**BAB IV**

**IMPLEMENTASI DAN PEMBAHASAN**

1. **Implementasi**

Dalam rencana penerapan sebuah sistem yang akan dibangun, agar sistem siap dioperasikan maka perlu diadakan kegiatan-kegiatan dari penerapannya. Adapun langkah-langkah yang akan dilakukan untuk penerapan sistem tersebut adalah pembuatan program, *testing* / uji coba sistem, dan sampai pada tahap pemeliharaan sistem. Berikut adalah tahap yang dilakukan penulis dalam penerapan sistem yang dibuat:

1. Uji Coba Sistem dan Program

Proses ujicoba sistem dan program merupakan hal yang sangat penting dilakukan mengingat sangat perlunya dilakukan pengujian untuk melihat kelayakan dan kemampuan maupun stabilitas program ketika dioperasikan.

Pengujian dilakukan dengan menggunakan metode *black box testing.* Pada tahap pengujian dengan metode *black box* dilakukan testing terhadap semua fungsi. Dengan metode *black box* *testing* ini maka dapat diketahui kesalahan yang terjadi seperti ketidaksesuaian *interface*, kesalahan kinerja, dan kesalahan fungsi baik dari fungsi program maupun fungsi lainnya. Dengan memasukkan semua kemungkinan input dan menjalankan semua fitur yang tersedia pada aplikasi kemudian diperiksa apakah fungsi yang diujikan tersebut menghasilkan output sesuai dengan yang diharapkan sehingga tidak ada ditemukan kesalahan *interface*, kesalahan kinerja dan fungsi.

Adapun *black box testing* yang dilakukan meliputi beberapa pengujian yang dapat dikelompokkan seperti pada Tabel 6.

Tabel 6. Pengujian Sistem Metode *Black Box*

|  |  |  |
| --- | --- | --- |
| Nama Pengujian | Kriteria Evaluasi Hasil | Jenis Pengujian |
| 1 | 2 | 3 |
| Pengujian *login* admin pada website server | Website dapat menampilkan halaman *dashboard* admin jika login berhasil | *Black Box Testing* |
| Pengujian *login* dosen pada website server | Website dapat menampilkan halaman *dashboard* dosen jika login berhasil | *Black Box Testing* |
| Pengujian terhadap tambah data | Website akan dapat menampilkan hasil proses tambah data jadwal dengan munculnya pesan “data berhasil ditambahkan” dan adanya data baru yang ditambahkan pada bagian tabel data | *Black Box Testing* |
| Pengujian terhadap edit data | Website akan dapat menampilkan hasil proses edit data jadwal dengan adanya data yang berubah pada bagian tabel data | *Black Box Testing* |
| Pengujian terhadap hapus data | Website tidak dapat menampilkan lagi data yang dihapus | *Black Box Testing* |
| Pengujian terhadap upload file | Website dapat menyimpan file didalam direktori materi pada server | *Black Box Testing* |
| Pengujian *login* mahasiswa pada aplikasi client | aplikasi dapat menampilkan halaman informasi dan menu aplikasi jika login berhasil | *Black Box Testing* |
| Pengujian terhadap menu jadwal | Aplikasi dapat menampilkan data jadwal yang dikirim oleh server pada client | *Black Box Testing* |
| Pengujian terhadap menu tugas | Aplikasi dapat menampilkan data tugas yang dikirim oleh server pada client | *Black Box Testing* |
| Pengujian terhadap menu materi | Aplikasi dapat menampilkan data materi yang dikirim oleh server pada client | *Black Box Testing* |
| Pengujian terhadap menu informasi | Aplikasi dapat menampilkan data informasi yang dikirim oleh server pada client | *Black Box Testing* |
| Pengujian terhadap *download file materi* | *User* dapat menyimpan *file* dalam *smartphone* atau *client* | *Black Box Testing* |

Pengujian pada sistem ini merupakan tahap-tahap apakah program yang diuji sesuai dengan hasil yang diharapkan, adapun hasil pengujian pada aplikasi ini dapat dilihat dengan beberapa pengujian.

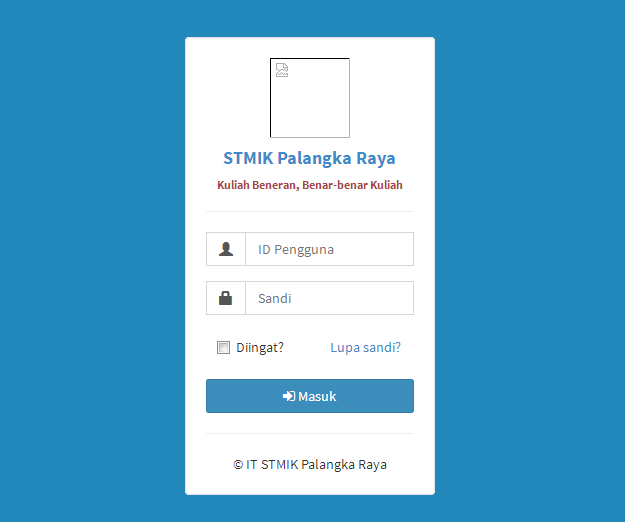
1. Pengujian *login* pada *website server*

Hasil pengujian ini dilakukan untuk mengetahui apakah fungsi login pada *website server* sudah berjalan dengan baik dan sesuai harapan. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel hasil pengujian login

Tabel 8. Hasil Pengujian *login*

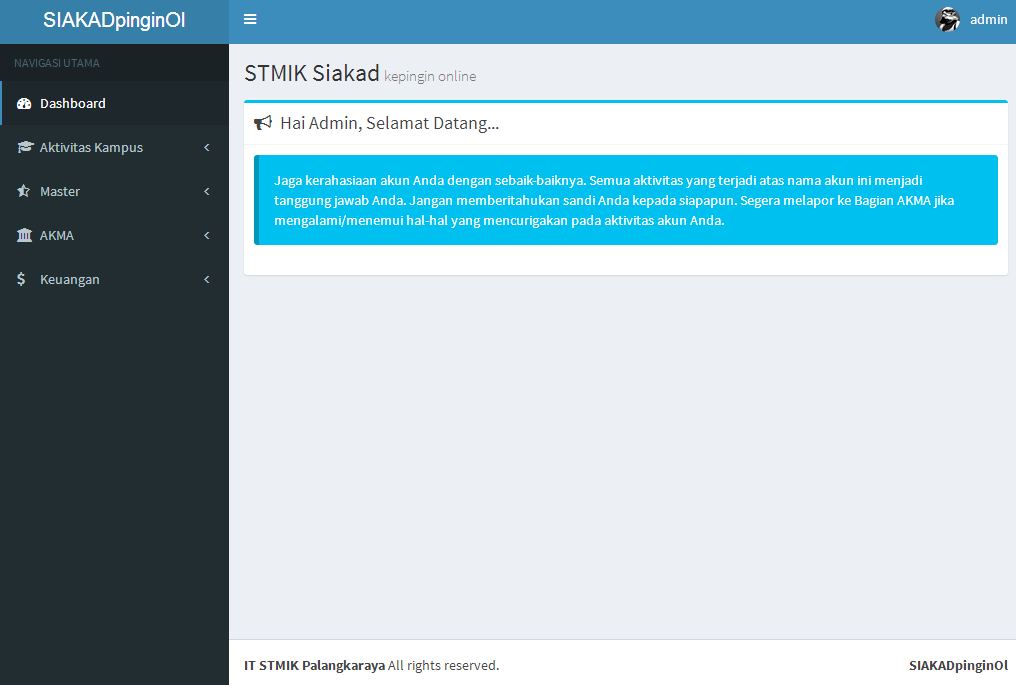
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian *login* admin | Pengguna memasukkan *username* dan *password* kemudian menekan tombol login | Website akan menampilkan halaman *dashboard* untuk admin | Halaman *dashboard* admin tampil setelah tombol login ditekan | Sesuai |

Adapun tampilan dari *implementasi* *login* pada *website server* adalah sebagai berikut :



Gambar 49. *form login* pada *website server*

Dalam pengujian ini akan tampil halaman *dashboard* setelah meng-*input* *username* dan *password* dengan benar. Berikut tampilan dashboard setelah proses login selesai:



Gambar 50. Halaman dashboard

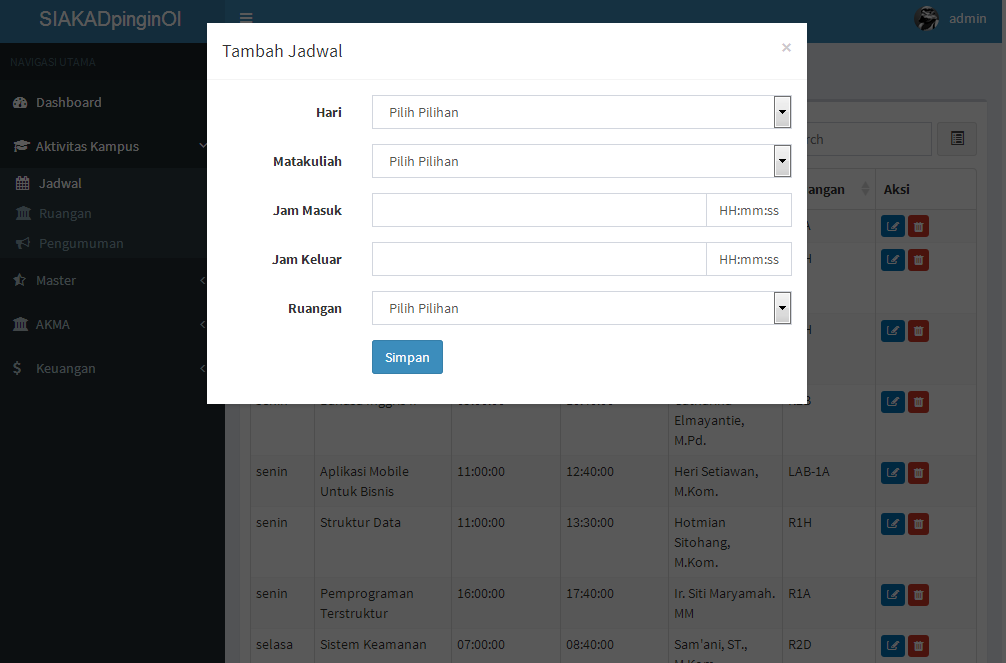
1. Pengujian tambah jadwal

Hasil pengujian ini dilakukan untuk mengetahui fungsi tambah jadwal telah berjalan dengan baik dan benar. Adapun hasil dari pengujian tersebut dapat dilihat pada tabel

Tabel 9. Hasil pengujian tambah jadwal

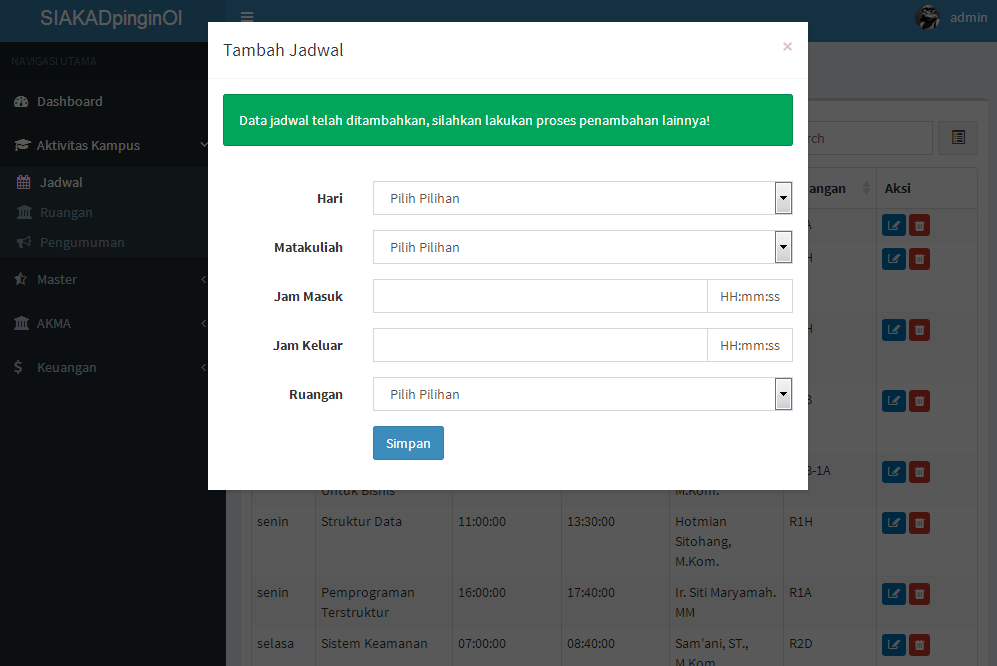
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian tambah jadwal | Admin memasukan data jadwal pada *form* tambah jadwal | Data jadwal bertambah | Muncul pesan “jadwal telah ditambahkan” dan data jadwal telah bertambah | Sesuai |

Adapun tampilan dari *implementasi* tambah jadwalpada *website server* adalah sebagai berikut :



Gambar 51. form tambah jadwal

Dalam pengujian ini akan bertambah data jadwal setelah admin meng-input data pada form tambah jadwal. Berikut tampilan setelah admin berhasil menambahkan data jadwal :



