

SURAT PERNYATAAN

Yang bertanda tangan dibawah ini :

N a m a : ADITYO PERMANA WIBOWO
Kewarganegaraan : WNI
Alamat : PERUM SUMBERADI ASRI BLOK A. 13 JUMENENG KIDUL
RT 008 / RW 039 SUMBERADI, MLATI, SLEMAN, DIY

N a m a : DONNY AVIANTO
Kewarganegaraan : WNI
Alamat : PERUM NOGOTIRTO III JL KAPUAS B29 RT 008 / RW 017
GAMPING, SLEMAN, DIY

N a m a : IMANTOKO
Kewarganegaraan : WNI
Alamat : KOMP PELANGI JAYA LESTARI BLOK C 51A RT 038 / RW 011
CEMPAKA, BANJAR BARU, KALIMANTAN SELATAN

Dengan ini menyatakan bahwa :

1. Karya Cipta yang kami mohonkan :

Berupa : Program Komputer
Berjudul : ALGEN Scheduling (Automatically Generator Scheduling)

- Tidak meniru dan tidak sama secara esensial dengan Karya Cipta milik pihak lain atau obyek kekayaan intelektual lainnya sebagaimana dimaksud dalam Pasal 68 ayat (2);
 - Bukan merupakan Ekspresi Budaya Tradisional sebagaimana dimaksud dalam Pasal 38;
 - Bukan merupakan Ciptaan yang tidak diketahui penciptanya sebagaimana dimaksud dalam Pasal 39;
 - Bukan merupakan hasil karya yang tidak dilindungi Hak Cipta sebagaimana dimaksud dalam Pasal 41 dan 42;
 - Bukan merupakan Ciptaan seni lukis yang berupa logo atau tanda pembeda yang digunakan sebagai merek dalam perdagangan barang/jasa atau digunakan sebagai lambang organisasi, badan usaha, atau badan hukum sebagaimana dimaksud dalam Pasal 65 dan;
 - Bukan merupakan Ciptaan yang melanggar norma agama, norma susila, ketertiban umum, pertahanan dan keamanan negara atau melanggar peraturan perundang-undangan sebagaimana dimaksud dalam Pasal 74 ayat (1) huruf d Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.
2. Sebagai pemohon mempunyai kewajiban untuk menyimpan asli contoh ciptaan yang dimohonkan dan harus memberikan apabila dibutuhkan untuk kepentingan penyelesaian sengketa perdata maupun pidana sesuai dengan ketentuan perundang-undangan.
3. Karya Cipta yang kami mohonkan pada Angka 1 tersebut di atas tidak pernah dan tidak sedang dalam sengketa pidana dan/atau perdata di Pengadilan.
4. Dalam hal ketentuan sebagaimana dimaksud dalam Angka 1 dan Angka 3 tersebut di atas kami langgar, maka kami bersedia secara sukarela bahwa :
- a. Permohonan karya cipta yang saya ajukan dianggap ditarik kembali;
Karya Cipta yang telah terdaftar dalam Daftar Umum Ciptaan Direktorat Hak Cipta, Direktorat Jenderal Hak Kekayaan Intelektual, Kementerian Hukum Dan Hak Asasi Manusia R.I dihapuskan sesuai dengan ketentuan perundang-undangan yang berlaku.
 - b. Dalam hal kepemilikan Hak Cipta yang dimohonkan secara elektronik sedang dalam berperkara dan/atau sedang dalam gugatan di Pengadilan maka status kepemilikan surat pencatatan elektronik tersebut ditangguhkan menunggu putusan Pengadilan yang berkekuatan hukum tetap.

Demikian Surat pernyataan ini kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.

Yogyakarta, 21 Oktober 2019

Yang menyatakan,





PROGRAM KOMPUTER

“ALGEN Scheduling (*Automatically Generator Scheduling*)”
APLIKASI PENJADWALAN UJIAN PENDADARAN PROYEK TUGAS
AKHIR MENGGUNAKAN ALGORITMA GENETIKA

Oleh:

Adityo Permana Wibowo

Donny Avianto

Imantoko

PROGRAM UTAMA

1. Source Code *Generator* Utama Algen untuk Penjadwalan

```
1. public StringBuilder generateSchedule()
2.     {
3.         Stopwatch timer = new Stopwatch(); Random rn = new Random();
4.         Generator demo = new Generator(); StringBuilder schedule = new StringBuilder();
5.         demo.population.initializePopulation();
6.         //Calculate fitness of each individual
7.         demo.population.calculateFitness();
8.         //show all individu in population
9.         demo.population.ShowEachIndividuByDay();
10.        Console.WriteLine("=====");
11.        timer.Start();
12.        do
13.        {
14.            ++demo.generationCount;
15.            demo.selection(); // pilih 2 individu terbaik sebagai induk
16.            Console.WriteLine("1st fittest ");
17.            //demo.population.ShowIndividuByObj(demo.fittest);
18.            Console.WriteLine("2nd fittest ");
19.            //demo.population.ShowIndividuByObj(demo.secondFittest);
20.            Console.WriteLine("=====");
21.
22.            Individual best_offspring = demo.crossover();
23.            demo.population.calculateFitness();
24.            //Add fittest offspring to population
25.            demo.addFittestOffspring(best_offspring);
26.            demo.population.calculateFitness();
27.            Console.WriteLine("=====POPULASI SETELAH CROSS=====");
28.
29.            //demo.population.ShowEachIndividuByDay();
30.            Console.WriteLine("=====");
31.
32.            if (rn.Next() % 50 == 0)
33.            {
34.                demo.mutation(demo.population.individuals);
35.            }
36.            //Calculate new fitness value
37.            demo.population.calculateFitness();
38.            Console.WriteLine("=====POPULASI SETELAH MUTASI=====");
39.
40.            //demo.population.ShowEachIndividuByDay();
41.            Console.WriteLine("=====");
42.
43.            //Select the fittest
44.            demo.fittest = demo.population.getFittest();
45.            Console.WriteLine("Terbaik dari Generasi " + demo.generationCount);
46.            schedule = demo.population.ShowIndividuByObj(demo.fittest);
47.            Console.WriteLine("=====");
48.
49.        } while (demo.generationCount <= 999 && demo.fittest.fitness != 0);
50.        timer.Stop();
51.        #region timer adjusment codes
52.        double frequency = Stopwatch.Frequency;
53.        double nanosecPerTick = (1000 * 1000 * 1000) / frequency;
54.        double durNanosec = timer.ElapsedTicks * nanosecPerTick;
55.        double durMilisec = durNanosec * 0.000001;
56.        double durSec = durMilisec * 0.001;
57.        Console.WriteLine("\n\nDuration: " + String.Format("{0:0.000}", durSec) + " Second");
58.
59.        #endregion
60.        return schedule;
61.    }
62. }
```

