



IMPLEMENTASI APLIKASI SISTEM PENYUSUNAN JADWAL KULIAH PADA JURUSAN TEKNIK ELEKTRO UIN SULTAN SYARIF KASIM

Oktaf Brillian Kharisma^{a,1,*}, Reski Riandi^{b,2}

^a Fakultas Sains dan Teknologi, UIN Sultan Syarif Kasim Pekanbaru

^b Fakultas Sains dan Teknologi, UIN Sultan Syarif Kasim Pekanbaru

¹ brilliankhar@gmail.com *; ²reskiriandi@gmail.com

* corresponding author

ARTICLE INFO

ABSTRACT

Keywords

Penjadwalan kuliah

Aplikasi

PHP

MySQL

Proses penyusunan jadwal perkuliahan merupakan kegiatan rutin setiap universitas dalam menghadapi setiap semester baru. Jurusan Teknik Elektro UIN Suska Riau, penyusunan jadwal perkuliahan masih menggunakan program Microsoft Excel. Sehingga, pembuatan jadwal tersebut masih kurang efisien karena prosesnya membutuhkan waktu yang lama dan harus melakukan revisi jadwal yang bentrok. Maka, dibutuhkan sebuah sistem yang dapat menangani proses penjadwalan perkuliahan tersebut dengan menerapkan metode asosiasi berbasis web. Dengan adanya aplikasi ini nantinya proses perkuliahan di jurusan teknik elektro dapat dilakukan tanpa adanya jadwal perkuliahan yang bentrok dan proses penyusunannya dapat dilakukan dengan cepat. Aplikasi yang dibangun akan mengenerate otomatis dan menampilkan jadwal berupa tabel penjadwalan secara keseluruhan yaitu berupa data dosen, data waktu, data mata kuliah, dan data ruangan. Dari hasil uji beta yang dilakukan oleh 100 dosen sebagai subjek penguji menyatakan bahwa keberhasilan sistem sebesar 75%.

1. Pendahuluan

Penyusunan jadwal kuliah merupakan kegiatan rutin dalam sistem akademik di Universitas atau Perguruan Tinggi yang dilakukan dalam menghadapi setiap semester baru. Penyusunan jadwal juga sesuatu hal yang penting dalam mengatur proses perkuliahan agar tidak terjadinya bentrok antara satu mata kuliah dengan mata kuliah yang lain.

Permasalahan yang dihadapi dalam penyusunan jadwal kuliah khususnya di jurusan Teknik Elektro UIN Suska Riau masih menggunakan cara manual dengan mencocokkan satu persatu mata kuliah dengan lokal, jam, ruangan, dan dosen yang ada, meski menggunakan logika/formula yang ada pada *Microsoft Excel* tetapi ini masih kurang efektif dan masih ada mata kuliah yang bentrok antara satu dengan yang lainnya. Hal ini mengakibatkan perkuliahan di awal semester kurang efektif karena harus melakukan revisi jadwal berulang kali sampai sesuai dengan kebutuhan Akademik.

Penjadwalan juga perlu adanya optimasi. Oleh karena itu, pembuatan sistem penjadwalan kuliah ini dilakukan dengan berbagai perbaikan untuk menghindari terjadinya bentrok pada penjadwal. Pembuatan aplikasi ini menggunakan bahasa pemrograman *Hypertext PreProcessor* (PHP) untuk membangun sistem, dan MySQL sebagai database untuk menyimpan data yang diperlukan dalam pembuatan aplikasi ini.

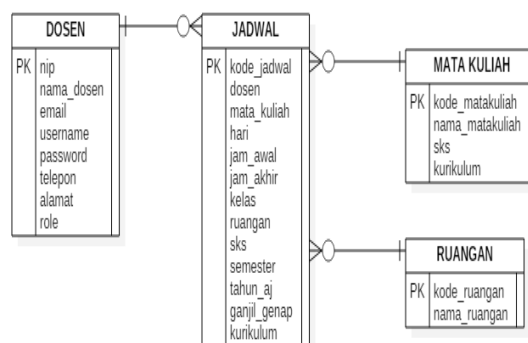
Aplikasi penyusunan jadwal Kuliah ini akan sangat membantu bagi Universitas dalam pembuatan jadwal kuliah. Juga akan sangat efektif yaitu tidak akan terdapat mata kuliah yang bentrok karena aplikasi ini sudah dibuat untuk mengatasi hal tersebut.

