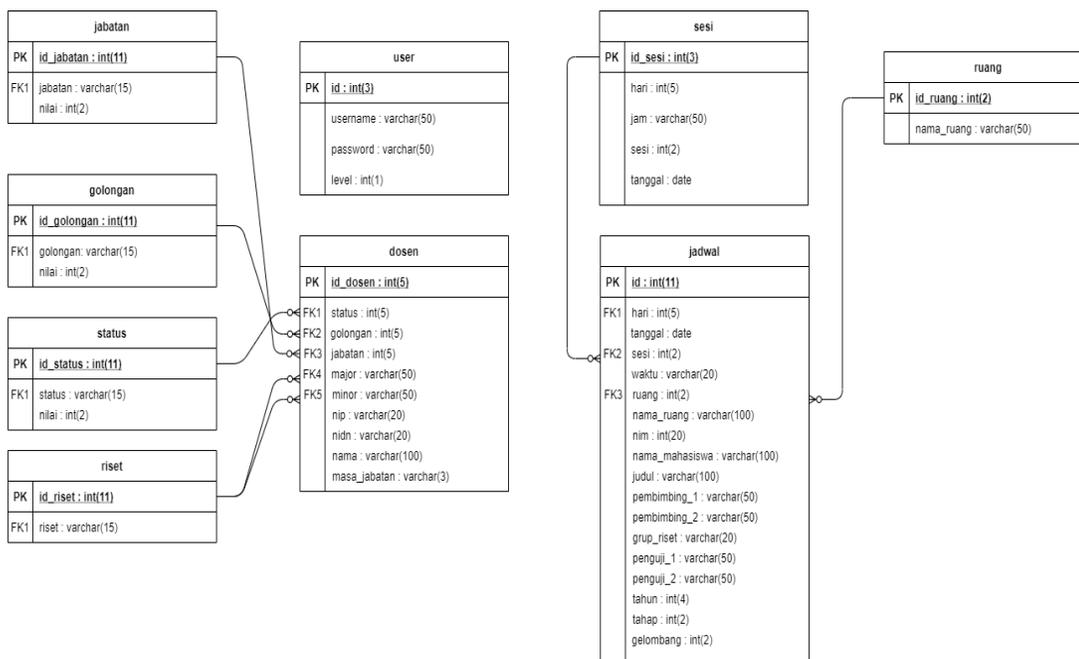


BAB V IMPLEMENTASI DAN PENGUJIAN

5.1. Implementasi Basis Data

Implementasi basis data yang digunakan untuk menunjang kebutuhan dalam proses penjadwalan pada sistem yang dibangun terdapat 9 tabel, diantaranya tabel *user*, dosen, status, golongan, jabatan, sesi, ruang, jadwal, dan riset seperti yang terdapat pada Gambar 5.1.



Gambar 5. 1 Implementasi Basis Data

5.2. Implementasi Proses Penjadwalan

Aplikasi dibuat menggunakan Bahasa pemrograman php dengan kombinasi bootstrap dan menggunakan *framework codeigniter*. Implementasi proses penjadwalan melalui beberapa proses, dari *upload file* peserta *.xls*, kemudian *generate* jadwal hingga *generate* penguji. Berikut adalah proses – proses yang akan dilakukan pada proses penjadwalan.

5.2.1. Upload *File* Peserta

Pada tahap upload peserta merupakan langkah awal untuk melakukan suatu penjadwalan yang selanjutnya akan digunakan untuk data peserta yang akan masuk ke data jadwal, berikut adalah potongan kode proses *upload file* peserta pada Gambar 5.1.

```
function UploadPeserta() {
    var fileUpload = document.getElementById("filePeserta");

    var regex = /^[a-zA-Z0-9\s_\.\-:]+(\.xls|.xlsx)$/;
    if (regex.test(fileUpload.value.toLowerCase())) {
        if (typeof (FileReader) !== "undefined") {
            var reader = new FileReader();
            if (reader.readAsBinaryString) {
                reader.onload = function (e) {
                    ProcessShowPeserta(e.target.result);
                };
                reader.readAsBinaryString(fileUpload.files[0]);
            } else {
                reader.onload = function (e) {
                    var data = "";
                    var bytes = new Uint8Array(e.target.result);
                    for (var i = 0; i < bytes.byteLength; i++) {
                        data += String.fromCharCode(bytes[i]);
                    }
                    ProcessShowPeserta(data);
                };
                reader.readAsArrayBuffer(fileUpload.files[0]);
            }
        } else {
            alert("This browser does not support HTML5.");
        }
    } else {
        alert("Please upload a valid Excel file.");
    }
};
```