2. Source Code proses Crossover

```
1. public Individual crossover()
2.     {
3.         //Select a random crossover point
4.         Random rn = new Random();
5.         int crossOverPoint = rn.Next(population.individuals[0].genelength);
6.
7.         Individual offspring1 = new Individual(fittest);
8.         Individual offspring2 = new Individual(secondFittest);
9.
10.        //Swap value 2
11.        for (int i = 0; i < crossOverPoint; i++)
12.            {
13.                int[] arrTempOffsp1 = new int[3];
14.                int[] arrTempOffsp2 = new int[3];
15.
16.                for (int j = 0; j < 3; j++)
17.                    {
18.                        arrTempOffsp1[j] = offspring1.jadwals[i].Tim_penguji[j].Id_dosen;
19.                        arrTempOffsp2[j] = offspring2.jadwals[i].Tim_penguji[j].Id_dosen;
20.                    }
21.
22.                    offspring1.jadwals[i].Tim_penguji[0].Id_dosen = arrTempOffsp2[0];
23.                    offspring1.jadwals[i].Tim_penguji[1].Id_dosen = arrTempOffsp2[1];
24.                    offspring1.jadwals[i].Tim_penguji[2].Id_dosen = arrTempOffsp2[2];
25.
26.                    offspring2.jadwals[i].Tim_penguji[0].Id_dosen = arrTempOffsp1[0];
27.                    offspring2.jadwals[i].Tim_penguji[1].Id_dosen = arrTempOffsp1[1];
28.                    offspring2.jadwals[i].Tim_penguji[2].Id_dosen = arrTempOffsp1[2];
29.                }
30.
31.                offspring1.calcFitness();
32.                offspring2.calcFitness();
33.
34.                //tambahan: cetak keturunan
35.                Console.WriteLine("1st offspring");
36.                //population.ShowIndividuByObj(offspring1);
37.                Console.WriteLine("2nd offspring");
38.                //population.ShowIndividuByObj(offspring2);
39.                Console.WriteLine("=====");
40.        };
41.
42.        if (offspring1.fitness >= offspring2.fitness)
43.            {
44.                return offspring1;
45.            }
46.        else
47.            {
48.                return offspring2;
49.            }
50.    }
```

3. Source Code Proses Mutasi Individu

```
1. public void mutation(Individual[] arr_pops)
2.     {
3.         Random rn = new Random();
4.         for (int x = 0; x < population.popSize; x++)
5.             {
6.                 for (int i = 0; i < 18; i++) //< 5 karena ada 5 hari
7.                     {
8.                         Dosen[] dsn1;
9.                         Dosen[] dsn2;
10.                        dsn1 = arr_pops[x].jadwals[i].Tim_penguji;
11.                        dsn2 = arr_pops[x].jadwals[i + 2].Tim_penguji; // + 2 karena ruangan ada 2
12.
13.                        if (dsn1.Intersect(dsn2, new DosenComparer()).Any())
14.                            {
15.                                //fitness = fitness - 1;
16.                                Individual proto = new Individual(rn);
17.                                for (int idx = 0; idx < proto.jadwals.Length; idx++)
18.                                    {
19.                                        if (!(dsn1.Intersect(proto.jadwals[idx].Tim_penguji, new DosenCompare
20.                                        r()).Any()))
21.                                            {
22.                                                arr_pops[x].jadwals[i].Tim_penguji = proto.jadwals[idx].Tim_penguji;
23.                                                break;
24.                                            }
25.                                        }
26.                                    }
27.                                }
28.                            }
29.                    }
30.            }
```

4. Source Code Proses Pemilihan Tim Dosen Penguji

```
1. string PrintTimPenguji(Dosen[] arr)
2.     {
3.         StringBuilder sb = new StringBuilder();
4.         for (int i = 0; i < 3; i++)
5.             {
6.                 sb.Append(arr[i].Id_dosen);
7.             }
8.         return sb.ToString();
9.     }
```


5. Source Code Penentuan Populasi Genetik untuk Penjadwalan

```
1. //otomatis membuat 10 individu (1 populasi = 10 individu)
2. public void initializePopulation()
3. {
4.     for (int i = 0; i < individuals.Length; i++)
5.     {
6.         individuals[i] = new Individual(rn);
7.     }
8. }
```

```
1. //Get the fittest individual
2. public Individual getFittest()
3. {
4.     int maxFitIdx = 0;
5.     for (int i = 0; i < individuals.Length; i++)
6.     {
7.         if (individuals[maxFitIdx].fitness < individuals[i].fitness)
8.         {
9.             maxFitIdx = i;
10.        }
11.    }
12.    fittest = individuals[maxFitIdx].fitness;
13.    idx_1stFst = maxFitIdx;
14.    return individuals[maxFitIdx];
15. }
```

```
1. //Get the second most fittest individual
2. public Individual getSecondFittest()
3. {
4.     int maxFit1 = 0;
5.     int maxFit2 = 1;
6.     for (int i = 0; i < individuals.Length; i++)
7.     {
8.         if (individuals[i].fitness > individuals[maxFit1].fitness && i != maxFit1)
9.         {
10.            maxFit2 = maxFit1;
11.            maxFit1 = i;
12.        }
13.        else if (individuals[i].fitness > individuals[maxFit2].fitness)
14.        {
15.            maxFit2 = i;
16.        }
17.    }
18.    idx_2ndFst = maxFit2;
19.    return individuals[maxFit2];
20. }
```

```

1. //Get index of least fittest individual
2. public int getLeastFittestIndex()
3. {
4.     int minFitIdx = 0;
5.     for (int i = 0; i < individuals.Length; i++)
6.     {
7.         if (individuals[minFitIdx].fitness >= individuals[i].fitness)
8.         {
9.             minFitIdx = i;
10.        }
11.    }
12.    return minFitIdx;
13. }
14.
15. //Calculate fitness of each individual
16. public void calculateFitness()
17. {
18.     for (int i = 0; i < individuals.Length; i++)
19.     {
20.         individuals[i].calcFitness();
21.     }
22.     getFittest();
23. }