2. Metode Penelitian

Metode penelitian diawali dengan observasi terhadap sistem penyusunan jadwal perkuliahan yang ada di jurusan teknik elektro UIN Sultan syarif kasim. Hal ini dilakukan untuk untuk mengumpulkan informasi untuk kebutuhan pembuatan sistem. Sistem di buat berdasarkan informasi yang diperoleh, antara lain data ruang, data mata kuliah, data dosen, data kelas, hari dan jam sebagai *input* untuk pembuatan jadwal. Sedangkan *ouput* yang diperoleh adalah berupa tabel penjadwalan kuliah. Kemudian, tahapan selanjutnya adalah desain sistem. Sistem ini dirancang menggunakan metode *scramble* yang berarti bahwa setiap dosen akan dapat menentukan jadwalnya masing-masing secara acak sebelum nantinya di generate untuk menentukan waktu dan jadwal perkuliahan.

2.1 Perancangan Database

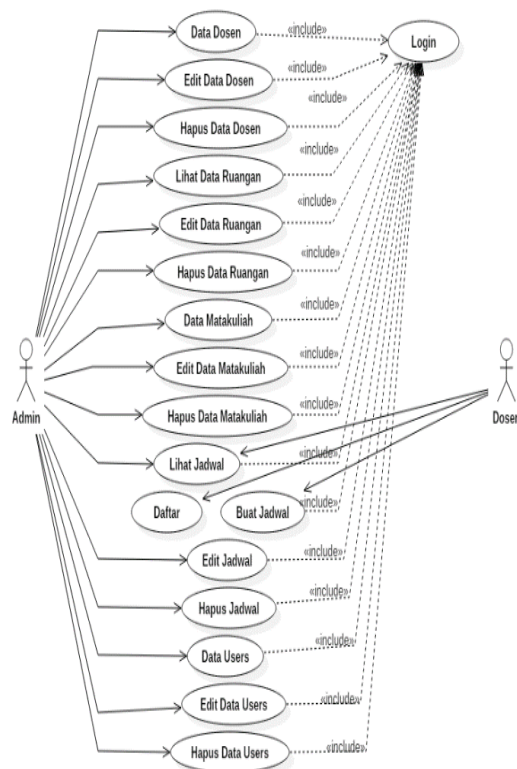
Hal mendasar yang harus dipenuhi dalam pembuatan sistem ini adalah Perancangan database yang menggunakan ERD (*Entity Relationship Diagram*).



Gambar 1. ERD (*Entity Relationship Diagram*)

2.2 Perancangan Aplikasi

Proses perancangan digambarkan menggunakan *usecase diagram*. Sebagaimana berikut:

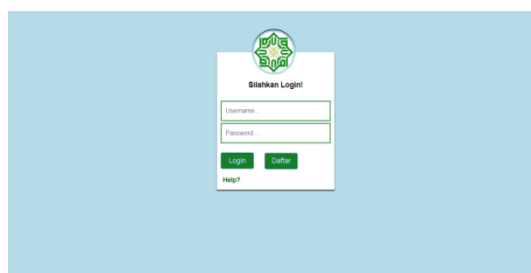


Gambar 2. Usecase diagram

3. Hasil Dan Pembahasan

3.1 Implementasi Sistem

Hasil implementasi setiap halaman aplikasi adalah sebagaimana berikut ini :