Gambar 5. 2 Potongan Kode Proses *Upload File* Peserta

5.2.2. Generate Jadwal

Pada proses ini merupakan kelanjutan dari *upload file* peserta, yaitu *generate* jadwal dengan mengambil data sesi dan ruang yang telah tersedia pada database, berikut adalah potongan kode proses *generate* jadwal pada Gambar 5.2.

```

if(sesi==data_sesi.length){
    sesi=0;
    ruang++
}
if(ruang==data_ruang.length){
    ruang=0;
    hari++
}
if(hari==data_hari.length){
    hari=0;
}
var unix1 = data_hari[hari].tanggal+'_'+data_sesi[sesi].ses
i+'_'+data_ruang[ruang].id_kelas+'_'+penguji_1.id_dosen
var unix2 = data_hari[hari].tanggal+'_'+data_sesi[sesi].ses
i+'_'+data_ruang[ruang].id_kelas+'_'+penguji_2.id_dosen
var unix3 = data_hari[hari].tanggal+'_'+data_sesi[sesi].ses
i+'_'+data_ruang[ruang].id_kelas
if((!check1.includes(unix1) && (!check2.includes(unix2)) &&
!check3.includes(unix3))){
    check1.push(unix1)
    check2.push(unix2)
    check3.push(unix3)

    data_jadwal.row.add([
        i+1,
        data_hari[hari].hari,
        data_hari[hari].tanggal,
        data_sesi[sesi].sesi,
        data_sesi[sesi].jam,
        data_ruang[ruang].id_kelas,

```

Gambar 5. 3 Potongan Kode Proses *Generate* Jadwal

5.2.3. Generate Penguji

Pada tahap ini merupakan kelanjutan dari *generate* jadwal, yaitu *generate* penguji, dimana ketika *generate* jadwal telah berhasil, maka selanjutnya menentukan penguji sesuai aturan yang telah ditentukan sebelumnya, berikut potongan kode proses *generate* penguji pada Gambar 5.3.

```

if(role_1==true){
    // console.log('masuk 1')
    if((iterasi_1 >= maks_iterasi) || ((tmp_peng
uji[id_1].length<maksimal_menguji) && (data_penguji_1.major).tol
owerCase() == (peserta.group_riset).toLowerCase() && ((data_peng
uji_1.status) == '1') && ((peserta.pembimbing_1!=data_penguji_1.
nama) && (peserta.pembimbing_2!=data_penguji_1.nama) ))){
        tmp_penguji[id_1].push(id_1)
        penguji_1 = data_penguji_1
        loop = false
    }
}
}else{
    // console.log('masuk 2')
    if((iterasi_1 >= maks_iterasi) || ((tmp_peng
uji[id_1].length<maksimal_menguji) && (data_penguji_1.major).tol
owerCase() == (peserta.group_riset).toLowerCase() && ((peserta.p
embimbing_1!=data_penguji_1.nama) && (peserta.pembimbing_2!=data
_penguji_1.nama) ))){
        tmp_penguji[id_1].push(id_1)
        penguji_1 = data_penguji_1
        loop = false
    }
}
}
}
var loop2 = true
var data_penguji_2,id_2;
while (loop2) {