Gambar 1. Tampilan setelah berhasil tambah data

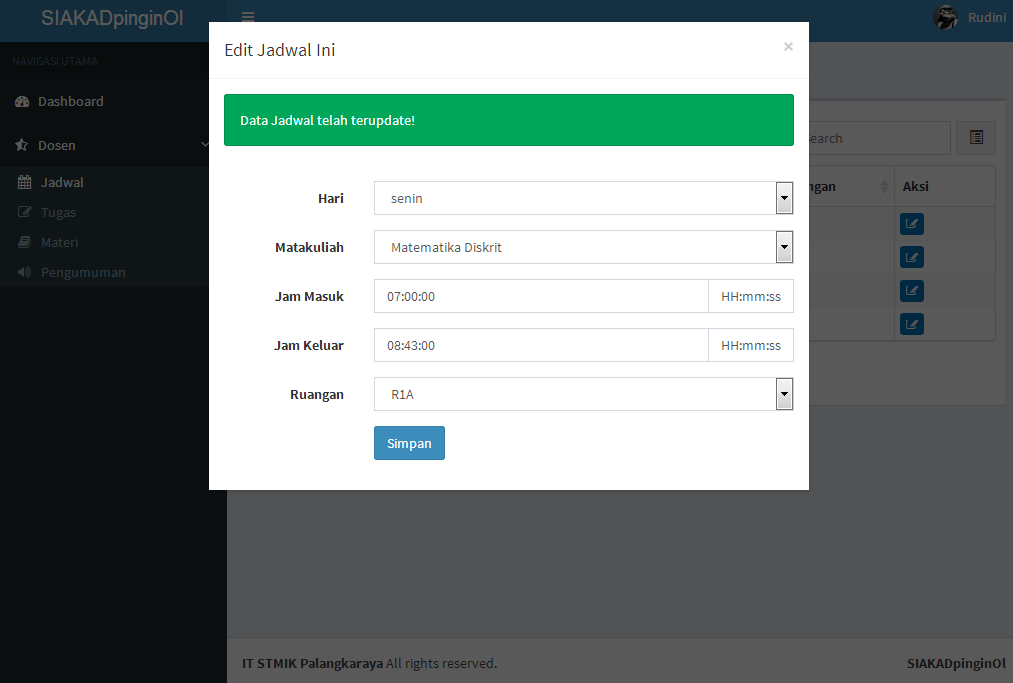
1. Pengujian edit jadwal

Hasil pengujian ini dilakukan untuk mengetahui apakah fungsi edit jadwal sudah berjalan dengan baik atau tidak. Berdasarkan hasil pengujian yang dilakukan, fitur edit jadwal sudah berjalan dengan baik. Adapun hasil dari pengujian tersebut dapat dilihat pada tabel 8.

Tabel 9. Hasil Pengujian edit jadwal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian edit jadwal | Dosen memilih data jadwal yang hendak dirubah dan mengubah data jadwal pada form edit. | Data jadwal berubah sesuai dengan perubahan yang dilakukan dosen | Muncul pesan “jadwal telah ter-*update*” pada form edit jadwal dan data jadwal telah berubah | Sesuai |

Dalam pengujian ini data jadwal akan berubah setelah dosen meng-*edit* data pada form edit jadwal. Berikut tampilan saat dosen mengubah data jadwal pada form edit jadwal :



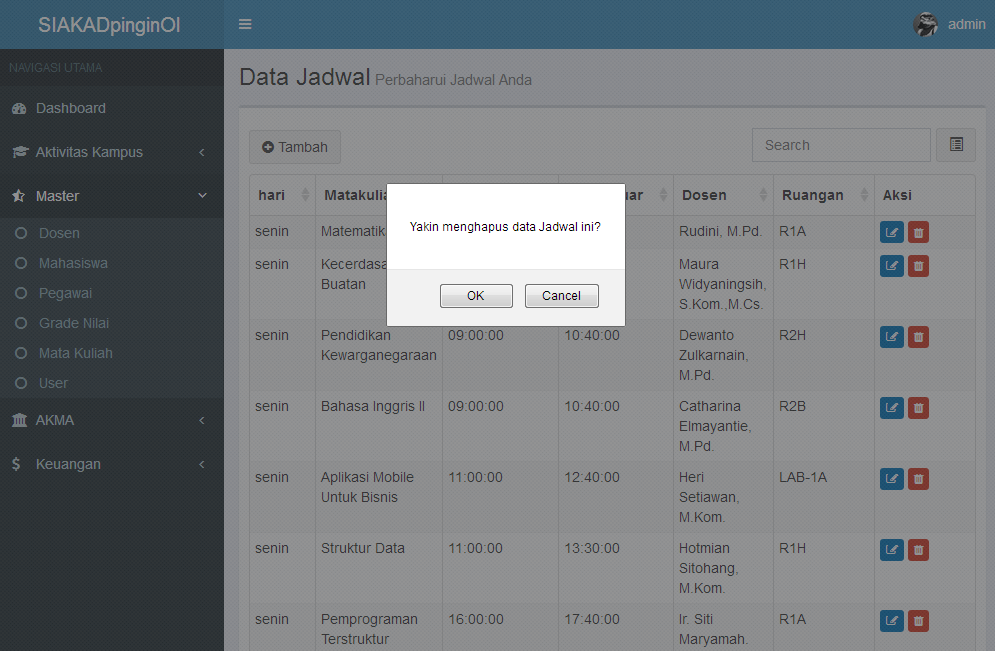
1. Pengujian hapus data jadwal

Hasil pengujian ini dilakukan untuk mengetahui fungsi hapus jadwal telah berjalan dengan baik dan benar. Adapun hasil dari pengujian tersebut dapat dilihat pada tabel

Tabel 10. Hasil pengujian hapus jadwal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian hapus data jadwal | Admin menekan tombol hapus | Data jadwal terhapus | Data jadwal terhapus setelah menekan tombol yes pada kotak dialog peringatan | Sesuai |

Dalam pengujian ini admin menekan tombol hapus, kemudian akan muncul kotak dialog untuk memvalidasi apakah admin memang ingin menghapus jadwal, apabila iya maka tekan tombol yes pada kotak dialog dan data jadwal akan terhapus. Adapun tampilan dari *implementasi* hapus jadwalpada *website server* adalah sebagai berikut :



Gambar 2. implementasi hapus jadwal

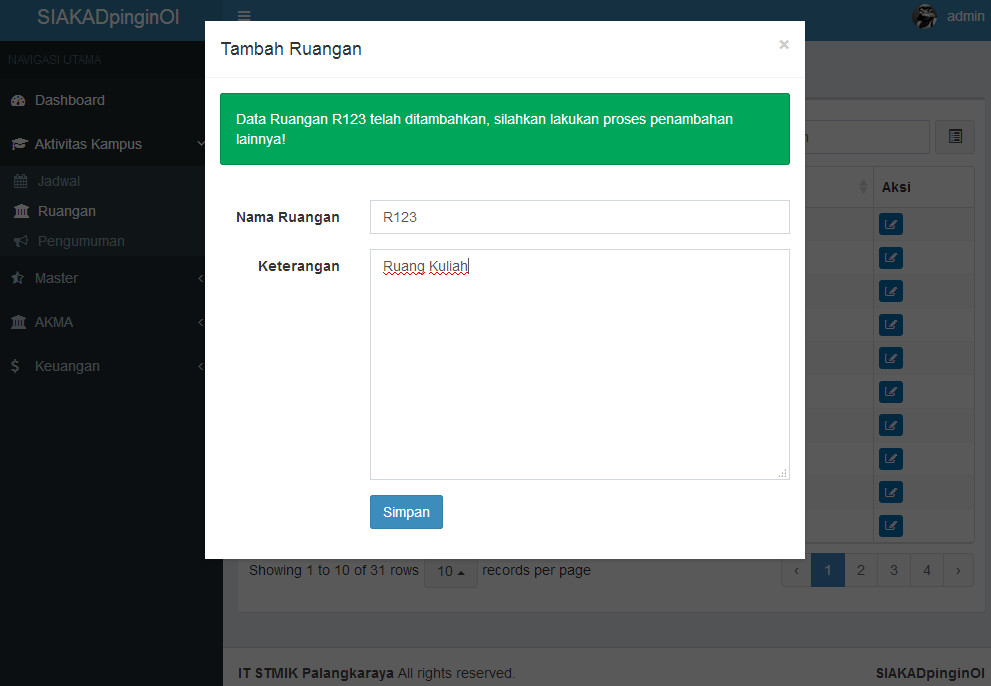
1. Pengujian tambah ruangan

Hasil pengujian ini dilakukan untuk mengetahui fungsi tambah ruangan telah berjalan dengan baik dan benar. Adapun hasil dari pengujian tersebut dapat dilihat pada tabel

Tabel 11. Hasil pengujian tambah ruangan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian tambah ruangan | Admin menekan tombol tambah ruangan | Data ruangan bertambah | Muncul pesan “ruangan telah ditambahkan” dan data ruangan telah bertambah pada tabel daftar ruangan | Sesuai |

Dalam pengujian ini data ruangan akan bertambah setelah admin meng-input data pada form tambah ruangan. Adapun tampilan dari *implementasi* tambah ruangan adalah sebagai berikut :

Gambar . implementasi tambah ruangan

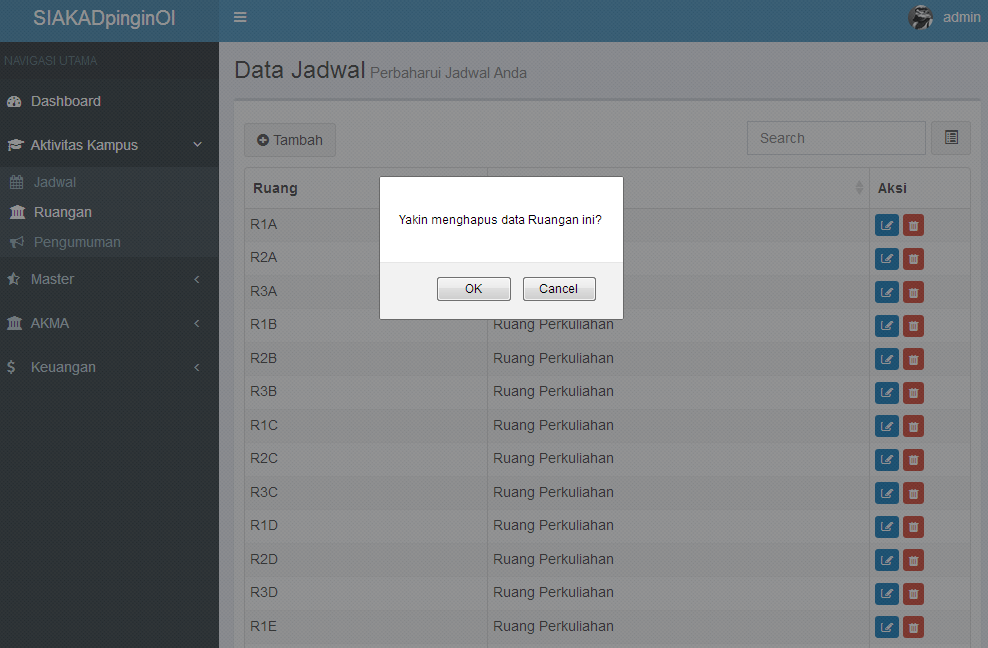
1. Pengujian hapus ruangan

Hasil pengujian ini dilakukan untuk mengetahui fungsi hapus jadwal telah berjalan dengan baik dan benar. Adapun hasil dari pengujian tersebut dapat dilihat pada tabel

Tabel 10. Hasil pengujian hapus jadwal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian hapus data jadwal | Admin menekan tombol hapus | Data jadwal terhapus | Data jadwal terhapus setelah menekan tombol yes pada kotak dialog peringatan | Sesuai |

Dalam pengujian ini admin menekan tombol hapus, kemudian akan muncul kotak dialog untuk memvalidasi apakah admin memang ingin menghapus ruangan, apabila iya maka tekan tombol yes pada kotak dialog dan data ruangan akan terhapus. Adapun tampilan dari *implementasi* hapus ruangan adalah sebagai berikut :

Gambar . implementasi hapus ruangan

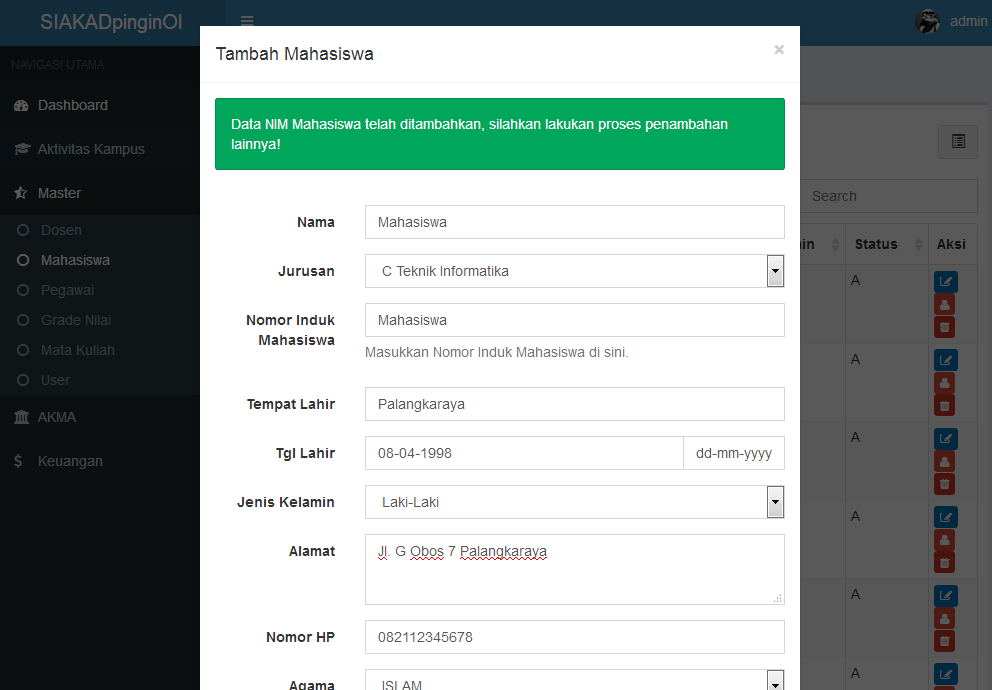
1. Pengujian tambah mahasiswa

Hasil pengujian ini dilakukan untuk mengetahui fungsi tambah mahasiswa telah berjalan dengan baik dan benar. Adapun hasil dari pengujian tersebut dapat dilihat pada tabel