```

```

1. public void ShowEachIndividuByDay()
2. {
3.     for (int pop_idx = 0; pop_idx < individuals.Length; pop_idx++)
4.     {
5.         Console.WriteLine("Individu " + (pop_idx + 1));
6.         Jadwal[] jd = individuals[pop_idx].jadwals;
7.         StringBuilder sb_chrome = new StringBuilder();
8.         for (int i = 0; i < jd.Length; i++)
9.         {
10.            for (int j = 0; j < 3; j++)
11.            {
12.                if ((jd[i].Tim_penguji[j].Id_dosen.ToString()).Length < 2)
13.                {
14.                    sb_chrome.Append(" ");
15.                }
16.                sb_chrome.Append(jd[i].Tim_penguji[j].Id_dosen + "-");
17.            }
18.            if (jd[i].Peserta_ujian.Id_mhs < 10)
19.            {
20.                sb_chrome.Append(" ");
21.            }
22.            sb_chrome.Append(jd[i].Peserta_ujian.Id_mhs + "-");
23.            if (jd[i].Ruang_ujian.Id_ruang < 10)
24.            {
25.                sb_chrome.Append(" ");
26.            }
27.            sb_chrome.Append(jd[i].Ruang_ujian.Id_ruang + "|");
28.            if ((i + 1) % 4 == 0)
29.            {
30.                sb_chrome.Append("\n");
31.            }
32.            Console.Write(sb_chrome.ToString());
33.            Console.WriteLine("Fitness: " + individuals[pop_idx].fitness + "\n");
34.        }
35.    }
36. }

```

```

1. public void ShowFittestIndividuByDay()
2.     {
3.         Individual idv_fittest = getFittest();
4.         Jadwal[] jd = idv_fittest.jadwals;
5.         StringBuilder sb_chrome = new StringBuilder();
6.         for (int i = 0; i < jd.Length; i++)
7.         {
8.             for (int j = 0; j < 3; j++)
9.             {
10.                if ((jd[i].Tim_penguji[j].Id_dosen.ToString()).Length < 2)
11.                {
12.                    sb_chrome.Append(" ");
13.                }
14.
15.                sb_chrome.Append(jd[i].Tim_penguji[j].Id_dosen + "-");
16.            }
17.
18.
19.            if (jd[i].Peserta_ujian.Id_mhs < 10)
20.            {
21.                sb_chrome.Append(" ");
22.            }
23.            sb_chrome.Append(jd[i].Peserta_ujian.Id_mhs + "-");
24.
25.
26.            if (jd[i].Ruang_ujian.Id_ruang < 10)
27.            {
28.                sb_chrome.Append(" ");
29.            }
30.            sb_chrome.Append(jd[i].Ruang_ujian.Id_ruang + "|");
31.
32.            if ((i + 1) % 4 == 0)
33.            {
34.                sb_chrome.Append("\n");
35.            }
36.        }
37.
38.        Console.Write(sb_chrome.ToString());
39.        Console.WriteLine("Fitness: " + idv_fittest.fitness + "\n");
40.    }

```



```

1.     public StringBuilder ShowIndividuByObj(Individual ind)
2.     {
3.         Individual idv_toshow = ind;
4.         Jadwal[] jd = idv_toshow.jadwals;
5.         StringBuilder sb_chrome = new StringBuilder();
6.         for (int i = 0; i < jd.Length; i++)
7.         {
8.             for (int j = 0; j < 3; j++)
9.             {
10.                if ((jd[i].Tim_penguji[j].Id_dosen.ToString()).Length < 2)
11.                {
12.                    sb_chrome.Append(" ");
13.                }
14.
15.                sb_chrome.Append(jd[i].Tim_penguji[j].Id_dosen + "-");
16.            }
17.
18.
19.            if (jd[i].Peserta_ujian.Id_mhs < 10)
20.            {
21.                sb_chrome.Append(" ");
22.            }
23.            sb_chrome.Append(jd[i].Peserta_ujian.Id_mhs + "-");
24.
25.
26.            if (jd[i].Ruang_ujian.Id_ruang < 10)
27.            {
28.                sb_chrome.Append(" ");
29.            }
30.            sb_chrome.Append(jd[i].Ruang_ujian.Id_ruang + "|");
31.
32.            if ((i + 1) % 4 == 0)
33.            {
34.                sb_chrome.Append("\n");
35.            }
36.        }
37.
38.        Console.Write(sb_chrome.ToString());
39.        Console.WriteLine("Fitness: " + idv_toshow.fitness + "\n");
40.        return sb_chrome;
41.    }

```

6. Source Code Penyimpanan Data Dosen

```
1. private void btnSubmit_Click(object sender, EventArgs e)
2.     {
3.         string nik, nama, no_telp, alamat;
4.         nik = txtNIK.Text;
5.         nama = txtName.Text;
6.         no_telp = txtPhoneNumber.Text;
7.         alamat = txtAddress.Text;
8.
9.         if (string.IsNullOrEmpty(nik) || string.IsNullOrEmpty(nama) || string.IsNullOrEmpty
10. (no_telp) || string.IsNullOrEmpty(alamat))
11.         {
12.             MessageBox.Show("Tidak dapat menyimpan data dosen.\nSilakan mengisi semua field.");
13.         }
14.         else
15.         {
16.             string query = "insert into dosen (nik,nama,no_telp,alamat) values('" + nik + "
17. ', '" + nama + "', '" + no_telp + "', '" + alamat + "')";
18.             sqlite.openConnection();
19.             sqlite.dataManipulator(query);
20.             sqlite.closeConnection();
21.             loadFLP();
22.             clearField();
23.         }
24.     }
```

7. Source Code Penyimpanan Data Mahasiswa

```
1. private void btnSubmit_Click(object sender, EventArgs e)
2.     {
3.         try
4.         {
5.             string nim, nama, judul, no_telp, alamat;
6.             nim = txtNIM.Text;
7.             nama = txtName.Text;
8.             judul = txtThesisTitle.Text;
9.             no_telp = txtPhoneNumber.Text;
10.            alamat = txtAddress.Text;
11.
12.            if (string.IsNullOrEmpty(nim) || string.IsNullOrEmpty(nama) || string.IsNullOrEmpty
13. (judul) || string.IsNullOrEmpty(no_telp) || string.IsNullOrEmpty(alamat))
14.            {
15.                MessageBox.Show("Tidak dapat menyimpan data mahasiswa.\nSilakan mengisi semua
16. field.");
17.            }
18.            else
19.            {
20.                string query = "insert into mhs values('" + nim + "', '" + nama + "', '" + judul
21. + "', '" + no_telp + "', '" + alamat + "')";
22.                ConnectionClass connect = new ConnectionClass();
23.                connect.openConnection();
24.                connect.dataManipulator(query);
25.                connect.closeConnection();
26.                clearField();
27.                loadFLP();
28.            }
29.        }
30.        catch (Exception err)
31.        {
32.            MessageBox.Show(err.ToString());
33.        }
34.    }
```