```

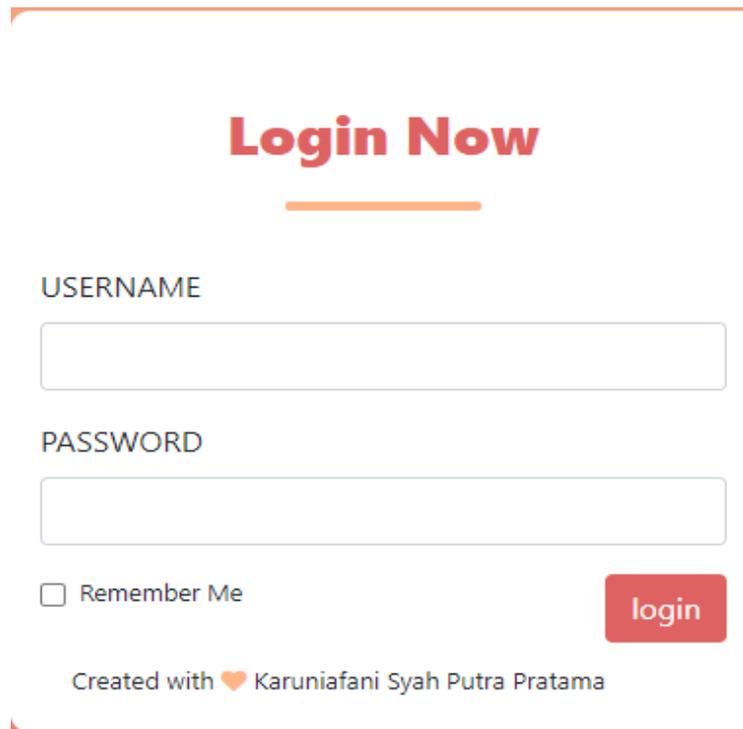
Gambar 5. 4 Potongan Kode Proses *Generate* Penguji

5.3. Implementasi Tampilan

Implementasi tampilan berdasarkan perancangan antarmuka yang telah dibuat sebelumnya adalah sebagai berikut.

5.3.1. Implementasi Tampilan *Login*

Pada halaman *login* terdapat kolom *username* dan *password* dan tampilan halaman *login* sudah sesuai dengan perancangan yang telah dibuat sebelumnya, berikut tampilan *login* pada Gambar 5.4.



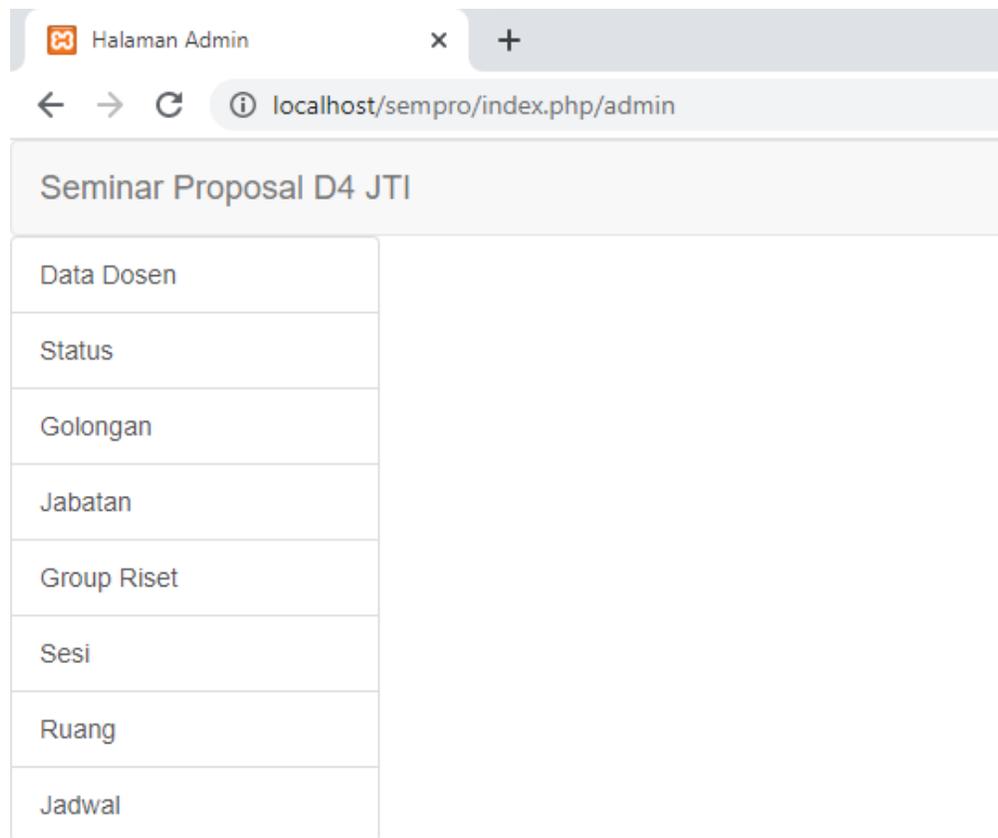
The image shows a login form with the following elements:

- Title:** "Login Now" in a large, bold, red font, centered at the top.
- Username Field:** A text input field labeled "USERNAME" in all caps.
- Password Field:** A text input field labeled "PASSWORD" in all caps.
- Remember Me:** A checkbox labeled "Remember Me".
- Login Button:** A red button with the text "login" in white.
- Footer:** "Created with ❤️ Karunifani Syah Putra Pratama" in a small font at the bottom.

