

BAB IV

HASIL DAN PEMBAHASAN

4.1 Software concept

Sistem informasi akademis berbasis WAP memungkinkan *user* sebagai penggunanya untuk mengakses pengisian KRS dan melihat KHS melalui *mobile device* (*handphone, PDA*) menggunakan *micro browser* yang ada pada *handphone*. Karena berbasis *mobile* maka sistem ini dapat diakses kapan saja, dimana saja di area yang terdapat sinyal GSM.

Sistem ini tidak jauh berbeda dengan Siakad *Online* berbasis *website* yang sekarang masih digunakan oleh UNILA dalam proses langkah- langkah memilih menu seperti mengisi KRS dan menampilkan KHS. Tampilan sistem yang dikembangkan dibuat dengan *user friendly*, simpel sehingga *user* langsung dapat menggunakan tanpa harus di adakan pelatihan sebelumnya.

Karena merupakan sebuah *prototype* pengembangan Siakad *online* yang berbasis *web*, sistem berbasis WAP ini tidak selengkap seperti Siakad *online* berbasis *web*. Sistem ini dikembangkan untuk kemudahan mengakses melalui ponsel. Siakad *online* berbasis *website* dikembangkan untuk diakses melalui PC bukan melalui ponsel.

Penggunaan metode *prototyping* sangat cocok untuk sistem ini karena dibutuhkan masukan, saran, kritik dari sisi konsumen dalam lingkup ini yaitu *user* dan dosen pembimbing untuk dilakukan proses perbaikan sistem agar konsumen puas dan sistem layak digunakan.

4.2 Preliminary requirement analysis

Pengumpulan *user requirement* menggunakan 3 metode yaitu :

a. Observasi

Observasi dilakukan dengan melihat kembali sistem yang sudah ada yaitu melihat Siakad *online* yang berbasis *web*. Adapun langkah-langkahnya sebagai berikut :

1. Melihat cara menampilkan Kartu Rencana Studi (KRS) pada Siakad *online*.
2. Melihat cara menampilkan Kartu Hasil Studi (KHS) pada Siakad *online*.

b. Metode Literatur

Metode literatur dilakukan dengan mempelajari buku-buku pemrograman WML, PHP yang membantu dalam pengembangan Siakad berbasis WAP ini. Adapun langkah-langkahnya sebagai berikut :

1. Mempelajari sistem basis data.
2. Mempelajari struktur bahasa pemrograman *WML*.
3. Mempelajari struktur bahasa pemrograman *PHP*.
4. Mempelajari sistem basis data pada *MySQL*.

5. Mempelajari konfigurasi atau penggabungan *script code* bahasa pemrograman *WML* dan *PHP* serta *database* melalui *MySQL*.
6. Mempelajari teknik kompilasi program pada *WAP emulator*.
7. Mempelajari teknik membuat *wap server* menggunakan *OPERA UNITE*

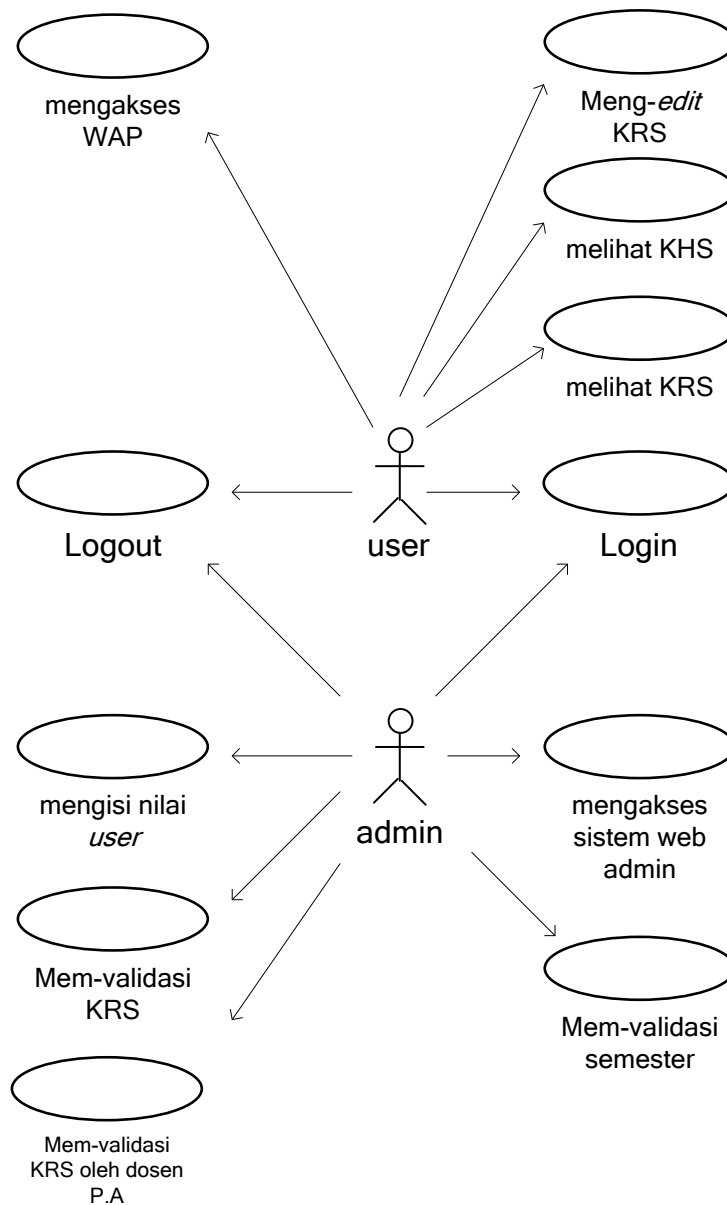
c. Wawancara (*interview*)

Metode ini untuk mengetahui bagaimana respon dari *user* tentang sistem informasi akademik yang dapat di akses melalui telepon seluler. Selain itu juga melakukan bimbingan dengan dosen pembimbing.

4.2.1 Use case Siakad berbasis WAP

Dalam pengumpulan *user requirement* ini perlu dibuat *use case*. Dalam *use case* ini digambarkan apa saja interaksi yang dapat dilakukan dengan sistem.

Dalam sistem ini terdapat dua *user* yang berinteraksi dengan sistem.



Gambar 4.1 Use case diagram Pengembangan Siakad berbasis WAP

Dari Gambar 4.1, *user* berinteraksi dengan sistem yang berbasis WAP. *User* dapat melakukan *login* untuk mengakses menu *edit* KRS, lihat KRS, dan lihat

KHS. Dari setiap menu yang diakses oleh *user*, mula-mula *user* diberikan tombol untuk memilih semester berapa yang di-*edit* atau yang dilihat. Perbedaan menu hanya terdapat di pilihan *edit* KRS. Pada menu *edit* KRS *user* diberikan tombol tambahan yaitu tombol untuk memilih mata kuliah yang akan dipilih.

Sedangkan *admin* dan bagian kemahasiswaan berinteraksi dengan sistem yang berbasis *web*. *Admin*, bagian kemahasiswaan melakukan *maintenance* sistem, *maintenance* nilai *user* yang otomatis langsung merubah nilai pada sistem yang berbasis WAP.

4.3 Design of architecture and system core

Pendekatan dengan metode *evolutionary prototype* diterapkan dari awal perancangan, hingga *coding* sampai testing. Perancangan dimulai dari analisa kebutuhan awal, dan membuat *use case*. Selanjutnya pengembangan Siakad berbasis WAP dimulai dengan mendesain arsitektur dan inti membuat DFD pengembangan sistem, kemudian dibuat *prototype* melalui pengumpulan data menganalisa basis data yang digunakan, hingga tahap dimana *user* dapat mencoba mengakses langsung dari *handphone*-nya sendiri. Setelah *user* menggunakan sistem tersebut, maka didapatkan masukan atau bagaimana keinginan *user* tersebut agar sistem layak digunakan. Setelah *prototype* melalui tahap ini maka dilanjutkan dengan membangun kembali sesuai dengan permintaan *user*. Proses perbaikan ini terus berlangsung hingga *user* merasa puas hingga sistem mencapai *state final*.

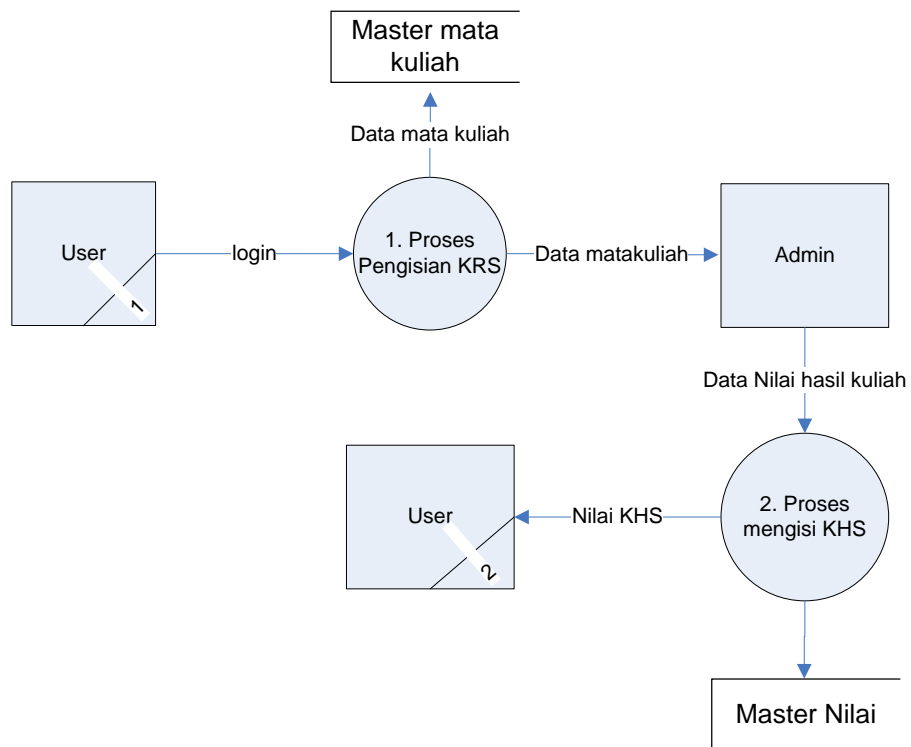
Pada pengembangan SIAKAD berbasis WAP ini melalui beberapa langkah yaitu :

