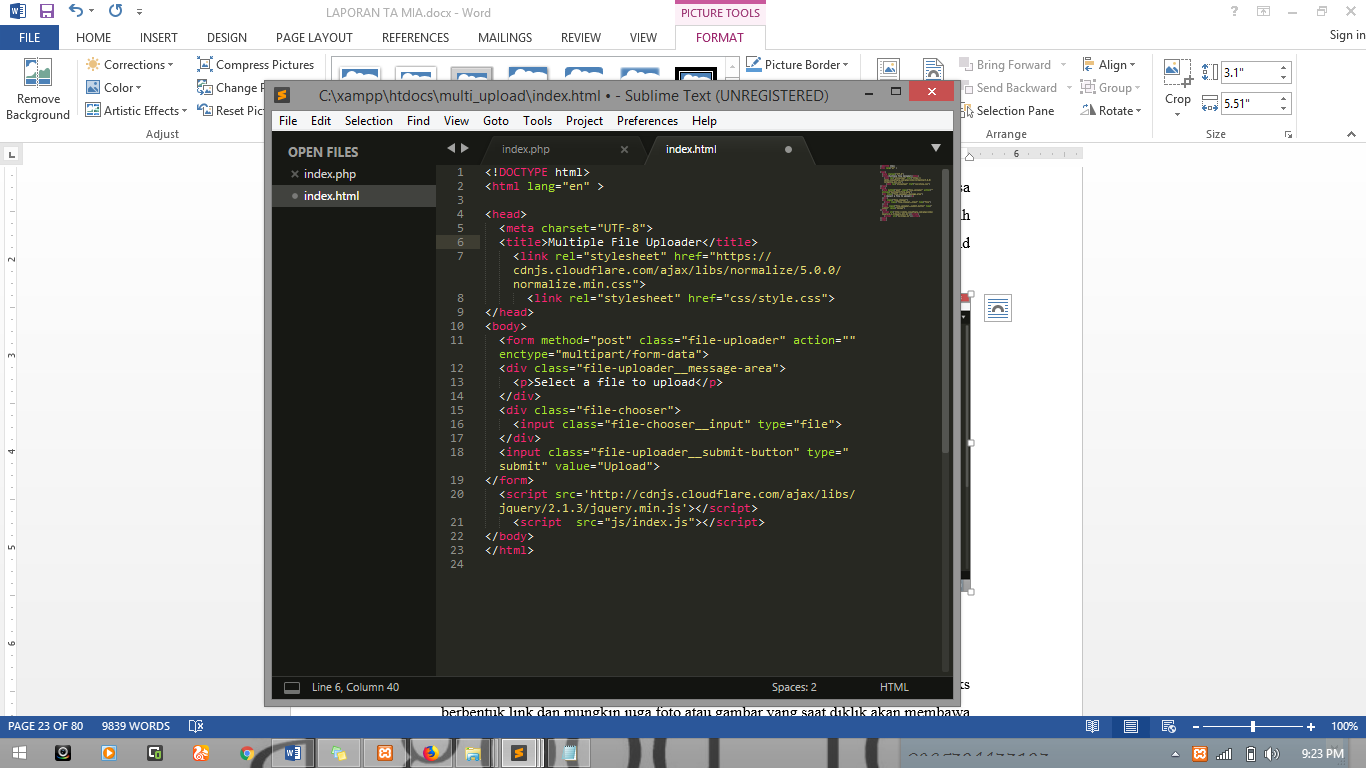
Gambar 2.2 Proses Bisnis Surat Keluar

* 1. **Tools pemrograman**

Dalam pengembangan suatu aplikasi, tentunya membutuhkan suatu tools atau alat berupa software dan bahasa pemrograman. Tools dan alat yang dipakai adalah sebagai berikut:

* + 1. **Sublime Text**

Sublime Text adalah aplikasi editor untuk kode dan teks yang dapat berjalan di berbagai *platform operating system* dengan menggunakan teknologi Phyton API. Terciptanya aplikasi ini terinspirasi dari aplikasi Vim, Aplikasi ini sangatlah fleksibel dan *powerfull*. Fungsionalitas dari aplikasi ini dapat dikembangkan dengan menggunakan *sublime-packages*. Sublime Text mendukung berbagai bahasa pemrograman dan mampu menyajikan fitur syntax highlight hampir di semua bahasa pemrogramman yang didukung ataupun dikembangkan oleh komunitas seperti; C, C++, C#, CSS, D, Dylan, Erlang, HTML, Groovy, Haskell, Java, JavaScript, LaTeX, Lisp, Lua, Markdown, MATLAB, OCaml, Perl, PHP, Python, R, Ruby, SQL, TCL, Textile and XML. Biasanya bagi bahasa pemrograman yang didukung ataupun belum terdukung secara default dapat lebih dimaksimalkan atau didukung dengan menggunakan add-ons yang bisa didownload sesuai kebutuhan user (Anhar, 2010).



Gambar 2.3 *Interface* Sublime Text

* + 1. **Definisi html**

HTML merupakan sebuah bahasa markah yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi di dalam sebuah penjelajah web Internet dan pemformatan hiperteks sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegerasi. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan dalam format ASCII normal sehingga menjadi halaman web dengan perintah-perintah HTML. Bermula dari sebuah bahasa yang sebelumnya banyak digunakan di dunia penerbitan dan percetakan yang disebut dengan SGML *(Standard Generalized Markup Language),* HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman web. HTML saat ini merupakan standar Internet yang didefinisikan dan dikendalikan penggunaannya oleh *World Wide Web Consortium* (W3C). HTML dibuat oleh kolaborasi Caillau TIM dengan Berners-lee Robert ketika mereka bekerja di CERN pada tahun 1989 CERN adalah lembaga penelitian fisika energi tinggi di Jenewa (Karma, 2004).

* + 1. **PHP**

PHP adalah singkatan dari *Hypertext Preprosesor* yaitu bahasa pemrograman web *server-Side* yang bersifat *open source*. PHP merupakan *Script* yang terintegrasi dengan HTML dan berada pada server. PHP adalah Script yang digunakan untuk membuat halaman website yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh client. Mekanisme ini menyebabkan informasi yang diterima client selalu yang terbaru atau *up to date*. Semua *Script* PHP dieksekusi pada server dimana script tersebut dijalankan (Putri, 2014).

* + 1. **CSS**

CSS (*Cascading Style Sheet*)merupakan suatu bahasa *style sheet* yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa Markup. CSS bekerja sebagai pelengkap pada HTML dalam memformat dokumen web atau untuk mempercantik tampilan web. Penulisan kode CSS disisipkan pada tag HTML. Kode CSS ditulis dalam tag <style> dan </style> dengan membuat suatu style baru yang kemudian dapat digunakan berulang kali. Penulisan kode CSS dapat langsung pada dokumen HTML atau disimpan dalam dokumen tersendiri kemudian dipanggil untuk digunakan (Augury, 2009).

* + 1. ***Javascript***

*JavaScript* adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*. Client Side Programming Language adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh client. Aplikasi client yang dimaksud adalah web browser seperti Google Chrome dan Mozilla Firefox. (Kabuhung, 2016).

*Javascript* diperkenalkan pertama kali oleh **Netscape** pada tahun 1995. Pada awalnya bahasa ini dinamakan *“LiveScript”* yang berfungsi sebagai bahasa sederhana untuk browser Netscape Navigator 2. Pada masa itu bahasa ini banyak di kritik karena kurang aman, pengembangannya yang terkesan buru buru dan tidak ada pesan kesalahan yang di tampilkan setiap kali membuat kesalahan pada saat menyusun suatu program. Kemudian sejalan dengan sedang giatnya kerjasama antara Netscape dan Sun (pengembang bahasa pemrograman *“Java”* ) pada masa itu, maka Netscape memberikan nama *“JavaScript”* kepada bahasa tersebut pada tanggal 4 desember 1995. Pada saat yang bersamaan Microsoftsendiri mencoba untuk mengadaptasikan teknologi ini yang mereka sebut sebagai *“Jscript”* di browser Internet Explorer 3 (Yatini, 2014).

* + 1. **xampp**

XAMPP server adalah paket web server yang didalamnya terdapat software MySQL Server, Apache web server, PHP, dan tools PHPMyAdmin. Untuk memudahkan dalam mengelola database dalam aplikasi Inventaris ini, maka dibutuhkan tools atau program aplikasi PHPMyAdmin, yaitu aplikasi berbasis web yang dapat dijalankan pada web server. Maka dari itu digunakanlah aplkasi XAMPP Server (Yatini, 2014).

* + 1. **Mysql**

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau yang dikenal dengan DBMS (*Database Management System),* database ini *multithread, multi user.* MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU General Public Licence (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus yang bersifat khusus. Kekuatan MySQL tidak ditopang oleh sebuah komunitas, seperti Apache, yang dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh pemillik masing-masing, tetapi MySQL didukung penuh oleh sebuah perusahaan profesional dan komersial, yakni MySQL AB dari Swedia. MySQL adalah *Relational Database Mangement System* (RDBMS) yang didistribusikan secara gratis di bawah lisensiGPL (General Public License). Di mana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat *closed Source* atau komersial (Miftakhul, 2010).

* + 1. **Bootstrap**

Menurut Husein (2015), *Bootstrap* adalah *framework* ataupun tools untuk membuat aplikasi *web* ataupun *website* yang bersifat *responsive* secara cepat, mudah dan gratis. Kata *responsive* disini berarti bahwa tampilan *web* (lebar dan susunan isinya dapat berubah secara otomatis sesuai dengan lebar layar yang ditampilkannya).

* + 1. **FPDF**

Untuk kepentingan membuat dokumen PDF melalui PHP bisa memanfaatkan FPDF. FPDF adalah suatu pustaka yang dibuat dengan menggunakan bahasa PHP. FPDF menawarkan beberapa keunggulan dan kemudahan dalam membuat dokumen PDF. Beberapa hal yang bisa diperoleh dengan menggunakan pustaka FPDF diantaranya adalah sebagai berikut (Ester Laekha, 2017):

1. Tersedia banyak metode untuk menyusun dokumen PDF.
2. Mendukung penyisipan gambar kedalam dokumen PDF.
3. Terdapat dukungan dari orang-orang lain yang berkontribusi untuk membuat contoh-contoh di luar pustaka yang berguna untuk membuat dokumen PDF (misalnya membuat *barcode*).
   1. ***Use Case* Diagram**

*Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya *login* ke sistem, meng-*create* sebuah daftar belanja, dan sebagainya. Seorang aktor merupakan sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu (Ester, 2017).

Berikut tabel simbol *use case* diagram yang dapat dilihat pada tabel 2.1.

Tabel 2.1 Simbol *Use Case* Diagram

|  |  |
| --- | --- |
| **Simbol** | **Keterangan** |
|  | **Use Case**,Kelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan oleh sebuah aktor. |

(Sumber : Suyasari, 2012)

Tabel 2.2 Simbol *Use Case* Diagram (Lanjutan)

|  |  |
| --- | --- |
|  | **Generalisasi**, Menggambarkan hubungan khusus dalam obyek anak/child yang menggantikan obyek parent/induk. |
|  | **Realisasi**, Merupakan hubungan semantik antara kelompok yang menjamin adanya ikatan diantaranya. |
|  | **Actor**, merupakan kesatuan eksternal yang berinteraksi dengan sistem. |
|  | **System**, Menspesifikasikan paket yang menampilkan sistem secara terbatas. |
|  | **Collaboration**, Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi). |

(Sumber : Suyasari, 2012)

* 1. ***Activity* Diagram**

Pada dasarnya diagram *activity* sering digunakan oleh *flowchart*. Diagram ini berhubungan dengan diagram *Statechart*. Diagram *Statechart* berfokus pada obyek yang dalam suatu proses (atau proses menjadi suatu obyek), diagram *activity* berfokus pada aktifitas-aktifitas yang terjadi yang terkait dalam suatu proses tunggal. Jadi dengan kata lain, diagram ini menunjukkan bagaimana aktifitas-aktifitas tersebut bergantung satu sama lain.

Tabel 2.3 Simbol *Activity* Diagram

|  |  |
| --- | --- |
| **Simbol** | **Keterangan** |
|  | **Initial**, Titik awal, untuk memulai suatu aktivitas. |
|  | **Dependency**, Titik akhir, untuk mengakhiri aktivitas. |
|  | **Activity**, Menandakan sebuah aktivitas. |
|  | **Decision**,Pilihan untuk mengambil keputusan. |

(Sumber : Suyasari, 2012)

Tabel 24 Simbol *Activity Diagram* (Lanjutan)

|  |  |
| --- | --- |
|  | **Fork/ join**,Menunjukan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu. |
|  | **Rake**,Menunjukan adanya dekomposisi. |
|  | **Fork**,Digunakan untuk menunjukkan kegiata yang dilakukan secara parallel. |
|  | **Join**,Digunakan untuk menunjukkan kegiatan yang di gabungkan. |

* 1. ***Sequence* Diagram**

Diagram *sequence* merupakan salah satu diagram Interaction yang menjelaskan bagaimana suatu operasi itu dilakukan; *message* (pesan) apa yang dikirim dan kapan pelaksanaannya. Diagram ini diatur berdasarkan waktu. Obyek-obyek yang berkaitan dengan proses berjalannya operasi diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang terurut.

Tabel 2.5 Simbol *Sequence* Diagram

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Keterangan** |
|  | Actor | Menggambarkan orang yang sedang berinteraksi dengan sistem. |
|  | Entity class | Menggambarkan hubungan kegiatan yang akan dilakukan. |
|  | Boundary class | Menggambarkan sebuah penggambaran dari form. |
|  | Control class | Menggambarkan penghubung antara boundary dengan tabel. |
|  | A focus of control & a life line | Menggambarkan tempat mulai dan berakhirnya sebuah pesan. |
|  | A massage | Menggambarkan pengiriman pesan. |

(Sumber : Suyasari, 2012)

* 1. **Flowchart**

*Flowchart* adalah bagan yang menggambarkan urutan instruksi proses dan hubungan satu proses dengan proses lainnya menggunakan simbol-simbol tertentu. Bagan alir digunakan sebagai alat bantu komunikasi dan dokumentasi. *System flowchart* merupakan bagan yang menunjukkan pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urut-urutan dari prosedur-prosedur yang ada di dalam sistem dan menunjukkan apa saja yang dikerjakan pada sistem. *Document flowchart* atau *form flowchart* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya (Chrystanti, 2011).