Gambar 5. 5 Implementasi Tampilan *Login*

5.3.2. Implementasi Tampilan Admin

Pada tampilan admin terdapat beberapa fitur yang sebelumnya sudah dirancang, pada halaman admin juga merupakan tampilan awal ketika admin berhasil *login*, berikut adalah tampilan halaman admin pada Gambar 5.5.



Gambar 5. 6 Implementasi Tampilan Admin

5.3.3. Implementasi Tampilan Data Dosen

Pada tampilan data dosen berisikan data – data dosen yang dibutuhkan untuk *men-generate* penguji, berikut tampilan data dosen pada Gambar 5.6.

No	NIP	NIDN	Nama	Status	Golongan	Jabatan	Masa Jabatan	Major	Minor	Total Nilai	Aksi
1	196107052005011002	0005078102	Ahmadi Yuli Ananta, ST., M.M.	PNS	IIIB	LEKTOR	15	INFORMATION SYSTEM	COMPUTER NETWORK	25	Edit Hapus
2	196108102005012002	0010086101	Ariadi Retno Reti, S.Kom., M.Kom.	PNS	IIIC	LEKTOR	15	SISTEM CERDAS	INFORMATION SYSTEM	26	Edit Hapus
3	197903130008121002	0013037905	Arief Prasetyo, S.Kom., M.Kom.	PNS	IIIB	ASISTEN AHLJ	12	COMPUTER NETWORK		21	Edit Hapus
4	197606250005012001	0025067607	Aliqan Nurul Asri, S.Pd., M.Pd.	PNS	IIID	LEKTOR	15			27	Edit Hapus
5	196106092010121002	0009088107	Banni Satna Andoko, S.Kom., M.MSI	PNS	IIIC	LEKTOR	10			21	Edit Hapus
6	196201051990031002	0005016211	Budi Harjanto, ST., M.MKom.	PNS	IVA	LEKTOR KEPALA	30	INFORMATION SYSTEM		44	Edit Hapus
7	197202022005011002	002027214	Cahya Rahmad, ST., M.Kom., Dr. Eng	PNS	IVA	LEKTOR KEPALA	15	COMPUTER VISION		29	Edit Hapus
8	19621128198811001	0028116204	Deddy Kusbiyanto PA, Ir., M.Mkom	PNS	IVA	LEKTOR KEPALA	32	INFORMATION SYSTEM	SISTEM CERDAS	46	Edit Hapus
9	196311092014042001	0009116305	DhebyS Suryani, S.Kom., MT	PNS	IIIB	ASISTEN AHLJ	6	MULTIMEDIA AND GAME	SISTEM CERDAS	15	Edit Hapus

Gambar 5. 7 Implementasi Tampilan Data Dosen

5.3.4. Implementasi Tampilan Status

Pada tampilan status berisikan data – data status yang dibutuhkan untuk men-generate pengujian, berikut tampilan status pada Gambar 5.7.

Id_status	Status	Nilai	Aksi
1	PNS	6	Edit Hapus
2	CPNSK	5	Edit Hapus
3	CPNSM	4	Edit Hapus
4	KONTRAK	3	Edit Hapus
5	LB	2	Edit Hapus
6	MKU	1	Edit Hapus

Gambar 5. 8 Implementasi Tampilan Status

5.3.5. Implementasi Tampilan Golongan

Pada tampilan golongan berisikan data – data golongan yang dibutuhkan untuk *men-generate* penguji, berikut tampilan golongan pada Gambar 5.8.

id_golongan	golongan	Nilai	Aksi
1	IVE	9	Edit Hapus
2	IVD	8	Edit Hapus
3	IVC	7	Edit Hapus
4	IVB	6	Edit Hapus
5	IVA	5	Edit Hapus
6	IID	4	Edit Hapus
7	IIC	3	Edit Hapus
8	IIB	2	Edit Hapus
9	IIA	1	Edit Hapus