4.3.1 Membuat Data Flow Daigram

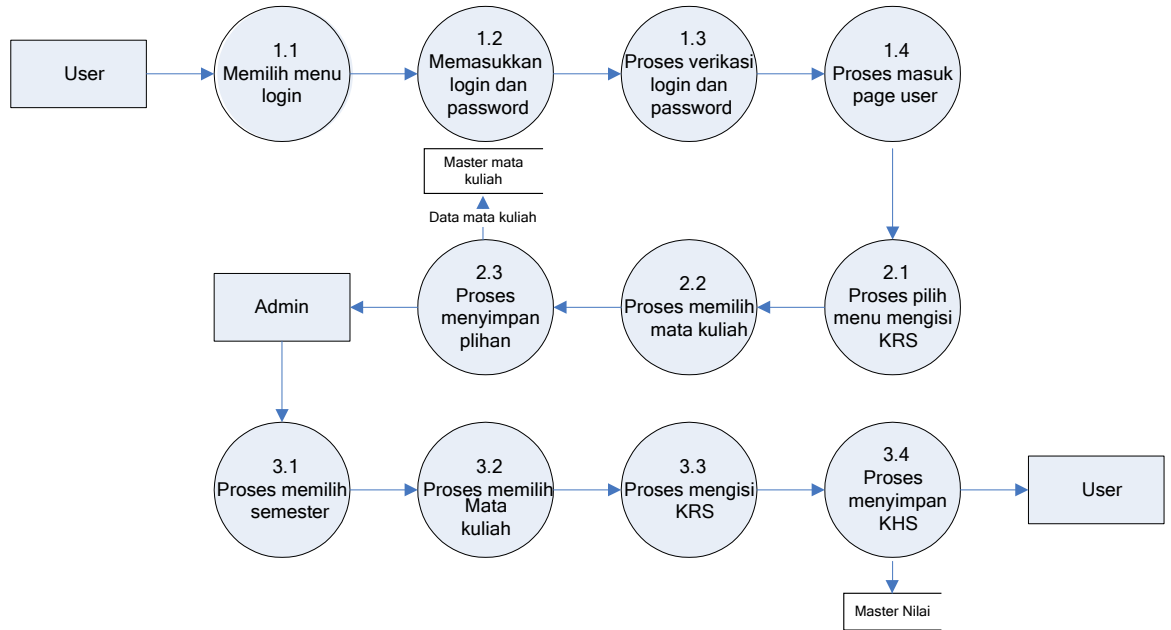
Dalam sistem ini terdapat tahapan bagaimana *user* dapat berinteraksis dengan sistem yang dikembangkan.



Gambar 4.2 Diagram konteks pengembangan Siakad berbasis WAP



Gambar 4.3 DFD level nol Pengembangan Siakad berbasis WAP



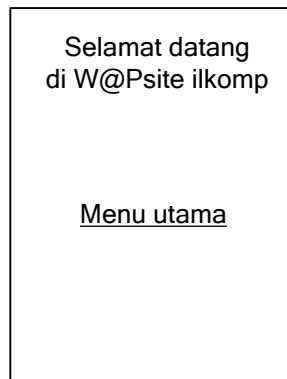
Gambar 4.4 DFD level 1 pengembangan Siacad berbasis WAP

Proses pertama yaitu *user* memasuki menu utama, lalu diberi pilihan untuk *login*. Setelah *user* memilih menu *login* maka sistem merespon dengan menanyakan dengan memberikan *form* kepada *user* yang berisikan *user name* dan *password*. Selanjutnya *user* dipersilahkan mengisi *field* yang diberikan, apabila berhasil, ada pesan balasan yang menyatakan berhasil *login*, dan apabila data *user* belum ada maka sistem memberikan pernyataan bahwa data belum terdaftar. Setelah berhasil login maka *user* masuk ke menu pilihan untuk mengisi KRS atau melihat KHS.

4.3.2 Membuat tampilan desain awal *prototype*

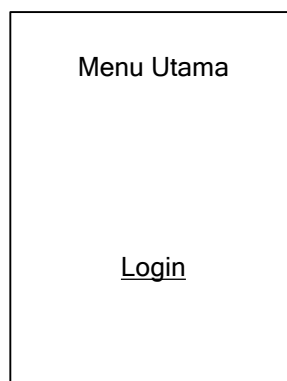
Desain awal ini merupakan langkah pertama dalam membuat *interface*. Tampilan desain awal *prototype* dapat dilihat dari Gambar 4.5 sampai dengan 4.29 berikut ini :

4.3.2.1 Desain awal mengakses WAP site



Gambar 4.5 Desain awal halaman selamat datang

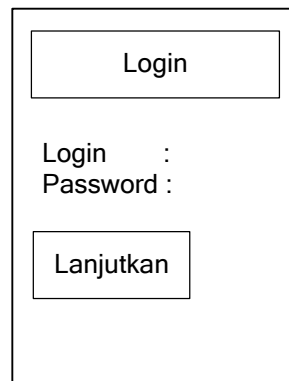
Desain menu selamat datang adalah tampil awal / *default* disaat *user* mengakses sistem berbasis WAP. Pada halaman ini diberikan pilihan untuk masuk ke menu utama.



Gambar 4.6 Desain awal menu utama

Pada tampilan menu utama, diberikan pilihan kepada *user* untuk melakukan proses *login*.

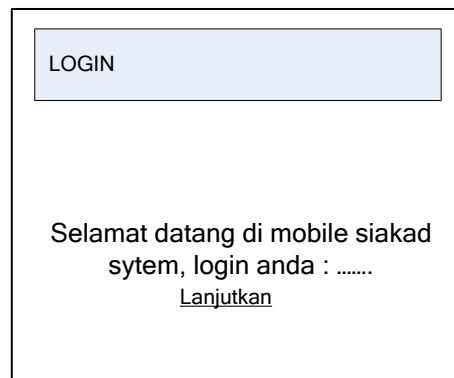
4.3.2.2 Desain awal *Login WAP site*



The image shows a rectangular frame representing a mobile screen. At the top, there is a button labeled "Login". Below it, there are two labels: "Login :" and "Password :". At the bottom, there is another button labeled "Lanjutkan".

Gambar 4.7 Desain awal halaman *Login*

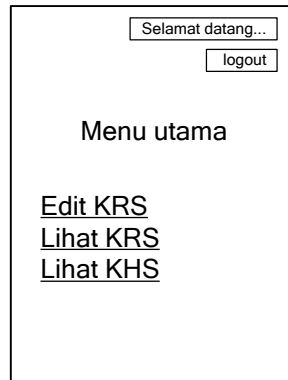
Pada Gambar 4.7 *user* yang memilih menu *login*, diberikan *field* untuk memasukkan *login* dan *password user*. Setelah memasukkan *login* dan *password*, *user* dapat lanjut ke tahap berikutnya dengan menekan tombol lanjutkan.



The image shows a rectangular frame representing a mobile screen. At the top, there is a button labeled "LOGIN". Below it, there is a message: "Selamat datang di mobile siakad sytem, login anda :". At the bottom, there is a button labeled "Lanjutkan".

Gambar 4.8 Desain awal konfirmasi berhasil *login*

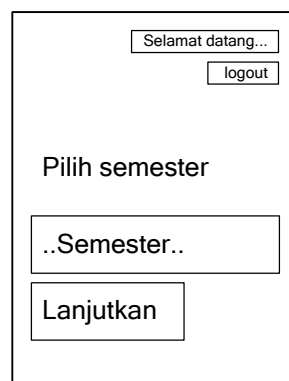
Gambar 4.8 merupakan desain awal tampilan konfirmasi sistem saat *user* berhasil memasukkan *login* dan *password* dengan benar.



Gambar 4.9 Desain awal memilih menu setelah berhasil *login*

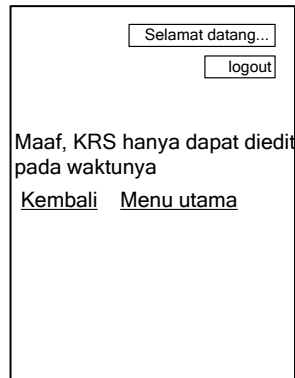
Gambar 4.9 adalah desain tampilan awal sistem apabila sistem berhasil mengenali *user* yang berhasil *login*, sistem akan menampilkan menu *edit* KRS, lihat KRS, dan lihat KHS.

4.3.2.3 Desain awal meng-*edit* KRS



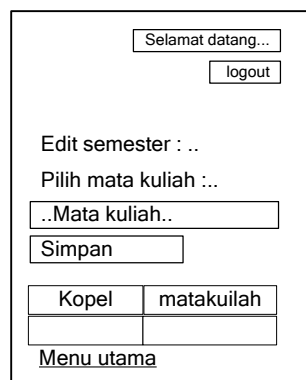
Gambar 4.10 Desain awal tampilan pilih semester pada *option edit* KRS

Pada Gambar 4.10 di atas, tampilan memilih semester akan tampil untuk menentukan mata kuliah yang ada pada semester yang dipilih.



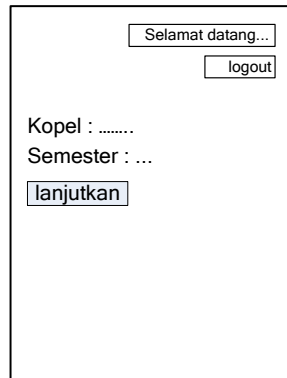
Gambar 4.11 Desain awal pemberitahuan KRS tidak dapat di-*edit*

Gampilan gambar 4.11 dibuat agar sistem dapat menolak *user* meng-*edit* KRS apabila waktu pengisian KRS telah berakhir.



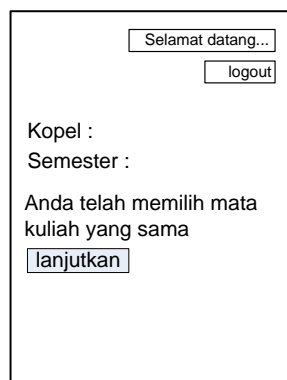
Gambar 4.12 Desain awal tampilan pilih mata kuliah

Gambar 4.12 merupakan desain awal sistem memilih mata kuliah setelah *user* memilih semester yang di-*edit*.



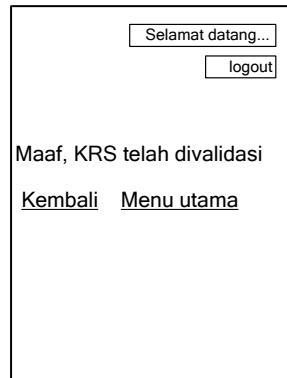
Gambar 4.13 Desain awal konfirmasi mata kuliah yang dipilih

Desain gambar 4.13 diatas merupakan tampilan pemberitahuan atau konfirmasi dari sistem tentang mata kuliah yang dipilih oleh *user*.