8. Source Code Penyimpanan Data Ruang

```
1. private void btnSubmit_Click(object sender, EventArgs e)
2.     {
3.         try
4.         {
5.             string nama;
6.             nama = txtRoom.Text;
7.
8.             if (string.IsNullOrEmpty(nama))
9.             {
10.                MessageBox.Show("Tidak dapat menyimpan data ruangan.\nSilakan mengisi semua field.");
11.            }
12.            else
13.            {
14.                string query = "insert into ruang values('" + nama + "')";
15.                ConnectionClass connect = new ConnectionClass();
16.                connect.openConnection();
17.                connect.dataManipulator(query);
18.                connect.closeConnection();
19.                loadFLP();
20.            }
21.        }
22.        catch (Exception err)
23.        {
24.            MessageBox.Show(err.ToString());
25.        }
26.    }
```

9. Source Code Generate Penjadwalan

```
1. private void btnGenerate_Click(object sender, EventArgs e)
2.     {
3.         if (btnGenerate.Text.Equals("Buat Jadwal"))
4.         {
5.             setButtonReport(true);
6.             setButtonReset(true);
7.             Bitmap bmp = new Bitmap(this.ClientRectangle.Width, this.ClientRectangle.Height)
8.             ;
9.             using (Graphics G = Graphics.FromImage(bmp))
10.            {
11.                G.CompositingMode = System.Drawing.Drawing2D.CompositingMode.SourceOver;
12.                G.CopyFromScreen(this.PointToScreen(new Point(0, 0)), new Point(0, 0), this.ClientRectangle.Size
13.                );
14.                double percent = 0.60;
15.                Color darken = Color.FromArgb((int)(255 * percent), Color.Black);
16.                using (Brush brsh = new SolidBrush(darken))
17.                {
18.                    {
19.                        G.FillRectangle(brsh, this.ClientRectangle);
20.                    }
21.                }
22.                // put the darkened screenshot into a Panel and bring it to the front:
23.                using (Panel p = new Panel())
24.                {
25.                    p.Location = new Point(0, 0);
26.                    p.Size = this.ClientRectangle.Size;
27.                    p.BackgroundImage = bmp;
28.                    this.Controls.Add(p);
29.                    p.BringToFront();
30.                    // display your dialog somehow:
31.                    View.LoadingForm loadingForm = new View.LoadingForm(this);
32.                    loadingForm.StartPosition = FormStartPosition.CenterParent;
33.                    btnGenerate.Text = "Simpan Jadwal";
34.                    loadingForm.ShowDialog();
35.                }
36.            }
37.        }
38.        else
39.        {
40.            Bitmap bmp = new Bitmap(this.ClientRectangle.Width, this.ClientRectangle.Height)
41.            ;
42.            using (Graphics G = Graphics.FromImage(bmp))
43.            {
44.                G.CompositingMode = System.Drawing.Drawing2D.CompositingMode.SourceOver;
45.                G.CopyFromScreen(this.PointToScreen(new Point(0, 0)), new Point(0, 0), this.
46.                ClientRectangle.Size);
47.                double percent = 0.60;
48.                Color darken = Color.FromArgb((int)(255 * percent), Color.Black);
49.                using (Brush brsh = new SolidBrush(darken))
50.                {
51.                    {
52.                        G.FillRectangle(brsh, this.ClientRectangle);
53.                    }
54.                }
55.                // put the darkened screenshot into a Panel and bring it to the front:
56.                using (Panel p = new Panel())
57.                {
58.                    p.Location = new Point(0, 0);
59.                    p.Size = this.ClientRectangle.Size;
60.                    p.BackgroundImage = bmp;
61.                    this.Controls.Add(p);
62.                    p.BringToFront();
63.                    // display your dialog somehow:
64.                    View.SaveScheduleForm saveForm = new View.SaveScheduleForm(this);
65.                    saveForm.StartPosition = FormStartPosition.CenterParent;
66.                    saveForm.ShowDialog();
67.                }
68.            }
69.        }
70.    }
```

MANUAL PENGGUNAAN PROGRAM

Aplikasi ALGEN Scheduling memiliki kurang lebih 13 tampilan dari masing-masing prosesnya. Tampilan tersebut terdiri dari tampilan halaman input dan output (cetak).

Urutan tampilan program saat pertama kali di running adalah sebagai berikut:

1. Tampilan halaman utama

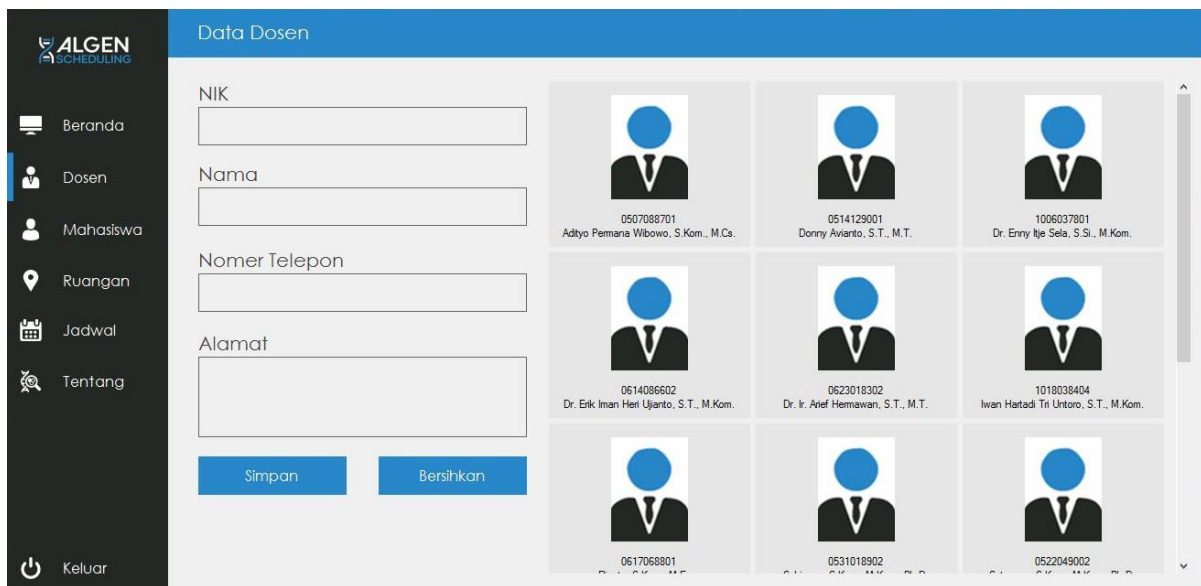
Halaman utama aplikasi ALGEN Scheduling terdiri dari dua bagian, yaitu bagian utama dan bagian menu. Bagian menu berada disebelah kiri yang berisi menu beranda, menu dosen, menu mahasiswa, menu ruangan, menu jadwal dan menu tentang. Bagian utama terdapat label nama program pada bagian tengah (bagian utama). Tampilan aplikasi yang muncul saat pertama kali di *running* seperti terlihat pada Gambar 1.