Tujuan dari *flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi dan jelas menggunakan simbol-simbol yang standar.

Tabel 2.6 Simbol-Simbol *Flowchart*

|  |  |
| --- | --- |
| **Simbol** | **Fungsi** |
|  | **Terminal**, Menyatakan permulaan atau akhir suatu program. |
|  | **Input** / **Output**, Menyatakan proses input atau output tanpa tergantung jenis peralatannya. |
|  | **Process**,Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer. |
|  | **Manual Operation,** Menunjukkan pengolahan yang tidak dilakukan oleh komputer. |
|  | **Decision**, Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua pilihan jawaban: ya / tidak. |
|  | **Connector**, Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama. |
|  | **Offline Connector**, Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda. |
|  | **Predefined** **Process**, Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal. |
|  | **Punched Card**, Menyatakan input berasal dari kartu atau output ditulis ke kartu. |
|  | **Document**, Mencetak keluaran dalam bentuk dokumen (melalui printer). |

(Sumber : Suyasari, 2012)

Tabel 2.7 Simbol-Simbol *Flowchart* (Lanjutan)

|  |  |
| --- | --- |
|  | **Offline Storage,** Menyatakan bahwa data akan disimpan. |
|  | **Flow**, Menyatakan jalannya arus suatu proses. |
|  | **Magnetik Disk**, Untuk menyimpan data. |

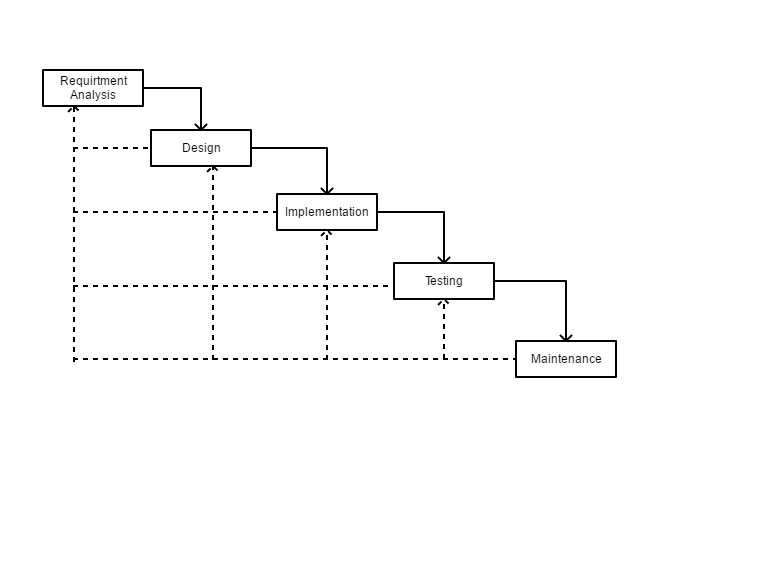
(Sumber : Suyasari, 2012)

**BAB III**

**METODOLOGI PENELITIAN**

* 1. **Metodologi Penelitian**

Metodelogi penelitian dan pengembangan yang digunakan adalah menggunakan metodologi *System Development Life Cycle* (SDLC). Siklus hidup pengembangan Sistem atau SDLC adalah metodologi untuk merancang, membangun, dan memelihara informasi dan proses sistem. Terdapat banyak model SDLC, salah satunya adalah model *Waterfall* yang akan digunakan dalam penelitian ini.



Gambar 3.1 Model *Waterfall* ( Nurasiah, 2014)

Konsep dasar dalam perancangan dan pengembangan sebuah sistem pakar adalah Waterfall SDLC (*System Development Life Cycle*). Tahapan-tahapan Waterfall SDLC (Nurasiah, 2014) :

1. ***Requirtmen Analysis***

*Requirtmen Analysis* atau analisis kebutuhan adalah tahapan menganalisis kebutuhan data, mengumpulkan data yang akan digunakan dan mempersiapkan alat serta bahan penelitian. Persiapan yang baik, akan memudahkan dalam perancangan sistem.

1. ***System Design***

*System design* atau desain sistem merupakan tahapan untuk merancang dan mendesain database, mendesain alur program dan rancangansistem lainnya. Jadi tahapan ini merancang rancangan sistem dan jugamendesign tampilan antarmuka aplikasi. Diagram-diagram yangdigunakan, mencakup pembuatan *use case diagram, activity diagram, sequence diagram,* dan *class diagram*.

1. ***Implementation***

*Implementation* atau implemntasi adalah suatu tahapan sebagai penerapan

pengkodean atau penulisan bahasa program agar aplikasi dapat dijalankan oleh mesin.

1. ***Testing***

*Testing* atau pengujian merupakan tahapan yang dilakukan untuk mengoreksi atau menguji kelayakan dari tahapan sebelumnya, apakah sudah sesuai dengan design yang sudah dibuat atau belum.

1. ***Deployment***

*Deployment* merupakan suatu tahapan untuk memperkenalkan aplikasi yang telah dibuat kepada masyarakat luas, atau kepada mitra yang telah ditentukan sebelumnya.

1. ***Maintenance***

*Maintenance* atau perawatan merupakan tahap akhir pada metode ini, dimana pada tahap ini diperlukan perbaikan atau pengembangan dari aplikasi yang telah dibuat.

* 1. **Metode Pengumpulan Data**
     1. **Studi Pustaka**

Penulis mencari bahan-bahan atau materi sumber yang dapat digunakan sebagai referensi penulis baik dari jurnal ataupun sumber lainnya yang dapat digunakan dalam membuat aplikasi pengelolaan surat di internal polindra berbasis web.

* 1. **Kebutuhan Sistem**

Dalam hal ini mengenai kebutuhan sistem berupa perangkat keras (*hardware*) dan perangkat lunak (*software*) yang dibutuhkan untuk membangun aplikasi pengelolaan surat di internal Polindra berbasis web.

* + 1. **Kebutuhan *Hardware***

Berikut adalah spesifikasi *hardware* pendukung yang digunakan dalam membangun aplikasi ini dapat dilihat pada tabel 3.1.

Tabel 3.1 Kebutuhan *Hardware*

|  |  |  |
| --- | --- | --- |
| **No** | **Jenis *Hardware*** | **Kebutuhan *Hardware*** |
| 1 | Laptop/PC | Intel. |
| 2 | Processor | 2.0 GHz *up to above.* |
| 3 | ram | 2 GB *up to above.* |
| 4 | Hard Disk | 500 GB *up to above.* |

* + 1. **Kebutuhan *Software***

Berikut adalah spesifikasi *software* pendukung yang digunakan dalam penelitian ini dapat dilihat pada tabel 3.2.

Tabel 3.2 Kebutuhan *Software*

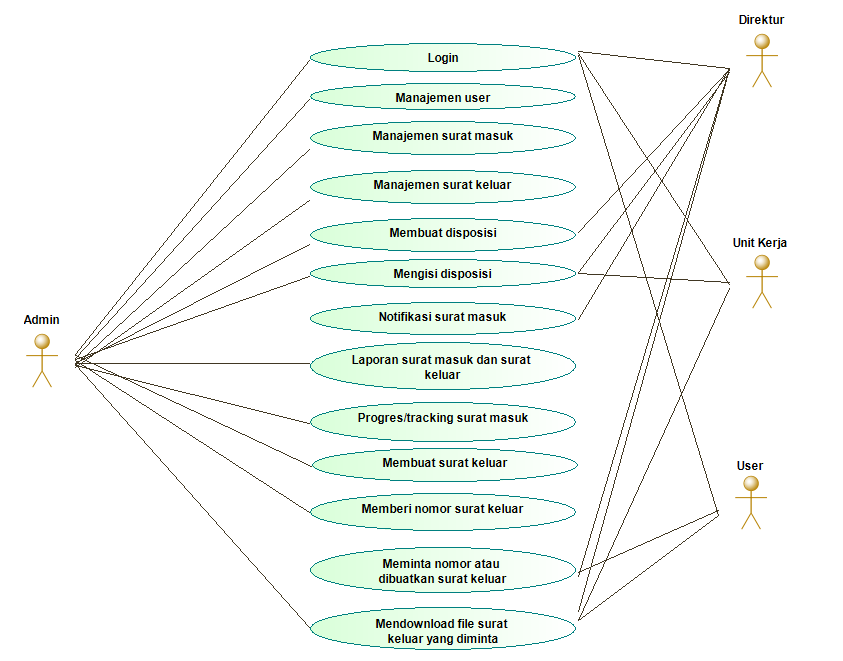
|  |  |  |
| --- | --- | --- |
| No | Jenis *Software* | Kebutuhan *Software* |
| 1 | Sistem Operasi | Windows 8.1 *Pro up to above.* |
| 2 | Text Editor | Sublime Text *version* 3.0. |
| 3 | Local Server | Xampp Versi 3.2.2. |
| 4 | Browser | Google Chrome dan Mozila Firefox. |

* 1. **Perancangan Sistem**

Pada perancangan sistem ini menggambarkan garis besar mengenai aplikasi pengelolaan surat di internal polindra berbasis web.

1. **Use Case Diagram**

*Use Case* Diagram merupakan pemodelan yang digunakan untuk menggambarkan tingkah laku yang dapat dikerjakan oleh sistem. Berikut ini adalah *use case* diagram dari aplikasi pengelolaan surat di internal Polindra berbasis web dapat dilihat pada gambar 3.2.



Gambar 3.2 *Use Case* Diagram

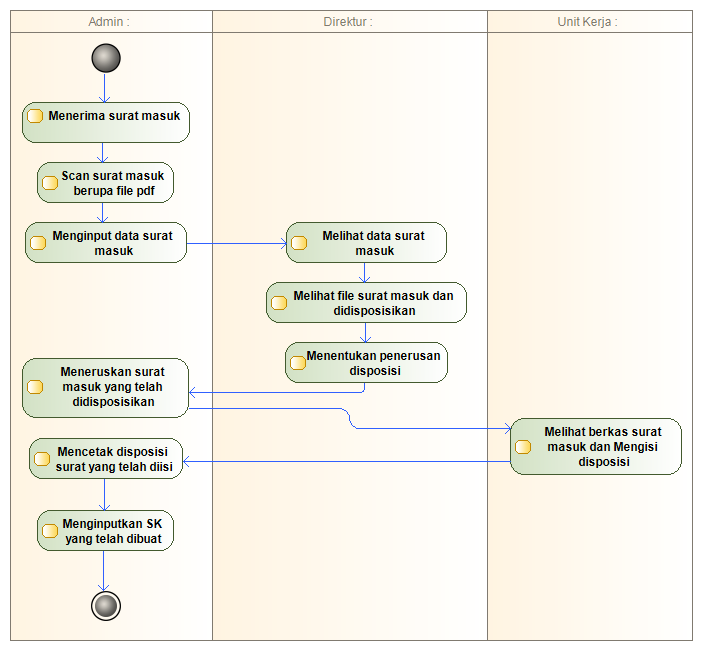
Adapun keterangan *use case* diagram akan dijelaskan pada tabel 3.12.

Tabel 3.3 Keterangan *Use Case* Diagram

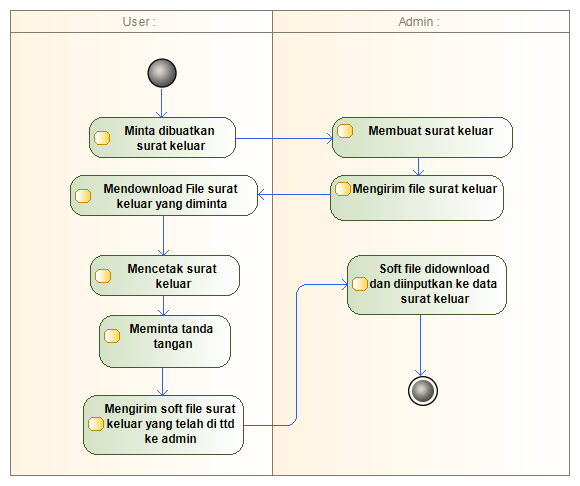
|  |  |
| --- | --- |
| Nama Use Case | Keterangan Use Case |
| Login | Admin, direktur, unit kerja, dan user dapat melakukan login dengan mengisiskan username dan password. |
| Manajemen User | Admin dapat mengelola akun user dengan menambah, mengedit, dan menghapus. |
| Manajemen Surat Masuk | Admin dapat mengelola data surat masuk dengan menambah, mengedit, dan menghapus. |
| Manajemen Surat Keluar | Admin dapat mengelola surat keluar dengan menambah, mengedit, dan menghapus. |
| Membuat Disposisi | Admin dan direktur dapat membuat disposisi surat masuk. |
| Mengisi Disposisi | Admin dan unit kerja dapat mengisi disposisi. |
| Notifikasi Surat Masuk | Direktur mendapatkan notifikasi surat masuk dari data yang admin inputkan untuk dibuatkan disposisinya. |
| Laporan surat masuk dan surat keluar | Admin dapat mengekspor laporan surat masuk dan surat keluar dalam bentuk excel. |
| Progress/Tracking Surat Masuk | Admin dapat melihat progress surat masuk sudah sampai mana. |
| Membuat Surat Keluar | Admin dapat membuat surat keluar pada aplikasi web ini yang diekspor dalam bentuk file pdf. |
| Memberi Nomor Surat Keluar | Admin dapat memberi nomor surat keluar. |
| Meminta Nomor Atau  Dibuatkan Surat Keluar | Direktur, unit kerja, dan user dapat meminta nomor surat keluar atau dibuatkan. |
| Mendownload File Surat Keluar yang diminta | Direktur, unit kerja, dan user dapat mendownload file surat yang diminta. |

* + 1. **Activity Diagram**

Activity Diagram merupakan gambaran aktifitas-aktifitas *user* dalam berinteraksi dengan sistem. Berikut ini *activity* diagram pada aplikasi pengelolaan surat di Internal Polindra berbasis web. Dalam pembuatan activity diagram ini bertujuan untuk membantu memahami proses yang terjadi pada sistem secara berurutan yang dapat dilihat pada gambar 3.3.