Gambar 4.14 Desain awal pemberitahuan memilih mata kuliah yang sama

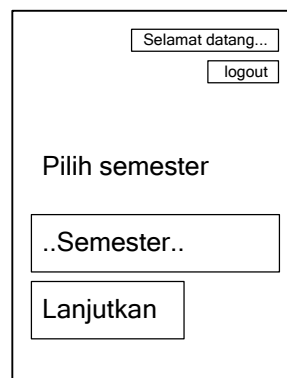
Gambar 4.14 merupakan desain awal pemberitahuan sistem apabila *user* memilih mata kuliah yang telah dipilh sebelumnya.



Gambar 4.15 Desain awal KRS telah divalidasi

Gambar 4.15 merupakan tampilan awal KRS yang tidak dapat di-*edit* oleh *user* karena telah divalidasi oleh *admin* atau dosen pembimbing asuh.

4.3.2.4 Desain tampilan awal melihat KRS



Gambar 4.16 Desain awal tampilan pilih semester pada *option* lihat KRS

Pada Gambar 4.16 di atas, tampilan memilih semester akan tampil untuk menampilkan mata kuliah yang telah di-*edit* oleh *user*.

Kopel	Mata kuliah

Gambar 4.17 Desain awal lihat KRS

Gambar 4.17 adalah desain awal tampilan sistem yang menampilkan mata kuliah yang dipilih oleh *user* dalam proses *edit* KRS.

4.3.2.5 Desain tampilan awal melihat KHS

Selamat datang...
logout

Pilih semester

..Semester..

Lanjutkan

Gambar 4.18 Desain awal tampilan pilih semester pada *option* lihat KHS

Pada Gambar 4.18 di atas, tampilan memilih semester akan tampil untuk menentukan mata kuliah dan huruf mutu dari semester yang dijalankan oleh *user*

Selamat datang...	
logout	
matakuliah	Huruf Mutu

Gambar 4.19 Desain awal lihat KHS

Gambar 4.19 merupakan desain awal dari *option* lihat KHS. *Option* dan tampilan ini dapat dilihat bila *user* sudah melakukan proses *edit* KRS dan telah dimasukkan huruf mutu hasil perkuliahannya oleh dosen.

4.4.2.6 Desain awal *logout* WAP site

lanjutkan'."/>

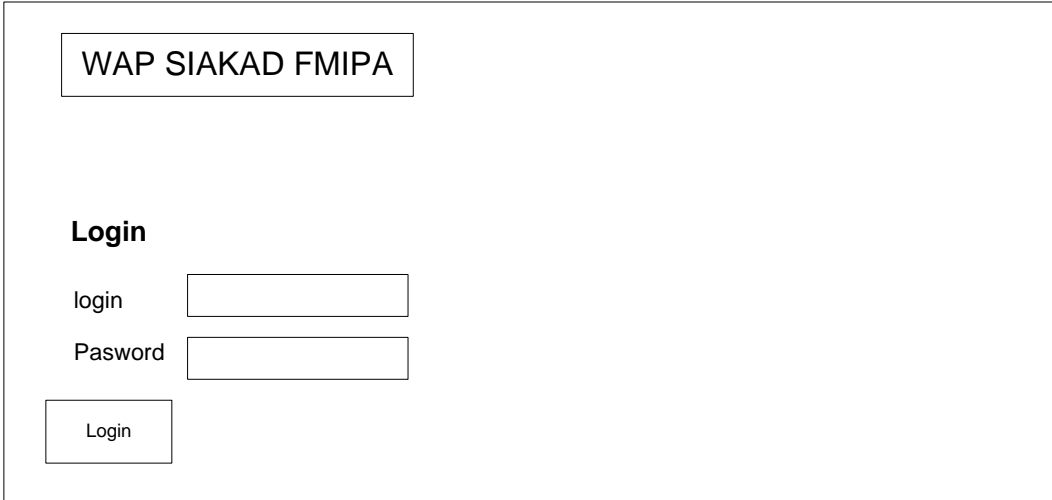
LOGOUT

Anda telah berhasil Logout,
Terimakasih telah
menggunakan layanan
kami, [lanjutkan](#)

Gambar 4.20 Desain awal halaman *logout*

Gambar 4.20 merupakan desain awal halaman logout apabila *user* memilih *option* *logout*.

4.3.2.7 Desain awal mengakses sistem *administrator*



WAP SIAKAD FMIPA

Login

login

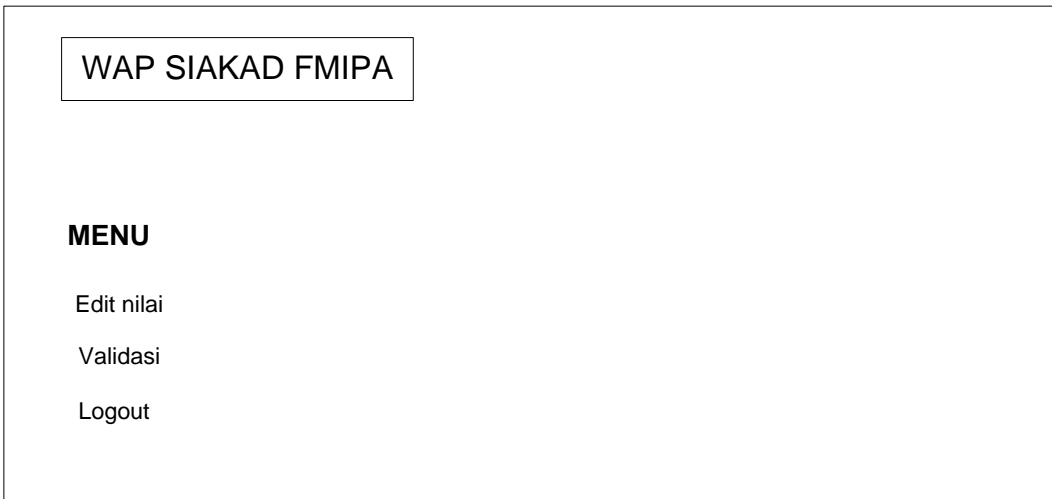
Password

Login

Gambar 4.21 Desain awal menu *admin*

Gambar 4.21 merupakan desain awal dari halaman *admin* yang akan berjalan berbasis *website*. Sistem berbasis web ini hanya diperuntukkan untuk *admin*, dosen pembimbing asuh dan bagian akademik. Pada halaman ini hanya terdapat *field* yang diberikan untuk melakukan *login*.

4.3.2.8 Desain awal *login* sistem *administrator*



WAP SIAKAD FMIPA

MENU

Edit nilai

Validasi

Logout

Gambar 4.22 Desain awal menu *admin* setelah *login*

Gambar 4.22 menunjukkan menu-menu yang ada setelah *administrator* berhasil melakukan proses *login*. Pada halaman ini terdapat *option edit* nilai, *validasi* dan *logout*.

4.3.2.9 Desain awal *edit* nilai

WAP SIAKAD FMIPA

MENU

- Edit nilai
- Validasi
- Logout

Isi nilai

1 lanjutkan

Gambar 4.23 Desain awal pilih semester pada *option edit* nilai

Pada gambar 4.23, tampilan memilih semester akan tampil untuk menentukan mata kuliah yang ada pada semester tersebut.

WAP SIAKAD FMIPA

MENU

- Edit nilai
- Validasi
- Logout

Isi nilai

no	kopel	Mata kuliah

Gambar 4.24 Desain awal tampilan mata kuliah

Setelah menentukan semester maka sistem akan menampilkan daftar mata kuliah yang ada pada semester yang telah dipilih seperti pada gambar 4.24.

WAP SIAKAD FMIPA

Isi nilai

	no	NPM	Nama	Huruf mutu
MENU				
Edit nilai				
Validasi				
Logout				

Gambar 4.25 Desain awal daftar *user* yang mengambil mata kuliah

Pada gambar 4.25 sistem menampilkan daftar *user* yang telah memilih mata kuliah yang ada pada semester tersebut.

4.3.2.10 Desain awal validasi semester

VALIDASI

- Pilih validasi -

no	Semester	Edit
1	1	Non aktifkan
2	2	Non aktifkan
3	2p	aktifkan

Gambar 4.26 Desain awal validasi semester

Gambar 4.26 merupakan desain awal dari halaman validasi semester yang dilakukan oleh bagian kemahasiswaan untuk memperbolehkan *user* pada sistem berbasis WAP melakukan proses peng-*editan* KRS atau tidak

4.3.2.11 Desain awal validasi oleh dosen pembimbing asuh

VALIDASI

- Pilih validasi -

- Pilih semester -

no	NPM	Nama	Valid

Gambar 4.27 Desain awal validasi pembimbing asuh

Pada gambar 4.27 merupakan halaman untuk pembimbing asuh untuk memvalid KRS dari *user* yang merupakan tanggung jawab dari pembimbing asuh tersebut.

4.3.2.12 Desain awal validasi SPP

VALIDASI

- Pilih validasi -

- Pilih semester -

no	NPM	Nama	Status

Gambar 4.28 Desain awal validasi SPP

Halaman validasi pada gambar 4.28 merupakan halaman untuk bagian kemahasiswaan untuk memvalidasi status SPP dari *user* yang terdaftar pada semester yang dipilih.

4.3.2.13 Desain awal *logout* sistem *administrator*

The image shows a wireframe for a login page. At the top, there is a rectangular box containing the text 'WAP SIAKAD FMIPA'. Below this, the word 'Login' is written in a bold font. Underneath, there are two input fields: the first is labeled 'login' and the second is labeled 'Pasword'. At the bottom left of the form area, there is a rectangular button labeled 'Login'.

Gambar 4.29 Desain awal halaman *logout*

Halaman *logout* sistem *administrator* ini akan kembali kehalaman utama saat pertama kali *admin* mengakses sistem ini.