Tabel 11. Hasil pengujian tambah mahasiswa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian tambah mahasiswa | Admin meng-input data tugas pada form tambah mahasiswa | Data mahasiswa bertambah | Muncul pesan “mahasiswa telah ditambahkan” dan data mahasiswa telah bertambah pada tabel daftar mahasiswa | Sesuai |

Dalam pengujian ini data mahasiswa akan bertambah setelah admin meng-input data pada form tambah mahasiswa. Adapun tampilan dari *implementasi* tambah matakuliah adalah sebagai berikut :



Gambar . implementasi tambah mahasiswa

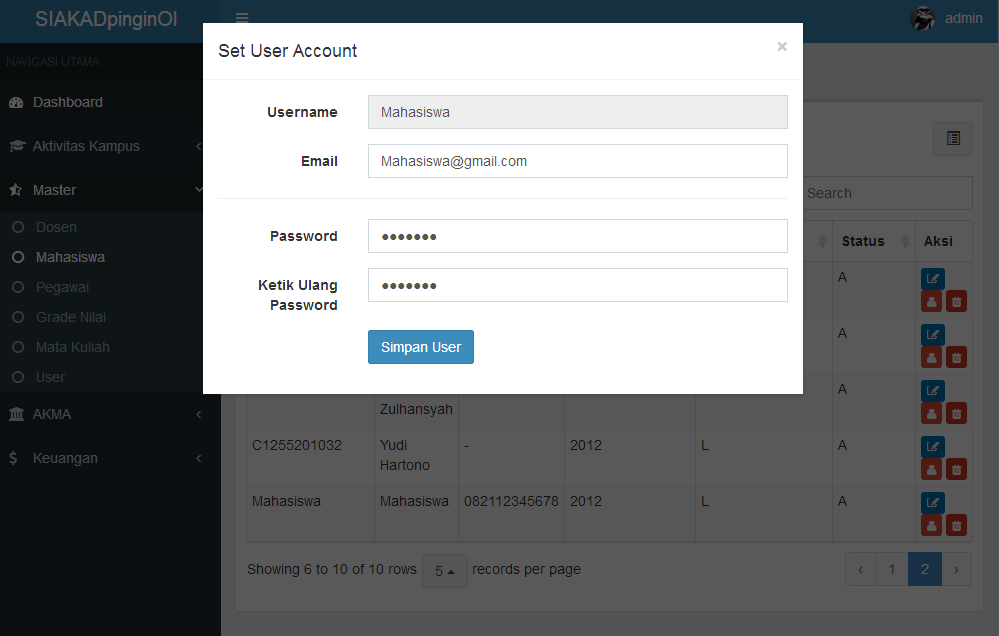
1. Pengujian *Set User Account*

Hasil pengujian ini dilakukan untuk mengetahui fungsi set user account telah berjalan dengan baik dan benar. Set user account adalah menu untuk menambahkan atau mengelola username dan password mahasiswa serta dosen. Adapun hasil dari pengujian tersebut dapat dilihat pada tabel

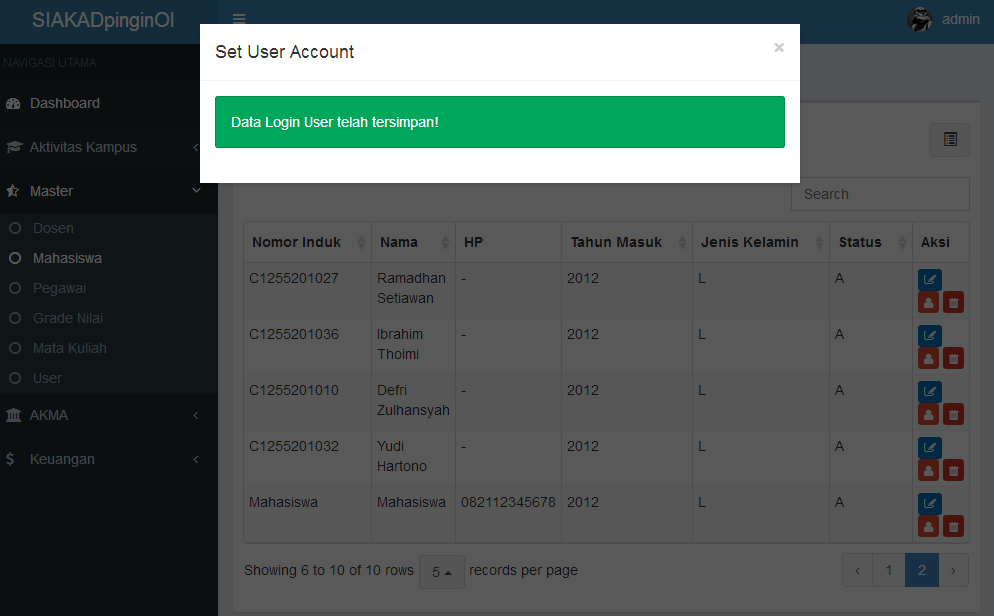
Tabel 11. Hasil pengujian *set user account*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian *set user account* | Admin meng-input data username dan password pada form *set user account* | Data account user bertambah atau berubah | Muncul pesan data login user telah tersimpan dan data *user account* telah bertambah | Sesuai |

Dalam pengujian ini data pada tabel *user* akan bertambah atau berubah setelah admin meng-input data pada form *set user account*. Adapun tampilan dari *implementasi* *set user account* adalah sebagai berikut :



Gambar . implementasi set user account



Gambar . pesan setelah set user account berhasil

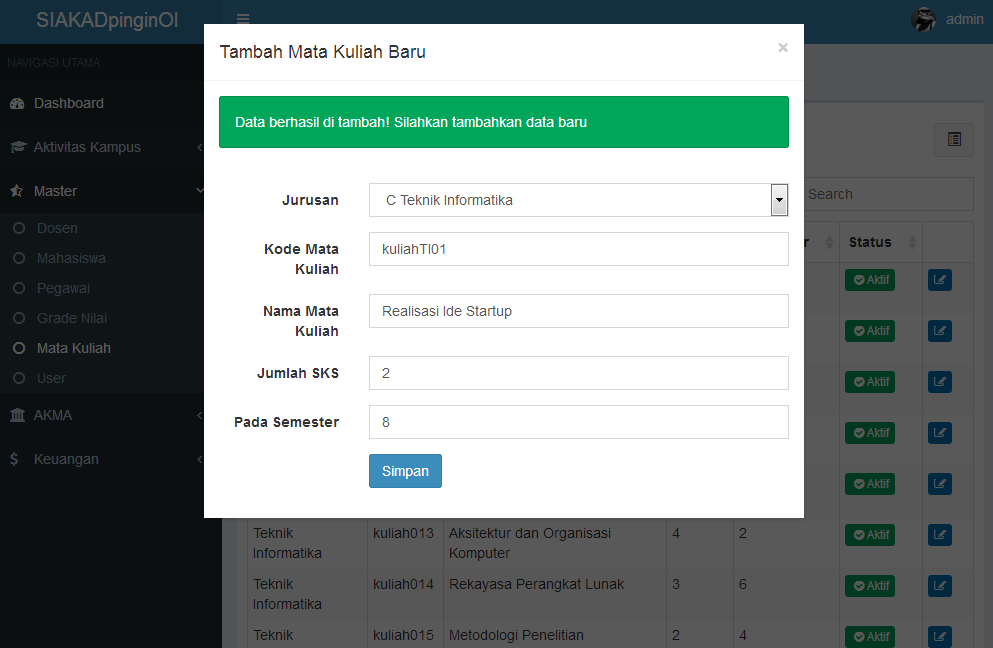
1. Pengujian tambah matakuliah

Hasil pengujian ini dilakukan untuk mengetahui fungsi tambah matakuliah telah berjalan dengan baik dan benar. Adapun hasil dari pengujian tersebut dapat dilihat pada tabel

Tabel 11. Hasil pengujian tambah tugas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian tambah matakuliah | Admin meng-input data tugas pada form tambah matakuliah | Data matakuliah bertambah | Muncul pesan “matakuliah telah ditambahkan” dan data matakuliah telah bertambah pada tabel daftar matakuliah | Sesuai |

Dalam pengujian ini data matakuliah akan bertambah setelah admin meng-input data pada form tambah matakuliah. Adapun tampilan dari *implementasi* tambah matakuliah adalah sebagai berikut :



Gambar 8. implementasi tambah matakuliah

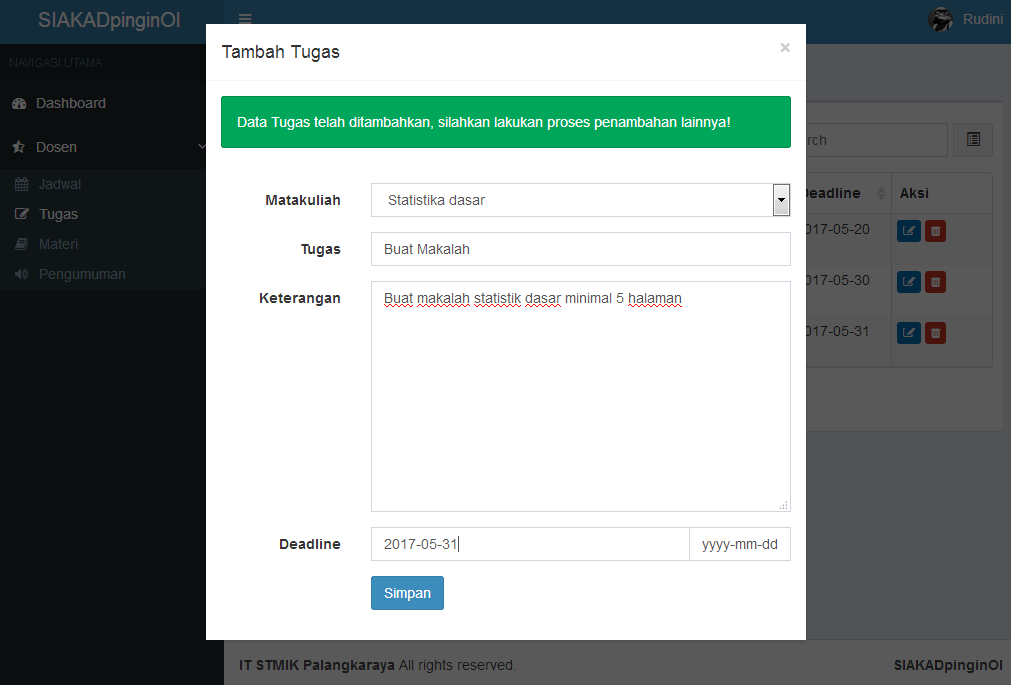
1. Pengujian tambah tugas

Hasil pengujian ini dilakukan untuk mengetahui fungsi tambah tugas telah berjalan dengan baik dan benar. Adapun hasil dari pengujian tersebut dapat dilihat pada tabel

Tabel 11. Hasil pengujian tambah tugas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian tambah tugas | Dosen meng-input data tugas pada form tambah tugas | Data tugas bertambah | Muncul pesan “tugas telah ditambahkan” dan data tugas telah bertambah pada tabel daftar tugas | Sesuai |

Dalam pengujian ini data tugas akan bertambah setelah dosen meng-input data pada form tambah tugas. Adapun tampilan dari *implementasi* tambah tugas adalah sebagai berikut :