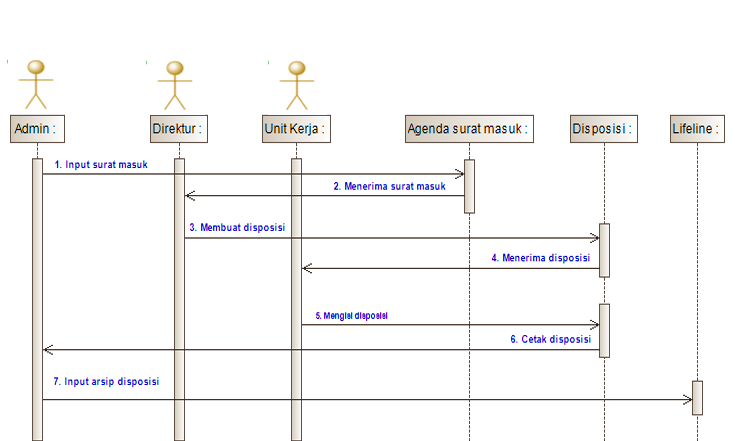
Gambar 3.3 Activity Diagram Surat Masuk



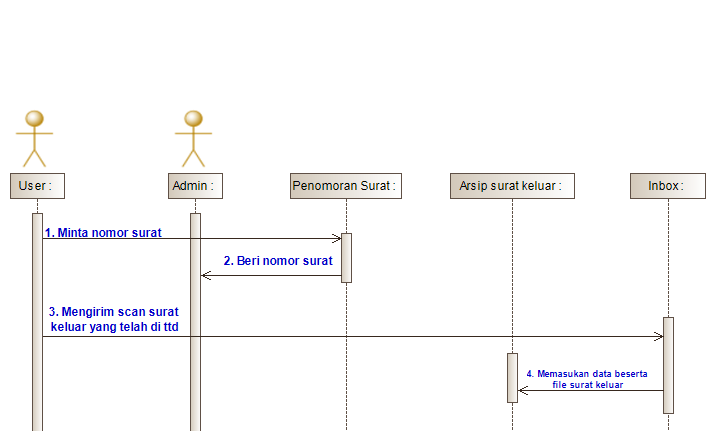
Gambar 3.4 Activity Diagram Surat Keluar

* + 1. **Sequence Diagram**

Sequence diagram merupakan serangkaian interaksi yang menjelaskan bagaimana suatu operasi itu dilakuakn, obyek-obyek yang berkaitan dengan proses berjalannya opersai diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang berurutan dapat dilihat pada gambar 3.4.



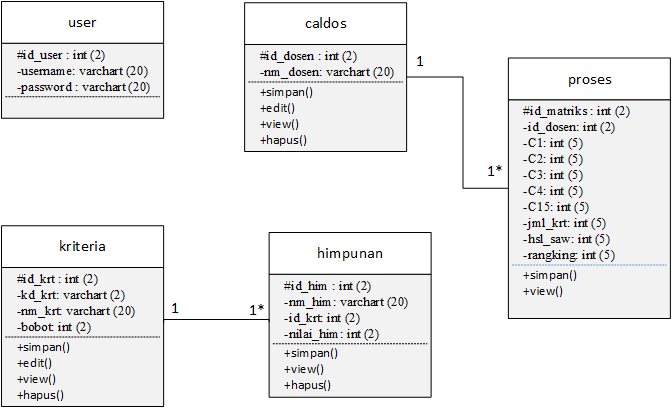
Gambar 3.4 Sequence Diagram Surat Masuk



Gambar 3.4 Sequence Diagram Surat Keluar

* + 1. ***Class* Diagram**

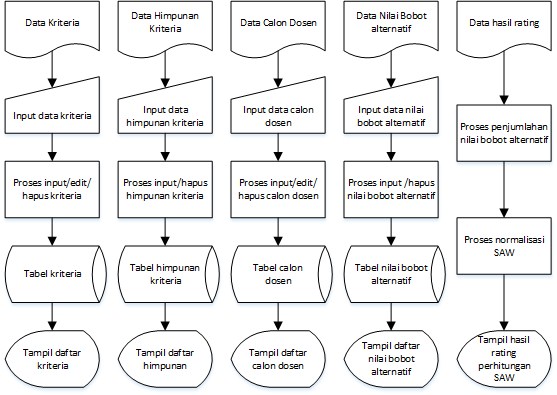
*Class* Diagrammenunjukkan interaksi antar kelas dalam sistem. *Class* diagram juga menampilkan struktur dari sebuah sistem tersebut akan menampilkan sistem kelas, atribut, dan hubungan antar kelas ketika suatu sistem telah selesai membuat diagram. Adapun *class* diagram dari aplikasi ini dapat dilihat pada gambar 3.5.



**Gambar 3.5 *Class Diagram***

* + 1. **Flowcahart Sistem**

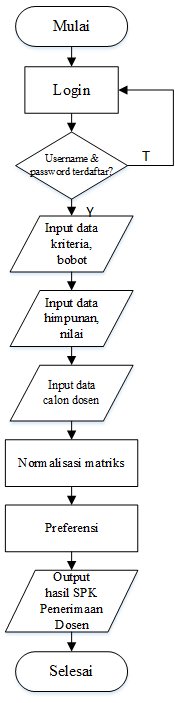
Flowchart Sistem terdiri dari data yang mengalir melalui sistem dan proses yang mentransformasikan data itu. Data dan proses dalam flowchart sistem dapat digambarkan secara online (dihubungkan langsung dengan komputer) atau offline (tidak dihubungkan langsung dengan komputer, misalnya mesin tik, cash register atau kalkulator). Berikut ini adalah *flowchart* sistem yang dapat dilihat pada gambar 3.6.

****

Gambar 3.6 Flowchart Sistem

* + 1. **Flowchart Program**

*Flowchart* program merupakan keterangan yang lebih rinci tentang bagaimana setiap langkah prosedur yang dilakukan oleh pengguna aplikasi ini. *Flowchart* ini menunjukan setiap langkah prosedur dalam urutan yang tepat saat terjadi. Berikut ini adalah *flowchart* program yang dapat dilihat pada gambar 3.7.



Gambar 3.7 Flowchart Program

* 1. **Perancangan *Database***

Berikut ini merupakan perancangan dari tabel-tabel dalam *database* pembuatan SPK seleksi penerimaan dosen dengan metode Simple Additive Weigthing (SAW) :

* + 1. **Tabel User**

Berikut ini dari tabel user, dapat dilihat pada tabel 3.13.

Tabel 3.13 Tabel User

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama field | Tipe data | keterangan |
| 1 | Id\_user | Int (5) | Primary key |
| 2 | Username | Varchar (20) |  |
| 3 | Password | Varchar (20) |  |

* + 1. **Tabel Kriteria**

Berikut ini dari tabel kriteria, dapat dilihat pada tabel 3.14.

Tabel 3.14 Tabel Kriteria

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama field | Tipe data | Keterangan |
| 1 | Id\_krt | Int (5) | Primary key |
| 2 | Kd\_krt | Varchar (20) |  |
| 3 | Nm\_krt | Varchar (20) |  |
| 4 | Bobot | Int (5) |  |

* + 1. **Tabel Himpunan**

Berikut ini dari tabel himpunan, dapat dilihat pada tabel 3.15.

Tabel 3.15 Tabel himpunan

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe data | Keterangan |
| 1 | Id\_him | Int (5) | Primary key |
| 2 | Id\_krt | Int (5) |  |
| 3 | Nm\_him | Varchar (25) |  |
| 4 | Nilai\_him | Int (5) |  |

* + 1. **Tabel Caldos**

Berikut ini dari tabel caldos, dapat dilihat pada tabel 3.16.

Tabel 3.16 Tabel Caldos

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe Data | Keterangan |
| 1 | Id\_dosen | Int (5) | Primary key |
| 2 | Nik\_dosen | Int (10) |  |
| 3 | Nm\_dosen | Varchar (20) |  |
| 4 | Tempat\_lahir | Varchar (20) |  |
| 5 | Tgl\_lahir | Date |  |
| 6 | Penempatan | Varchar (30) |  |

* + 1. **Tabel Proses**

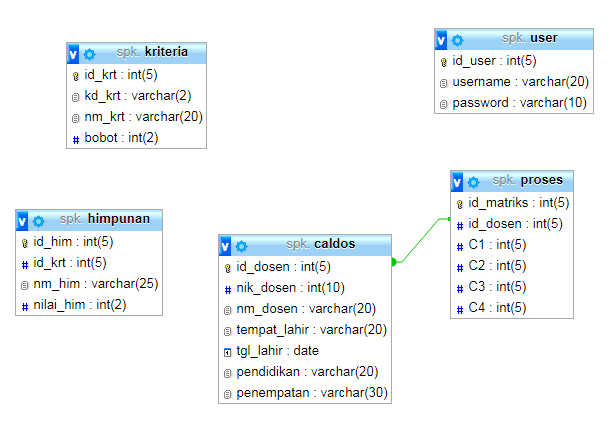
Berikut ini dari tabel proses, dapat dilihat pada tabel 3.17.

Tabel 3.17 Tabel Proses

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe Data | Keterangan |
| 1 | Id\_matriks | Int (5) | Primary key |
| 2 | Id\_dosen | Int (5) |  |
| 3 | C1 | Int (5) |  |
| 4 | C2 | Int (5) |  |
| 5 | C3 | Int (5) |  |
| 6 | C4 | Int (5) |  |

* + 1. **Relasi *Database***

Berikut gambar relasi atau hubungan antar tabel *database* yang digunakan untuk membangun aplikasi ini.



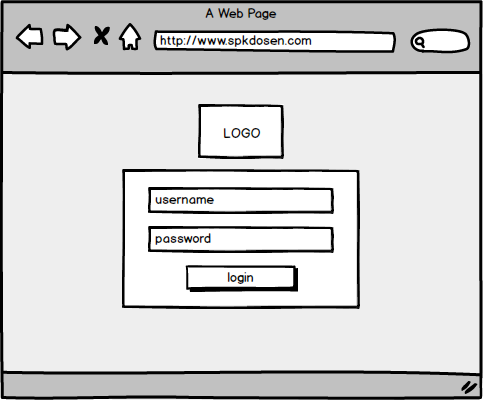
Gambar 3.8. Relasi Tabel

* 1. **Perancangan Antar Muka**

Ada beberapa rancangan dari aplikasi SPK penerimaan dosen berbasis web yang akan dijelaskan pada bagian sub bab berikut.

* + 1. **Rancangan Halaman Login**

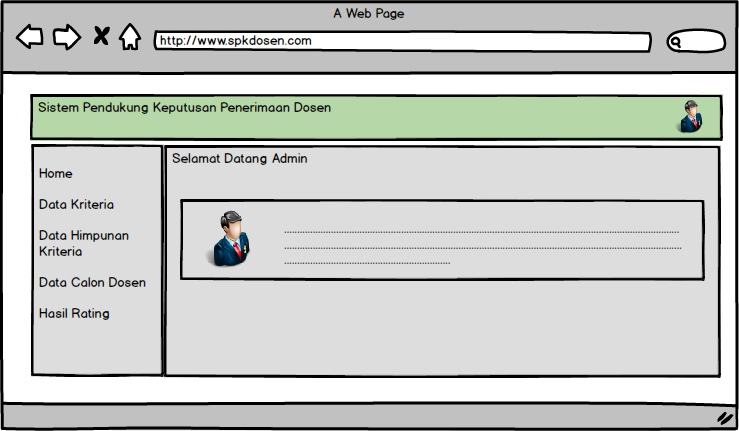
Tampilan ketika pengguna admin mengakses website SPK penerimaan dosen maka halaman ini yang akan ditampilkan petama kali. Berikut rancangan halaman login dapat dilihat pada gambar 3.9.

****

Gambar 3.9 Rancangan Halaman Login

* + 1. **Rancangan Halaman *Home***

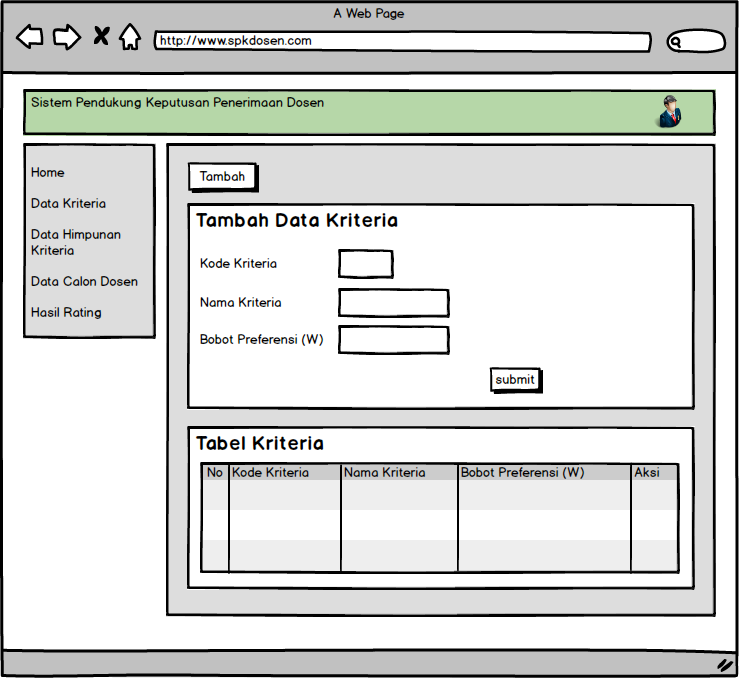
Setelah admin melakukan akses login maka tampilan selanjutnya adalah halaman *home*. Berikut rancangan halaman *home* dapat dilihat pada gambar 3.10.