Gambar 3. Form Login Admin dan Member



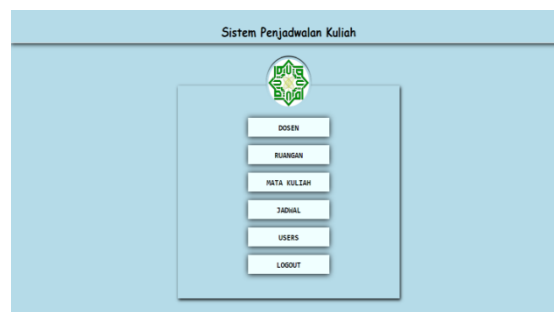
Gambar 4. *Form Daftar Member*



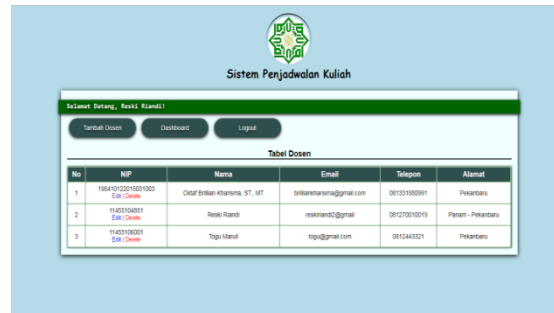
Gambar 5. *Tampilan Halaman Utama*



Gambar 6 *Form Buat Jadwal*



Gambar 7. *Tampilan Halaman Dashboard*



Gambar 8. Tampilan Data Dosen



Gambar 9. Tampilan Data Mata Kuliah



Gambar 10. Tampilan Data Ruang



Gambar 11. Tampilan Report Jadwal

3.2 Pengujian Aplikasi

a. Pengujian Black Box



Black box testing merupakan teknik pengujian yang berfokus pada keluaran hasil dari respon masukan, atau secara sederhana *black box* merupakan proses menjalankan aplikasi untuk mengetahui apakah ada *error* atau ada fungsi yang tidak berjalan sesuai harapan. Berikut hasil pengujian menggunakan metode *black box*:

Tabel 1. Pengujian *Login Admin dan Dosen*

Kasus			
<i>Login Admin</i>			
Skenario Uji	Hasil yang Diharapkan	Pengamatan	Hasil Uji
Masukkan <i>Username</i> dan <i>Password</i> Admin yang benar.	Ketika data <i>login</i> dimasukkan dan tombol <i>Login</i> diklik maka akan dilakukan proses pengecekan data <i>login</i> . Jika data <i>login</i> benar maka akan masuk ke halaman <i>dashboard</i> .	Masuk ke halaman <i>dashboard</i> .	Diterima [X] Ditolak []
Kasus			
<i>LoginMember/Dosen</i>			
Masukkan <i>Username</i> dan <i>Password</i> Dosen yang benar.	Ketika data <i>login</i> dimasukkan dan tombol <i>Login</i> diklik maka akan dilakukan proses pengecekan data <i>login</i> . Jika data <i>login</i> benar maka akan masuk ke halaman utama.	Masuk ke halaman utama.	Diterima [X] Ditolak []
Kasus			
<i>Kesalahan data login</i>			
Masukkan <i>Username</i> dan <i>Password</i> Admin dan Dosen yang salah.	Tidak bisa <i>login</i> dan menampilkan pesan.	Tampil pesan " <i>Username</i> atau <i>Password</i> belum terdaftar!"	Diterima [X] Ditolak []

Tabel 2. Pengujian *Form Daftar*

Kasus			
<i>Tombol Simpan</i>			
Skenario Uji	Hasil yang Diharapkan	Pengamatan	Hasil Uji
Masukkan data <i>member/dosen</i> pengampu yang ingin membuat jadwal.	Ketika data dosen dimasukkan dan tombol <i>Daftar</i> diklik, maka akan langsung masuk ke halaman <i>login</i> .	Masuk ke halaman <i>login</i> dan keluar pesan "Selamat anda sudah terdaftar, silahkan login".	Diterima [X] Ditolak []
Kasus			



