

**PENJADWALAN KULIAH PROGRAM STUDI SISTEM
INFORMASI UNIVERSITAS ARS MENGGUNAKAN
ALGORITMA GENETIKA
(Studi Kasus: Universitas ARS)**

TESIS

Disusun sebagai salah satu syarat untuk
memperoleh gelar Magister Komputer
dari Sekolah Tinggi Manajemen Informatika dan Komputer LIKMI

Oleh:

AHSANI TAKWIM

NPM: 2019210045



**PROGRAM STUDI PASCASARJANA
MAGISTER SISTEM INFORMASI
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER LIKMI
BANDUNG
2021**

**PENJADWALAN KULIAH PROGRAM STUDI SISTEM
INFORMASI UNIVERSITAS ARS MENGGUNAKAN
ALGORITMA GENETIKA
(Studi Kasus: Universitas ARS)**

Oleh:

AHSANI TAKWIM

NPM: 2019210045

Bandung, 26 Januari 2021

Mengetahui,

Dr. Hery Heryanto, S.Kom., M.Kom

Pembimbing

**PROGRAM STUDI PASCASARJANA
MAGISTER SISTEM INFORMASI
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER LIKMI
BANDUNG
2021**

**Dipersembahkan untuk keluarga tercinta terkhusus kepada Ibu Siti, Pak Uyun
Pak Aleh, Teh Yuni, A Fyan, A Jaka, Teh Aas, Ajmalina, Rakaditya dan Abyan.**

ABSTRAK

PENJADWALAN MATAKULIAH MENGGUNAKAN ALGORITMA GENETIKA PROGRAM STUDI SISTEM INFORMASI UNIVERSITAS ARS

Oleh:
AHSANI TAKWIM
NPM: 2019210045

Proses pembuatan jadwal perkuliahan di Universitas ARS saat ini pengerjaan pembuatan jadwal masih secara manual dengan menggunakan Microsoft Office Excel, sehingga memerlukan waktu pengerjaan yang cukup lama karena terdapat kendala dalam pengalokasian mata kuliah, dosen, mahasiswa yang mengambil mata kuliah beserta ruangnya agar tidak terjadi kesalahan penjadwalan yang ada. Saat ini dikelola oleh staf dengan acuan kesediaan mengajar seluruh dosen yang mengajar dengan berbagai ketentuan dari masing-masing Kepala Program Studi. Teknis penjadwalannya adalah dengan menginputkan dosen, matakuliah, ruangan, kelas dan waktu dalam aplikasi Microsoft Excel, setelahnya disesuaikan dengan ketersediaan mengajar dosen, dengan demikian banyak jadwal yang berbenturan pada awal penginputan sehingga diperlukan penyesuaian-penyesuaian yang memakan cukup banyak waktu.

Universitas Adhirajasa Reswara Sanjaya merupakan sebuah perguruan tinggi swasta yang terdapat di Kota Bandung. Universitas ARS pertama kali resmi didirikan dengan nama Universitas ARS Internasional dengan pada tanggal 6 Oktober 2000. Menjadi perguruan tinggi yang unggul berbasis teknologi informasi yang mampu menghasilkan tenaga profesional yang berjiwa kewirausahaan dalam menghadapi persaingan global merupakan visi dari Universitas ARS. Melaksanakan pendidikan tinggi berbasis ilmu pengetahuan teknologi dan desain Menyelenggarakan penelitian dan pengabdian masyarakat yang memberikan kontribusi positif terhadap peningkatan kualitas akademik dan masyarakat. Melaksanakan tata kelola universitas yang transparan, kredibel, bertanggung jawab dan adil.

Pada penelitian ini akan dibuat sebuah aplikasi penjadwalan mata kuliah menggunakan algoritma genetika dengan metode penelitian *Agile Iteration*. Algoritma Genetika merupakan teknik optimasi dalam ilmu komputer berfungsi untuk memberikan solusi terbaik dengan menggunakan teknik yang terinspirasi oleh Biologi Evolusioner yang mempertimbangkan kondisi dan tahapan seperti warisan, mutasi (*mutation*), seleksi alam (*selection*) dan rekombinasi (*crossover*). Pendekatan yang diambil oleh algoritma ini adalah dengan menggabungkan secara acak berbagai pilihan solusi terbaik. *Agile Development Methods* adalah sekelompok metodologi pengembangan perangkat lunak yang didasarkan pada pengembangan berulang, dimana persyaratan dan solusinya berkembang melalui kolaborasi antara tim lintas fungsi yang mengatur sendiri. Jadi, bisa disimpulkan jika *agile* merupakan model pengembangan jangka pendek yang memerlukan adaptasi cepat dan pengembangan terhadap perubahan dalam bentuk apapun. Metode *Agile Iteration* mempunyai tahapan *requirement*, *planing*, *desain*, *implementation*, *realease* dan *control/maintenance* tahapan tersebut memberikan gaya tersendiri dalam metode pengembangan aplikasi yang dinamis.