4.3.3 Mengkolabirasikan WML dengan PHP.

File WML dapat digabungkan atau dimasukkan dalam *script* PHP. Ini berfungsi untuk menambah fungsi-fungsi yang tidak ada pada bahasa pemrograman WML. Penulisan *file* WML yang digabungkan dengan PHP harus mengikuti peraturan penulisan *script* PHP dimulai dengan tanda `<?php` dan di akhiri dengan `?>` serta mencantumkan *script* penanda bahwa *file* itu merupakan *file* WML yang digabungkan dengan PHP. Berikut ini merupakan contoh *file* PHP yang digabungkan dengan WML.

```
<?php
Header('Content-type:text/vnd.wap.wml');
Echo('<?xml version="1.0"?>');
Echo('<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML
1.1/EN">');
Echo "<wml>";
```

```

Echo "<card title=\".:: Dasar PHP WML ::.\"
newcontext=\"true\">";
Echo "<p> Menggunakan PHP </p>";
Echo "</card>";
Echo "</wml>";
?>

```

Baris ke dua sampai ke tiga merupakan penanda bahwa *file* tersebut merupakan *file* yang menggunakan bahasa WML namun digabungkan dengan PHP.

4.3.4 Menjalankan pada WAP emulator



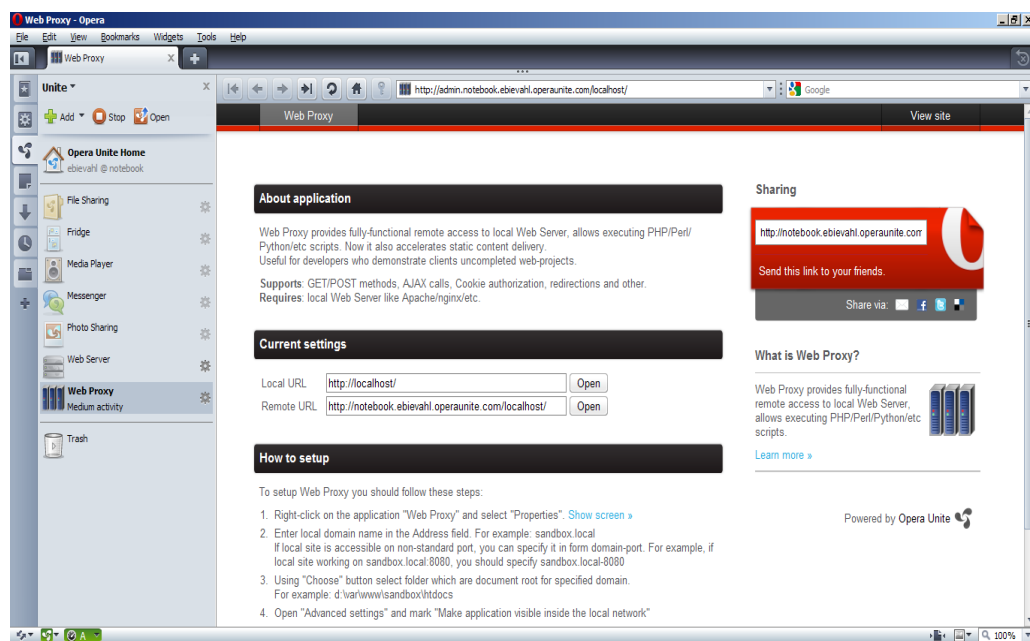
Gambar 4.30 Menjalankan pada WAP emulator

Gambar 4.15 memperlihatkan hasil *compile* bahasa WML yang dijalankan melalui *Internet explorer* v.6 lalu saat *file* dibuka maka secara otomatis M3 gate emulator menampilkan *output* dari program tersebut.

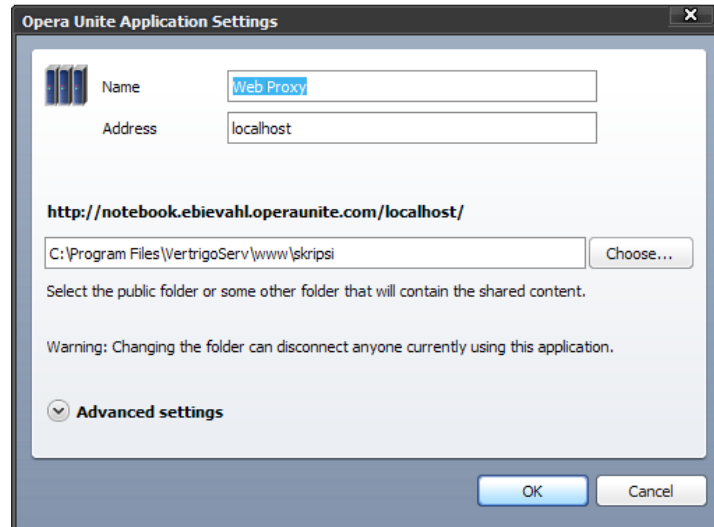
4.3.5 Mempersiapkan WAP server menggunakan *OPERA UNITE*.

Opera Unite merupakan layanan tambahan dari Opera 10, yang memungkinkan komputer dapat dijadikan sebagai *webserver*. Dalam *Opera Unite* terdapat beberapa layanan yang dapat digunakan. Dalam pengembangan sistem ini menggunakan layanan *web proxy* yang memungkinkan *user* membangun sebuah *server* menggunakan *web server* Apache.

Sebelum dapat menggunakan layanan *Opera Unite*, *user* diharuskan untuk mendaftar terlebih dahulu untuk menjadi pemakai *Opera Unite*. Setelah mendaftar langkah selanjutnya adalah mengaktifkan *webserver* Apache yang ada pada komputer maupun *laptop*.



Gambar 4.31 Tampilan halaman utama *Opera Unite*



Gambar 4.32 Konfigurasi *localhost* pada *Opera Unite*

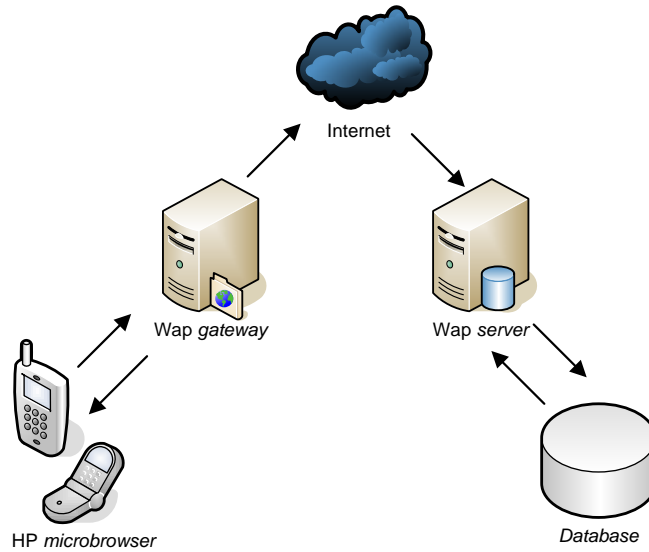
Setelah diaktifkan maka selanjutnya melakukan konfigurasi pada *web proxy*. Opera Unite memberikan alamat *website* yang dapat di akses sehingga *notebook* dan PC pribadi dapat dijadikan *web server* tanpa melakukan *hosting* di *Internet*.

4.4 Develop a version

Setelah mendesain arsitektur dan inti sistem, maka sistem dikembangkan dan mulai dibangun. Hasil pengembangan tampak seperti penjelasan berikut:

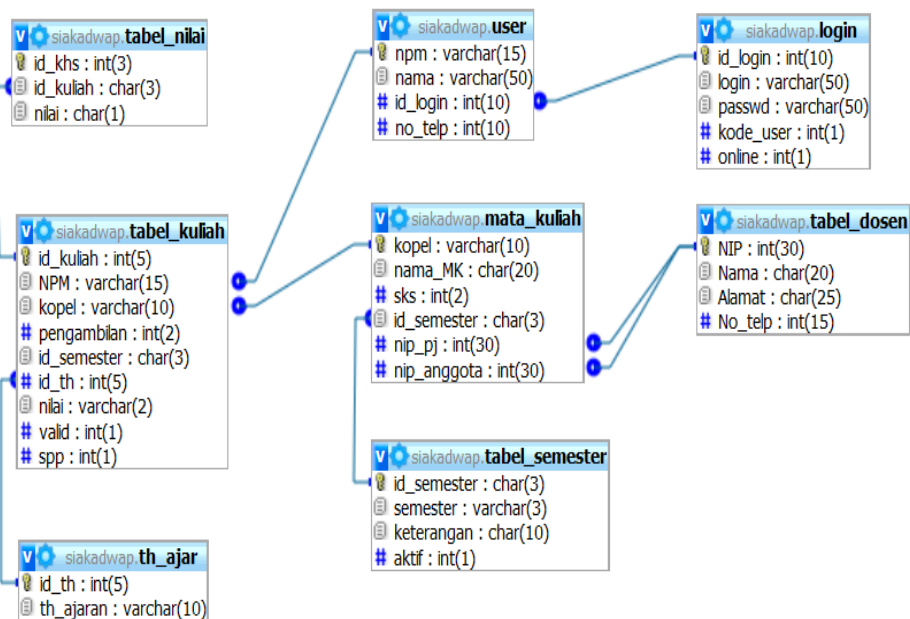
4.4.1 Rancangan basis data dan ERD

Perancangan Siakad berbasis WAP ini memerlukan *database* untuk menyimpan data-data *user*, data mata kuliah, dll. *Database* yang digunakan yaitu MySQL. Sistem ini menggunakan MySQL karena bahasa WML yang digabungkan dengan PHP dapat membaca *database* MySQL, dan *web server* Apache yang sudah tergabung dalam satu aplikasi yaitu Vertrigo. Dalam pengembangan sistem berbasis WAP ini digunakan tabel-tabel.



Gambar 4.33 Arsitektur basis data pengembangan Siakad berbasis WAP

Berikut ini adalah ERD dari tabel yang digunakan dalam pengembangan Siakad berbasis WAP yang menunjukkan relasi antar tabel.



Gambar 4.34 ERD Pengembangan Siakad berbasis WAP

Berikut beberapa tabel yang digunakan dalam pengembangan Siakad berbasis WAP :

Tabel 4.1 *Login*

No	Nama <i>field</i>	Tipe data
1	<i>Id_login</i>	Int(10)
2	<i>Login</i>	Varchar(50)
3	<i>Passwd</i>	Varchar(50)
4	<i>Kode_user</i>	Int(1)
5	<i>Online</i>	Int(1)

Tabel 4.1 dipergunakan dalam sistem untuk menyimpan *login* dan *password user*. Pada tabel ini terdapat *field* *Kode_user* yang digunakan untuk verifikasi *level user* yang digunakan sistem. *Field Online* pada tabel 4.1 digunakan sebagai kondisi dimana *user* yang *online* akan didaftar pada *field* ini.