****

Gambar 3.10 Rancangan Halaman *Home*

* + 1. **Rancangan Halaman Kriteria**

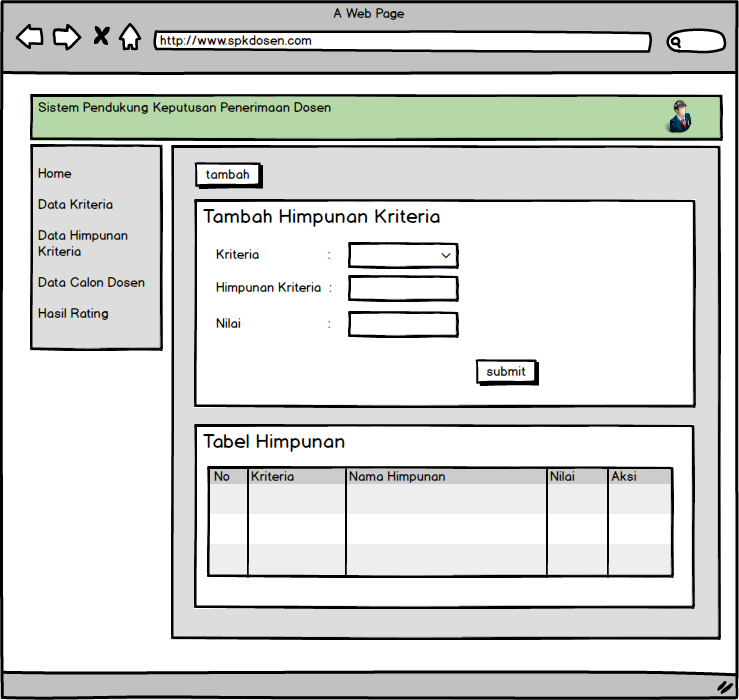
Pada halaman kriteria terdapat form untuk untuk menginputkan kriteria-kriteria apa saja yg dibutuhkan dalam penyeleksian penerimaan dosen serta pemberian bobot untuk masing-masing kriteria tersebut dapat dilihat pada gambar 3.11.

****

Gambar 3.11 Rancangan Halaman Kriteria

* + 1. **Rancangan Halaman Himpunan Kriteria**

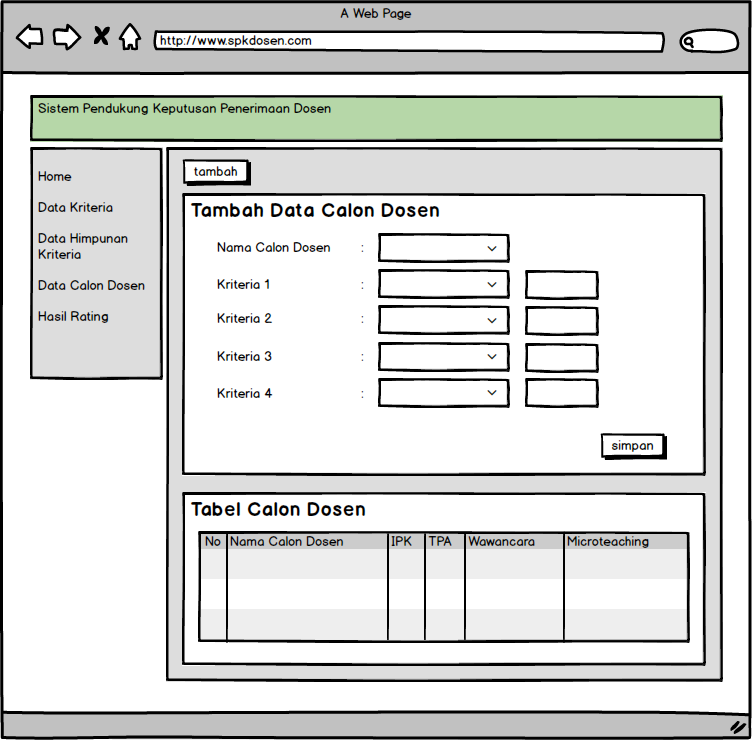
Pada halaman himpunan kriteria terdapat form untuk memasukkan himpunan kriteria apa saja yg dibutuhkan dalam penyeleksian penerimaan dosen. Tampilan menu himpunan kriteria ditunjukkan pada gambar 3.12.

****

Gambar 3.12 Rancangan Halaman Himpunan Kriteria

* + 1. **Rancangan Halaman Data Calon Dosen**

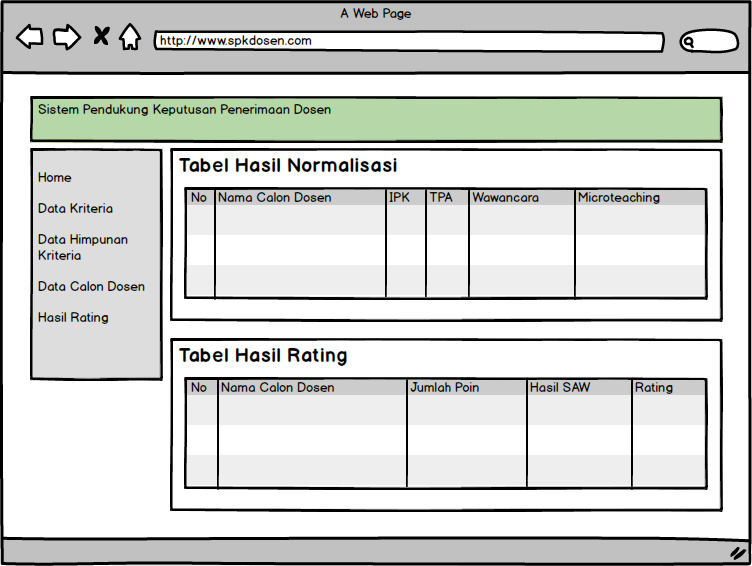
Pada halaman alternatif terdapat form untuk memasukkan nama calon dosen serta menentukan penilaian pada tiap-tiap kriteria. Tampilan menu alternatif penilaian ditunjukkan pada gambar 3.13.

****

Gambar 3.13 Rancangan Halaman Calon Dosen

* + 1. **Rancangan Halaman Hasil Rating**

Pada halaman hasil rating berisikan data dari hasil perhitungan dengan menggunakan metode saw yang menghasilkan urutan rating dari calon dosen. Berikut gambar halaman hasil rating ditunjukkan pada gambar 3.14.

****

Gambar 3.14 Rancangan Halaman Hasil Rating

**BAB IV**

**HASIL DAN PEMBAHASAN**

* 1. **Hasil**

Hasil dari penelitian ini menghasilkan sebuah aplikasi sistem pendukung keputusan seleksi penerimaan dosen dengan menggunkaan metode *Simple Additive Weigthing.* Hasil akhir dari penelitian adalah terletak pada bagaimana hasil implementasi dari metode SAW pada studi kasus penerimaan dosen dengan menggunkan beberapa kriteria yang telah ditentukan. Metode Saw digunakan untuk menentukan solusi pelamar atau dosen yang lolos seleksi.

* 1. **Pembahasan**

Seelah mengumpulkan, menganalisis dan merancang, maka penulis mengimplemntasikan semuanya untuk terciptanya apleikasi sistem pendukung keputusan seleksi penerimaan dosen dengan menggunakan metode *Simple Additive Weighting* (saw)

* + 1. **Implementasi Basis Data**

Pembuatan *database* SPK seleksi penerimaan dosen dengan metode SAW terdiri dari 5 tabel yang terdiri dari tabel caldos, tabel kriteria, tabel himpunan, tabel proses dan tabel user. Berikut implementasi pembuatan basis data untuk SPK seleksi penerimaan dosen dengan metode SAW :

* + 1. ***Database* one403024**

*Database* one403024 adalah nama *database* yang sudah *default* berdasarkan *username server. Database* oneone403024 merupakan sebuah tempat untuk menampung tabel-tabel *database* yang dibuat.

Berikut *query* membuat *database-nya:*

CREATE DATABASE

* + 1. **Tabel Caldos**

Tabel caldos berfungsi untuk menampung data-data yang berkaitan dengan data pelamar atau calon dosen. Berikut query untuk membuatnya:

CREATE TABLE IF NOT EXISTS `caldos` (

`id\_dosen` int(5) NOT NULL,

`nik\_dosen` int(10) NOT NULL,

`nm\_dosen` varchar(20) NOT NULL,

`tempat\_lahir` varchar(20) NOT NULL,

`tgl\_lahir` date NOT NULL,

`pendidikan` varchar(20) NOT NULL,

`penempatan` varchar(30) NOT NULL,

)

* + 1. **Tabel Kriteria**

Tabel kriteria berfungsi untuk menampung data-data yang berkaitan dengan data kriteria seleksi dosen. Berikut query untuk membuatnya:

CREATE TABLE IF NOT EXISTS `kriteria` (

`id\_krt` int(5) NOT NULL,

`kd\_krt` varchar(2) NOT NULL,

`nm\_krt` varchar(20) NOT NULL,

`bobot` int(2) NOT NULL

)

* + 1. **Tabel Himpunan Kriteria**

Tabel himpunan kriteria berfungsi untuk menampung data-data yang berkaitan dengan data himpunan kriteria seleksi dosen. Berikut query untuk membuatan.

CREATE TABLE IF NOT EXISTS `himpunan` (

`id\_him` int(5) NOT NULL,

`id\_krt` int(5) NOT NULL,

`nm\_him` varchar(25) NOT NULL,

`nilai\_him` int(2) NOT NULL

)

* + 1. **Tabel Proses**

Tabel proses berfungsi untuk menampung data-data dasil inputan form calon dosen beserta nilai masing-masing kriteria. Berikut query untuk membuatnya:

CREATE TABLE IF NOT EXISTS `proses` (

`id\_matriks` int(5) NOT NULL,

`id\_dosen` int(5) NOT NULL,

`C1` int(5) NOT NULL,

`C2` int(5) NOT NULL,

`C3` int(5) NOT NULL,

`C4` int(5) NOT NULL

)

* + 1. **Tabel User**

Tabel user berfungsi untuk menampung data yang berkaitan dengan *username* dan *password*. Berikut query untuk membuatnya:

`id\_user` int(5) NOT NULL,

`username` varchar(20) NOT NULL,

`password` varchar(10) NOT NULL

)

* + 1. **Implementasi Desain Interface**

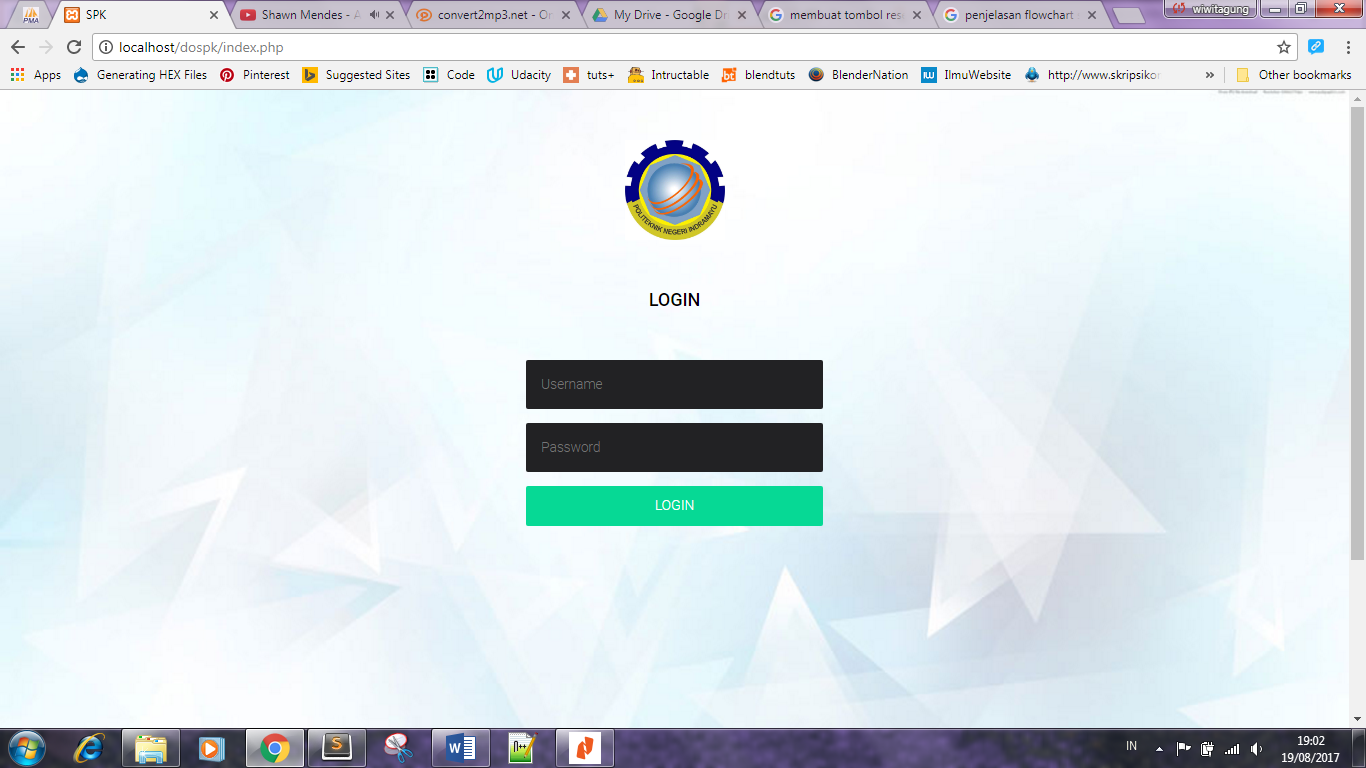
Tahap ini akan mengimplementasikan desain yang telah dirancang pada

bab sebelumnya. Desain halaman *form* dibuat menggunakan *sublime text 2.*

Antarmuka atau *interface* dapat memudahkan *user* dalam melakukan operasi-operasi prosedural tanpa harus berurusan dengan *coding-coding* program yang rumit. Untuk *sintaks* program sendiri dapat dilihat pada lampiran. Berikut tampilan *interface* dalam aplikasi SPK seleksi penerimaan dosen dengan metode SAW *:*

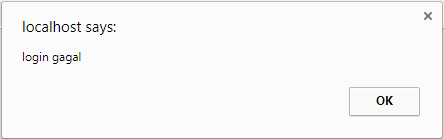
1. **Tampilan login aplikasi SPK seleksi penerimaan dosen metode SAW**

Halaman login berisi *username* dan *password* yang harus di isikan oleh admin untuk melanjutkan halaman selanjutnya. Jika username dan password yang di inputkan benar maka menampilkan halaman home.