Gambar 5. 9 Implementasi Tampilan Golongan

5.3.6. Implementasi Tampilan Jabatan

Pada tampilan jabatan berisikan data – data jabatan yang dibutuhkan untuk *men-generate* penguji, berikut tampilan jabatan pada Gambar 5.9.

id_jabatan	jabatan	Nilai	Aksi
1	GURU BESAR	4	Edit Hapus
2	LEKTOR KEPALA	3	Edit Hapus
3	LEKTOR	2	Edit Hapus
4	ASISTEN AHLI	1	Edit Hapus

Gambar 5. 10 Implementasi Tampilan Jabatan

5.3.7. Implementasi Tampilan *Group Riset*

Pada tampilan *group riset* berisikan data – data *group riset* yang dibutuhkan untuk *men-generate* penguji, berikut tampilan *group riset* pada Gambar 5.10.

Seminar Proposal D4 JTI		Keluar
Data Dosen	Tambah Data	
Status		
Golongan	No	Group Riset
Jabatan	1	INFORMATION SYSTEM Edit Hapus
Group Riset	2	SISTEM CERDAS Edit Hapus
Sesi	3	COMPUTER VISION Edit Hapus
Ruang	4	MULTIMEDIA AND GAME Edit Hapus
Jadwal	5	COMPUTER NETWORK Edit Hapus

Gambar 5. 11 Implementasi Tampilan *Group Riset*

5.3.8. Implementasi Tampilan Sesi

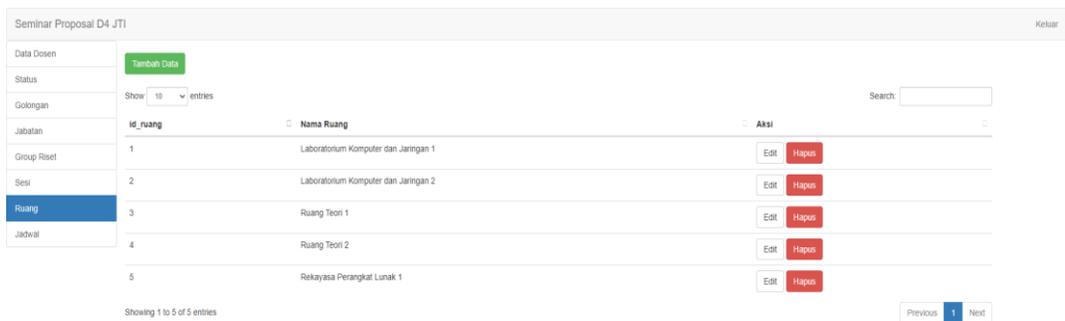
Pada tampilan Sesi berisikan data – data Sesi yang dibutuhkan untuk *men-generate* jadwal, berikut tampilan Sesi pada Gambar 5.11.

Seminar Proposal D4 JTI		Keluar
Data Dosen	Tambah Data	
Status		
Golongan	Id_sesi	Hari Tanggal Sesi Jam
Jabatan	3	1 2020-07-07 1 08.00-08.30 Edit Hapus
Group Riset	4	1 2020-07-07 2 08.45-09.15 Edit Hapus
Sesi	5	1 2020-07-07 3 09.30-10.00 Edit Hapus
Ruang	6	1 2020-07-07 4 10.15-10.45 Edit Hapus
Jadwal	7	2 2020-07-08 1 08.00-08.30 Edit Hapus
	8	2 2020-07-08 2 08.45-09.15 Edit Hapus
	9	2 2020-07-08 3 09.30-10.00 Edit Hapus
	10	2 2020-07-08 4 10.15-10.45 Edit Hapus
	11	3 2020-07-09 1 08.00-08.30 Edit Hapus
	12	3 2020-07-09 2 08.45-09.15 Edit Hapus
	13	3 2020-07-09 3 09.30-10.00 Edit Hapus
	14	3 2020-07-09 4 10.15-10.45 Edit Hapus