Gambar . implementasi tambah tugas

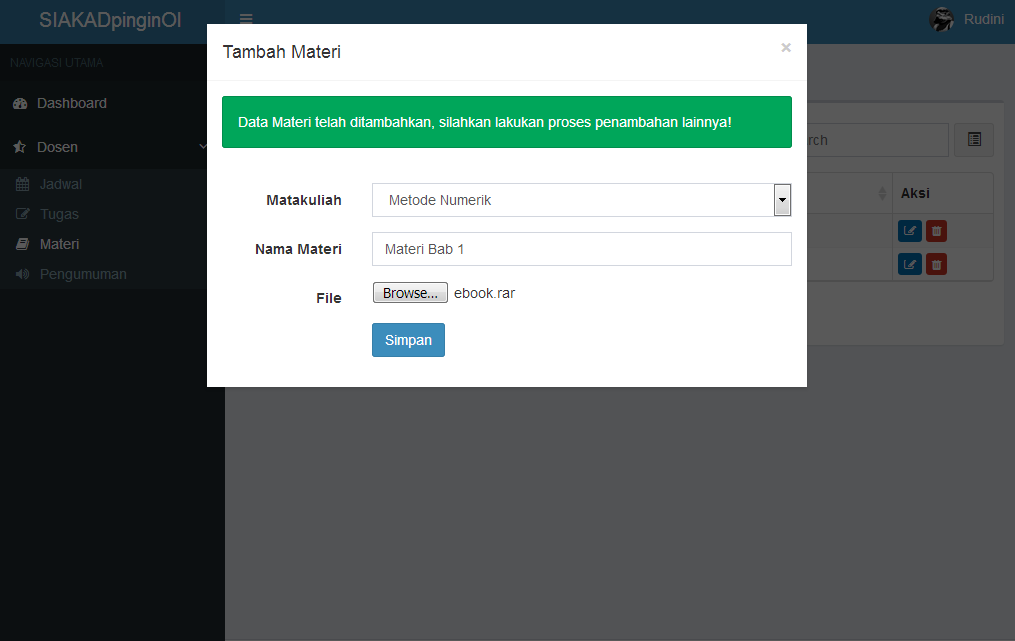
1. Pengujian upload materi

Hasil pengujian ini dilakukan untuk mengetahui fungsi upload materi telah berjalan dengan baik dan benar. Adapun hasil dari pengujian tersebut dapat dilihat pada tabel

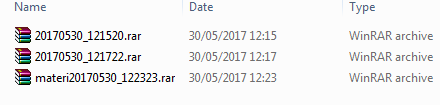
Tabel 11. Hasil pengujian tambah tugas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian upload materi | Dosen meng-input file materi pada form upload materi | Data materi bertambah dan file materi tesimpan pada folder materi | Muncul pesan “materi telah ditambahkan” dan file materi telah tersimpan pada folder materi | Sesuai |

Dalam pengujian ini data materi akan bertambah dan file materi akan tersimpan dalam folder materi di dalam server setelah dosen meng-input data pada form upload materi. Adapun tampilan dari *implementasi* tambah tugas adalah sebagai berikut :

Gambar . implementasi upload materi

Adapun hasil file yang di upload kedalam server akan tersimpan pada folder materi, isi folder materi dapat dilihat pada gambar

Gambar . file materi yang tersimpan dalam server

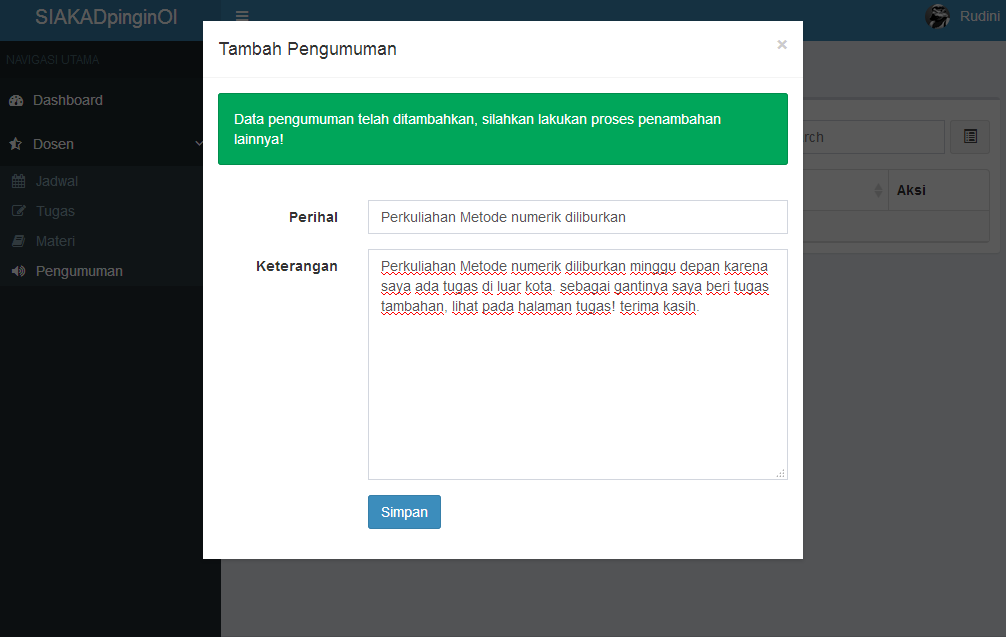
1. Pengujian tambah pengumuman

Hasil pengujian ini dilakukan untuk mengetahui fungsi tambah oengumuman telah berjalan dengan baik dan benar. Adapun hasil dari pengujian tersebut dapat dilihat pada tabel

Tabel 11. Hasil pengujian tambah pengumuman

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian tambah pengumuman | Dosen meng-input data pengumuman pada form pengumuman | Data pengumuman bertambah | Muncul pesan “ pengumuman telah ditambahkan” dan data pengumuman bertambah pada tabel daftar pengumuman | Sesuai |

Dalam pengujian ini data pengumuman akan bertambah setelah dosen meng-input data pada form tambah pengumuman. Adapun tampilan dari *implementasi* tambah pengumuman adalah sebagai berikut :



Gambar . implementasi tambah pengumuman

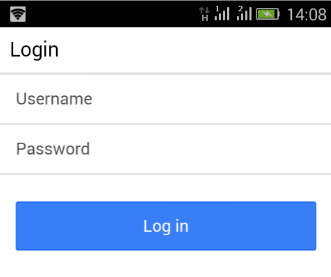
1. Pengujian login aplikasi client

Hasil pengujian ini dilakukan untuk mengetahui apakah fungsi login pada *website server* sudah berjalan dengan baik dan sesuai harapan. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel hasil pengujian login

Tabel 8. Hasil Pengujian *login* aplikasi client

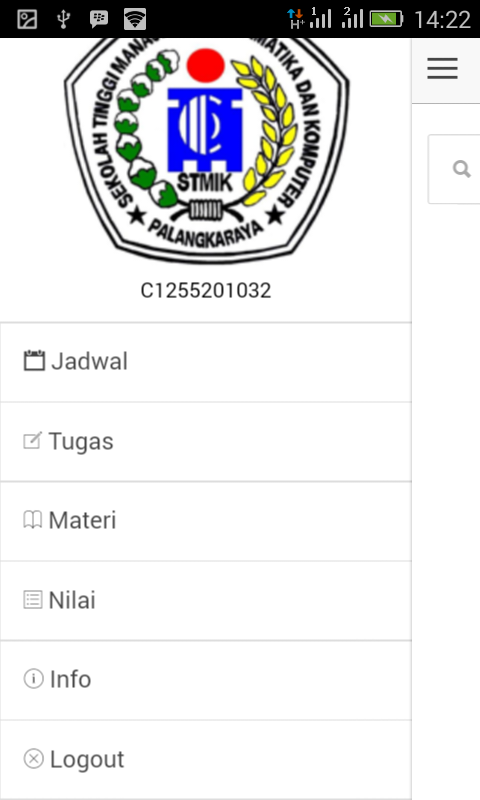
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian *login* mahasiswa pada aplikasi client | Pengguna memasukkan *username* dan *password* kemudian menekan tombol login | aplikasi akan menampilkan side menu aplikasi client | Halaman side menu aplikasi muncul | Sesuai |

Adapun tampilan dari *implementasi* *login* pada aplikasi client adalah sebagai berikut :



Gambar . implementasi login aplikasi client

Adapun tampilan dari *side menu* pada aplikasi client adalah sebagai berikut :



Gambar . implementasi side menu

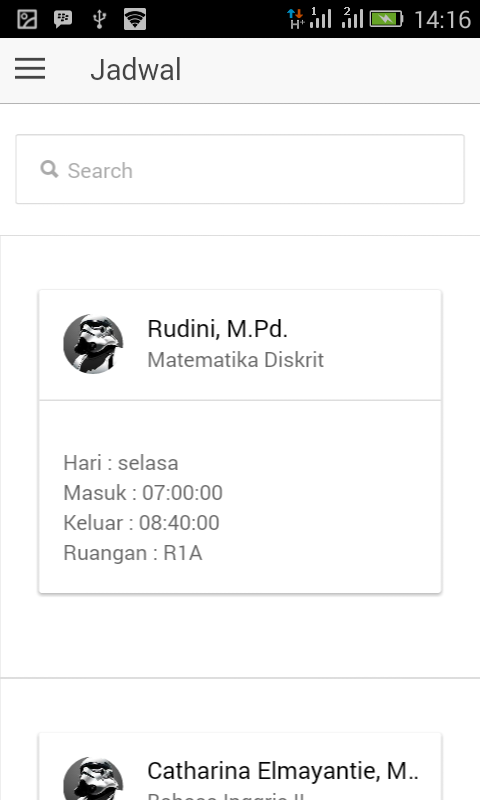
1. Pengujian menu jadwal aplikasi client

Hasil pengujian ini dilakukan untuk mengetahui apakah fungsi menu jadwal pada aplikasi client sudah berjalan dengan baik dan sesuai harapan. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel hasil pengujian login

Tabel 8. Hasil Pengujian menu jadwalaplikasi client

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian menu jadwal aplikasi client | Pengguna menekan menu jadwal | aplikasi akan menampilkan halamanjadwal | Halaman jadwal tampil setelah loading | Sesuai |

Adapun tampilan dari *implementasi* menu jadwal pada aplikasi client adalah sebagai berikut :



Gambar . implementasi menu jadwal

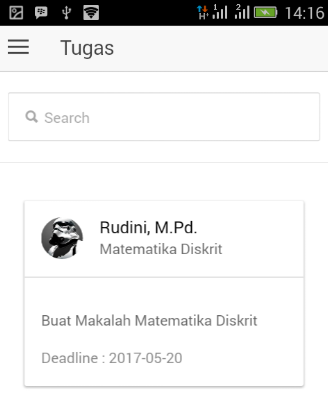
1. Pengujian menu tugas aplikasi client

Hasil pengujian ini dilakukan untuk mengetahui apakah fungsi menu jadwal pada aplikasi client sudah berjalan dengan baik dan sesuai harapan. Adapun hasil dari pengujian tersebut dapat dilihat pada Tabel hasil pengujian login

Tabel 8. Hasil Pengujian menu jadwalaplikasi client

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hasil Pengujian** | | | | |
| **Nama pengujian** | **Data Masukan** | **Harapan** | **Pengamatan** | **Kesimpulan** |
| Pengujian menu jadwal aplikasi client | Pengguna menekan menu jadwal | aplikasi akan menampilkan halamanjadwal | Halaman jadwal tampil setelah loading | Sesuai |

Adapun tampilan dari *implementasi* menu tugas pada aplikasi client adalah sebagai berikut :



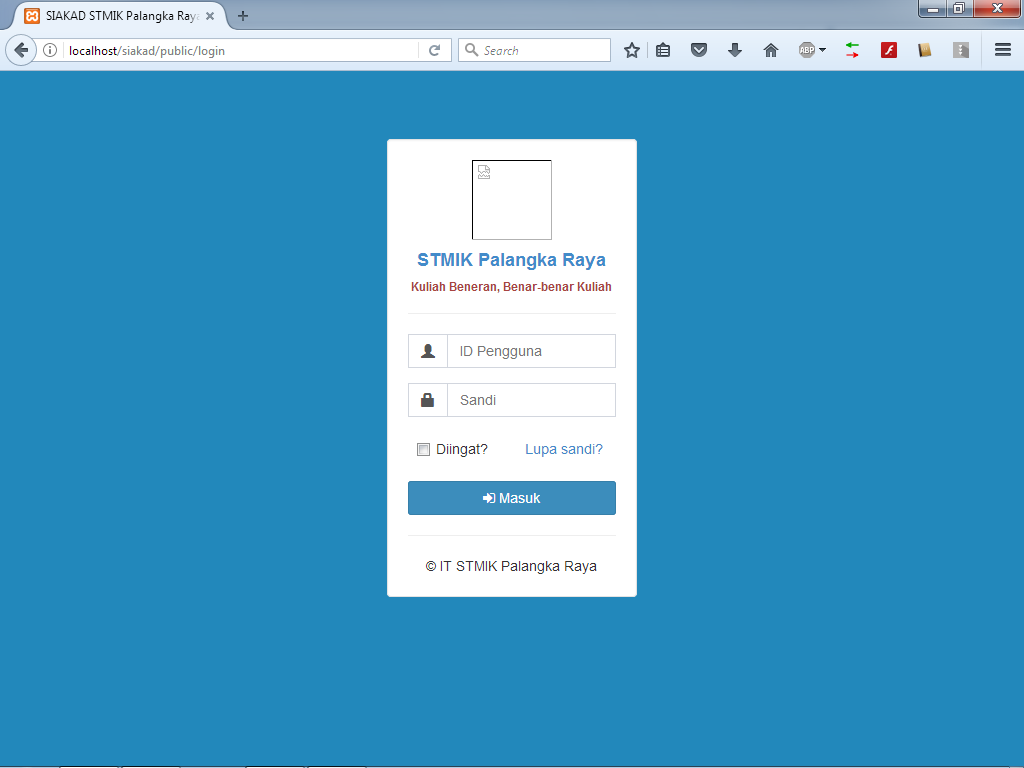
Gambar . implementasi menu tugas

1. Pengujian menu nilai aplikasi client
2. Pengujian menu informasi / pengumuman aplikasi client
3. Pengujian download materi aplikasi client
4. Manual program