Gambar 1. Tampilan Halaman Utama

2. Tampilan Halaman data Dosen dan Input Dosen

Tampilan halaman data dosen dan input dosen dibuat menjadi satu. Dimaksudkan untuk memudahkan dan membuat lebih efisien dalam tampilan. Untuk menginputkan data dosen dengan cara mengisikan NIK, Nama, Nomor Telepon, dan Alamat pada komponen input *text* kemudian klik *button* simpan. Data yang sudah tersimpan akan tampil pada bagian sebelah kanan dari inputan data dosen. Tampilan halaman data dosen dan input dosen terlihat pada Gambar 2.



Gambar 2 Halaman Data Dosen dan Input Data Dosen

3. Tampilan Detail Data Dosen

Tampilan detail data dosen di buat untuk menampilkan data dosen sesuai dengan kriteria yang diinginkan. Tampilan ini juga bisa digunakan untuk merubah data dosen. Untuk menampilkan tampilan ini, caranya dengan klik data dosen yang diinginkan. Tampilan detail data dosen seperti terlihat pada Gambar 3.



Gambar 3 Tampilan Detail Data Dosen

4. Tampilan Data Mahasiswa

Sama halnya dengan halaman data dosen, halaman data mahasiswa juga terdiri dari dua bagian yaitu bagian inputan dan bagian daftar data mahasiswa. Pada tampilan inputan terdapat 2 button yang digunakan untuk menyimpan data dan membersihkan data.

Sedangkan tampilan data mahasiswa berisi data mahasiswa yang sudah melakukan pendaftaran ujian pendadaran Proyek Tugas Akhir. Tampilan data mahasiswa terlihat pada Gambar 4.

NIM	Nama	Judul Tugas Akhir
5130411177	Siska Fanda	Analisis dan Pengujian di Jaringan 3G dan 4G dalam Layanan Quality of Services (QoS)
5130411189	Handians	Deteksi Muka Depan Manusia dari Sebuah Citra Berwarna dengan Template Matching
5130411179	Resky Nuseptadi	Implementasi SMS gateway dengan menggunakan bahasa alami dalam sistem informasi perdagangan (Studi Kasus Toko Amarna)
3115111438	Ahmad Mutadho	Studi dan Implementasi Konsep Business to Costumer dengan Teknologi M-Commerce berbasis WAP
5130411109	Ichsan Ananto	Aplikasi Logika Fuzzy untuk Prediksi Penyakit dengan Metode Criteria Decision Making
5130411368	Yusri Iha	Sistem Informasi Eksekutif untuk Perencanaan Tata Letak Anak Cabang Perusahaan (Studi Kasus Bank PD. BPR Bapas GS Magelang)

Gambar 4. Tampilan Data Mahasiswa

5. Tampilan Detail Data Mahasiswa

Untuk mengetahui detail data mahasiswa, ada tampilan yang dibuat khusus untuk menampilkan detail data tersebut. Detail data yang ditampilkan terdiri dari NIM, Nama Mahasiswa, Judul Tugas Akhir, nomor telepon, alamat mahasiswa, serta foto dari mahasiswa tersebut. Untuk menampilkan halaman detail data mahasiswa, dengan cara klik data mahasiswa yang diinginkan pada tampilan data mahasiswa Tampilan detail data mahasiswa seperti terlihat pada Gambar 5.

The screenshot shows a form titled "Detail Mahasiswa" with a blue header. On the left is a placeholder for a student profile picture. The form contains the following fields:

- NIM:** 5130411177
- Nama:** Siska Farida
- Judul Tugas Akhir:** Analisis dan Pengujian di Jaringan 3G dan .
- Nomer Telepon:** 081221445786
- Alamat:** Yogyakarta

At the bottom of the form are three buttons: "Konfirmasi" (blue), "Hapus" (black), and "Batal" (black).

Gambar 5. Tampilan Detail Data Mahasiswa

6. Tampilan Data Ruangan

Data ruangan yang digunakan untuk tempat ujian pendadaran, disimpan melalui tampilan data ruangan. Proses penyimpanan yang terjadi dengan cara mengisi edit text nama ruangan, kemudian klik button simpan. Button bersihkan digunakan untuk menghapus dan membersihkan edit text pada nama ruangan. Data yang tersimpan akan tampil pada bagian sebelah kanannya. Tampilan data ruangan terlihat pada Gambar 6.