Validasi data sama

Masukkan data <i>member</i> /dosen pengampu yang ingin membuat jadwal tetapi <i>username</i> dan email sama dengan tersimpan di <i>database</i> .	Tidak bisa mendaftar dan menampilkan pesan.	Tampil pesan " <i>Username</i> atau email sudah terdaftar!"	Diterima [X]
			Ditolak []

Tabel 3. Pengujian *Form* Data Dosen

Kasus			
Tombol Tambah			
Skenario Uji	Hasil yang Diharapkan	Pengamatan	Hasil Uji
Pada halaman dosen klik, menu Tambah Dosen dan masukkan data dosen pengampu mata kuliah.	Ketika data dosen dimasukkan dan tombol tambah diklik, maka data dosen akan tersimpan dan langsung ke halaman dosen dan data dosen pada tabel bertambah, dan menampilkan pesan.	Langsung ke halaman dosen dan data baru dimasukkan muncul pada tabel dosen, dan keluar pesan "Data berhasil disimpan".	Diterima [X]
			Ditolak []
Kasus			
Tombol Edit			
Pilih data dosen yang akan diedit dengan cara mengklik tombol Edit pada baris dosen yang akan diedit.	Ketika edit diklik, maka akan masuk kehalaman edit dosen. Dan ketika data dosen diedit dan tombol <i>Update</i> diklik, maka akan langsung ke halaman dosen dan data dosen yang diedit berubah, dan menampilkan pesan.	Langsung ke halaman dosen dan data dosen yang diedit berubah, dan keluar pesan "Data berhasil diupdate".	Diterima [X]
			Ditolak []
Kasus			
Tombol Hapusa dan Pesan Konfirmasi Hapus			
Pilih data dosen yang akan dihapus dengan cara mengklik tombol Hapus pada baris dosen yang akan dihapus.	Ketika tombol edit diklik, maka akan menampilkan pesan konfirmasi. Ketika memilih <i>Yes</i> maka data dosen akan hilang dari <i>database</i> ditandai dengan jumlah baris tabel pada halaman dosen berkurang.	Data dosen yang dipilih hilang dari <i>database</i> berarti data berhasil dihapus. Dan keluar pesan "Data berhasil dihapus".	Diterima [X]
			Ditolak []

Tabel 4. Pengujian *Form* Data Mata Kuliah

Kasus			
Tombol Tambah			
Skenario Uji	Hasil yang Diharapkan	Pengamatan	Hasil Uji
Pada halaman mata kuliah, klik menu Tambah Mata Kuliah dan masukkan data dosen mata kuliah.	Ketika data mata kuliah dimasukkan dan tombol tambah diklik, maka data mata kuliah akan tersimpan dan langsung ke halaman Mata Kuliah dan data mata kuliah pada tabel bertambah, dan menampilkan pesan.	Langsung ke halaman Mata Kuliah dan data baru dimasukkan muncul pada tabel mata kuliah, dan keluar pesan "Data berhasil disimpan".	Diterima [X]
			Ditolak []
Kasus			
Tombol Edit			
Pilih data mata kuliah yang akan diedit dengan cara mengklik tombol Edit pada baris mata kuliah yang akan diedit.	Ketika edit diklik, maka akan masuk kehalaman edit mata kuliah. Dan ketika data mata kuliah diedit dan tombol <i>Update</i> diklik, maka akan langsung ke halaman mata kuliah dan data mata kuliah yang diedit berubah, dan menampilkan pesan.	Langsung ke halaman mata kuliah dan data mata kuliah yang diedit berubah, dan keluar pesan "Data berhasil diupdate".	Diterima [X]
			Ditolak []
Kasus			



Tombol Hapus dan Pesan Konfirmasi Hapus

Pilih data mata kuliah yang akan dihapus dengan cara mengklik tombol Hapus pada baris mata kuliah yang akan dihapus.	Ketika tombol edit diklik, maka akan menampilkan pesan konfirmasi. Ketika memilih <i>Yes</i> maka data mata kuliah akan hilang dari <i>database</i> .	Data mata kuliah yang dipilih hilang dari <i>database</i> berarti data berhasil dihapus. Dan keluar pesan "Data berhasil dihapus".	Diterima [X] Ditolak []
--	---	--	---------------------------------------