Tahap ini akan menjelaskan bagaimana penggunaan / manual dari Aplikasi Pengelolaan Mata Kuliah Teknik Informatika Berbasis *Client Server* Pada STMIK Palangkaraya secara keseluruhan, sehingga dapat memudahkan pengguna untuk menggunakan aplikasi. Pada tahap ini akan dijelaskan menu – menu yang terdapat pada aplikasi. Adapun penjelasan masing-masing menu adalah sebagai berikut :

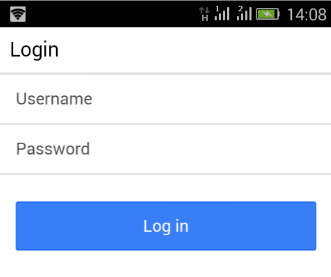
1. Login

Untuk menggunakan Aplikasi Pengelolaan Mata Kuliah Teknik Informatika Berbasis *Client Server* Pada STMIK Palangkaraya, pengguna harus melakukan login terlebih dahulu dengan memasukkan username dan password pada halaman login. Untuk pengguna dengan hak akses sebagai admin atau dosen maka pengguna login pada website server. Berikut tampilan login server :



Gambar 50. login website server

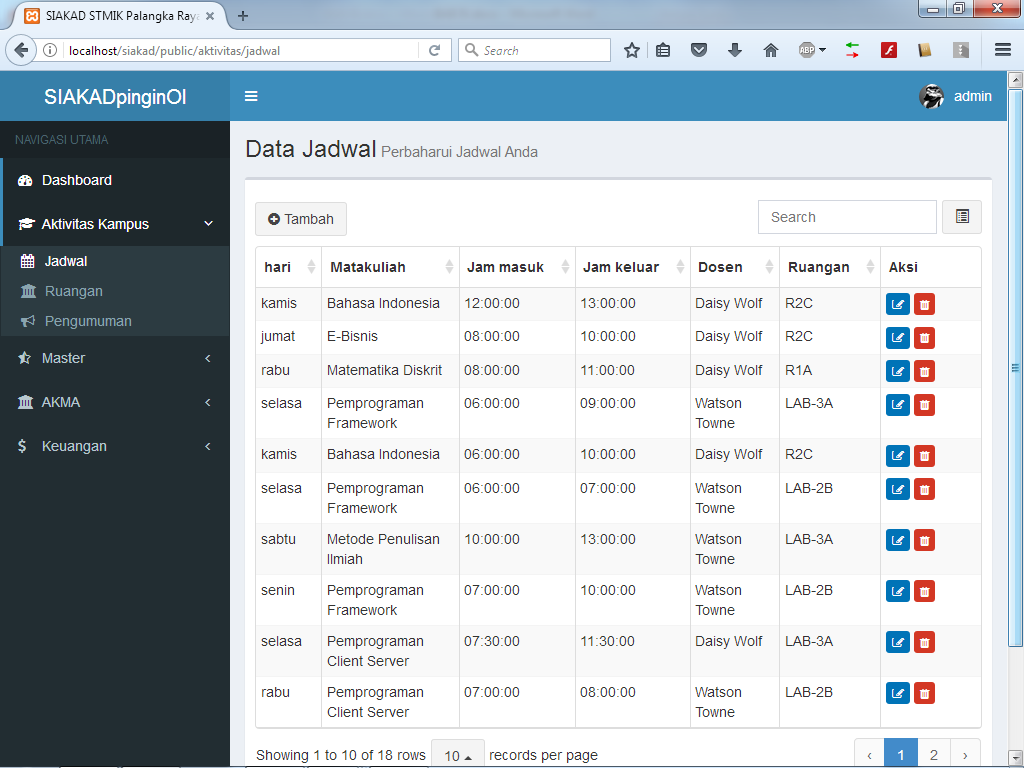
Sedangkan untuk pengguna dengan hak akses mahasiswa maka harus login pada aplikasi client pada smartphone android. Berikut tampilan halaman login aplikasi client :



Gambar 51. Halaman login aplikasi client

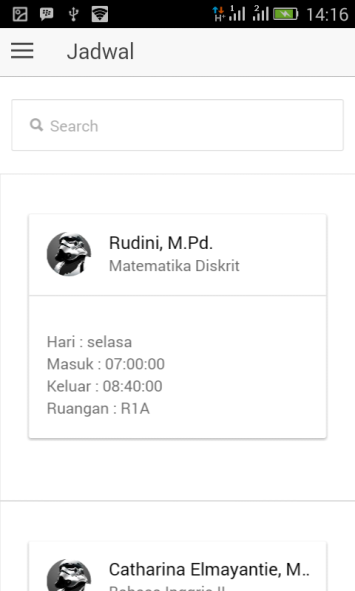
1. Menu jadwal matakuliah

Pada menu jadwal matakuliah pengguna yang memiliki hak akses admin dapat menambahkan, mengedit dan menghapus jadwal matakuliah. Sedangkan pengguna yang memiliki hak akses sebagai dosen hanya dapat mengedit jadwal matakuliah saja. Tampilan halaman jadwal pada website server dapat dilihat pada gambar 52



Gambar 52 halaman untuk mengelola jadwal

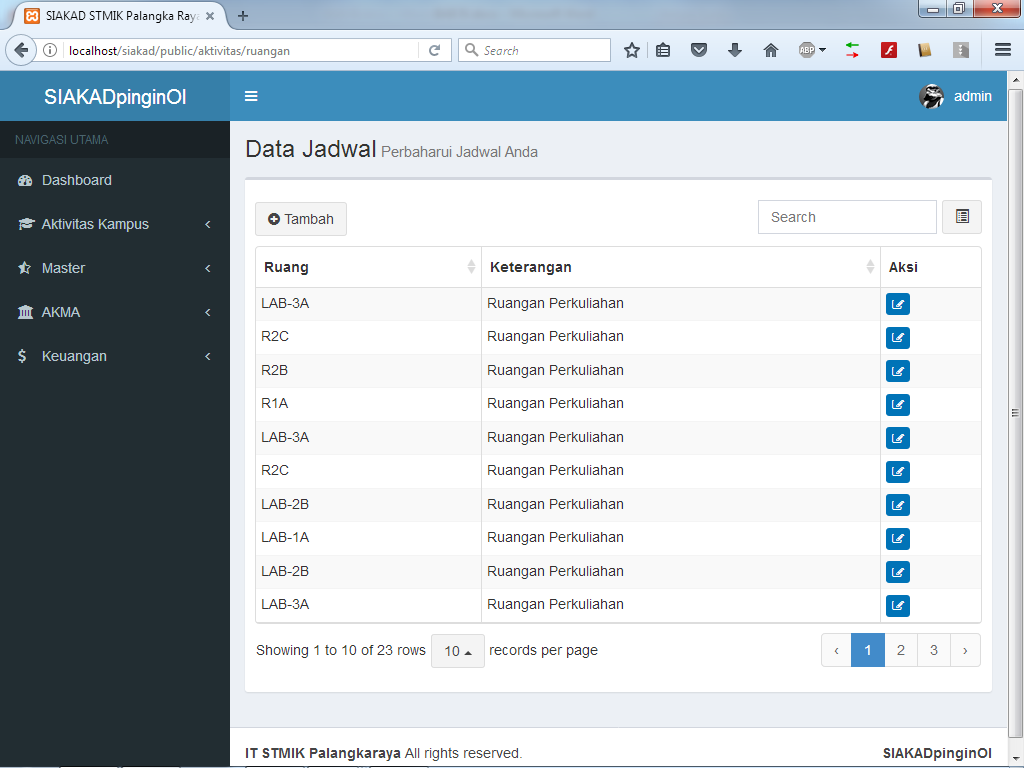
Untuk pengguna yang memiliki hak akses sebagai mahasiswa maka dapat melihat data jadwal dari server lewat aplikasi client pada smartphone android. Tampilan menu jadwal aplikasi client dapat dilihat pada gambar berikut



Gambar 17. menu jadwal aplikasi client

1. Menu ruangan

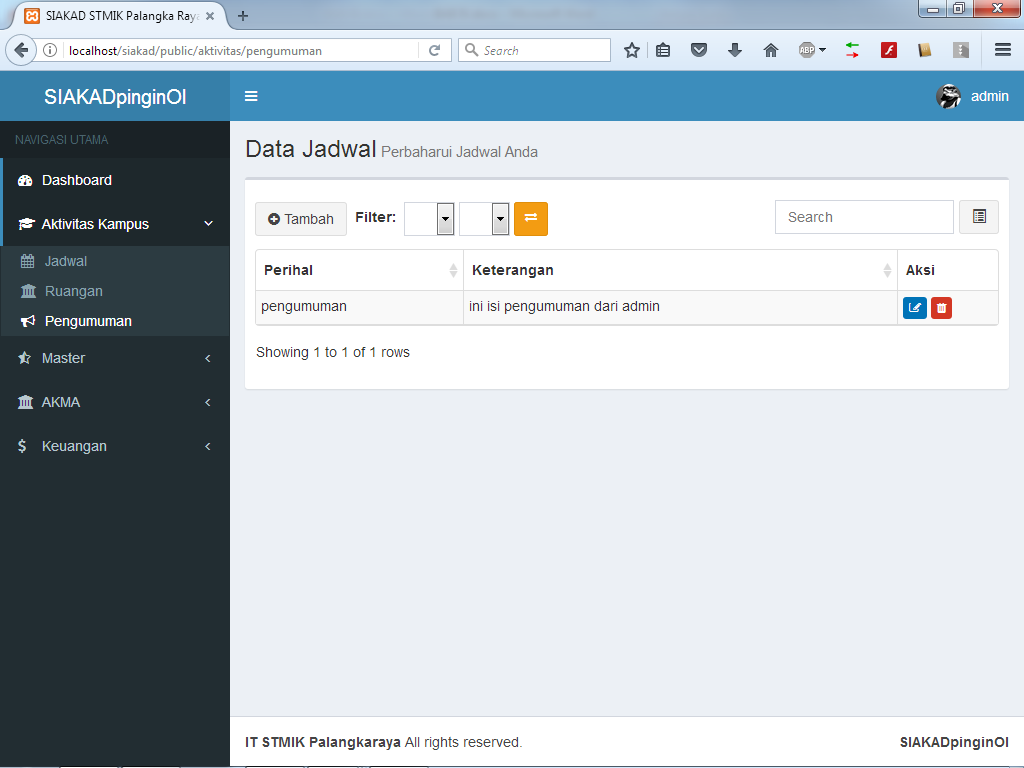
Menu ruangan hanya dapat digunnakan oleh admin untuk mengelola ruangan, admin dapat menambah dan menghapus ruangan yang akan dipakai oleh jadwal matakuliah. Halaman ruangan dapat dilihat pada gambar 53



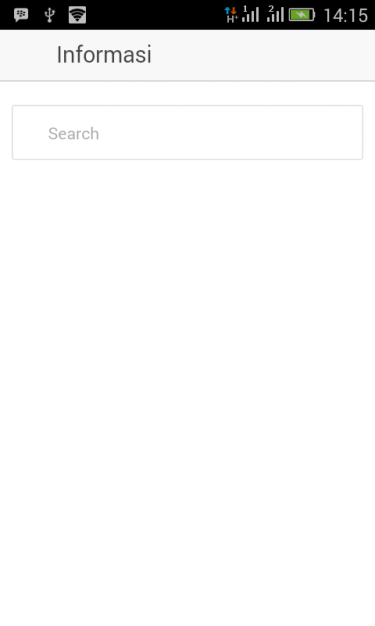
Gambar 53. Halaman ruangan

1. Menu pengumuman

Menu pengumuman digunnakan oleh pengguna dengan hak akses admin atau dosen untuk menambahkan dan menghapus pengumuman. Halaman pengumuman dapat dilihat pada gambar 54

Gambar 54. halaman pengumuman

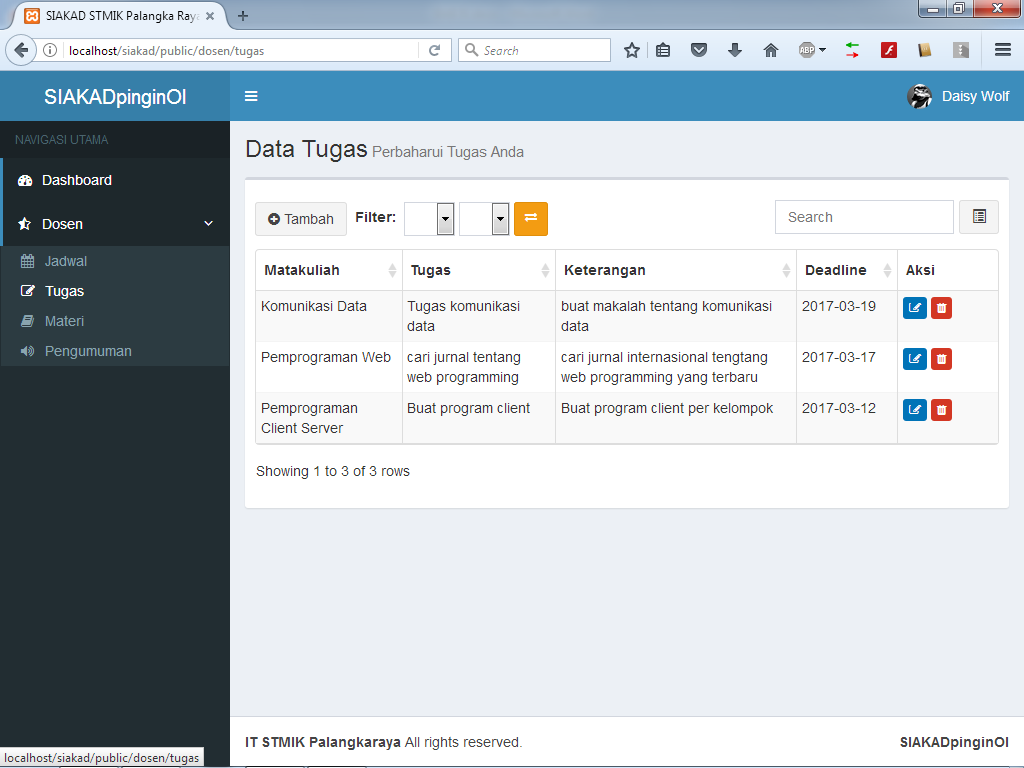
Sedangkan untuk pengguna dengan hak akses sebagai mahasiswa dapat melihat data pengumuman lewat menu informasi pada aplikasi client. berikut Tampilan menu pengumuman atau informasi dapat dilihat pada gambar