Gambar 4.1 Tampilan Login

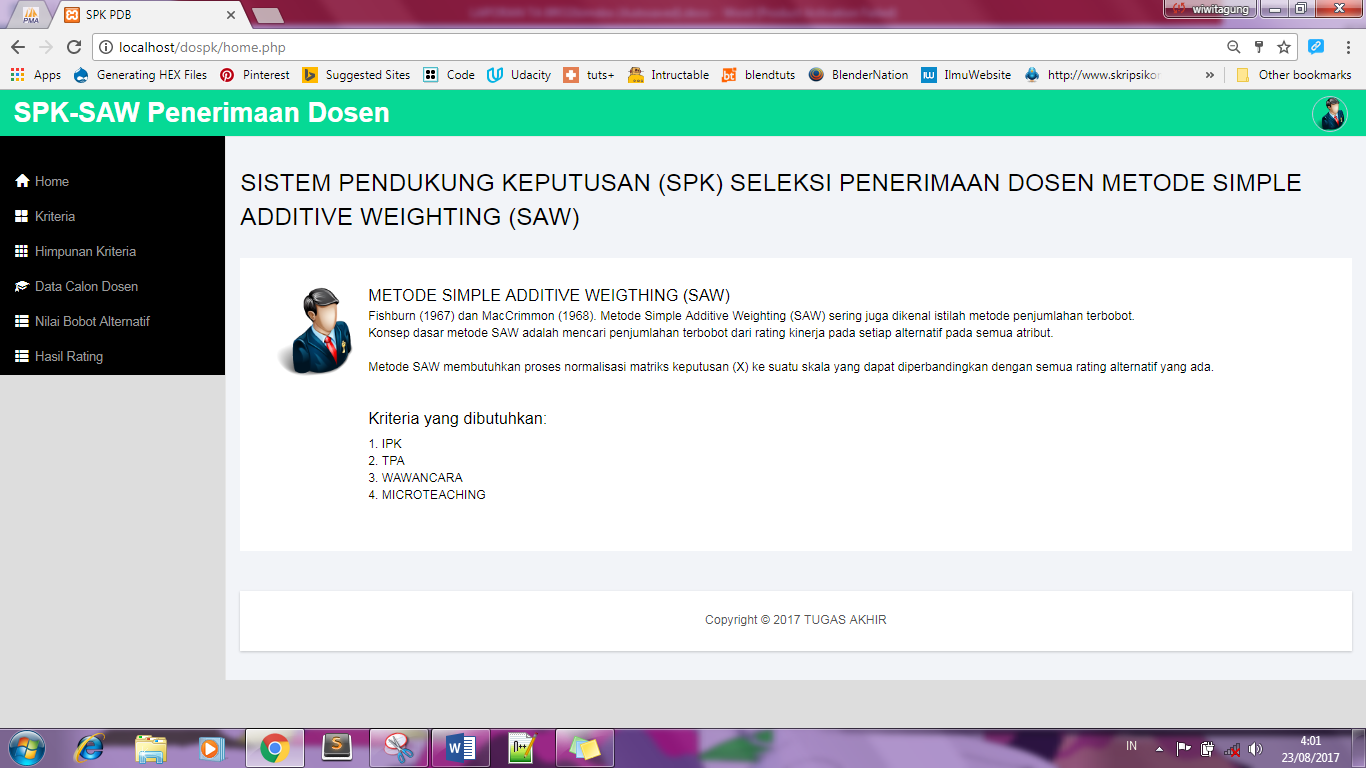
Jika *username* dan *password* yang di inputkan salah maka akan keluar *pop up* bahwa login gagal dilakukan yang dapat dilihat pada gambar 4.2.



Gambar 4.2 Tampilan *Pop Up Login* Gagal

1. **Tampilan Home**

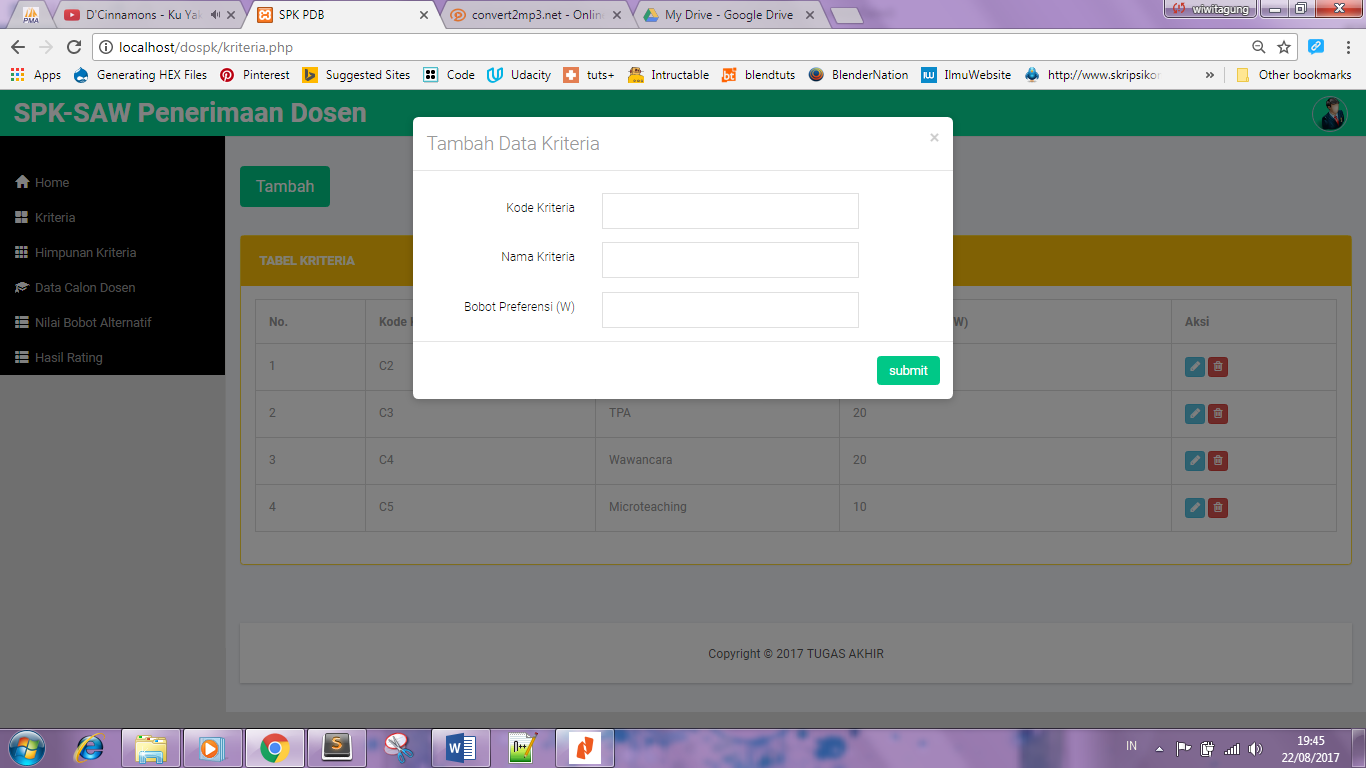
Halaman *home* berisi menu kriteria, menu himpunan kriteria, menu data calon dosen, menu nilai bobot alternatif dan menu hasil rating



Gambar 4.3 Tampilan *Home*

1. **Tampilan Kriteria**

Halaman kriteria terdapat tombol tambah jika diklik akan menampilkan form tambah kriteria dengan isi form inputan berupa kode kriteria, nama kriteria dan bobot preferensi atau bobot kriteria tersebut. Setelah melakukan inputan form kriteria klik tombol *submit* untuk menyimpan data tersebut yang dapat dilihat pada gambar 4.4.



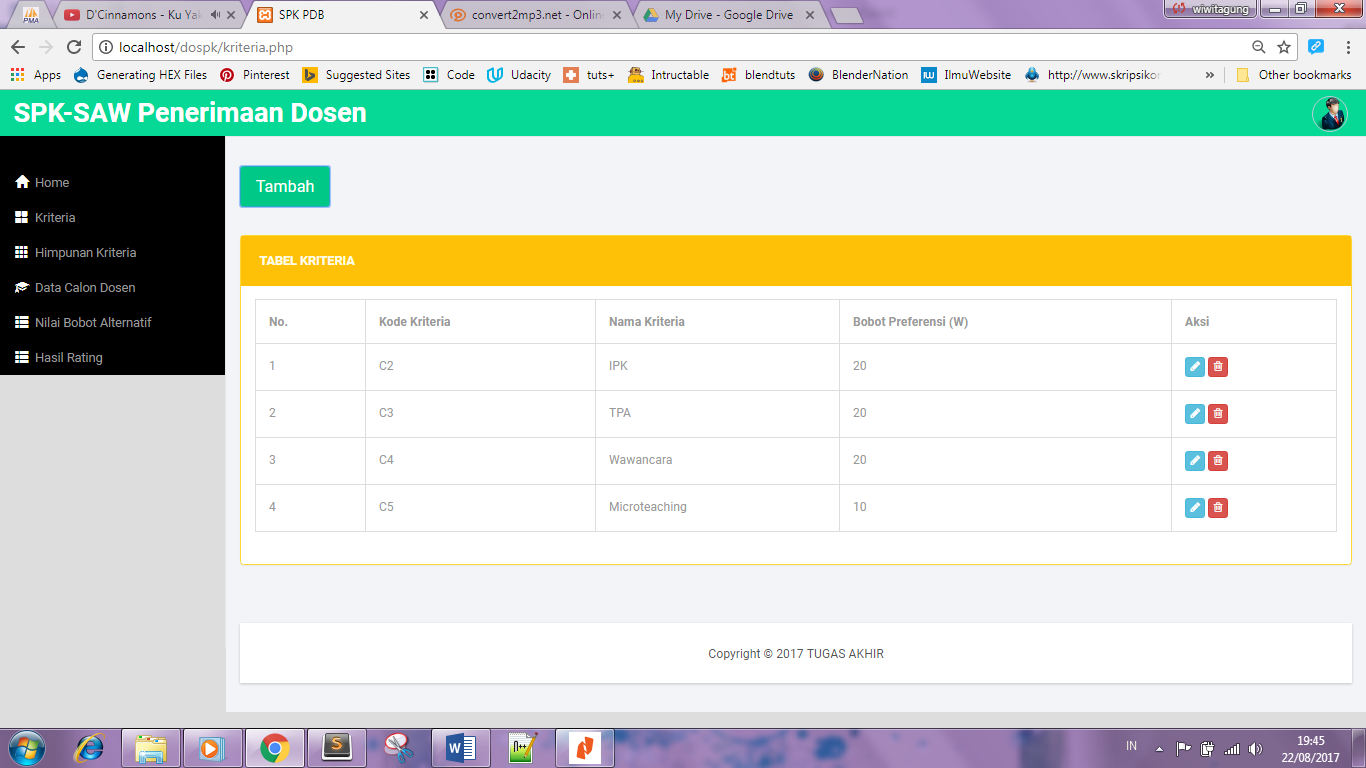
Gambar 4.4 Tampilan Form Tambah Kriteria

Setelah pilih tombol *submit* selanjutnya tampilan akan berganti dengan menampilkan pop up bahwa data yang inputkan itu berhasil disimpan/tambahkan halaman yang berisikan tabel hasil inputan dari form tadi yang dapat dilihat pada gambar 4.5.



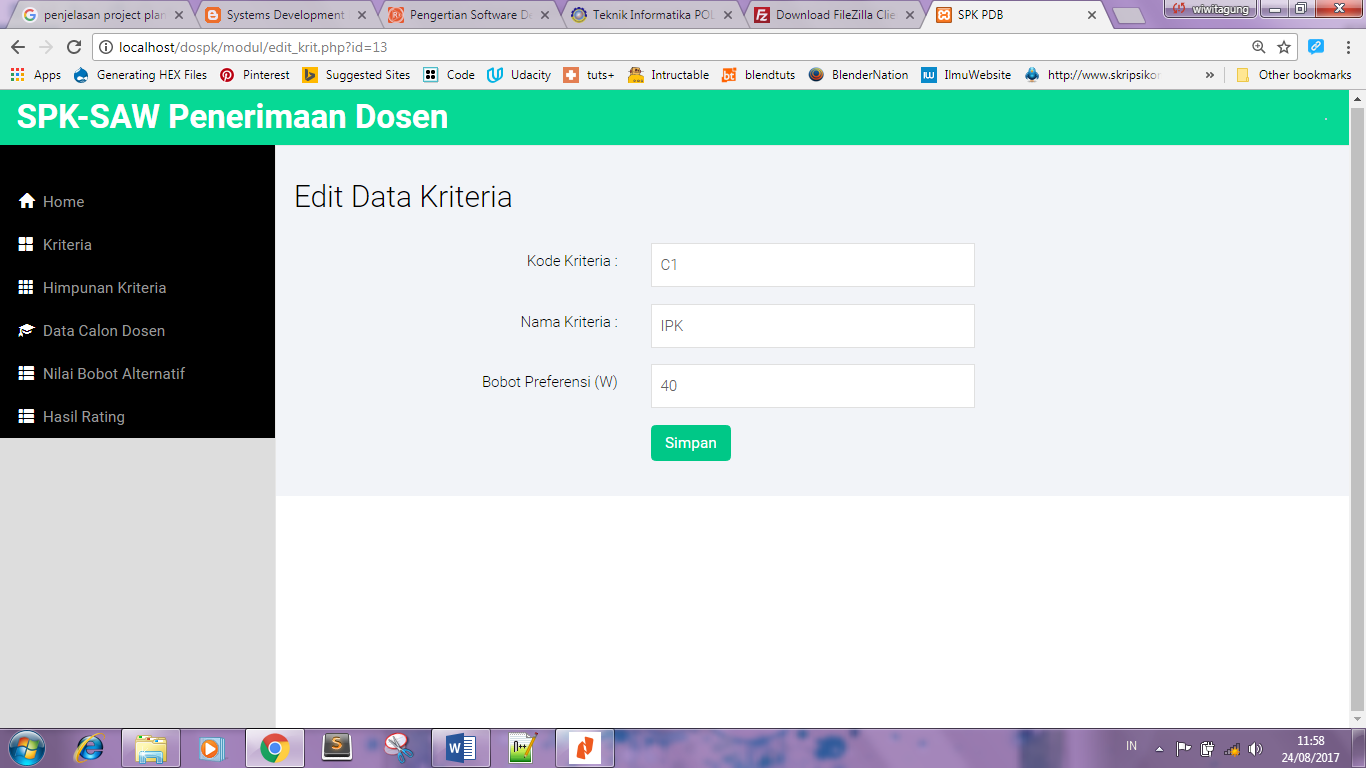
Gambar 4.5 Tampilan Pop Up Data Berehasil Ditambahkan

Selanjutkan pilih tombol **ok** maka halaman yang akan ditampilkan adalah tabel kriteria yang berisikan data dari hasil inputan tadi. Pada tabel kriteria terdapat pula *button* edit dan hapus yang dapat dilihat pada gambar 4.6.