Tabel 4.2 Mata kuliah

No	Nama <i>field</i>	Tipe data
1	<i>Kopel</i>	Varchar(10)
2	<i>Nama_MK</i>	Char(20)
3	<i>Sks</i>	Int(2)
4	<i>Id_semester</i>	Char(3)
5	<i>Nip_pj</i>	Int(30)
6	<i>Nip_anggota</i>	Int(30)

Dalam sistem saat memilih mata kuliah, maka dimunculkan beberapa mata kuliah sesuai dengan semesternya dan dosen pengajarnya. Nama-nama mata kuliah beserta keterangan mata kuliah tersebut disimpan dalam tabel 4.2.

Tabel 4.3 Tabel Dosen

No	Nama <i>field</i>	Tipe data
1	NIP	Int(30)
2	Nama	Char(20)
3	Alamat	Char(25)
4	No_telp	Int(15)

Tabel 4.3 dipergunakan untuk menyimpan informasi dosen yang mengajar matakuliah.

Tabel 4.4 Tabel kuliah

No	Nama <i>field</i>	Tipe data
1	Id_kuliah	Int(5)
2	NPM	Varchar(15)
3	Kopel	Varchar(10)
4	Pengambilan	Int(2)
5	Id_semester	Char(3)
6	Id_th	Int(5)
7	Valid	Int(1)
8	Spp	Int(1)

Kontrak kuliah yang dilakukan oleh dosen dan mahasiswa dilakukan dengan memilih mata kuliah sesuai dengan semesternya yang disimpan dalam tabel 4.4. pada tabel ini terdapat beberapa informasi seperti matakuliah yang diambil, mahasiswa yang bersangkutan serta informasi validasi dosen dan SPP yang harus dilakukan untuk memvalidasi kontrak kuliah.

Tabel 4.5 Tabel nilai

No	Nama <i>field</i>	Tipe data
1	Id_khs	Int(3)
2	Id_kuliah	Char(3)
3	Nilai	Char(1)

Nilai yang didapat setelah perkuliahan selesai disimpan dalam tabel 4.5. nilai yang disimpan dimunculkan dalam format huruf mutu.

Tabel 4.6 Tabel semester

No	Nama <i>field</i>	Tipe data
1	Id_semester	Char(3)
2	Semester	Varchar(3)
3	Keterangan	Char(10)
4	Aktif	Int(1)

Semester pada pada sistem dibuat dinamis dengan mengisikan kode semester pada tabel 4.6. pada tabel ini semester yang aktif pada waktunya dapat di-*set* dengan mengubah nilai pada *field* Aktif.

Tabel 4.7user

No	Nama <i>field</i>	Tipe data
1	NPM	Varchar(15)
2	Nama	Varchar(50)
3	Id_login	Int(1)
4	No_telp	Int(10)

Mahasiswa yang terdaftar dalam sistem disimpan dalam tabel 4.7. dalam tabel ini terdapat informasi mahasiswa yang digunakan sebagai identitas mahasiswa.

Tabel 4.8 Tahun ajar

No	Nama <i>field</i>	Tipe data
1	Id_th	Int(5)
2	Th_ajaran	Varchar(10)

Dalam *database* sistem Siakad berbasis WAP ini, tahun ajar digunakan dalam menentukan mata kuliah dalam semester disesuaikan dengan angkatan mahasiswa yang mengambil mata kuliah. Tahun ajaran terus di-*update* sesuai dengan pengisian mata kuliah dalam semester.

4.4.2 Membuat Tampilan WAP *site*

Pada tahap ini sistem dibuat berdasarkan proses-proses yang telah dilakukan sebelumnya.

4.4.2.1 Tampilan awal akses WAP *site*

Disini *user* memasuki menu awal selamat datang saat WAP *site* di akses melalui *handphone*, dan diberikan *link* untuk memasuki menu utama.



Gambar 4.35 Tampilan halaman selamat datang

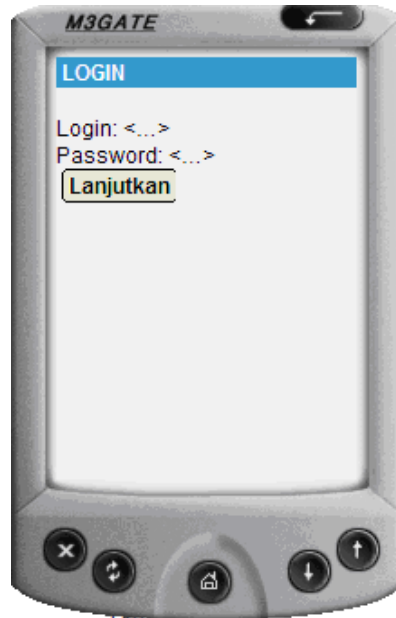
Di menu utama ini *user* diberikan pilihan untuk langsung *login* ke sistem dengan memilih *link Login*.



Gambar 4.36 Tampilan menu utama

4.4.2.2 Tampilan *Login WAP site*

Pada halaman ini *user* di berikan *field* untuk mengisi *login* dan *password* untuk memasuki menu pengisian KRS dan melihat KHS.



Gambar 4.37 Tampilan halaman *Login*

Pada tampilan ini sistem menyatakan *Login* anda berhasil. Pada proses *Login* berhasil ini sistem meng-*update field online* pada tabel *login* dengan mengubah nilai 0 dengan nilai 1 yang menyatakan bahwa *user* ini sedang *online*. Prosedur ini menggantikan fungsi *session* yang biasanya digunakan pada sistem berbasis *website* . perubahannya dapat dilihat pada tabel 4.9 Berikut ini merupakan potongan *coding* tersebut.

```
$baca = mysql_query("select * from login where login='$login' and
passwd='$passwd'");
$ketemu = mysql_num_rows($baca); // cek ada apa tidak (jika ada maka
nilainya >0 )
while($data = mysql_fetch_array($baca))
{
    $sid = $data['0'];
    $login = $data['1'];
    $password = $data['2'];
    $kode_user = $data['3'];
}
```

```

if ($ketemu > 0)
{
    //menentukan spesifikasi user
    if ($kode_user=='2')
    {
        //buat kondisi dinyatakan online
        $update = mysql_query("update login set online = '1'
where id_login = '$id'");
        echo "Selamat datang di Mobile siacad system, login
anda: $login <br/>";
        echo "<a href='user.php?id=$id'>Lanjutkan</a>";
    }
}

```



Gambar 4.38 Tampilan konfirmasi berhasil *Login*

Id_login	Login	Passwd	Kode_user	online
101	Febri	Kingston	2	1
103	Upin	Upin	2	0

Tabel 4.9 Perubahan status *online* pada tabel *login*

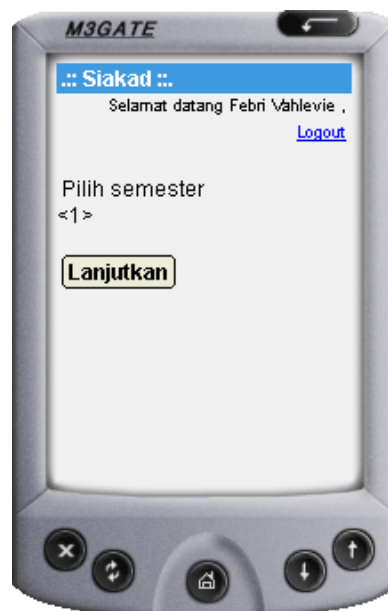
Setelah berhasil *login* maka *user* dapat masuk ke menu utama *edit KRS*, lihat KRS dan lihat KHS. Seperti pada Gambar 4.39 berikut ini



Gambar 4.39 Tampilan memilih menu setelah berhasil *login*

4.4.2.3 Tampilan meng-*edit* KRS

Setelah memilih *edit* KRS maka sistem menampilkan layar memilih semester seperti pada Gambar 4.40 berikut ini.



Gambar 4.40 Tampilan pilih semester pada *option edit* KRS

Pada saat memilih semester maka sistem menjalankan *script* yang menampilkan daftar semester yang dipilih.

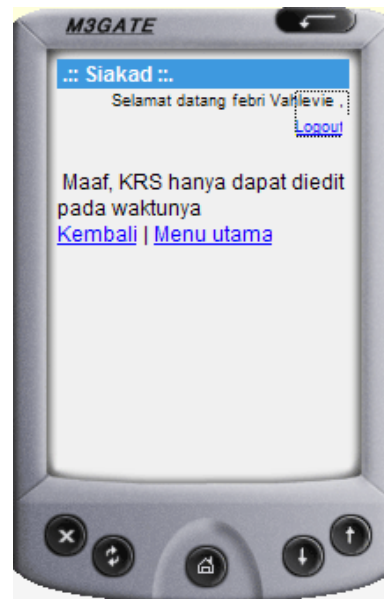
```

echo "Pilih semester<br/>";
$krss = mysql_query("select * from tabel_semester");
echo "<select name='semester'>";
//loop drop down yang isinya dari tabel
while(list($id_semester,$semester) = mysql_fetch_array($krss))
    {
        echo "<option value =
'$id_semester'>$semester</option>";
    }
echo "</select>";
?>
<br/>
<br/>
<!-- kirim variabel pake button-->
<do type="accept" label="Lanjutkan">
<!-- baca method-->
        <go href="editkrs2.php" method="get">
                <postfield name="id" value="<? echo
"$id";?>" />
                <postfield name="semester" value="$(semester)" />
        </go>
</do>
</p>

```

Sistem melanjutkan ke menu pemilihan mata kuliah berdasarkan semester.

Apabila waktu untuk mengisi KRS sudah habis atau belum memasuki waktu untuk mengisi KRS maka sistem menolak atau memberikan respon bahwa pengisian KRS hanya dapat di-*edit* pada waktunya seperti pada Gambar 4.41 berikut ini.