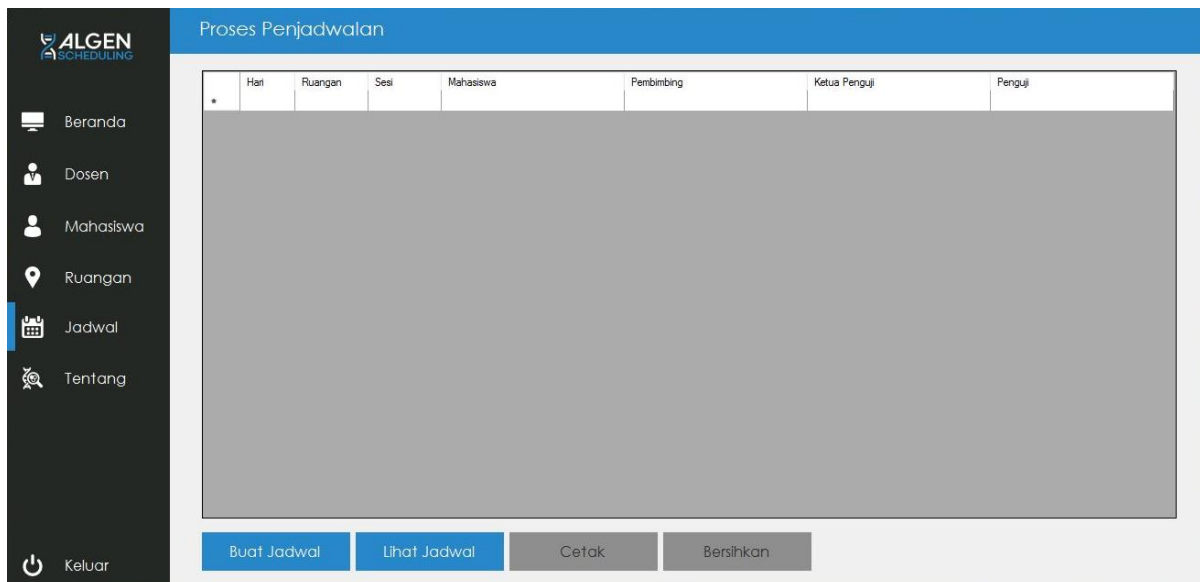
The screenshot shows the "Data Ruangan" form with a blue header. On the left is a dark sidebar menu with the "ALGEN SCHEDULING" logo and the following items: Beranda, Dosen, Mahasiswa, Ruangan (highlighted), Jadwal, Tentang, and Keluar. The main form area contains:

- Nama Ruangan:** An empty text input field.
- Buttons:** "Simpan" (blue) and "Bersihkan" (blue).
- Room Cards:** Two cards labeled "Ruangan 1" and "Ruangan 2", each featuring a location pin icon.

Gambar 6. Tampilan Data Ruangan

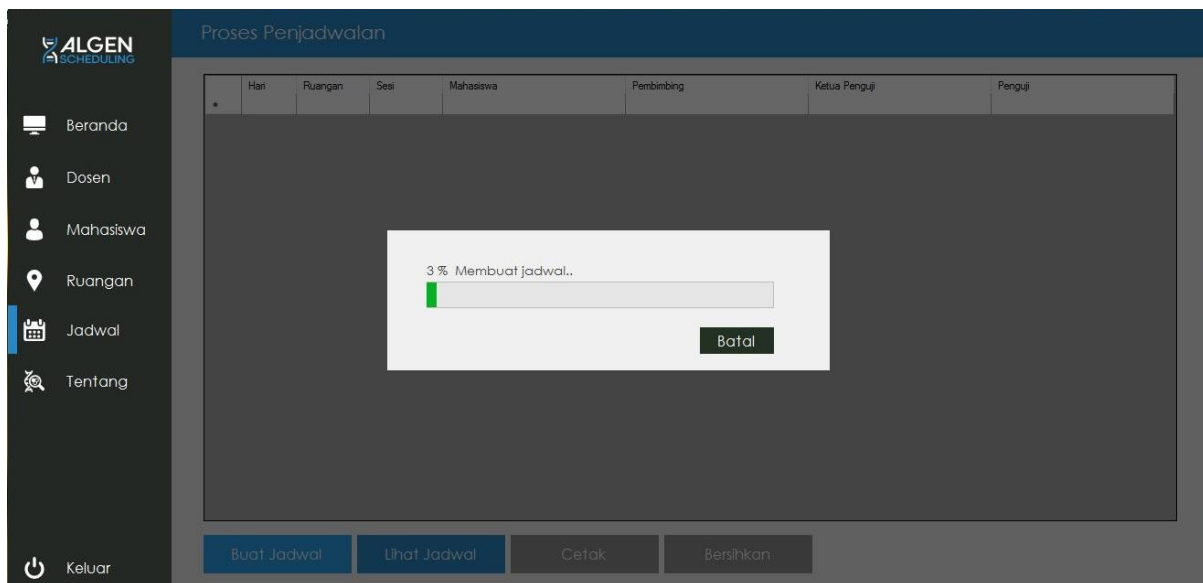
7. Tampilan Proses Penjadwalan

Setelah semua data diinputkan dan disimpan dalam database, selanjutnya akan diolah untuk melakukan proses penjadwalan. Tampilan proses penjadwalan muncul jika di klik menu jadwal yang terletak pada bagian sebelah kiri tampilan. Tampilan proses penjadwalan terdiri dari beberapa button dan tampilan tabel yang berisi data penjadwalan. Untuk membuat/men-generate jadwal dengan melakukan klik button Buat Jadwal. Button lihat jadwal digunakan untuk melihat jadwal yang sudah degenerate. Button cetak digunakan untuk mencetak jadwal yang sudah final dari hasil Buat Jadwal. Button bersihkan digunakan untuk menghapus dan membersihkan tampilan jadwal. Tampilan proses penjadwalan terlihat pada Gambar 7.



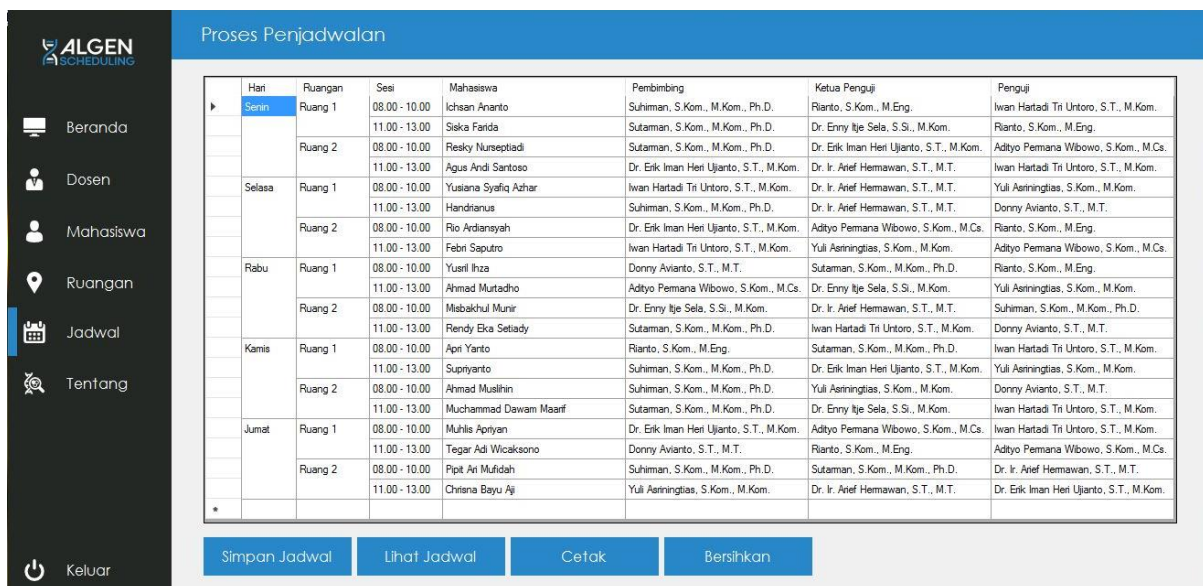
Gambar 7. Tampilan Proses Penjadwalan

Jika button Buat Jadwal di klik, maka akan terjadi proses generate jadwal menggunakan Algoritma Genetika. Proses *generate* jadwal yang terjadi seperti terlihat pada Gambar 8.