Gambar 18. menu pengumuman aplikasi client

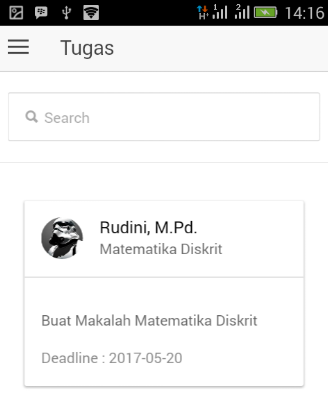
1. Menu tugas

Menu tugas digunakan oleh pengguna dengan hak akses dosen untuk mengelola tugas seperti menambah, mengedit dan mengahapus tugas. Berikut adalah tampilan halaman tugas yang dikelola dosen ditunjukkan pada gambar



Gambar 55. Menu tugas pada server

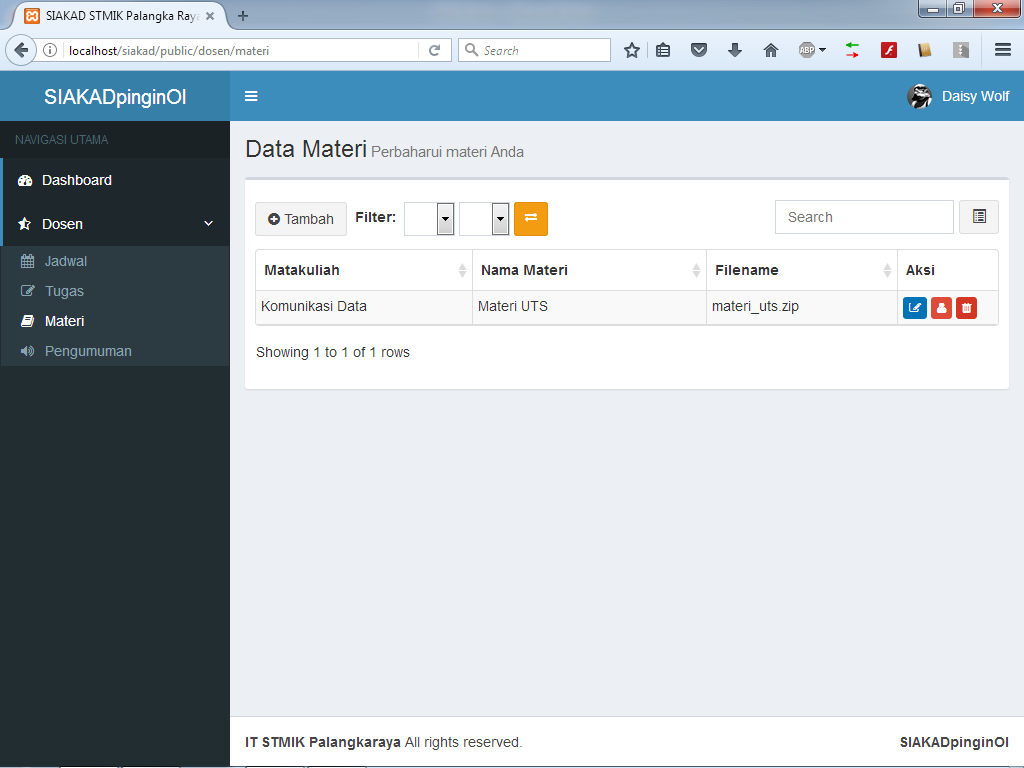
Sedangkan untuk pengguna dengan hak akses sebagai mahassiwa dapat melihat data tugas lewat aplikasi client pada smartphone. Berikut adalah tampilan menu tugas pada aplikasi client ditunjukkan pada gambar



Gambar 19. menu tugas pada aplikasi client

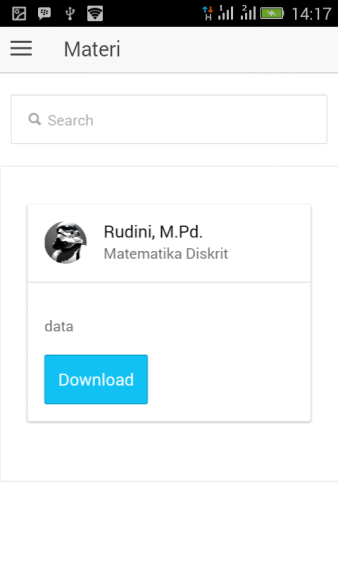
1. Menu materi

Menu materi digunakan oleh pengguna dengan hak akses sebagai dosen untuk mengunggah atau *upload* materi kuliah. Menu materi dapat dilihat pada gambar 56.



Gambar 56. Menu materi pada server

Sedangkan untuk pengguna yang memiliki hak akses sebagai mahasiswa dapat mendownload materi kuliah pada menu materi di dalam aplikasi client. Tampilan menu materi pada aplikasi client dapat dilihat pada gambar



Gambar 20. menu materi pada aplikasi client

1. Manual instalasi

Pada manual instalasi ini akan dijelaskan tentang bagaimana cara instalasi website server pada server lokal di komputer bersistem operasi *windows 7* dan instalasi aplikasi client pada sistem operasi *android*.

1. Manual instalasi website server
2. Copy file “siakad.rar” ke komputer dan ekstrak kedalam *folder* C:\xampp\htdocs
3. Kemudian jalankan aplikasi xampp dan buka command prompt, masuk kedalam path C:\xampp\htdocs\siakad lalu keitikkan perintah php artisan serve.
4. Manual instalasi aplikasi *client*
5. *Copy file* “*stmik.apk*” ke *SD card* atau *internal smartphone android* melalui kabel data, *bluetooth* atau melalui media lainnya.
6. Tap *file* “*stmik.apk*” yang sudah dipindahkan, maka instalasi aplikasi akan berjalan. Tunggu hingga proses instalasi selesai.
7. Lihat apakah *icon* doa Stmik palangkaraya sudah muncul pada layar *smartphone android* pengguna. Apabila sudah ada, aplikasi sudah siap dijalankan.
8. Pemeliharaan sistem

Pemeliharaan aplikasi perlu dilakukan agar sistem dapat berjalan lebih baik dan bertahan lebih lama, dilakukan penjagaan dari *asset* aplikasi itu sendiri maupun komponennya. Berikut kiat dan pemeliharaan aplikasi:

1. Nama *file* aplikasi asli sebaiknya tidak diganti karena akan mengalami masalah dalam kompresi *file* aslinya.
2. Penyimpanan *asset* aplikasi sangat penting, karena akan memudahkan *programmer* dalam menyusun kembali aplikasi apabila terjadi perubahan, *error,* maupun pengembangan ke tahap selanjutnya.
3. Melakukan *backup* aplikasi untuk menghindari kehilangan aplikasi.
4. Melakukan *backup database* untuk menhidari kehilangan data.
5. **Pembahasan**
6. Pembahasan *listing* kode program
7. Listing program koneksi database

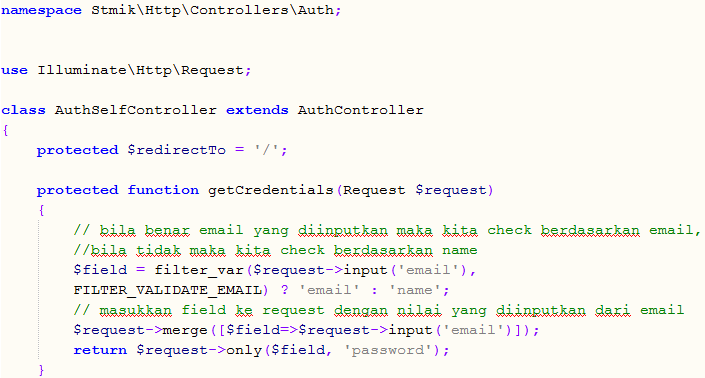
Untuk menampilkan data – data dari database server perlu melakukan koneksi dengan database. Kode program untuk melakukan koneksi dengan database dapat dilihat pada gambar

Gambar 21. listing program koneksi database

Pada listing program diatas program akan mengkoneksikan server dengan database yang digunakan. pada pengembangan Aplikasi ini penulis database yang digunakan adalah mysql. Variabel yang perlu di tentukan dalam program adalah nama database,username database, password database dan host atau tempat database di jalankan.

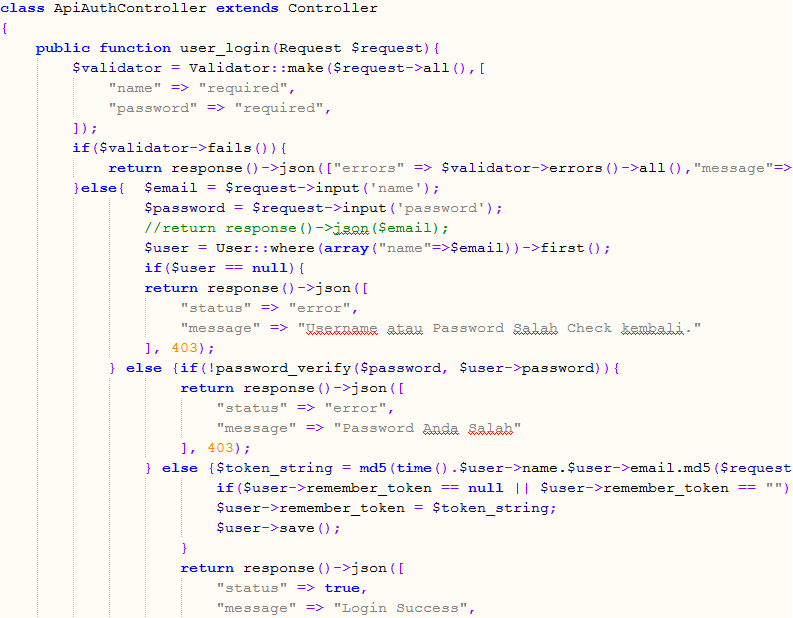
1. Listing program untuk login

Kode program untuk login dibagi menjadi dua bagian, yaitu bagian kode program login melalui server dan kode program login melalui aplikasi client. Listing kode program untuk login melalui server dapat dilihat pada gambar



Gambar 22. listing kode program login untuk server

Penjelasan dari kode program pada gambar diatas akan memeriksa login berdasarkan email atau nama yang di inputkan oleh pengguna dan mencocokkannya dengan email atau nama pengguna yang ada didalam database. Setelah itu program akan memerika password yang di input oleh pengguna. Jika yang di inputkan oleh pengguna cocok dengan yang ada di dalam database maka pengguna akan di arahkan ke halaman dashboard pengguna, jika salah maka akan kembali ke halaman login. sedangkan kode program untuk bagian login aplikasi client dapat dilihat pada gambar

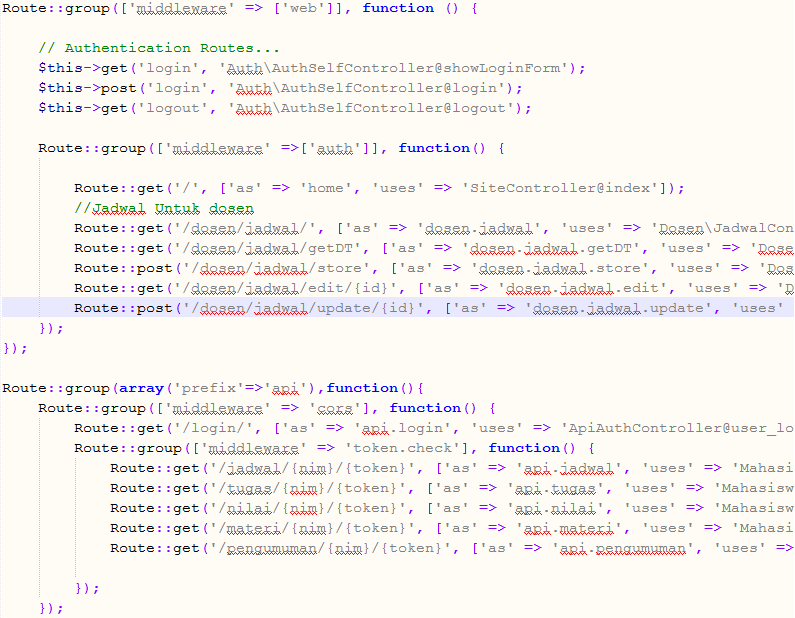


Gambar 23. listing program API login client

Penjelasan dari kode program pada gambar diatas akan memeriksa login berdasarkan email atau nama yang di inputkan oleh pengguna dan mencocokkannya dengan email atau nama pengguna yang ada didalam database. Jika email dan password benar maka program akan membuat token dan memberikannya pada aplikasi client dalam bentuk data json. Token digunakan layaknya seperti nomor indentitas client agar server dapat mengenali client, selanjutnya aplikasi client bisa meminta data kepada server dengan mengirimkan token saja tanpa perlu mengirim email atau password.