Gambar 4.41 Pemberitahuan KRS tidak dapat di-*edit*

Berikut ini merupakan tampilan *script* yang menunjukkan proses di atas.

```
<?
include "koneksi.php";

$cek = mysql_query("select aktif from tabel_semester where
id_semester = '$semester' ");
while($cross = mysql_fetch_array($cek))
{
    if($cross[0] == 0)
    {
        echo "Maaf, KRS hanya dapat di-edit pada waktunya
<br/>";
        echo "<a href='editkrs.php?id=$id'>Kembali</a> | <a
href='user.php?id=$id'>Menu utama</a>";
        ?>

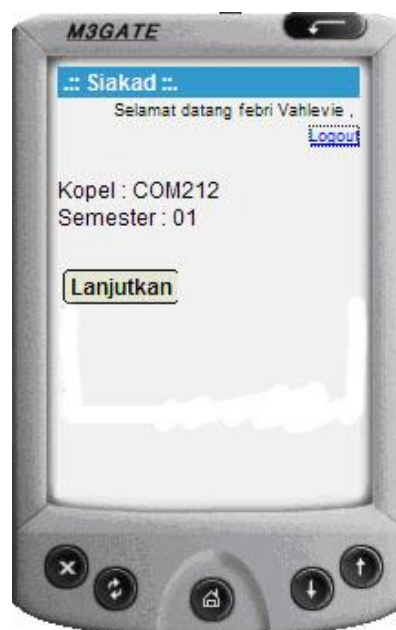
        </p>
        </card>
        </wml>
        <?
        exit();
    }
}
```

Setelah memilih semester yang di-*edit*, sistem akan lanjut ke langkah memilih mata kuliah yang ada berdasarkan semester. Seperti pada Gambar 4.42



Gambar 4.42 Tampilan memilih mata kuliah

Pada saat *user* memilih untuk menyimpan maka sistem mengkonfirmasi bahwa *user* mengambil mata kuliah tersebut dengan menampilkan kode pelajaran (kopel) dan semester mata kuliah tersebut seperti pada Gambar 4.43.



Gambar 4.43 Tampilan konfirmasi mata kuliah yang dipilih

Pada saat memilih mata kuliah maka sistem menjalankan *script* sebagai berikut ini :

```

echo "Edit semester: $semester";
//baca tabel kuliah
$kr = mysql_query("select * from mata_kuliah");
echo "<br/>Pilih matakuliah : <select name='kopel'>";
//looping buat drop down yang isinya baca dari tabel
while(list($kopel,$nama_MK) = mysql_fetch_array($kr))
    {
        echo "<option value = '$kopel'>$nama_MK</option>";
    }
echo "</select><br/>";

$simpan = mysql_query("select * from tabel_kuliah");

?>
<br/>
        <do type="accept" label="Simpan">
        <go href="editkr3.php" method="get">
            <postfield name="id" value="<? echo
"$id";?>"/>
            <postfield name="kopel" value="$(kopel)"/>
            <postfield name="semester" value="<? echo
"$semester";?>"/>
        </go>
        </do>
<br/>
//menampilkan mata kuliah mata kuliah dalam bentuk tabel
<table columns='2'>
    <tr>
        <td><b>Kopel</b></td>
        <td><b>Matakuliah</b></td>
        <td><b>Edit</b></td>
    </tr>
</table>
//baca 2 tabel query multi tabel pake inisialisasi nama tabel di-
form-nya
$bacakr = mysql_query("select a.kopel, b.nama_MK, a.id_kuliah from
mata_kuliah b, tabel_kuliah a where a.kopel = b.kopel and npm =
'$npm' and a.id_semester = '$semester'");
while($isinya = mysql_fetch_array($bacakr))
    {
        $kopel = $isinya['0'];
        $namaMK = $isinya['1'];
        $id_kuliah = $isinya['2'];
        $trim = "$id,$id_kuliah,$semester";

        echo "<tr><td>";
        echo "$kopel";
        echo "</td><td>";
        echo "$namaMK";
    }
?>
</table>

<? echo "<a href='user.php?id=$id'>Menu utama</a>"; ?>
</p>
</card>
</wml>

```

Apabila *user* melakukan pengisian mata kuliah yang sama maka sistem merespon dengan memberitahu bahwa *user* sudah mengambil mata kuliah tersebut seperti pada Gambar 4.44.



Gambar 4.44 Tampilan pemberitahuan memilih mata kuliah yang sama

Dari Gambar 4.44 di atas, sistem menjalankan *script* program sebagai berikut

```
<?php
header('Content-type:text/vnd.wap.wml');
echo('<?xml version="1.0"?>');
echo('<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1/EN">');
?>
<wml>
<card title=":: Siakad ::" newcontext="true">
<p align="right">
<small>
<?php
extract($_GET);
extract($_POST);
echo "Selamat datang ";
include "sesion.php";
echo " , <a href ='logout.php?id=$id'>Logout</a>";
?>
</small></p><p align="left">
<?
if(isset($kopel))
{
echo "<br/><br/>";
echo "Kopel : $kopel<br/>";
echo "Semester : $semester<br/><br/><br/><br/><br/>";
}
```

```

        $cek = mysql_query("select * from tabel_kuliah where
kopel = '$kopel' and npm = '$npm' and id_semester = '$semester'");
        //melihat jumlah isi yang akan dibaca
        $jum = mysql_num_rows($cek);
        //kondisi jika isi tabel kosong
        if($jum > 0)
            {
                ?>

                <? echo "Anda sudah memilih mata kuliah
yang sama";?>

                <br/>
                <do type="accept" label="Lanjutkan">
                <go href="editkrs2.php" method="get">
                    <postfield name="id" value="<? echo
"$sid";?>" />
                    <postfield name="semester"
value="<? echo "$semester";?>" />
                </go>
                </do>

                <?
                }
                else
                {

                    $simpan = mysql_query("insert into tabel_kuliah
(NPM,kopel,pengambilan,id_semester) values ('$npm', '$kopel', '$pengambil
an', '$semester')");
                    echo "<br/> ";

                ?>
                <br/>
                <do type="accept" label="Lanjutkan">
                    <go href="editkrs2.php" method="get">
                        <postfield name="id" value="<? echo "$sid";?>" />
                        <postfield name="semester" value="<? echo
"$semester";?>" />
                    </go>
                </do>

                <?
                    }
                }
            //buat extract karakter dengan jedah "," trus dimasukkan dalam
variaber secara berurutan (list)
            list($id,$id_kuliah,$semester) = explode(",", $id);
            if(isset($id_kuliah))
            {
                echo "<br/><br/>";
                $hapus = mysql_query("delete from tabel_kuliah where id_kuliah
= '$id_kuliah'");
                echo "Data telah dihapus <br/> id_kuliah = $id_kuliah";
            }
            ?>
            <br/>
            <do type="accept" label="Lanjutkan">
                <go href="editkrs2.php" method="get">
                    <postfield name="id" value="<? echo "$sid";?>" />
                    <postfield name="semester" value="<? echo
"$semester";?>" />
                </go>
            </do>

```

Selain itu sistem dapat melakukan penolakan bila pengisian KRS sudah di validasi oleh *administrator*, dosen seperti yang dapat dilihat pada Gambar 4.45.



Gambar 4.45 Tampilan KRS telah divalidasi

Berikut ini merupakan *script* yang menunjukkan proses seperti Gambar 4.45 di atas.

```

$ccek2 = mysql_query("select a.valid from tabel_kuliah a,user b where
a.NPM = b.npm and b.id_login = '$id'");
while($cross2 = mysql_fetch_array($cek2))
{
    $kros = $cross2[0];
}
if($kros == 1)
{
    echo "Maaf, KRS telah divalidasi <br/>";
    echo "<a href='editkrs.php?id=$id'>Kembali</a> | <a
href='user.php?id=$id'>Menu utama</a>";
    ?>

    </p>
</card>
</wml>
<?
exit();
}

```


4.4.2.4 Tampilan melihat KRS

Setelah KRS di-*edit* maka *user* dapat melihat KRS yang telah di-*edit* oleh *user* itu sendiri berdasarkan semester yang *user* pilih seperti pada Gambar 4.46



Gambar 4.46 Tampilan pilih semester pada *option* melihat KRS

Pada saat menjalankan proses melihat KRS maka sistem menjalankan *script* sebagai berikut ini :

```

echo "Pilih semester<br/>";
$krs = mysql_query("select * from tabel_semester");
echo "<select name='semester'>";

while(list($id_semester,$semester) = mysql_fetch_array($krs))
    {
        echo          "<option          value          =
'$id_semester'>$semester</option>";
    }
echo "</select>";

?>
<br/>

<br/>
<!-- button "lanjutkan" dan mengirim variabel dengan method get -->
<do type="accept" label="Lanjutkan">
    <go href="lihatkrs2.php" method="get">

```

```

        <postfield name="id" value="<? echo
"$sid";?>" />
        <postfield name="semester" value="$ (semester)" />
    </go>
</do>

```



Gambar 4.47 Tampilan melihat KRS

Pada gambar 4.47 matakuliah yang tampil adalah mata kuliah yang sebelumnya di pilih oleh *user* pada proses *edit* KRS.dari gambar 4.47, sistem menjalankan *script* program sebagai berikut ini :

```

<?
$bacakrs = mysql_query("select a.kopel, b.nama_MK from mata_kuliah b,
tabel_kuliah a where a.kopel = b.kopel and npm = '$npm' and
a.id_semester = '$semester'");
//melihat jumlah isi tabel yang akan dilihat
$num = mysql_num_rows($bacakrs);
//kondisi jika data kosong ($num) kosong
if($num == 0)
{
    echo "<br>KRS kosong<br>";
    echo "    <a href='lihatkrs.php?id=$id'>pilih
semester</a><br>
    <a href='user.php?id=$id'>Menu
utama</a></p></card></wml>";
    exit();
}
?>
<br/>
<!-- buat tabel-->
<table columns='2'>
<tr>
<td><b>Kopel</b></td>

```

```

        <td><b>Matakuliah</b></td>
    </tr>
<?
//baca 2 tabel

while($isinya = mysql_fetch_array($bacakrs))
    {
    $kopel = $isinya['0'];
    $namaMK = $isinya['1'];
    //looping isi tabel sekaligus baris tabelnya
    echo "<tr><td>";
    echo "$kopel";
    echo "</td><td>";
    echo "$namaMK";
    echo "</td></tr>";
    }
?>
</table>

<? echo "<a href='user.php?id=$id'>Menu utama</a>"; ?>

```

4.4.2.5 Tampilan melihat KHS

Setelah melakukan proses perkuliahan maka *user* akan mendapatkan huruf mutu hasil dari perkuliahan tersebut yang dapat dilihat pada pilihan lihat KHS pada semester yang dijalani oleh *user* tersebut seperti yang pada Gambar 4.48 dan 4.49.