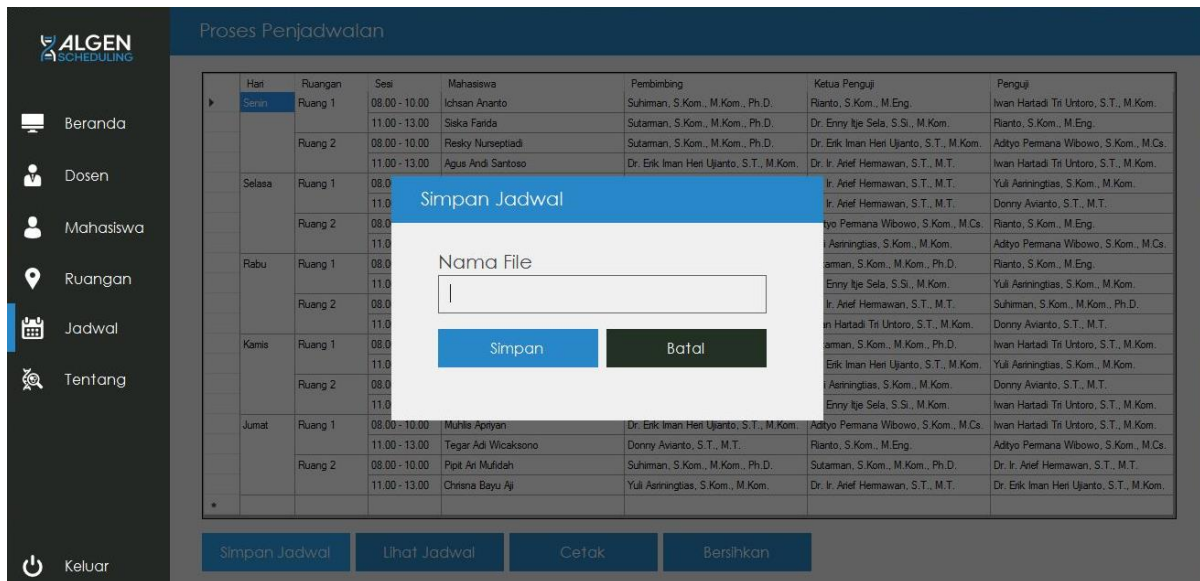
Gambar 8. Proses Generate Membuat Jadwal

Setelah proses *generate* membuat jadwal selesai, selanjutnya akan tampil jadwal keseluruhan yang siap untuk disimpan dan dicetak. Tampilan hasil jadwal keseluruhan terlihat pada Gambar 9.



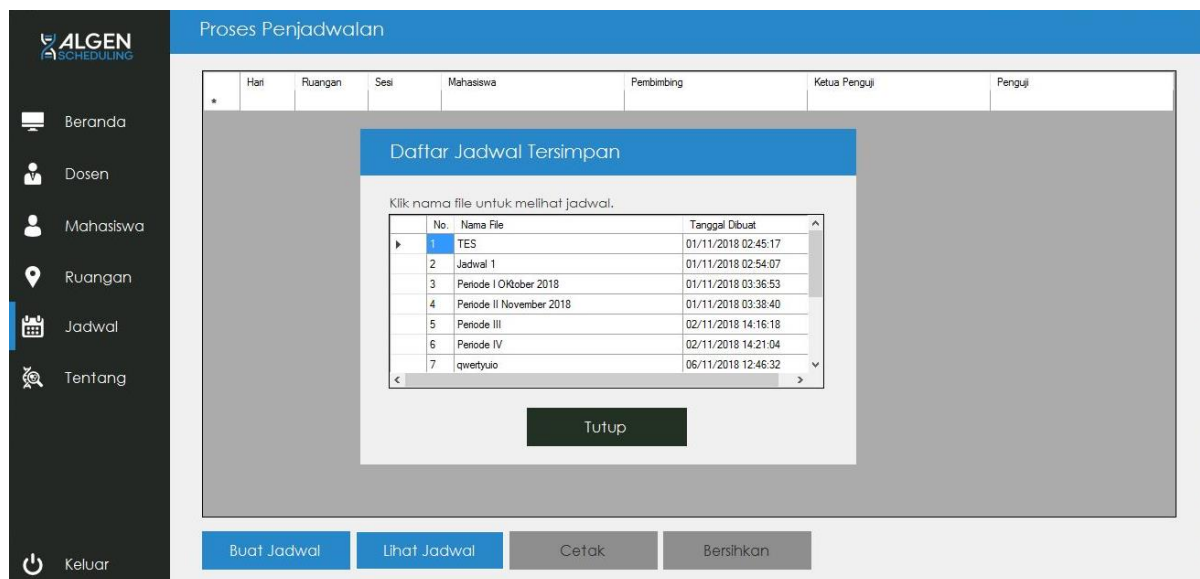
Gambar 9. Tampilan Jadwal Hasil *Generate* Algoritma Genetika

Untuk menyimpan jadwal, dengan cara klik button simpan jadwal. Selanjutnya akan muncul notif untuk mengisi nama file untuk kriteria jadwal. Notifikasi nama file jadwal terlihat pada Gambar 10.



Gambar 10. Tampilan Notifikasi Nama File Simpan Jadwal

File jadwal yang tersimpan bisa ditampilkan/dimunculkan sewaktu-waktu. Untuk memunculkan daftar nama file jadwal yang tersimpan dengan cara klik button Lihat Jadwal. Daftar nama file jadwal terlihat pada Gambar 11.