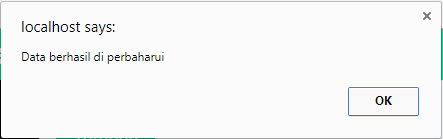
Gambar 4.6 Tampilan Tabel Kriteria

Jika ingin melakukan edit data kriteia, maka pilih tombol edit yang bergambar pensil yang menmpilkan form edit data kriteria yang dapat dilihat pada gambar 4.7.



Gambar 4.7 Tampilan Halaman Edit Kriteria

Jika data berhasil di edit maka akan muncul pop up bahwa data telah diperbaharui yang dapat dilihat pada gambar 4.8.



Gambar 4.8 tampilan pop up berhasil diperbaharui

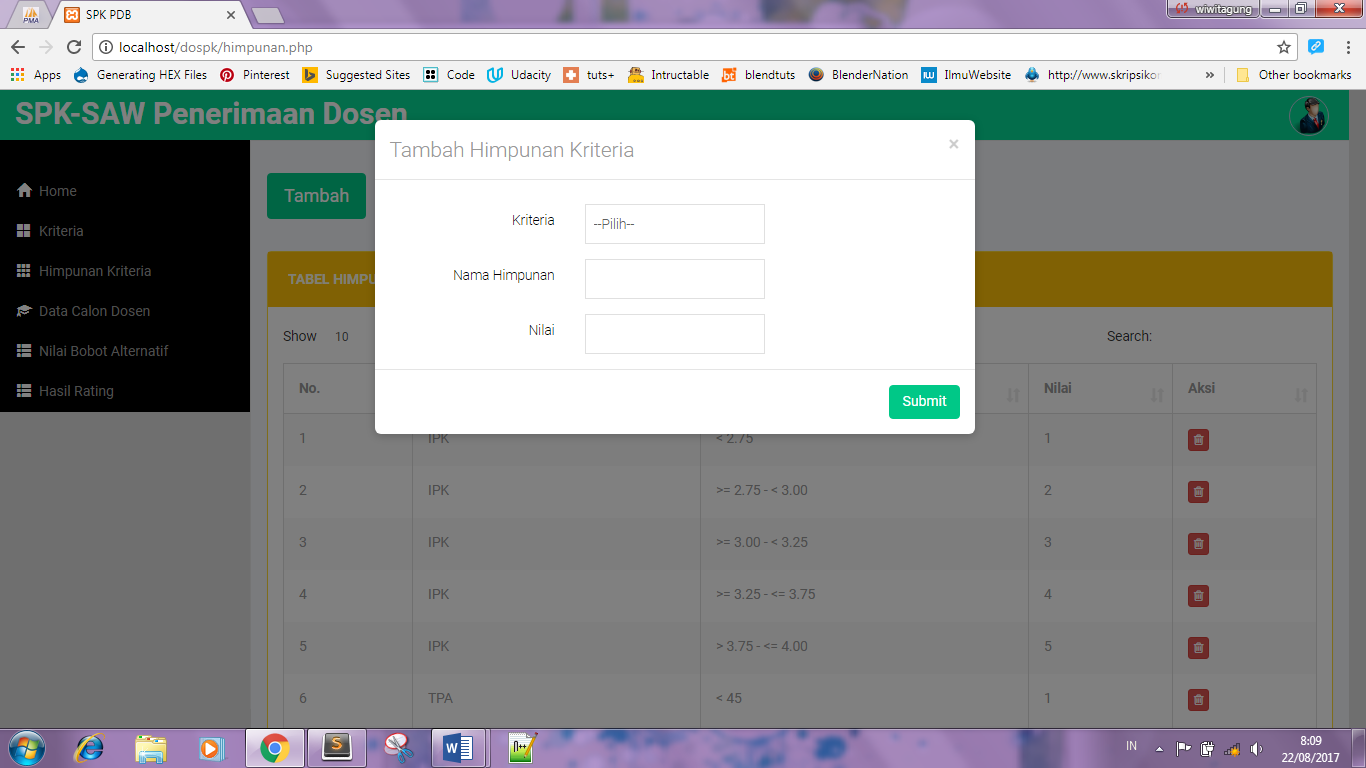
Jika ingin melakukan hapus data kriteria, pilih button dengan gambar tempat sampah. Bila berhasil akan menampilkan pop up bahwa data berhasil di hapus yang dapat dilihat pada gambar 4.9.



Gambar 4.9 Tampilan Pop Up Data Berhasil Dihapus

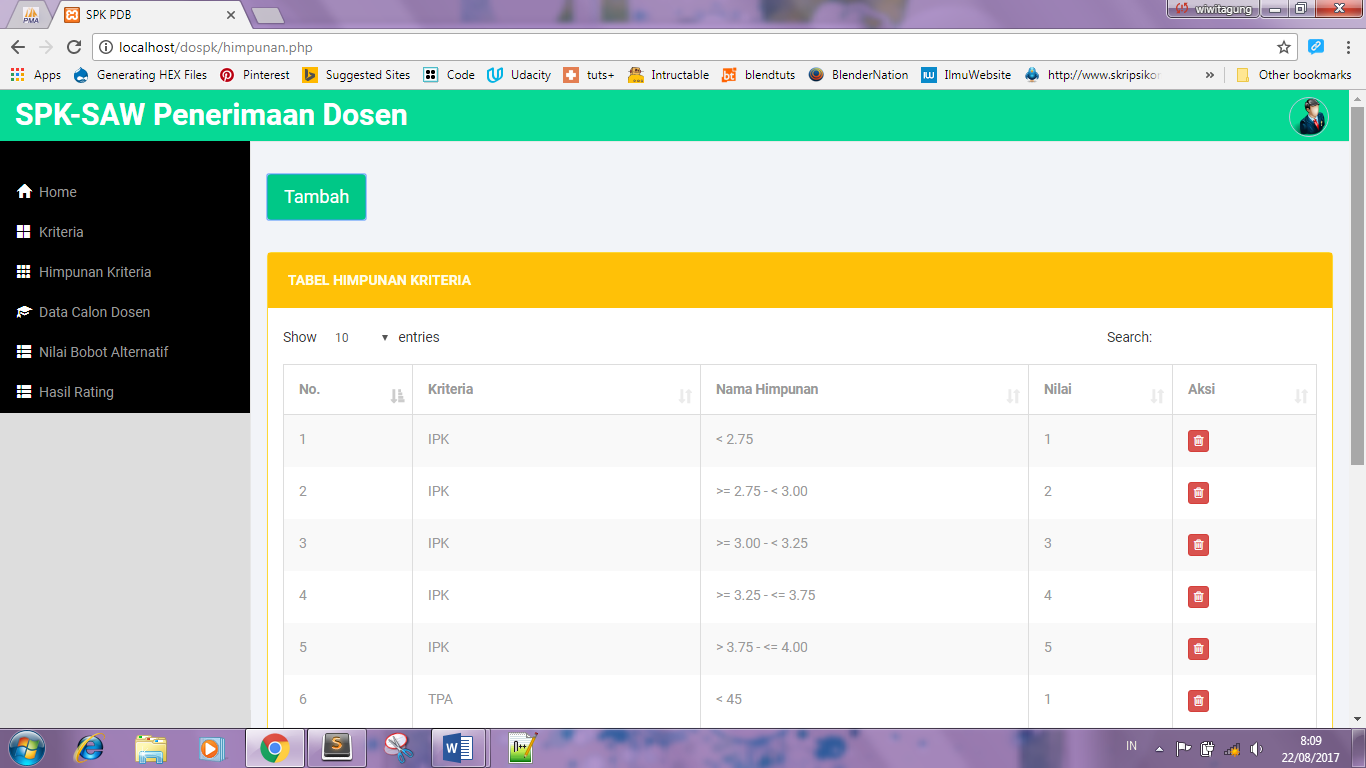
1. **Tampilan Himpunan Kriteria**

Halaman himpunan kriteria terdapat tombol tambah jika diklik akan menampilkan form tambah himpunan kriteria dengan isi form inputan berupa nama kriteria, nama himpunan dan nilai himpunan kriteria tersebut. Setelah melakukan inputan form himpunankriteria klik tombol *submit* untuk menyimpan data tersebut yang dapat dilihat pada gambar 4.10.



Gambar 4.10 Tampilan Form Tambah Himpunan Kriteria

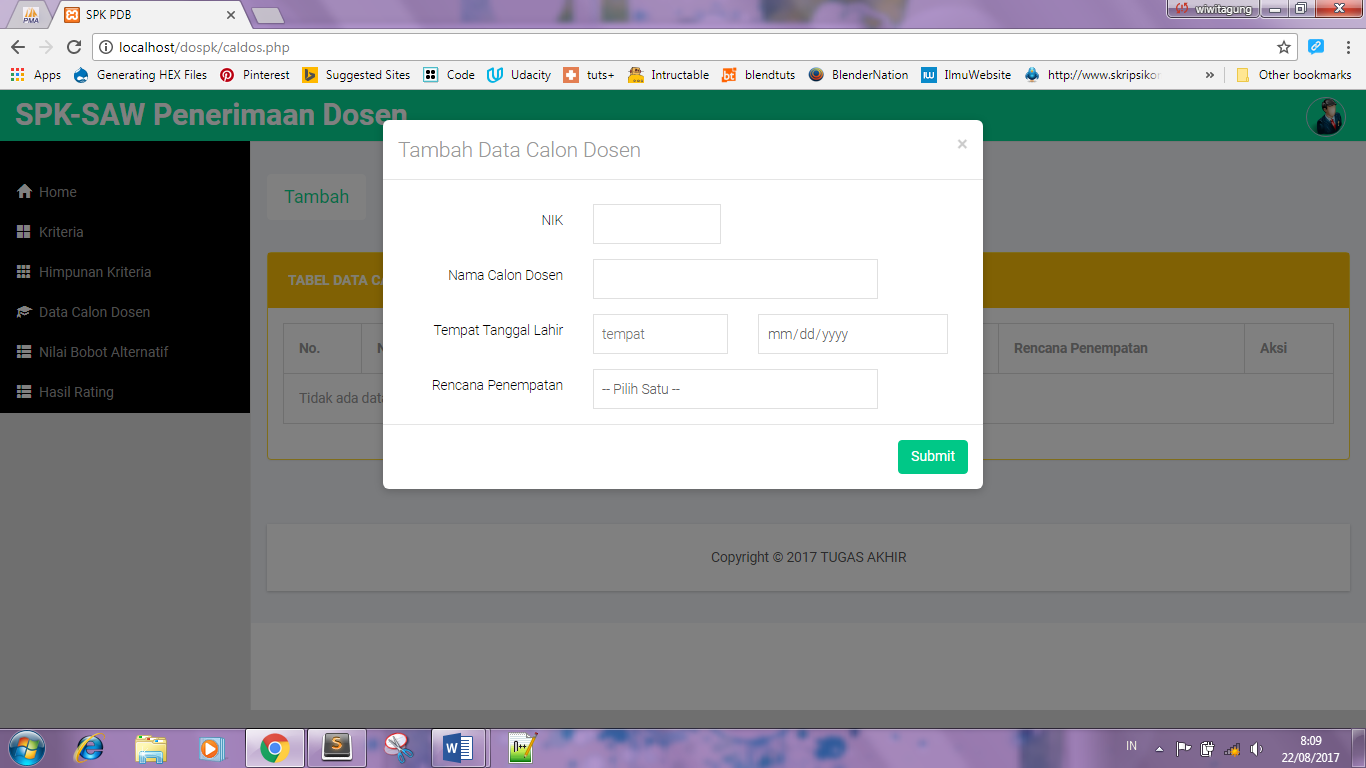
Setelah pilih tombol *submit* selanjutnya tampilan akan berganti dengan menampilkan halaman yang berisikan tabel hasil inputan dari form tadi. Pada tabel himpunan kriteria hanya terdapat *button* hapus yang dapat dilihat pada gambar 4.11.



Gambar 4.11 Tampilan Tabel Himpunan Kriteria

1. **Tampilan Data Calon Dosen**

Halaman data calon dosen terdapat tombol tambah jika diklik akan menampilkan form tambah data calon dosen dengan isi form inputan berupa NIK, nama calon dosen, tempat lahir, tanggal lahir, dan rencana penempatan. Setelah melakukan inputan form data calon dosen klik tombol *submit* untuk menyimpan data tersebut yang dapat dilihat pada gambar 4.12.

Gambar 4.12 Tampilan Form Data Calon Dosen

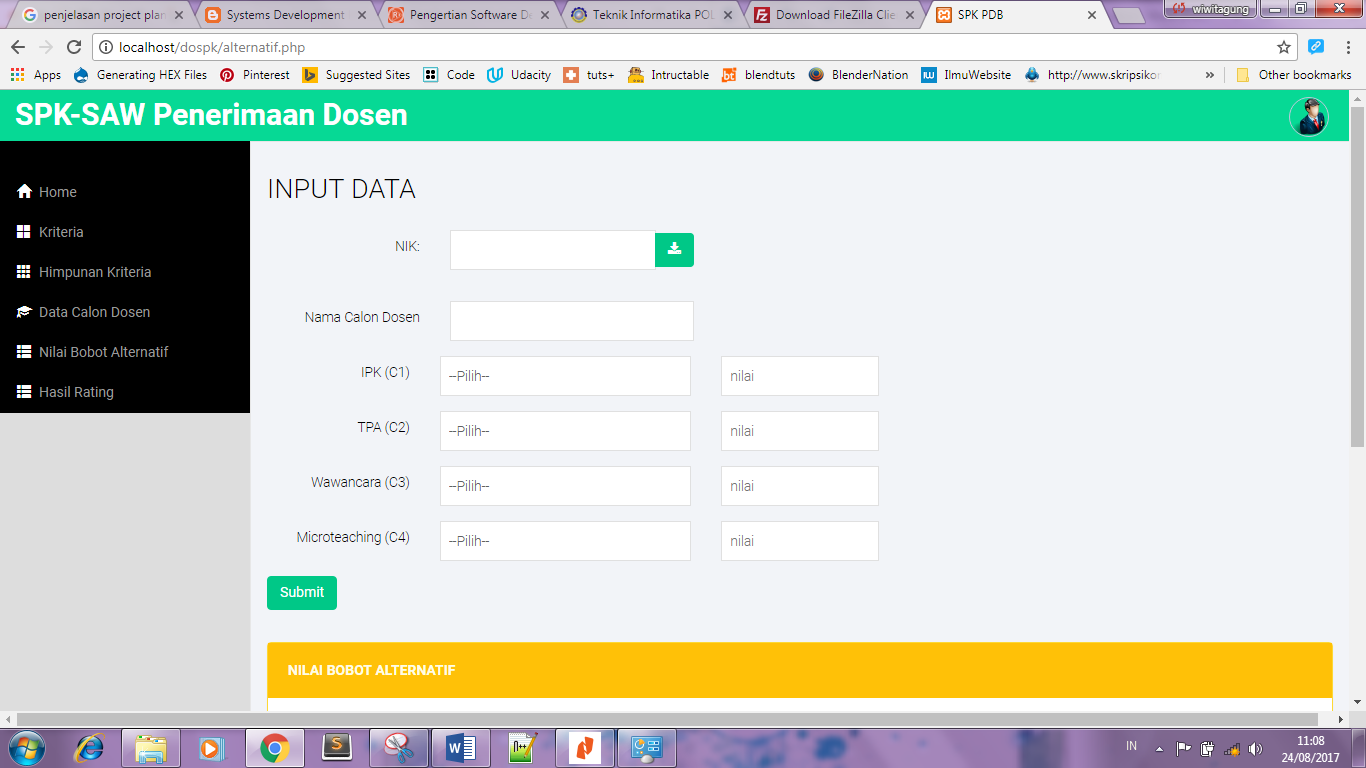
Setelah pilih tombol *submit* selanjutnya tampilan akan berganti dengan menampilkan halaman yang berisikan tabel hasil inputan dari form tadi. Pada tabel calon dosen yang dapat dilihat pada gambar 4.13.