Tabel 5. Pengujian *Form* Data Ruangan

Kasus			
Tombol Tambah			
Skenario Uji	Hasil yang Diharapkan	Pengamatan	Hasil Uji
Pada halaman Ruangan, klik menu Tambah Ruangan dan masukkan data dosen mata kuliah.	Ketika data ruangan dimasukkan dan tombol tambah diklik, maka data mata kuliah akan tersimpan dan langsung ke halaman Ruangan dan data ruangan pada tabel bertambah, dan menampilkan pesan.	Langsung ke halaman Ruangan dan data baru dimasukkan muncul pada tabel ruangan, dan keluar pesan "Data berhasil disimpan".	Diterima [X] Ditolak []
Kasus			
Tombol Edit			
Pilih data ruangan yang akan diedit dengan cara mengklik tombol Edit pada baris ruangan yang akan diedit.	Ketika edit diklik, maka akan masuk kehalaman edit ruangan. Dan ketika data ruangan diedit dan tombol <i>Update</i> diklik, maka akan langsung ke halaman Ruangan dan data ruangan yang diedit berubah, dan menampilkan pesan.	Langsung ke halaman Ruangan dan data ruangan yang diedit berubah, dan keluar pesan "Data berhasil diupdate".	Diterima [X] Ditolak []
Kasus			
Tombol Hapus dan Pesan Konfirmasi Hapus			
Pilih data ruangan yang akan dihapus dengan cara mengklik tombol Hapus pada baris ruangan yang akan dihapus.	Ketika tombol edit diklik, maka akan menampilkan pesan konfirmasi. Ketika memilih <i>Yes</i> maka data ruangan akan hilang dari <i>database</i> ditandai dengan jumlah baris tabel pada halaman ruangan berkurang.	Data ruangan yang dipilih hilang dari <i>database</i> berarti data berhasil dihapus. Dan keluar pesan "Data berhasil dihapus".	Diterima [X] Ditolak []

Tabel 6. Pengujian Proses Penjadwalan

Kasus			
Tombol Buat Jadwal			
Skenario Uji	Hasil yang Diharapkan	Pengamatan	Hasil Uji
Klik menu Buat Jadwal, masukkan jadwal dengan cara memasukkan data penjadwalan pada <i>form</i> buat jadwal.	Ketika data jadwal dimasukkan dan tombol Buat diklik, maka jadwal akan tersimpan di <i>database</i> ditandai dengan munculnya data pada tabel jadwal, dan keluar pesan.	Data yang baru saja dimasukkan muncul pada tabel jadwal, dan keluar pesan "Jadwal berhasil dibuat".	Diterima [X] Ditolak []
Kasus			
Tombol Hapus dan Pesan Konfirmasi Hapus			
Pilih data jadwal yang akan dihapus dengan cara mengklik tombol Hapus pada baris jadwal yang akan dihapus.	Ketika tombol Hapus diklik maka akan muncul pesan konfirmasi. Ketika memilih <i>Yes</i> maka data jadwal akan hilang dari <i>database</i> .	Data jadwal yang dipilih hilang dari <i>database</i> berarti data berhasil dihapus. Dan keluar pesan "Data berhasil dihapus".	Diterima [X] Ditolak []
Kasus			
Validasi Data Ganda/Bentrok			

Masukkan data jadwal yang sama dengan jadwal yang sudah ada tersimpan pada <i>database</i> .	Ketika tombol Buat diklik maka jadwal tidak tersimpan dan muncul pesan jadwal bentrok.	Sesuai dengan diharapkan, data jadwal tidak tersimpan dan keluar pesan.	Diterima [X] Ditolak []
--	--	---	---------------------------------