Gambar 4.48 Tampilan pilih semester yang pada *option* melihat KHS

Tampilan memilih semester pada gambar 4.48 menjalankan script sebagai berikut:

```

<?php
header('Content-type:text/vnd.wap.wml');
echo('<?xml version="1.0"?>');
echo('<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1/EN">');
?>
<wml>
<card title=".: Siakad :." newcontext="true">

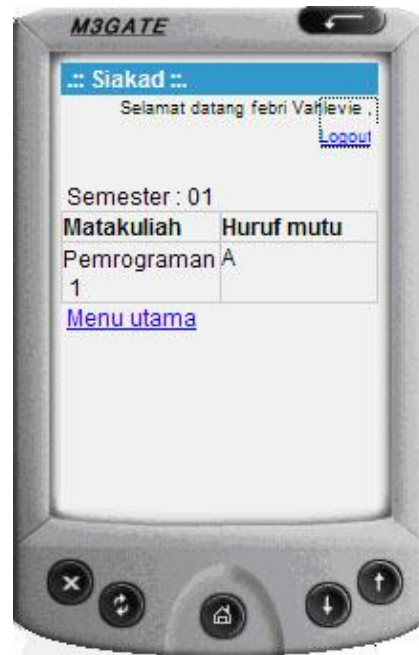
<p align="right">
<small>
<?php
    include "koneksi.php";
    extract($_GET);
    extract($_POST);
    include "sesion.php";
    echo "Selamat datang ";
    echo " , <a href ='logout.php?id=$id'>Logout</a>";
?>
</small>
</p>
<p align="left">
<br/><br/>
<?
echo "Pilih semester<br/>";
$krs = mysql_query("select * from tabel_semester");
echo "<select name='semester'>";
//loop drop down yang isinya dari tabel
while(list($id_semester,$semester) = mysql_fetch_array($krs))
    {
        echo "<option value = '$id_semester'>$semester</option>";
    }
echo "</select>";

?>
<br/>

<br/>
<!-- kirim variabel pake button-->
<do type="accept" label="Lanjutkan">
<!-- baca method-->
        <go href="khs2.php" method="get">
            <postfield name="id" value="<? echo
"$id";?>"/>
            <postfield name="semester" value="$(semester)"/>
        </go>
</do>
</p>

</card>
</wml>

```



Gambar 4.49 Tampilan lihat KHS

Pada saat melihat KHS *user* maka sistem menjalankan *script* program sebagai berikut :

```

echo "Semester : $semester";
$bacakrs = mysql_query("select a.nama_MK, a.nilai
from mata_kuliah a, tabel_kuliah c
where a.kopel = c.kopel and c.id_semester = '$semester'");
$jum = mysql_num_rows($bacakrs);
if($jum!=0)
{
?>
<table columns='2'>
  <tr>
    <td><b>Matakuliah</b></td>
    <td><b>Huruf mutu</b></td>
  </tr>
  <tr>
    <td><b>Pemrograman A</b></td>
    <td>1</td>
  </tr>
?<
//baca 3 tabel query multi tabel pake inisialisasi nama tabel di-
from-nya
while($isinya = mysql_fetch_array($bacakrs))
{
  $namaMK = $isinya['0'];
  $nilai = $isinya['1'];
  $trim = "$id,$id_kuliah,$semester";

  echo "<tr><td>";
  echo "$namaMK";
  echo "</td><td>";
  echo "$nilai";
  echo "</td></tr>";
}
?>
</table>

```

```

<?
}
else
{
echo "<br/><br/>Nilai belum ada<br/><br/>";
}
echo "<a href='user.php?id=$id'>Menu utama</a>"; ?>
</p>
</card>
</wml>

```

4.4.2.6 Tampilan *logout* WAP site

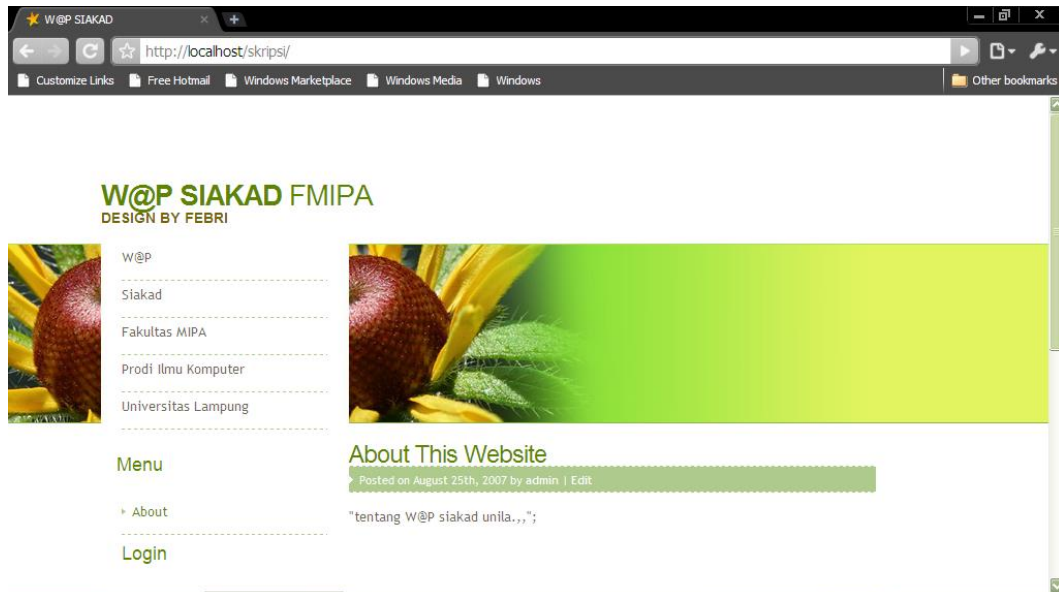
Setelah melakukan semua proses maka *user* dapat melakukan *logout* dari sistem. Seperti yang dilihatkan pada Gambar 4.33 berikut ini.



Gambar 4.50 Tampilan halaman *Logout*

4.4.2.7 Tampilan mengakses halaman *administrator*

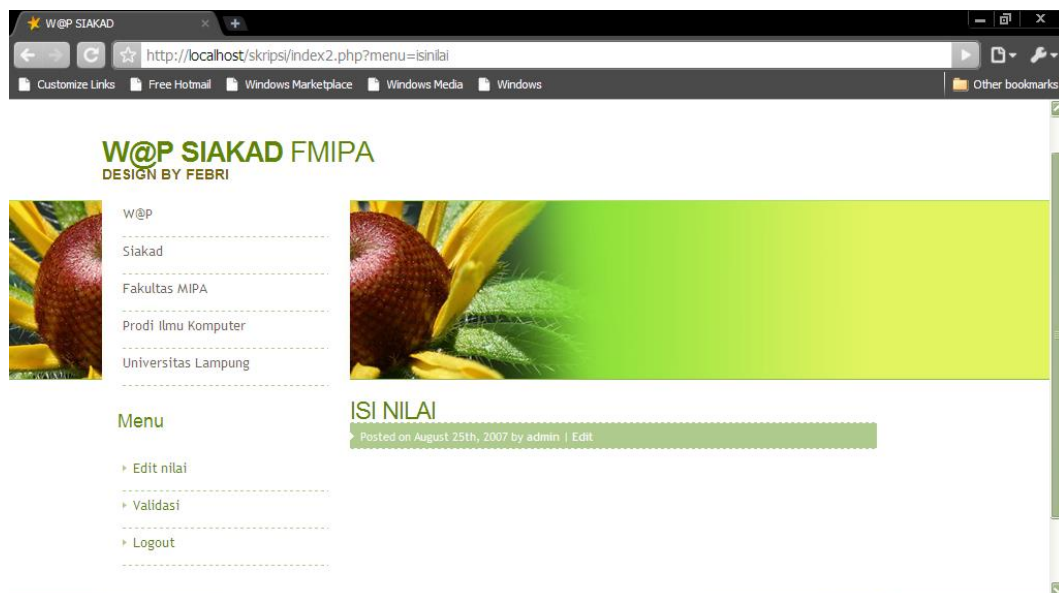
Pengisian nilai yang dilakukan oleh *admin* dilakukan melalui sistem berbasis *web*. *Admin* diharuskan untuk mengisi *form login* terlebih dahulu untuk masuk ke halaman pengisian nilai. Berikut ini adalah tampilan halaman *admin* seperti pada Gambar 4.34.



Gambar 4.51 Tampilan halaman menu *admin*

4.4.2.8 Tampilan *login* sistem *administrator*

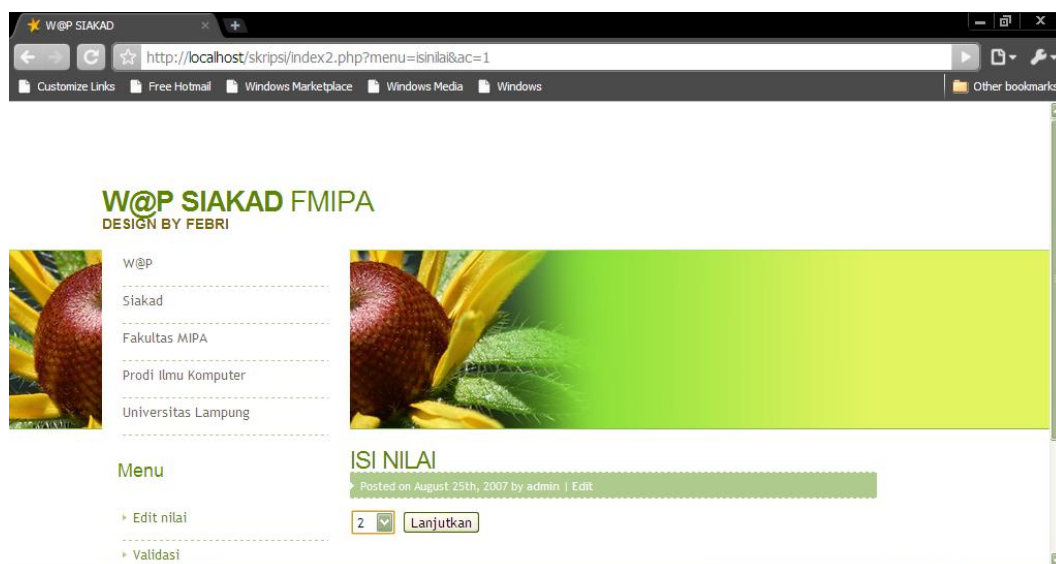
Setelah berhasil sistem memberikan *option edit* nilai, validasi, dan *logout* kepada *admin*



Gambar 4.52 Tampilan menu *admin* setelah *login*

4.4.2.9 Tampilan *edit* nilai

Apabila *admin* memilih *option edit* nilai maka sistem akan memberikan pilihan daftar semester yang akan diisi seperti pada Gambar 4.36 berikut ini :