Gambar 4.13 Tampilan Tabel Calon Dosen

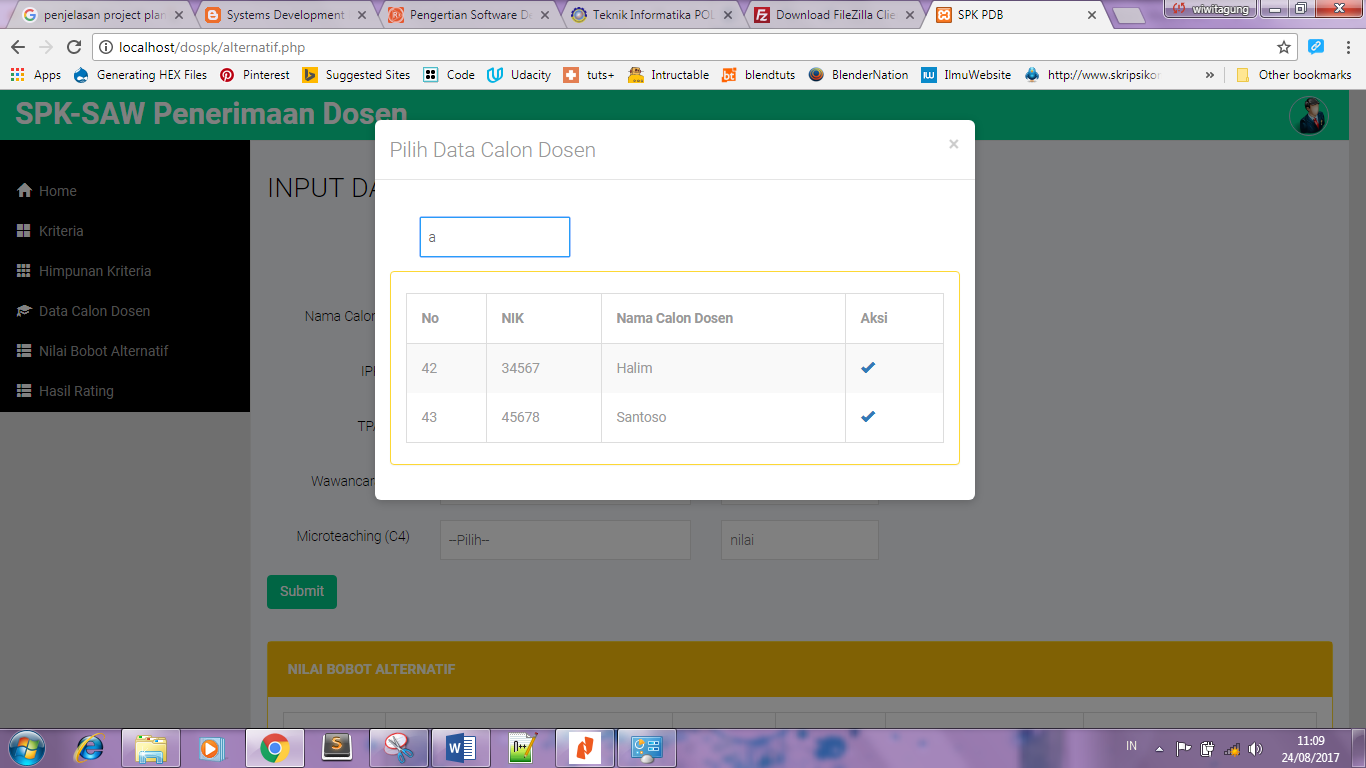
1. **Tampilan Nilai Bobot Alternatif**

Halaman nilai bobot alternatif terdapat dua tombol unduh dan tambah, form inputan nilai masing-masing kriteria dan tabel nilai bobot alternatif untuk menampilkan hasil dari inputan yang dapat dilihat pada gambar 4.14.



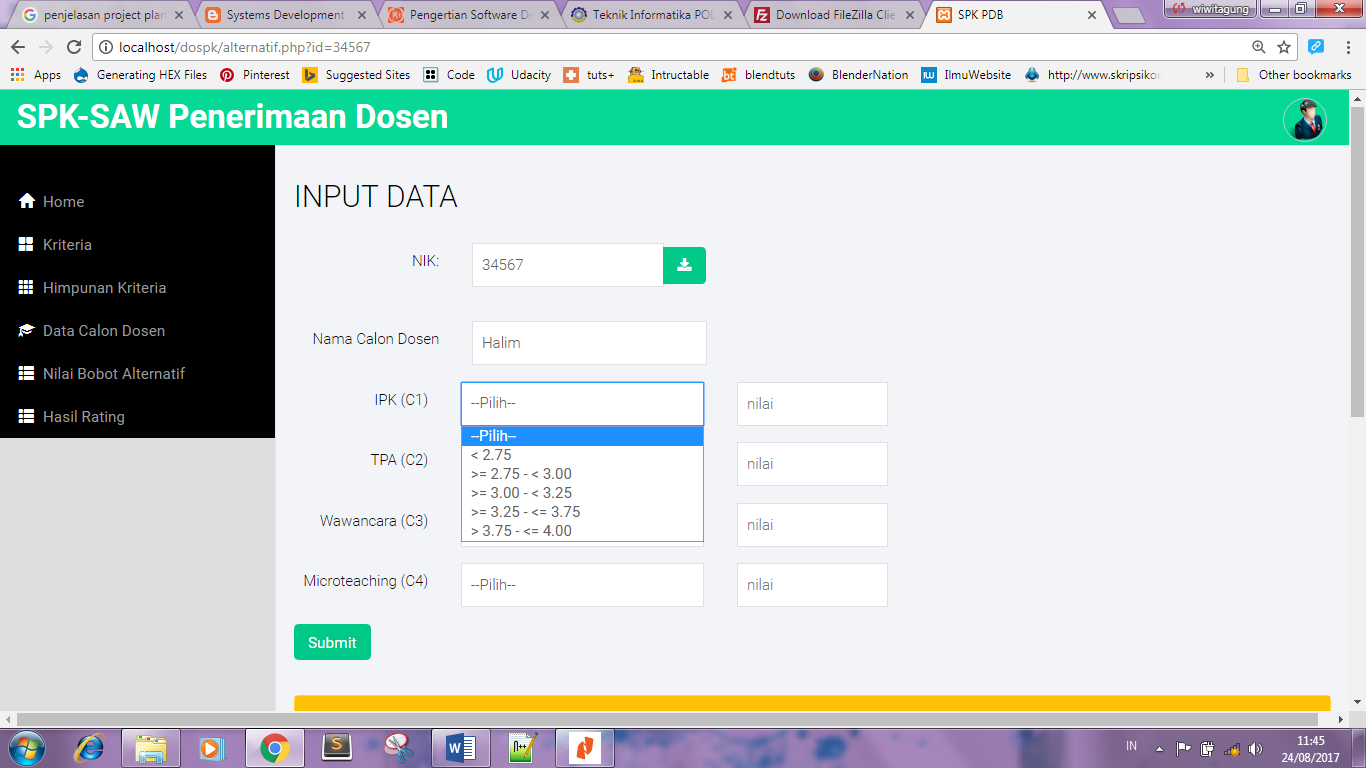
Gambar 4.14 Tampilan Nilai Bobot Alternatif

Jika pilih tombol unduh maka akan menampilkan *modals* yang berisikan form search dan tabel nama calon dosen untuk menampilkan data calon dosen yang dimaksud ada atau tidak. Jika nama calon dosen yang dimaksud benar lalu pilik tombol centang (√)yang dapat dilihat pada gambar 4.15.



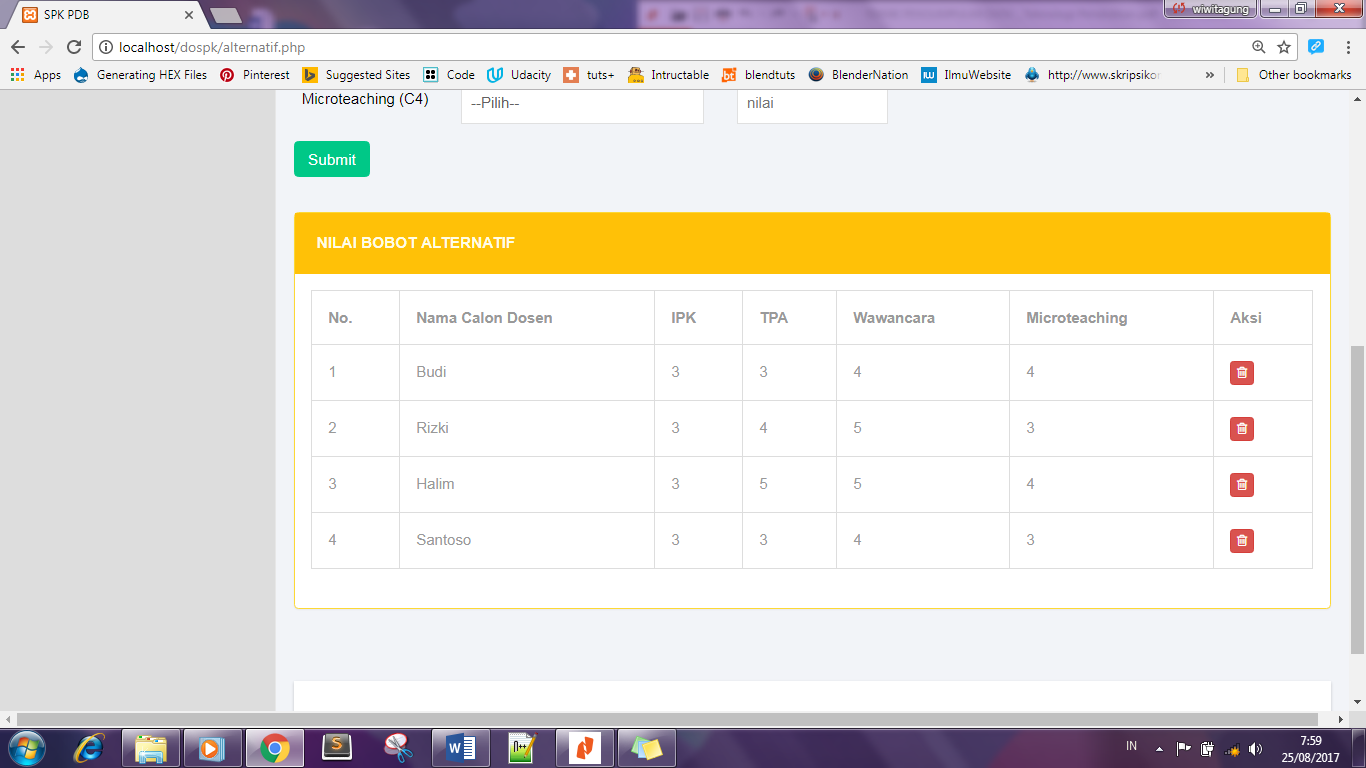
Gambar 4.15 Tampilan Modals Search Nama Calon Dosen

Setelah pilih nama calon dosen, selanjutnya akan menampilkan halaman dengan tampilan yang terdiri dari beberapa form inputan berupa form IPK, TPA, Wawancara, Microteaching yang berisikan data himpunan kriteria masing-masing kriteria yang dapat dilihat pada gambar 4.16.



Gambar 4.16 Tampilan Form Inputan Nilai Himpunan Kriteria

Setelah melakukan inputan form data calon dosen klik tombol *submit* untuk menyimpan data tersebut dan menampilkannya pada tabel data himpunan kriteria yang dapat dilihat pada gambar 4.17.



Gambar 4.17 Tampilan Tabel Data Himpunan Kriteria

1. **Tampilan Hasil Rating**

Halaman hasil rating terdapat 2 tabel yang dapat menampilkan hasil dari persamaan (rumus) normalisasi dan tabel hasil akhir perhitungan metode SAW beserta rating masing-masing calon dosen untuk menentukan nama calon dosen dengan kemungkingan diterima atau tidaknya yang dapat dilihat pada gambar 4.18.

Gambar 4.18 Tampilan Hasil Rating

* + 1. **Pengujian Aplikasi**

Setelah semuanya sudah diimplementasikan tahap selanjutnya untuk dilakukan pengujian. Pengujian aplikasi dilakukan menggunakan *black-box-testing. Black-box testing* adalah metode pengujian perangkat lunak yang tes fungsionalitas dari aplikasi yang bertentangan dengan struktur *internal* atau kerja. Pengetahuan khusus dari kode aplikasi dan pengetahuan pemrograman pada umumnya tidak diperlukan. Metode *black-box* memfokuskan pada keperluan fungsional dan *software* (Rusadi, 2013).