Gambar 5. 12 Implementasi Tampilan Sesi

5.3.9. Implementasi Tampilan Ruang

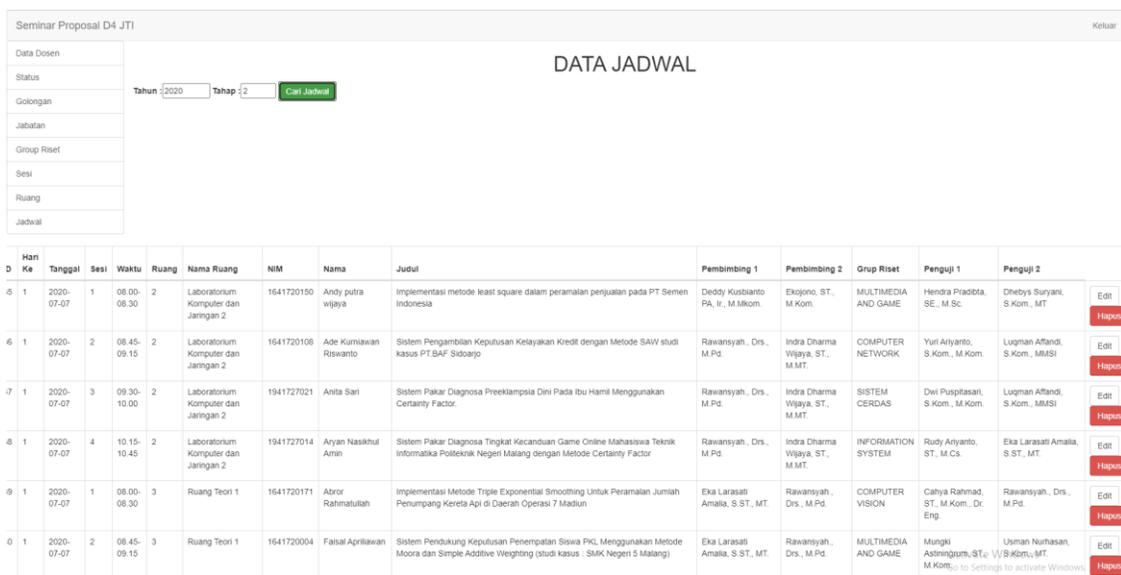
Pada tampilan Ruang berisikan data – data Ruang yang dibutuhkan untuk *generate* jadwal, berikut tampilan Ruang pada Gambar 5.12.



Gambar 5. 13 Implementasi Tampilan Ruang

5.3.10. Implementasi Tampilan Jadwal

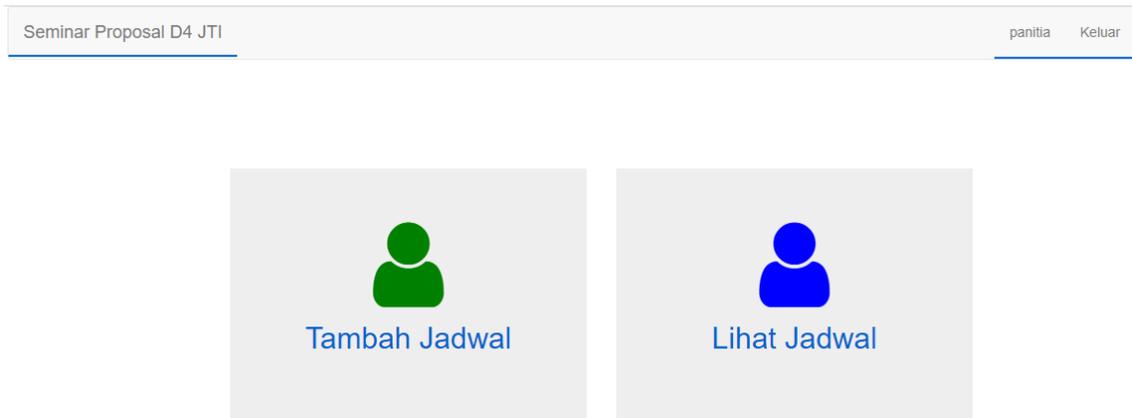
Pada tampilan Jadwal berisikan data – data jadwal yang telah berhasil di *generate* dan telah di simpan ke *database system*, berikut tampilan jadwal pada Gambar 5.13.



Gambar 5. 14 Implementasi Tampilan Jadwal

5.3.11. Implementasi Tampilan Panitia

Pada tampilan panitia terdapat beberapa fitur yang sebelumnya sudah dirancang, pada halaman panitia juga merupakan tampilan awal ketika panitia berhasil *login*, berikut adalah tampilan halaman panitia pada Gambar 5.14.