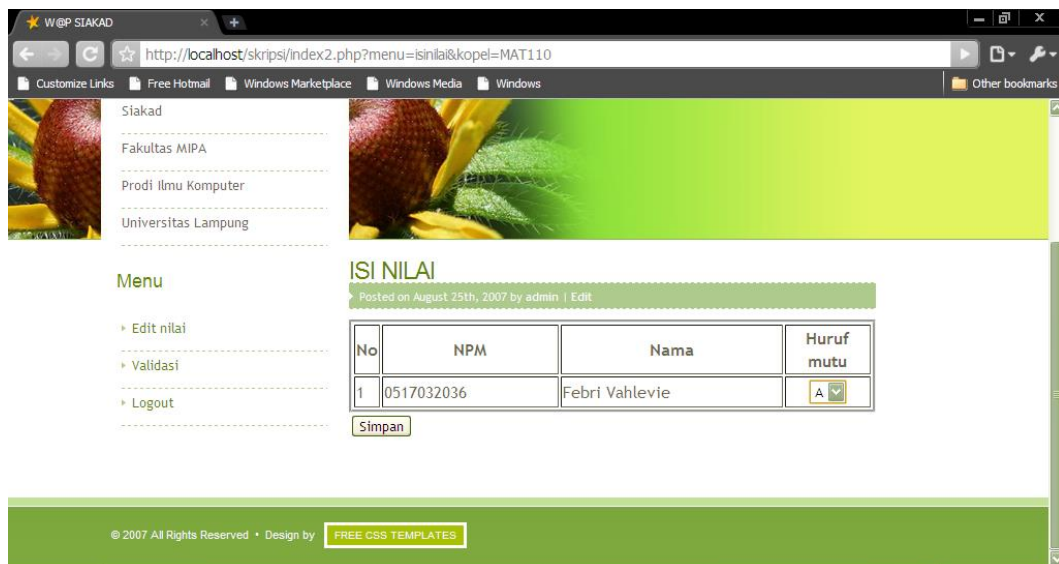
Gambar 4.53 Tampilan pilih semester pada *option edit* nilai

Selanjutnya sistem menampilkan mata kuliah yang ada pada semester tersebut untuk dipilih dan diisi nilainya oleh *admin* seperti pada Gambar 4.54 berikut ini.



Gambar 4.54 Tampilan mata kuliah

Setelah memilih mata kuliah maka sistem akan menampilkan daftar *user* atau *user* yang mengambil mata kuliah tersebut seperti pada Gambar 4.55 berikut ini.



Gambar 4.55 Tampilan daftar *user* yang mengambil mata kuliah

Berikut ini merupakan *script coding* dari Gambar 4.55 :

```
for($i= 1; $i<=$jum; $i++)
{
```

```

    $save = mysql_query("update tabel_kuliah set nilai =
    '$nilai[$i]' where id_kuliah = '$id_kuliah[$i]'");
}
echo "data telah disimpan";

```

4.4.2.10 Tampilan validasi semester

Bagian kemahasiswaan, *admin* dapat melakukan validasi pengaktifan semester yang dapat diakses oleh *user* pada saat itu (masa pengisian KRS, KPRS). Pengaktifan semester dapat dilihat pada Gambar 4.39 berikut ini.

VALIDASI

Posted on August 25th, 2007 by admin | Edit

- Pilih validasi -

No	Semester	Edit
1	1	Non Aktifkan
2	2	Aktifkan
3	2P	Aktifkan
4	3	Aktifkan
5	4	Aktifkan
6	4P	Aktifkan

Gambar 4.56 Validasi pengaktifan semester oleh *admin*

4.4.2.11 Tampilan validasi oleh dosen pembimbing asuh

Dosen pembimbing akademik juga dapat melakukan validasi terhadap KRS yang telah diisi oleh *user* pada semester yang telah ditentukan oleh *user*. Apabila tidak divalidasi, *user* tidak dapat mengakses menu lihat KHS. Validasi oleh *admin* dapat dilihat pada Gambar 4.39 berikut ini.

VALIDASI
Posted on August 25th, 2007 by admin | Edit

- Pilih validasi - ▾

- Pilih semester - ▾

No	NPM	Nama	Valid
1	0517032036	febri Vahlevie	<u>Non Aktifkan</u>

Gambar 4.57 Validasi pembimbing asuh

4.4.2.12 Tampilan validasi SPP

Bagian kemahasiswaan dapat melakukan validasi SPP *user*. Apakah sudah membayar atau belum. Apabila *user* belum membayar, *user* tidak dapat mengakses menu lihat KHS. Validasi SPP dapat dilihat pada Gambar 4.40 berikut ini.

VALIDASI
Posted on August 25th, 2007 by admin | Edit

- Pilih validasi - ▾

- Pilih semester - ▾

No	NPM	Nama	Status
1	0517032036	febri Vahlevie	Sudah bayar

Gambar 4.58 Validasi SPP

Berikut ini merupakan potongan *script* yang menunjukkan validasi oleh *admin*.

```
if(isset($semester))
```

```
{
```

```

        if($semester == 0)
        {
            echo "Semester $id_semester telah di
Non-Aktifkan";

            $seti3 = mysql_query("update
tabel_semester set aktif = '1' where id_semester = '$id_semester'");

            echo "<meta http-equiv=refresh
content=2;url='../index2.php?menu=validasi&id=1'>";

        }

        if($semester == 1)
        {
            echo "Semester $id_semester telah
diaktifkan";

            $seti2 = mysql_query("update
tabel_semester set aktif = '0' where id_semester = '$id_semester'");

            echo "<meta http-equiv=refresh
content=2;url='../index2.php?menu=validasi&id=1'>";

        }
    }

    if(isset($valid))
    {
        if($valid == 0)
        {
            echo "NPM $npm telah tidak valid";

            $seti = mysql_query("update tabel_kuliah
set valid = '0' where NPM = '$npm' and id_semester = '$id_semester'");

            echo "<meta http-equiv=refresh
content=2;url='../index2.php?menu=validasi&id=2'>";

        }

        if($valid == 1)
        {
            echo "NPM $npm telah valid";

            $seti = mysql_query("update tabel_kuliah
set valid = '1' where NPM = '$npm' and id_semester = '$id_semester'");

            echo "<meta http-equiv=refresh
content=2;url='../index2.php?menu=validasi&id=2'>";

        }
    }
}

```

```

if(isset($spp))

    {

        if($spp == 1)

            {

                echo "SPP $npm telah valid";

                $seti = mysql_query("update tabel_kuliah
set spp = '1' where NPM = '$npm' and id_semester = '$id_semester'");

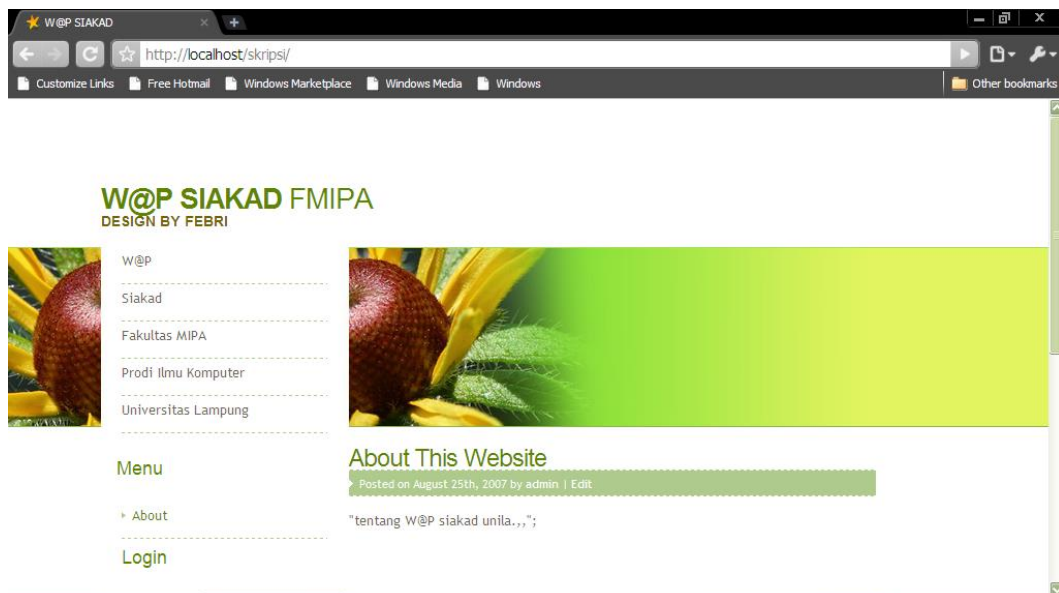
                echo "<meta http-equiv=refresh
content=2;url='../index2.php?menu=validasi&id=3'>";

            }

        }

```

4.4.2.13 Tampilan *logout administrator*



Gambar 4.59 Halaman *logout admin*

Halaman logout dari sistem administrator ini akan kembali ke halaman utama saat sistem ini di akses.

4.5 *Deliver the version*

Pada tahap ini sistem telah selesai dibangun dan dikembangkan dan siap digunakan oleh *user*.

4.6 Elicit customer feedback

Setelah digunakan oleh *user*, berikut ini merupakan hasil saran dan masukan yang dapat diterapkan ke sistem

Pengujian Tampilan	Hasil pengujian
Tampilan halaman awal	Halaman awal hanya terdapat menu <i>login</i> . Perlu ditambahkan menu <i>about</i> , <i>help</i> agar sistem terlihat lebih professional dan layak digunakan
Tampilan <i>edit</i> KRS	Perlu ditambahkan menu untuk menghapus mata kuliah yang dipilih. Apabila salah mengambil maka bisa menghapus mata kuliah tersebut.

Table 4.10 Masukan dari *user*

4.7 Incorporate customer feedback

Setelah masukan dari *user* diterima dan di data maka dilakukan perbaikan pada sistem. Berikut ini merupakan tampilan sistem sebelum dan sesudah diperbahurui.

4.7.1 Sebelum diperbaharui



Gambar 4.60 Tampilan sistem sebelum diperbaharui

Dari Gambar 4.41 di atas, tampilan menu utama hanya terdapat *option* untuk *login*, dan pada tampilan *edit* KRS tidak terdapat *option* untuk menghapus mata kuliah yang sudah dipilih.

4.7.2 Sesudah diperbaharui



Gambar 4.61 Tampilan sistem setelah diperbaharui

Dari gambar 4.42 di atas dapat dilihat adanya perbedaan dari gambar 4.41. Pada menu utama ditambahkan *option about* dan *help*. Sedangkan di tampilan *edit KRS* ditambahkan *option hapus* untuk menghapus mata kuliah yang hendak dihapus atau batal diambil oleh *user*