Penulis membagikan kuisioner ke pihak-pihak yang mensimulasikan aplikasi untuk diisi setelah *user* mengakses alamat *website* SPK seleksi penerimaan dosen menggunakan metode SAW*,* yaitu :one403024.ta.ti.polindra.ac.id. Untuk kuisioner dapat dilihat pada lampiran dan adapun hasil pengujian telah dirangkum pada Tabel 4.1.

Tabel 4.1 Tabel Hasil Pengujian Dengan *Black-Box Testing*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Deskripsi** | **Prosedur Pengujian** | **Keluaran yang diharapkan** | **Hasil yang dihapatkan** | **kesimpulan** |
| **Login Admin** | | | | | |
| 1 | admin dapat masuk ke aplikasi seteleah melakuakn *login* | * Masukkan *url* * Masukkan username * Masukkan password * Klik tombol login | Admin dapat melakukkan login dan dapat masuk ke halaman *home* | Halaman home dan diakses | Proses berjalan sesuai yang diharapkan |
| **Menu Kriteria** | | | | | |
| 2 | Admin ingin menambahkan kriteria | * Klik kriteria * Isi form * Klik tombol *submit* | Data kriteria dapat tersimpan, admin dapat menambah, mengubah, dan menghapus data kriteria | Halaman kriteria dapat diakses | Proses berjalan sesuai yang diharapkan |
| **Menu Himpunan Kriteria** | | | | | |
| 3 | Admin ingin menambahkan data himpunan kriteria | * Klik himpunan kriteria * Isi form * Klik tombol submit | Data himpunan kriteria dapat tersimpan, admin dapat menambah, dan menghapus data himpunan kriteria | Halaman himpunan kriteria dapat diakses | Proses berjalan sesuai yang diharapkan |

Tabel 4.2 Tabel Hasil Pengujian Dengan *Black-Box Testing* (Lanjutan)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Menu Data Calon Dosen** | | | | | |
| 4 | Admin ingin menambahkan data calon dosen | * Klik data calon dosen * Isi form * Klik tombol submit | Data calon dosen dapat tersimpan, admin dapat menambah, dan menghapus data calon dosen | Halaman calon dosen dapat diakses | Proses berjalan sesuai yang diharapkan |
| **Menu Nilai Bobot Alternatif** | | | | | |
| 5 | Admin ingin masukkandata dosen beserta nilai masing-masing kriteria | * Klik niali bobot alternatif * pilih nama calon dosen * pilih nilai masing-masing kriteria * Klik tombol submit | Data nilai bobot alternatif dapat tersimpan, dan menghapus nilai bobot alternatif | Halaman nilai bobot alternatif dapat diakses | Proses berjalan sesuai yang diharapkan |
| **Menu Hasil Rating** | | | | | |
| 6 | Admin ingin menampilkan hasil perhingan dan cetak pdf | * Klik hasil rating * Klik tombol cetak | Data tabel menampilkan hasil perhitungan metode SAW | Halaman hasil rating dapat diakses, cetak laporan tidak berhasil | Proses belum berjalan sesuai yang diharapkan |
| **Menu Setting** | | | | | |
| 7 | Admin mengubah *password* | * Klik *setting* * Isi form * Klik tombol ubah | Admin dapat mengakses halaman ganti *password* dan mengubah *password* | Halaman ganti *password* dapat diakses | Proses belum berjalan sesuai yang diharapkan |
| **Menu Logout** | | | | | |
| 8 | Admin ingin keluar dari aplikasi | * Klik menu logout * Klik ok | Admin dapat keluar dari apikasi | Halaman admin telah ditutup | Proses belum berjalan sesuai yang diharapkan |

* + 1. **Kesimpulan Pengujian**

Berdasarkan hasil *testing* yang dilakukan oleh *decision-maker* bahwa aplikasi yang dibuat penulis sudah berjalan dengan baik. Pada dasarnya hasil pengujian program apliksi sesuai harapan, program berjalan dengan lancar dan sangat membantu proses pengambilan keputusan. Penulis juga melakukan pengujian aplikasi sistem pendukung keputusan menggunakan metode *Simple Additive Weighting* dengan kuisioner yang dilakukan pada 10 orang penguji. Dari pengujian kuisioner dapat disimpulkan bahwa aplikasi ini amat baik karena diperoleh persentase pengujian sebesar82,51%. Untuk hasil kuisioner dapat dilihat pada lampiran 2.

**BAB V**

**PENUTUP**

1. **Kesimpulan**

Dari hasil penilitian dan pembahasan diatas maka dapat diambil beberapa kesimpulan antara lain:

* 1. Dengan adanya sistem pendukung keputusan untuk menentukan penerimaan dosen di Polindra akan membantu dalam memberikan rekomendasi dan pertimbangan dalam menentukan penerimaan dosen yang akan diterima melalui data perangkingan dari hasil yang telah diolah dalam sistem tersebut.
  2. Sistem pendukung keputusan untuk menentukan seleksi penerimaan dosen dengan membantu meningkatkan kualitas dalam menentukan penerimaan dosen sehingga dapat dmengurangi kesalahan-kesalahanyang dilakukan sebelum adanya sistem pendukung keputusan ini.
  3. Nilai kriteria dan bobot mempengaruhi hasil dari sistem pendukung keputusan ini. Kriteria yang dipakai pada aplikasi ini diantaranya nilai IPK, Tes Tulis, wawancara dan *microteaching*.
  4. **SARAN**

Berdasarkan pada hasil penelitian dan pembahasan maka dapat diberikan saran sebagai berikut:

1. Kriteria-kriteria dalam persyaratan seleksi dosen bisa berkembang dan mengalami perubahan maka penulis menyarankan sesuaikan syarat yang di butuhkan sesuai ketentuan masing-masing perguruan tinggi.
2. Penerapan metode yang penulis buat dapat dikembangkan dengan metode yang lain seperti Topsis, *Analitic Herarcy Process,* dan sebagainya.
3. Perlu dilakukan pemeliharaan dan pengawasan dari pihak yang bertanggungjawab terhadap sistem.

**DAFTAR PUSTAKA**

**LAMPIRAN**

Jawaban kuisoner:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Nama | jawaban | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 1 |  |  |  |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |  |  |  |  |  |
| 9 |  |  |  |  |  |  |  |  |  |  |  |
| 10 |  |  |  |  |  |  |  |  |  |  |  |

|  |  |
| --- | --- |
| Keterangan jawaban :  ST =  T =  R =  SR = | Keterangan Hasil :  Sangat baik = 80% - 100%  Baik = 60% - 79%  Cukup = 40% - 59%  Kurang = 20% - 39%  Gagal = 0% - 19% |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Pertanyaan** | **Sangat Tinggi** | **Tinggi** | **Rendah** | **Sangat Rendah** |
| **1** |  |  |  |  |  |
| **2** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **5** |  |  |  |  |  |
| **6** |  |  |  |  |  |
| **7** |  |  |  |  |  |
| **8** |  |  |  |  |  |
| **9** |  |  |  |  |  |
| **10** |  |  |  |  |  |
| **Total** | |  |  |  |  |

|  |  |
| --- | --- |
| Sangat Rendah =  Rendah =  Tinggi =  Sangat Tinggi = |  |
| Hasil |  |

**SINTAKS PROGRAM**

Sintaks Metode SAW

|  |
| --- |
| **<!--HEADER-->**  **<?php**  **include('session\_login.php');**  **include 'include/koneksi.php';**  **include 'include/header.php';**  **function getBobot($kriteria){**  **include 'include/koneksi.php';**  **$q = mysqli\_query($con, "SELECT bobot FROM kriteria WHERE kd\_krt='{$kriteria}'");**  **$output = mysqli\_fetch\_assoc($q);**  **return $output['bobot'];**  **}**  **?>**  **<!-- /.page-wrapper -->**  **<div id="page-wrapper">**  **<div class="col-md-12 graphs">**  **<!-- TABEL NORMALISASI -->**  **<div class="panel panel-warning" data-widget="{&quot;draggable&quot;: &quot;false&quot;}" data-widget-static="">**  **<div class="panel-heading">**  **<h2>Tabel Hasil Normalisasi</h2>**  **<div class="panel-ctrls" data-actions-container="" data-action-collapse="{&quot;target&quot;: &quot;.panel-body&quot;}"><span class="button-icon has-bg"><i class="ti ti-angle-down"></i></span></div>**  **</div>**  **<div class="panel-body no-padding" style="display: block;">**  **<table class="table table-bordered">**  **<thead>**  **<tr>**  **<th>No.</th>**  **<th>Nama Dosen</th>**  **<th>IPK</th>**  **<th>TPA</th>**  **<th>Wawancara</th>**  **<th>Microteaching</th>**  **</tr>**  **</thead>**  **<tbody>**    **<?php**  **$sql = mysqli\_query($con,"SELECT max(C1) as maxC1, max(C2) as maxC2, max(C3) as maxC3, max(C4) as maxC4 FROM proses");**  **$max = mysqli\_fetch\_array($sql);**  **$query = mysqli\_query($con, "SELECT \* FROM proses JOIN caldos ON proses.id\_dosen=caldos.id\_dosen ORDER BY id\_matriks ASC") or die (mysqli\_error());**  **if(mysqli\_num\_rows($query)==0){**  **echo '<tr><td colspan="7">Tidak ada data!</td></tr>';**  **}else{**  **$no = 0;**  **while ($data = mysqli\_fetch\_assoc($query)){**  **$no++;**  **$normal1 = round($data['C1']/$max['maxC1'],2);**  **$normal2 = round($data['C2']/$max['maxC2'],2);**  **$normal3 = round($data['C3']/$max['maxC3'],2);**  **$normal4 = round($data['C4']/$max['maxC4'],2);**  **?>**  **<tr>**  **<td><?php echo $no ?></td>**  **<td><?php echo $data['nm\_dosen']; ?></td>**  **<td><?php echo $normal1; ?></td>**  **<td><?php echo $normal2; ?></td>**  **<td><?php echo $normal3; ?></td>**  **<td><?php echo $normal4; ?></td>**  **</tr>**  **<?php**  **}**  **}**  **?>**  **</tbody>**  **</table>**  **</div>**  **</div>**  **<!-- TABEL HASIL SAW & RATING -->**  **<div class="panel panel-warning" data-widget="{&quot;draggable&quot;: &quot;false&quot;}" data-widget-static="">**  **<div class="panel-heading">**  **<h2>Tabel Hasil Rangking</h2>**  **<div class="panel-ctrls" data-actions-container="" data-action-collapse="{&quot;target&quot;: &quot;.panel-body&quot;}"><span class="button-icon has-bg"><i class="ti ti-angle-down"></i></span></div>**  **</div>**  **<div class="panel-body no-padding" style="display: block;">**  **<table class="table table-bordered">**  **<thead>**  **<tr>**  **<th>No.</th>**  **<th>Nama Dosen</th>**  **<th>jumlah poin</th>**  **<th>Hasil SAW</th>**  **<th>Rangking</th>**  **</tr>**  **</thead>**  **<tbody>**  **<?php**  **$tmp\_nilai = array(); //untuk menyimpan nilai yang didapatkan dari proses get query yang dimna datanya akan diurutkan di proses selanjutnya**  **$query = mysqli\_query($con, "SELECT \* FROM proses JOIN caldos ON proses.id\_dosen=caldos.id\_dosen") or die (mysqli\_error());**  **if(mysqli\_num\_rows($query)==0){**  **echo '<tr><td colspan="7">Tidak ada data!</td></tr>';**  **}else{**  **$no = 0;**  **while ($data = mysqli\_fetch\_assoc($query)){**  **$no++;**  **$jumlah = ($data['C1'])+($data['C2'])+($data['C3'])+($data['C4']);**  **$saw = round(**  **(($data['C1']/$max['maxC1'])\*getBobot('C1'))+**  **(($data['C2']/$max['maxC2'])\*getBobot('C2'))+**  **(($data['C3']/$max['maxC3'])\*getBobot('C3'))+**  **(($data['C4']/$max['maxC4'])\*getBobot('C4')),3);**  **//ini adalah proses memasukan data-data dari hasil get query ke dalam array yang sudah dibuat kemudian akan di urutkan menggunakan sorting di PHP function**  **array\_push($tmp\_nilai, array(**  **"saw" => $saw, //ini diletakan di array pertama supaya pas pengurutannya dapat diurutkan berdasarkan saw ini.**  **"nama\_dosen" => $data['nm\_dosen'],**  **"jumlah\_poin" => $jumlah**  **));**  **?>**  **<?php**  **} //end while**  **//sorting ascending menggunakan fungsi bawaaan dari PHP -> mengurutkan nilai berdasarkan "saw"**  **arsort($tmp\_nilai); //ini adalah proses pengurutan untuk menampilkan hasil yang menjadikan prioritas utama (rangking dosen)**  **$no2 = 1; //ini hanya untuk nomor pengurutan untuk ditampilkan di tabel**  **foreach ($tmp\_nilai as $r) {**  **?>**  **<tr>**  **<td><?php echo $no2 ?></td>**  **<td><?php echo $r['nama\_dosen']; ?></td>**  **<td><?php echo $r['jumlah\_poin']; ?></td>**  **<td><?php echo $r['saw'] ?></td>**  **<td><?php echo "Prioritas ".$no2; ?></td>**  **</tr>**  **<?php**  **$no2++;**  **}**  **}**  **echo "</table>";**  **?>**  **</tbody>**  **</table>**  **<div class="form-group">**  **<a href="html2pdf/hasil.php" type="submit" class="btn btn-success" name="cetak" ><span class="fa fa-print"></span> Laporan</a>**  **</div>**  **</div>**  **</div>**  **<!--FOOTER-->**  **<?php include 'include/footer.php'; ?>**  **<script type="text/javascript">**  **$(document).ready(function(){**  **$("#krit1").change(function(){**  **var nilai = $("option:selected", this).attr('attributNilai');**  **$("#krit1\_nilai").val(nilai);**  **});**  **$("#krit2").change(function(){**  **var nilai = $("option:selected", this).attr('attributNilai');**  **$("#krit2\_nilai").val(nilai);**  **});**  **$("#krit3").change(function(){**  **var nilai = $("option:selected", this).attr('attributNilai');**  **$("#krit3\_nilai").val(nilai);**  **});**  **$("#krit4").change(function(){**  **var nilai = $("option:selected", this).attr('attributNilai');**  **$("#krit4\_nilai").val(nilai);**  **});**  **});**  **</script>** |