1. listing program untk mengatur *url* server

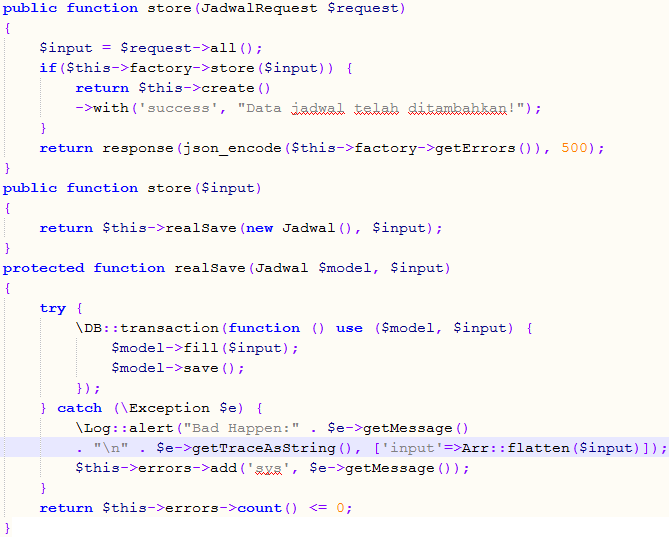
agar tidak sembarang orang bisa mengakses halaman server maka url server perlu diatur. url diatur lewat listing program seperti gambar berikut

Gambar 24. listing program untuk mengatur URL

Penjelasan dari kode program pada gambar diatas adalah grup web ialah grup yang berisi *url* – *url* yang bisa diakses oleh semua orang. Kemudian grup auth adalah grup yang berisi url yang hanya perlu login terlebih dahulu untuk mengakses url tersebut. Lalu grup api adalah grup yang berisi *url* – *url* untuk aplikasi client, perlu token untuk dapat mengakses *url* tersebut.

1. listing program tambah data

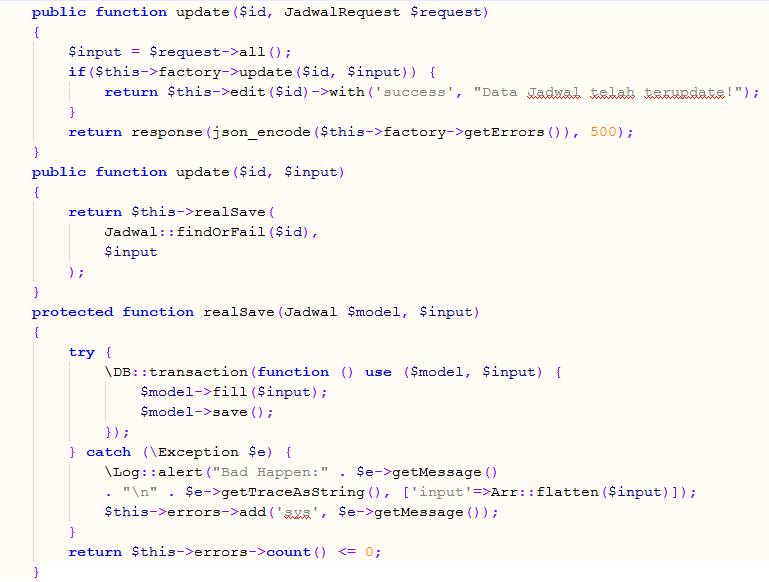
untuk menambah data diperlukan kode program yang bertugas manambahkan data ke dalam database. Listing program tambah data dapat dilihat pada gambar dibawah

Gambar 25. listing program tambah data

Kode program pada gambar diatas berfungsi untuk menambahkan data, data yang di inputkan oleh user akan di simpan jika sudah sesuai dengan diminta oleh *form* pengisian tambah data.

1. Listing program edit data

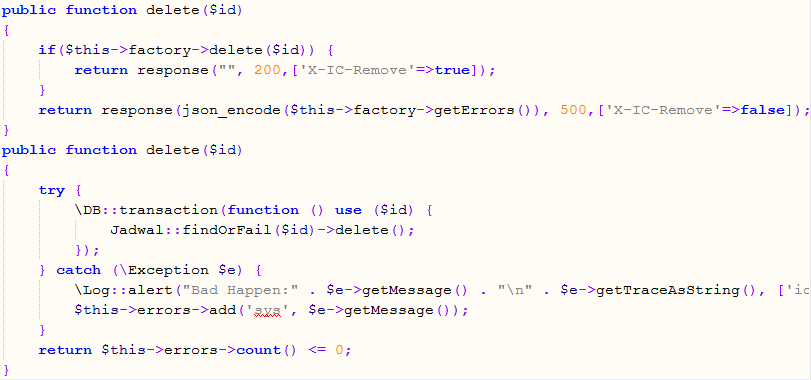
untuk keperluan merubah data maka diperlukan kode program yang dapat melakukan tugas merubah data dalam database. Berikut listing program untuk merubah atau edit data dapat dilihat pada gambar dibawah.

Gambar 26. listing program edit data

Kode program pada gambar diatas berfungsi untuk merubah data, cara kerja program tersebut pertama program akan mencari data sesuai id data yang ingin dirubah lalu program akan memunculkan form edit data kemudian user akan merubah data yang ada dalam form kemudian program akan mengganti data sebelumnya dengan data yang telah di isi dalam form edit data.

1. Listing program hapus data

Kode program untuk mengapus data dapat dilihat pada gambar dibawah berikut

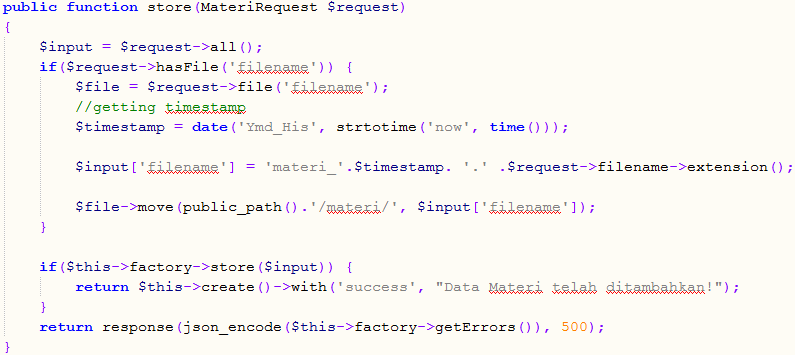


Gambar 27. listing program hapus data

cara kerja program pada gambar diatas adalah program akan mencari data sesuai id data, jika ada maka data itu akan dihapus dari dalam database.

1. Listing program upload materi

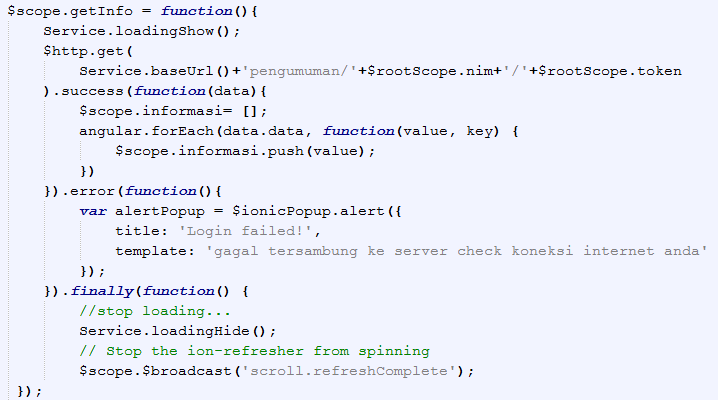
Agar materi dapat di unduh oleh mahasiswa maka dosen harus mengunggah atau upload file terlebih dahulu ke dalam server. Kode program untuk upload file dapat dilihat pada gambar dibawah ini.

Gambar 28. listing program upload materi

Cara kerja program tersebut adalah pertama program akan memeriksa apakah ada request file, jika ada maka program akan mengganti nama file tersebut dengan “materi” ditambah tanggal dan jam serta ekstensi file (rar, zip, pdf). Kemudian file akan disimpan ke dalam folder materi di server.

1. Listing program menampilkan data pada aplikasi client

Aplikasi client memerlukan data dari server untuk dapat menampilkan data pada mahasiswa. Listing program untuk melakukan request data kepada server dapat dilihat pada gambar

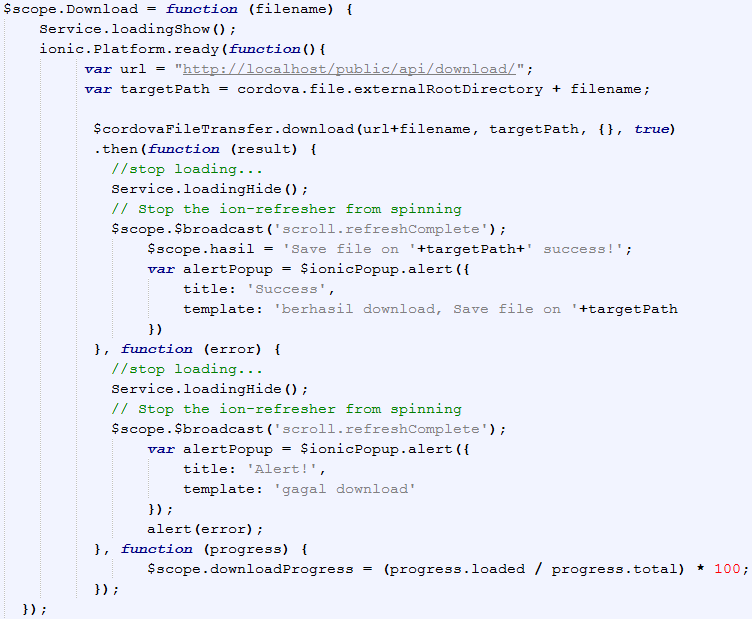


Gambar 29. listing program menampilkan data pada aplikasi client

Cara kerja kode program diatas ialah pertama aplikasi client akan mengirimkan nim dan token pada server, server akan memeriksa token jika token benar maka server akan mengirim data sesuai nim mahasiswa dalam bentuk json. Data yang diterima oleh aplikasi client akan dimasukkan dalam array setelah itu baru ditampilkan oleh aplikasi.

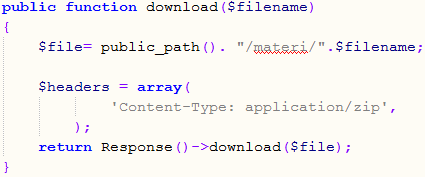
1. Listing program untuk download materi

Kode program yang berfungsi untuk mengunduh atau download materi dapat dilihat pada gambar dibawah berikut



Gambar 30. listing program download materi pada aplikasi client

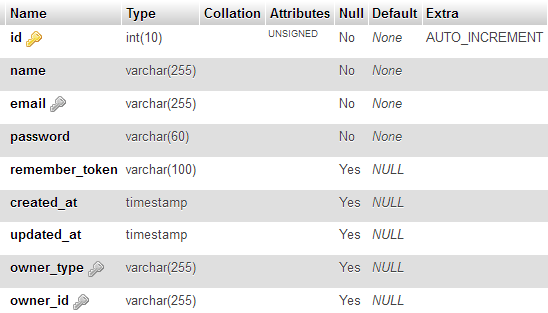
Cara kode program diatas ialah pertama program akan mengirimkan nama file yang ingin di download ke server, server akan mencari file tersebut jika file ditemukan maka akan di kirim oleh server kepada client. Selanjutnya aplikasi client akan melakukan proses download sampai selesai, jika download selesai maka akan muncul kotak dialog berisis pesan “berhasil download” dan jika gagal maka akan muncul kotak dialog berisis pesan “gagal download”. Listing program download materi pada bagian server dapat dilihat pada gambar berikut



Gambar 31. listing program download materi apada bagian server

1. Pembahasan database
2. Tabel users

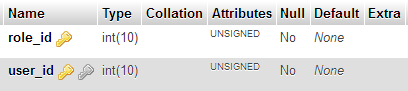
Tabel users digunakan untuk menyimpan data user account pengguna Aplikasi Pengelolaan Mata Kuliah Teknik Informatika Berbasis *Client Server* Pada STMIK Palangkaraya. Tabel users dapat dilihat pada gambar 66



Gambar 66. Tabel users

1. Tabel role\_user

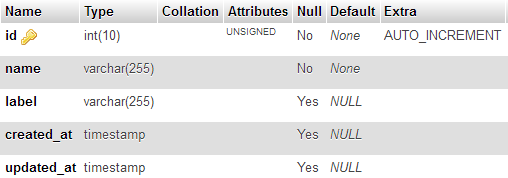
Tabel role\_user adalah tabel yang menyimpan data berisi informasi hak akses user berdasarkan kategori hak akses dalam tabel roles. Tabel role\_user dapat di lihat pada gambar 67



Gambar 67. Tabel role\_user

1. Tabel roles

Tabel roles adalah tabel yang menyimpan data kategori hak akses user



Gambar 68. Tabel roles

1. Tabel dosen

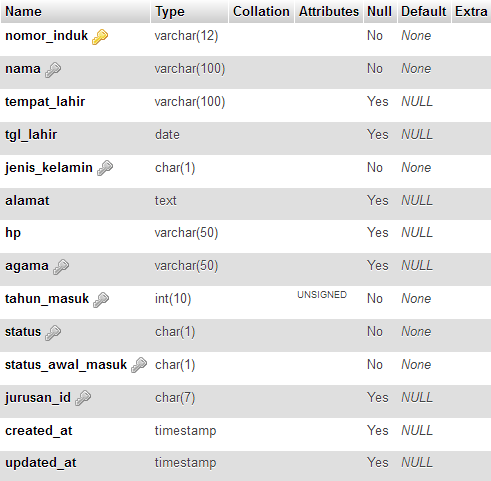
Tabel dosen digunakan untuk menyimpan data diri dosen - dosen Stmik palangkaraya. Tabel dosen dapat dilihat pada gambar 69.