Tabel 7. Pengujian *Report* Penjadwalan

Kasus			
Cetak Jadwal			
Skenario Uji	Hasil yang Diharapkan	Pengamatan	Hasil Uji
Pada halaman utama, klik tombol Cetak Jadwal.	Ketika tombol Cetak Jadwal diklik, maka akan tampil <i>report</i> penjadwalan keseluruhan dan diurutkan berdasarkan hari dan jam terkecil dalam bentuk <i>fpdf</i> .	Jadwal yang ditampilkan sesuai dengan yang diharapkan.	Diterima [X] Ditolak []

b. Analisis Beta

Pengujian *beta* merupakan pengujian untuk mendapatkan kesimpulan dari aplikasi yang telah dibuat, dengan cara memperoleh data dari kuesioner terhadap *users* atau pengguna secara langsung. Pengujian ini menggunakan kuisisioner yang terdiri dari 9 pertanyaan yang ditanya kepada 20 responden, dengan menggunakan skala *likert* dengan skala 1 sampai 4.

Tabel 8. Skala *Likert*

No.	Keterangan
1	Tidak Baik
2	Kurang Baik
3	Baik
4	Sangat Baik

Berdasarkan kuisisioner maka dicari persentase jawaban dengan rumus kuisisioner sebagai berikut:

$$c. Y = \frac{P}{Q} \times 100\%$$

Keterangan:

Y = Nilai persentase.

P = Banyaknya jawaban reponden dari tiap jawaban.

Q = Jumlah responden.

Setiap hasil persentase pertanyaan dari kuisisioner dapat di lihat pada Tabel 9.

Tabel 9. Deskripsi Tanggapan Responden

No	Item Pertanyaan	Tanggapan Responden								N	Total Skor	%	Kategori
		SB (4)		B (3)		KB (2)		TB (1)					
		f	%	f	%	f	%	f	%				
1	Item 1	8	40	10	50	2	10	0	0	20	66	82,50	Sangat Baik
2	Item 2	8	40	10	50	2	10	0	0	20	66	82,50	Sangat Baik
3	Item 3	8	40	9	45	3	15	0	0	20	65	81,25	Sangat Baik
4	Item 4	11	55	8	40	1	5	0	0	20	70	87,50	Sangat Baik
5	Item 5	12	60	6	30	2	10	0	0	20	70	87,50	Sangat Baik
6	Item 6	11	55	8	40	1	5	0	0	20	70	87,50	Sangat Baik
7	Item 7	9	45	10	50	1	5	0	0	20	68	85,00	Sangat Baik
8	Item 8	10	50	8	40	2	10	0	0	20	68	85,00	Sangat Baik
9	Item 9	8	40	9	45	3	15	0	0	20	65	81,25	Sangat Baik

Jumlah skor	608		
Rata-rata	68	85,00	Sangat Baik

Untuk menentukan kriteria atau kategori dari setiap pertanyaan, menggunakan rumus interval (I) sebagai berikut:

$$I = \frac{100}{\text{Jumlah Skor (Likert)}}$$

Berdasarkan rumus interval diatas, dimana jumlah skor (*likert*) = 4 maka didapat $I = \frac{100}{4} = 25$. Jadi, jarak atau interval dari 0% sampai 100% adalah 25%. Berikut kriteria interpretasi skornya berdasarkan interval:

Tabel 10. Persentase Kategori

No.	Interval	Kategori
1	0 % - 24,99 %	Tidak Baik
2	25 % - 49,99 %	Kurang Baik
3	50 % - 74,99 %	Baik
4	75 % - 100 %	Sangat Baik