Gambar 11. Tampilan Daftar Nama Jadwal Ujian

8. Tampilan Tentang

Tampilan tentang adalah tampilan terakhir yang ada pada aplikasi ALGEN Scheduling. Untuk menampilkan halaman tentang dengan cara klik menu Tentang yang ada di bagian sebelah kiri. Tampilan halaman Tentang terlihat pada Gambar 12.



Gambar 12. Tampilan Halaman Tentang

9. Tampilan Cetak Jadwal

Salah satu tampilan output/luaran pada aplikasi ALGEN Scheduling adalah tampilan cetak jadwal. Tampilan cetak jadwal muncul dari klik button cetak jadwal pada tampilan Proses Penjadwalan (Gambar 9). Tampilan cetak jadwal terlihat pada Gambar 13.

Cetak Jadwal

of 1 | Whole Page | Find | Next

Jadwal Ujian Pendadaran Tugas Akhir						
Hari	Ruang	Sesi	Mahasiswa	Pembimbing	Ketua Penguji	Penguji
Senin	R.Lengkung 1	08.00 - 10.00	Apri Yanto	Dr. Erik Iman Heri Ujianto, S.T., M.Kom.	Dr. Enny Ijte Sela, S.Si., M.Kom.	Rianto, S.Kom., M.Eng.
Senin	R.Lengkung 1	11.00 - 13.00	Resty Nurseptadi	Yuli Asningtas, S.Kom., M.Kom.	Iwan Hartadi Tri Untoro, S.T., M.Kom.	Dr. Enny Ijte Sela, S.Si., M.Kom.
Senin	R.Lengkung 2	08.00 - 10.00	Fabri Saputro	Dr. Ir. Anief Hermawan, S.T., M.T.	Yuli Asningtas, S.Kom., M.Kom.	Donny Arianbo, S.T., M.T.
Senin	R.Lengkung 2	11.00 - 13.00	Yusiana Syarif Azhar	Sutarnan, S.Kom., M.Kom., Ph.D.	Donny Arianbo, S.T., M.T.	Dr. Erik Iman Heri Ujianto, S.T., M.Kom.
Selasa	R.Lengkung 1	08.00 - 10.00	Ro Ardanejah	Iwan Hartadi Tri Untoro, S.T., M.Kom.	Sutarnan, S.Kom., M.Kom., Ph.D.	Suhman, S.Kom., M.Kom., Ph.D.
Selasa	R.Lengkung 1	11.00 - 13.00	Tegar Adi Wicaksono	Yuli Asningtas, S.Kom., M.Kom.	Dr. Enny Ijte Sela, S.Si., M.Kom.	Rianto, S.Kom., M.Eng.
Selasa	R.Lengkung 2	08.00 - 10.00	Rendy Eka Setady	Dr. Ir. Anief Hermawan, S.T., M.T.	Aditya Permiana Wibowo, S.Kom., M.Cs.	Rianto, S.Kom., M.Eng.
Selasa	R.Lengkung 2	11.00 - 13.00	Supriyanto	Dr. Erik Iman Heri Ujianto, S.T., M.Kom.	Aditya Permiana Wibowo, S.Kom., M.Cs.	Iwan Hartadi Tri Untoro, S.T., M.Kom.
Rabu	R.Lengkung 1	08.00 - 10.00	Agus Andi Santoso	Sutarnan, S.Kom., M.Kom., Ph.D.	Dr. Erik Iman Heri Ujianto, S.T., M.Kom.	Yuli Asningtas, S.Kom., M.Kom.
Rabu	R.Lengkung 1	11.00 - 13.00	Handrianus	Donny Arianbo, S.T., M.T.	Rianto, S.Kom., M.Eng.	Suhman, S.Kom., M.Kom., Ph.D.
Rabu	R.Lengkung 2	08.00 - 10.00	Yusli Inza	Dr. Ir. Anief Hermawan, S.T., M.T.	Rianto, S.Kom., M.Eng.	Suhman, S.Kom., M.Kom., Ph.D.
Rabu	R.Lengkung 2	11.00 - 13.00	Pipit Ari Mukdiah	Yuli Asningtas, S.Kom., M.Kom.	Aditya Permiana Wibowo, S.Kom., M.Cs.	Dr. Ir. Anief Hermawan, S.T., M.T.
Kamis	R.Lengkung 1	08.00 - 10.00	Misbahul Munir	Iwan Hartadi Tri Untoro, S.T., M.Kom.	Yuli Asningtas, S.Kom., M.Kom.	Donny Arianbo, S.T., M.T.
Kamis	R.Lengkung 1	11.00 - 13.00	Ahmad Murtadho	Rianto, S.Kom., M.Eng.	Sutarnan, S.Kom., M.Kom., Ph.D.	Dr. Erik Iman Heri Ujianto, S.T., M.Kom.
Kamis	R.Lengkung 2	08.00 - 10.00	Multia Aptijan	Sutarnan, S.Kom., M.Kom., Ph.D.	Dr. Erik Iman Heri Ujianto, S.T., M.Kom.	Suhman, S.Kom., M.Kom., Ph.D.
Kamis	R.Lengkung 2	11.00 - 13.00	Christina Bayu Aji	Dr. Ir. Anief Hermawan, S.T., M.T.	Dr. Enny Ijte Sela, S.Si., M.Kom.	Yuli Asningtas, S.Kom., M.Kom.
Jumat	R.Lengkung 1	08.00 - 10.00	Muhammad Dawam Makarfi	Dr. Ir. Anief Hermawan, S.T., M.T.	Donny Arianbo, S.T., M.T.	Aditya Permiana Wibowo, S.Kom., M.Cs.
Jumat	R.Lengkung 1	11.00 - 13.00	Ihsan Ananto	Aditya Permiana Wibowo, S.Kom., M.Cs.	Rianto, S.Kom., M.Eng.	Donny Arianbo, S.T., M.T.
Jumat	R.Lengkung 2	08.00 - 10.00	Ahmad Mugkhih	Suhman, S.Kom., M.Kom., Ph.D.	Iwan Hartadi Tri Untoro, S.T., M.Kom.	Yuli Asningtas, S.Kom., M.Kom.
Jumat	R.Lengkung 2	11.00 - 13.00	Marsodik	Iwan Hartadi Tri Untoro, S.T., M.Kom.	Dr. Erik Iman Heri Ujianto, S.T., M.Kom.	Dr. Ir. Anief Hermawan, S.T., M.T.

Yogyakarta, 19 Juli 2016

Dr. Enny Ijte Sela, S.Si., M.Kom.
NIK: 111118089

Gambar 13. Tampilan Cetak Jadwal Ujian Pendadaran