Gambar 5. 15 Implementasi Tampilan Panitia

5.3.12. Implementasi Tampilan Tambah Jadwal

Pada tampilan tambah jadwal, terdapat fitur upload peserta ujian dan *generate* jadwal, tampilan ini sudah disesuaikan dengan rancangan yang akan dibuat, berikut tampilan tambah jadwal pada Gambar 5.15.

Seminar Proposal D4 JTI panitia Keluar

PEMBUATAN JADWAL

1 UPLOAD PESERTA

Choose File Peserta Ujian.xlsx Lanjut Generate Jadwal

Upload

Show 10 entries Search:

Id	Nim	Nama	Judul	Pembimbing 1	Pembimbing 2	Group	Deskripsi
1	1641720163	ACHMAD MAHFUD	sistem peramalan modal kegiatan pembelian tebu pada UD juara agro perkasa di kecamatan bantur.	Budi Harijanto, ST., M.MKom.	Usman Nurhasan, S.Kom., MT.	INFORMATION SYSTEM	Sistem informasi yang digunakan untuk meramalkan jumlah penumpang kereta api di Daerah Operasi 7 Madiun dimasa mendatang dengan menggunakan data dari ...
2	1641720130	Anggik Ardianto	Pembelajaran Pada Ujian Online Menggunakan Forward Chaining Berdasarkan Tingkat Kesulitan Soal Menggunakan Metode Fuzzy Tsukamoto	Budi Harijanto, ST., M.MKom.	Usman Nurhasan, S.Kom., MT.	SISTEM CERDAS	Pada studi kasus kali ini saya telah mengembangkan bisnis penanaman porang di daerah kediri, akan tetapi terdapat sebuah permasalahan untuk pemilihan ...
3	1641720158	Allef Himawan	SISTEM INFORMASI PERAMALAN KETERSEDIAAN STOK SUKU CADANG SEPERA	Deddy Kusbianto PA, Ir., M.Mkom.	Ekojono, ST., M.Kom.	MULTIMEDIA AND GAME	Membuat chatbot yang bisa melayani pemesanan pijat bayi berdasarkan jadwal yang tersedia, karena saat ini hanya ada 1 admin yg melayani

Gambar 5. 16 Implementasi Tampilan Tambah Jadwal

5.3.13. Implementasi Tampilan *Generate* Penguji

Pada tampilan *generate* penguji terdapat fitur simpan jadwal, dan *generate* penguji ulang berdasarkan rule yang diinginkan, berikut tampilan *generate* penguji pada Gambar 5.16.

PEMBUATAN JADWAL

1 UPLoad PESERTA 2 GENERATE JADWAL

Tahun: Tahap:

Pilih Rule yang akan digunakan untuk menentukan penguji

- Status PNS
- Jabatan Fungsional
- Golongan
- Masa Jabatan

Show: entries Search:

No	Hari Ke	Tanggal	Sesi	Waktu	Ruang	Nama Ruang	NIM	Nama	Judul	Pembimbing 1	Pembimbing 2	Grup Riset	Penguji 1	Penguji 2
1	1	2020-07-07	1	08.00-08.30	1	Laboratorium Komputer dan Jaringan 1	1641720163	ACHMAD MAHFUD	sistem peramalan modal kegiatan pembelian tebu pada UD juara agro perkasa di kecamatan bantur.	Budi Harjanto, ST., M.MKom.	Usman Nurhasan, S.Kom., MT.	INFORMATION SYSTEM	Deddy Kusbianto, PA, Ir., M.Mkom.	Rudy Ariyanto, ST., M.Cs.
2	1	2020-07-07	2	08.45-09.15	1	Laboratorium Komputer dan Jaringan 1	1641720130	Anggik Ardianto	Pembelajaran Pada Ujian Online Menggunakan Forward Chaining Berdasarkan Tingkat Kesulitan Soal Menggunakan Metode Fuzzy Tsukamoto	Budi Harjanto, ST., M.MKom.	Usman Nurhasan, S.Kom., MT.	SISTEM CERDAS	Gunawan Budi Prasetyo, ST., MMT., Ph.D.	Eka Larasati Amalia, S.ST., MT.
3	1	2020-07-07	3	09.30-10.00	1	Laboratorium Komputer dan Jaringan 1	1641720158	Alief Himawan	SISTEM INFORMASI PERAMALAN KETERSEDIAAN STOK SUKU	Deddy Kusbianto, PA, Ir., M.MKom.	Ekojono, ST., M.Kom.	MULTIMEDIA AND GAME	Mungki Astiningrum, ST., M.Kom.	Hendra Pradibta SE., M.Sc.