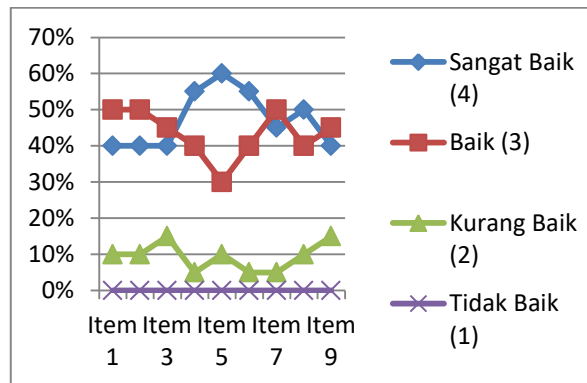
Agar mendapatkan hasil interpretasi, terlebih dahulu harus diketahui skor tertinggi (Y) dan skor terendah (X) untuk item penilaian dengan rumus sebagai berikut:

$$Y = \text{skor tertinggi likert} \times \text{jumlah reponden} \quad X = \text{terendah likert} \times \text{jumlah reponden}$$

Jumlah skor tertinggi untuk item “Sangat Baik” adalah $4 \times 20 = 80$, sedangkan item “Tidak Baik” adalah $1 \times 20 = 20$. Maka penilaian interpretasi responden dengan menggunakan rumus sebagai berikut, untuk menentukan kategori dari setiap pertanyaan:

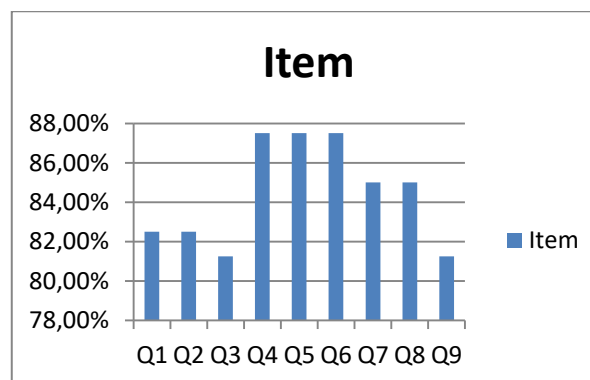
$$\text{Index \%} = \frac{\text{Total Skor}}{Y} \times 100$$

Berikut grafik persentase perbandingan item pertanyaan terhadap tanggapan responden:



Gambar 12. Persentase Perbandingan Item Pertanyaan Terhadap Tanggapan Responden

Berikut persentase kategori dari setiap item pertanyaan yang diberikan kepada responden:



Grafik 13. Kategori per Item Pertanyaan

4. Kesimpulan

Dari hasil pembahasan dapat diambil beberapa kesimpulan, antara lain:

1. Telah dirancang dan dibuat aplikasi sistem penyusunan jadwal kuliah berbasis *web* dengan bahasa pemrograman PHP dan MySQL.
2. Proses penyusunan jadwal di-*input* oleh dosen yang mengampumata kuliah yang bersangkutan dengan cara mengisi data pada *form* penjadwalan.
3. Dengan pengimplementasian hasil dari rancangan kedalam sistem berbasis *web* didapatkan sistem yang dapat mempermudah proses penjadwalan dan pengolahan data seperti memasukkan, penambahan dan pengecekan data. Serta dapat menghasilkan jadwal matakuliah yang anti bentrok.
4. Dengan adanya Aplikasi Penjadwalan Kuliah *WEB* ini dapat mempermudah penyusunan jadwal perkuliahan.

Daftar Pustaka

- [1] Hariyanto, B. 2004. *Sistem Manajemen Basis Data*. Bandung: Informatika
- [2] Kustiyahningsih, Yeni & Rosa Anamisa, Devie. 2011. *Pemrograman Basis Data Berbasis Web Menggunakan PHP & MySQL*. Yogyakarta: Graha Ilmu.

- [3] Raharjo, Budi dkk. 2012. *Modul Pemrograman WEB (HTML, PHP, & MySQL)*. Bandung: Modula.
- [4] Saputra, A. 2011. *Trik dan Solusi Jitu Pemrograman PHP*. Jakarta: Alex Media Komputindo.
- [5] Wicaksono, Y. 2008. *Membangun Bisnis Online dengan Mambo+*. Alex Media. Jakarta: Komputindo.