Gambar 69. Tabel dosen

1. Tabel mahasiswa

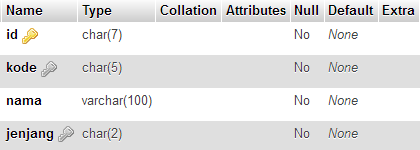
Tabel mahasiswa digunakan untuk menyimpan data diri semua mahasiswa Stmik palangkaraya. Tabel mahasiswa dapat dilihat pada gambar 70.



Gambar 70. Tabel mahasiswa

1. Tabel jurusan

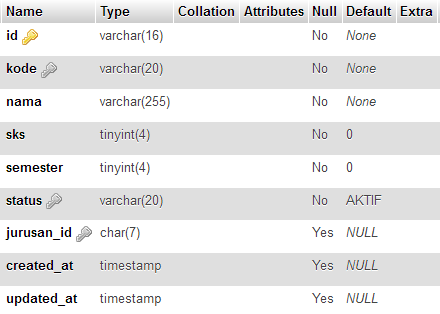
Tabel jurusan digunakan untuk menyimpan data jurusan. Tabel jurusan dapat dilihat pada gambar 71



Gambar 71. Tabel jurusan

1. Tabel mata\_kuliah

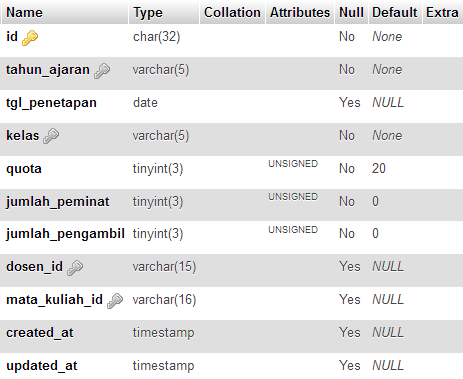
Tabel mata\_kuliah digunakan untuk menyimpan data matakuliah. Tabel mata\_kuliah dapat dilihat pada gambar 72



Gambar 72. Tabel mata\_kuliah

1. Tabel pengampu\_kelas

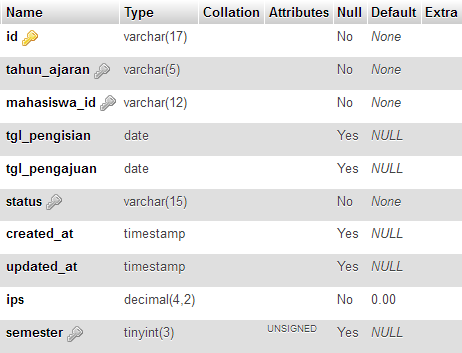
Tabel pengampu\_kelas digunakan untuk menyimpan data yang menghubungkan data dosen dalam tabel dosen dengan data matakuliah dalam tabel mata\_kuliah. Tabel pengampu\_kelas dapat dilihat pada gambar 73



Gambar 73. Tabel pengampu\_kelas

1. Tabel rencana\_studi

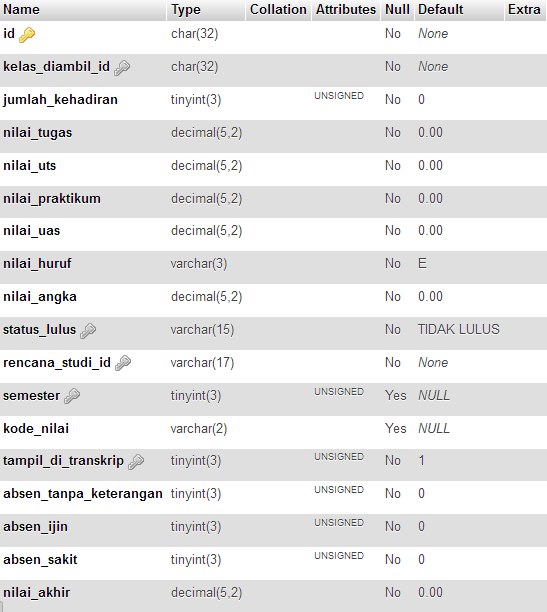
Tabel rencana\_studi digunakan untuk menyimpan data rencana stud mahasiswa stmik palangkaraya. Tabel rencana\_studi dapat dilihat pada gambar



Gambar 74. Tabel rencana\_studi

1. Tabel rincian\_studi

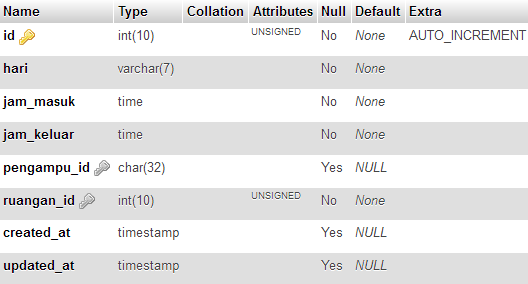
Tabel rincian\_studi digunakan untuk menyimpan rincian matakuliah seperti nilai dan jumlah kehadiran dari setiap matakuliah yang terdaftar dalam rencana studi mahasiswa. Tabel rincian\_studi dapat dilihat pada gambar 75.



Gambar 75. Tabel rincian\_studi

1. Tabel jadwal

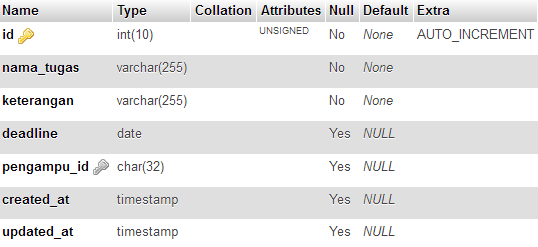
Tabel jadwal digunakan untuk menyimpan data jadwal matakuliah stmik palangkaraya. Tabel jadwal dapat dilihat pada gambar 76.



Gambar 76. Tabel jadwal

1. Tabel tugas

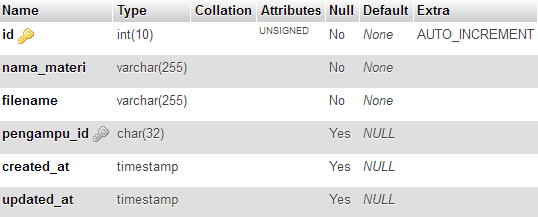
Tabel tugas digunakan untuk menyimpan data tugas yang dibuat oleh dosen. Tabel tugas dapat dilihat pada gambar 77.



Gambar 77. Tabel tugas

1. Tabel materi

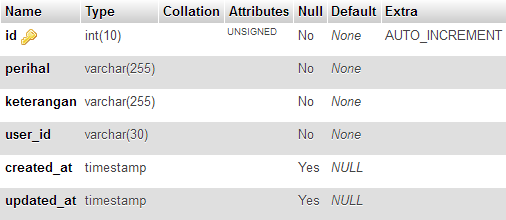
Tabel materi digunakan untuk menyimpan nama materi dan nama file materi yang telah di upload oleh dosen. Tabel materi dapat dlihat pada gambar 78.



Gambar 78. Tabel materi

1. Tabel pengumuman

Tabel pengumuman digunakan untuk menyimpan data pengumuman. Tabel pengumuman dapat dilihat pada gambar 79.



Gambar 79. Tabel pengumuman

1. Pembahasan Hasil Responden Pengguna

Penilain terhadap Aplikasi Pengelolaan Mata Kuliah Teknik Informatika Berbasis *Client Server* Pada STMIK Palangkaraya, penulis melakukan penyebaran kuisioner yang diberikan kepada 10 orang. Selanjutnya mereka menggunakan Aplikasi Pengelolaan Mata Kuliah Teknik Informatika Berbasis *Client Server* Pada STMIK Palangkaraya kemudian dilakukan pengumpulan data dengan mengisi kuesioner yang telah dibagikan sebelumnya oleh penulis.

1. Pernyataan kuesioner

Pernyataan kuesioner merupakan tahap untuk mengetahui pernyataan apa saja yang diajukan kepada responden terhadap Aplikasi Pengelolaan Mata Kuliah Teknik Informatika Berbasis *Client Server* Pada STMIK Palangkaraya. Berikut pernyataan kuesioner seperti pada Tabel 19.

Tabel 19. Pernyataan Kuesioner

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Pernyataan** | **Alternatif Jawaban** | | | | |
| **5** | **4** | **3** | **2** | **1** |
| **SB** | **B** | **CB** | **KB** | **TB** |
| 1. | Tampilan aplikasi sudah bersifat baik / *user friendly.* |  |  |  |  |  |
| 2. | Fungsional aplikasi sudah berjalan sesuai dengan fungsinya. |  |  |  |  |  |
| 3. | Aplikasi mampu melakukan *download file* materi. |  |  |  |  |  |
| 4. | informasi yang ditampilkan oleh aplikasi sesuai dengan data yang dibutuhkan mahasiswa. |  |  |  |  |  |
| 5. | Aplikasi terlihat menarik, baik dari segi tampilan dan teks. |  |  |  |  |  |
| 6. | Aplikasi ini dapat membantu ketika mencari informasi tentang matakuliah. |  |  |  |  |  |
| 7. | Tentang keseluruhan aplikasi. |  |  |  |  |  |

Keterangan bobot yang digunakan dalam penilaian alternatif jawaban dapat dilihat pada Tabel 20.

Tabel 20. Keterangan Bobot

|  |  |
| --- | --- |
| **Keterangan** | **Nilai** |
| Sangat Baik | 5 |
| Baik | 4 |
| Cukup Baik | 3 |
| Kurang Baik | 2 |
| Tidak Baik | 1 |

1. Hasil Kuesioner

Hasil kuesioner merupakan hasil perhitungan dari penilaian responden terhadap Aplikasi Pengelolaan Mata Kuliah Teknik Informatika Berbasis *Client Server* Pada STMIK Palangkaraya. Hasil kuesioner dapat dilihat pada Tabel 21.

Tabel 21. Hasil kuesioner

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Responden** | **Jawaban Pertanyaan** | | | | | | | **Jumlah** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| 1 |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |  |  |
| 9 |  |  |  |  |  |  |  |  |
| 10 |  |  |  |  |  |  |  |  |
| **Jumlah Skor Hasil Pengumpulan Data** | | | | | | | |  |

Sedangkan untuk bobot daftar pertanyaan dapat dilihat pada Tabel 22.

Tabel 22. Bobot daftar pertanyaan

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Keterangan** | **Daftar Pertanyaan** | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| Sangat Baik |  |  |  |  |  |  |  |
| Baik |  |  |  |  |  |  |  |
| Cukup Baik |  |  |  |  |  |  |  |
| Kurang Baik |  |  |  |  |  |  |  |
| Tidak Baik |  |  |  |  |  |  |  |

Berdasarkan hasil kuesioner pada tabel dapat dilihat menggunakan skala pengukuran, disini penulis menggunakan likert sebagai skala pengukuran yaitu sebagai berikut:

Jumlah skor kriteria = (skor tertinggi tiap item) x (jumlah pertanyaan) x (jumlah responden)

Jumlah skor kriteria = 5 x 7 x 10

= 350

Berdasarkan tabel untuk total jawaban responden adalah .... dan jumlah skor kriterium adalah 350. Dengan demikian Aplikasi Pengelolaan Mata Kuliah Teknik Informatika Berbasis *Client Server* Pada STMIK Palangkaraya menurut pendapat 10 orang responden yaitu:

Total hasil responden = .... / 350 x 100

= .... x 100

= .... %

Jadi, sikap responden terhadap aplikasi yang dibangun sebesar 91 % dan termasuk dalam interval sangat baik.

0 20% 40% 60% 80% 100%

TB KB CB B SB

Gambar 44. Persentase *range* kelompok responden

Keterangan :

TB = Tidak Baik

KB = Kurang Baik

CB = Cukup Baik

B = Baik

SB = Sangat Baik

Berdasarkan hasil perhitungan kuesioner dengan skala *likert* yang didapat dari 10 respon dan 7 pertanyaan maka didapat hasil interpretasi sebesar ....% atau dengan nilai .... dari nilai maksimal 350, maka dapat disimpulkan bahwa Aplikasi Pengelolaan Mata Kuliah Teknik Informatika Berbasis *Client Server* Pada STMIK Palangkaraya sangat baik dan layak untuk digunakan.