Gambar 5. 17 Implementasi Tampilan *Generate* Penguji

5.4. Pengujian Fungsional Sistem

Pengujian fungsionalitas sistem dilakukan dengan cara menjalankan tiap fitur dalam aplikasi dan melihat kesesuaian hasil yang terjadi dengan hasil yang diharapkan.

Tabel 5. 1 Tabel pengujian fungsional sistem

No	Use Case	Keterangan	Status
1.	Mengolah data admin	Admin dapat melihat, tambah, edit, hapus pada data admin pada fitur data admin.	Sesuai
2.	Mengelola data dosen	Admin dapat melihat, tambah, edit, hapus pada data dosen pada fitur data dosen.	Sesuai
3.	Mengelola status	Admin dapat melihat, tambah, edit, hapus pada data status pada fitur data status.	Sesuai

4.	Mengelola Golongan	Admin dapat melihat, tambah, edit, hapus pada data golongan pada fitur data golongan.	Sesuai
5.	Mengelola Jabatan	Admin dapat melihat, tambah, edit, hapus pada data jabatan pada fitur data jabatan.	Sesuai
6.	Mengelola <i>Group Riset</i>	Admin dapat melihat, tambah, edit, hapus pada data <i>Group</i> riset pada fitur data <i>Group</i> riset.	Sesuai
7	Mengelola Sesi	Admin dapat melihat, tambah, edit, hapus pada data sesi pada fitur data sesi.	Sesuai
8	Mengelola Ruang	Admin dapat melihat, tambah, edit, hapus pada data ruang pada fitur data ruang.	Sesuai
9	Mengelola Jadwal	Admin dan panitia dapat melihat, tambah, edit, hapus pada data jadwal pada fitur data jadwal.	Sesuai
10	Membuat Jadwal	Panitia dapat men- <i>generate</i> jadwal sesuai dengan aturan yang telah dibuat sebelumnya.	Sesuai

5.5. Pengujian Akurasi Sistem

Cara kerja penerapan *Rule based generator* pada penjadwalan ini diawali dengan membuat jadwal terlebih dahulu dengan aturan tidak boleh ada jadwal yang bentrok, kemudian setelah jadwal berhasil dibuat selanjutnya menentukan penguji 1 dan penguji 2. Pengujian seluruh *rule* pada penjadwalan dapat dilihat pada table di bawah ini :

Tabel 5. 2 Tabel Hasil Pengujian

No	Aturan – aturan pada <i>Rule based generator</i>	Cocok	Tidak
1	Semua sesi pada hari ke-1 akan masuk ke setiap data ruang yang tersedia, ketika data ruang telah habis, maka hari akan bertambah	✓	
2	Penguji 1 wajib PNS dan pada <i>Group</i> riset berstatus Major	✓	
3	Penguji 2 wajib PNS dan pada <i>Group</i> riset dapat berstatus Major/Minor	✓	
4	Jika penguji 1 dan 2 memiliki status yang sama (PNS) dan memiliki status <i>Group</i> riset yang sama (Major), maka perlu dipertimbangkan jabatan fungsionalnya, dan jika masih sama maka dapat di pertimbangkan dengan masa jabatan dosen	✓	
5	Jumlah menguji tiap dosen harus proporsional	✓	
6	Pembimbing 1 dan pembimbing 2 tidak sama dengan penguji 1 dan penguji 2	✓	

Pengukuran akurasi sistem dapat dilakukan dengan cara melakukan pembagian jumlah hasil benar dengan jumlah hasil benar ditambah hasil salah dikalikan 100% (Arman Dwi Jatmiko, 2017). Atau bisa melihat persamaan dibawah ini :

$$Akurasi = \frac{Hasil\ Benar}{(Hasil\ Benar + Hasil\ Salah)} \times 100\%$$

Hasil pengujian *Rule based generator* pada penjadwalan didapatkan 6 hasil benar dan juga 0 hasil kesalahan. Berdasarkan hasil tersebut kemudian dapat dihitung akurasi menggunakan persamaan diatas menjadi :

$$Akurasi = \frac{6}{(6+0)} \times 100\% = 100\%$$