Hasil penelitian ini menunjukkan bahwa dengan menggunakan Algoritma Genetika dan metode *Agile Iteration* menghasilkan sebuah aplikasi penjadwalan yang bisa memberikan hasil penjadwalan bahwa dengan *constraint* yang telah ditentukan aplikasi penjadwalan mata kuliah dengan algoritma genetika dapat berjalan secara maksimal, hal ini ditunjukkan dengan tidak adanya jadwal kuliah yang melewati waktu adzan yang menjadi *constraint* dipenelitian ini.

Kata Kunci: Penjadwalan, Algoritma Genetika, *Agile Iteration*, UML Diagram

ABSTRACT

COURSE SCHEDULING USING THE GENETIC ALGORITHM DEPARTMENT INFORMATION SYSTEM OF ARS UNIVERSITY

By:
AHSANI TAKWIM
NPM: 2019210045

The process of making lecture schedules at ARS University is currently still working on making schedules manually using Microsoft Office Excel, so it takes a long time to work because there are problems in allocating courses, lecturers, students who take courses and their rooms so that scheduling errors do not occur which exists. Currently managed by staff with reference to the willingness to teach all lecturers who teach with various provisions from each Head of the Study Program. The technical scheduling is by inputting lecturers, courses, rooms, classes and time in the Microsoft Excel application, after which it is adjusted to the availability of teaching lecturers, so that many schedules clash at the beginning of input so that adjustments are required that take quite a lot of time.

Adhirajasa Reswara Sanjaya University is a private university located in Bandung. ARS University was first officially established under the name ARS International University on October 6, 2000. To become a superior information technology-based university capable of producing entrepreneurial-spirited professionals in the face of global competition is the vision of ARS University. Carrying out higher education based on science, technology and design. Carrying out research and community service that makes a positive contribution to improving the quality of academics and society. Implementing transparent, credible, responsible and fair university governance.

Course scheduling application in this study will be made using a genetic algorithm with the Agile Iteration research method. Genetic Algorithms are optimization techniques in computer science that function to provide the best solutions by using techniques inspired by Evolutionary Biology which consider conditions and stages such as inheritance, mutation, natural selection and recombination (crossover). The approach taken by this algorithm is to randomly combine various best solutions. Agile Development Methods are a group of software development methodologies based on iterative development, where the requirements and solutions develop through collaboration between self-regulating cross-functional teams. So, it can be concluded that agile is a short-term development model that requires rapid adaptation and development to change of any kind. The Agile Iteration method has stages of requirements, planing, design, implementation, realease and control/maintenance, these stages provide its own style in dynamic application development methods.

The results of this study indicate that using the Genetic Algorithm and the Agile Iteration method produces a scheduling application that can provide scheduling results that with predetermined constraints the course scheduling application with a genetic algorithm can run optimally, this is indicated by the absence of a class schedule that passes through the time to call to prayer is the constraint in this study.

Key Words: Scheduling, Genetic Algorithm, Agile Iteration, UML Diagram

KATA PENGANTAR

Alhamdulillah wasyukrulillah atas berkat rahmat Allah SWT beserta dukungan dan berkat doa kedua orang tua yang kuat untuk dapat menyelesaikan penelitian dengan judul “Penjadwalan Mata Kuliah Program Studi Informasi Universitas ARS”. Pada kesempatan ini penulis mengucapkan terimakasih yang sebesar-besarnya kepada semua pihak yang turut serta membantu penyelesaian laporan tesis ini, yaitu kepada:

1. Bapak Dr. Herry Heryanto, S.Kom., M.Kom selaku pembimbing yang telah memberikan bimbingan dengan penuh perhatian, kesabaran, dan ketulusan sehingga dapat menyelesaikan tesis ini.
2. Seluruh Civitas akademika Universitas ARS yang sudah membantu terselesaikannya penelitian ini.
3. Seluruh Civitas akademika STMIK LIKMI yang membantu kelancaran proses perkuliahan.
4. Seluruh staff pengajar program pascasarjana (S2) STMIK LIKMI yang telah memberikan ilmu pengetahuan.
5. Seluruh keluarga yang telah memberikan dukungan dan motivasi serta doa sehingga dapat terselesaikannya penelitian ini.
6. Rekan rekan seperjuangan terkhusus Kak Yunita, Pak Dekky, Hena, Fadila, Clarissa, Yasti, Viqi, Rifki, Rissa, yang selalu membantu dan banyak rekan yang lainnya yang tidak bisa saya sebutkan satu persatu.

Dalam penyusunan tesis ini, penulis menyadari masih banyaknya terdapat kelemahan dan kekurangan yang perlu dilengkapi. Oleh karena itu, dengan rendah hati penulis mengharapkan masukan, dan saran untuk perbaikan penelitian ini. Dan semoga penelitian ini dapat bermanfaat bagi pihak yang membutuhkannya.

Bandung, 26 Januari 2021

Penulis

DAFTAR ISI

ABSTRAK.....	iii
ABSTRACT	iv
KATA PENGANTAR	v
DAFTAR ISI.....	vi
DAFTAR GAMBAR	viii
DAFTAR TABEL	xii
DAFTAR RUMUS.....	xiii
DAFTAR SIMBOL	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan Penelitian.....	4
1.4 Batasan Masalah.....	4
1.5 Sistematika Penulisan	4
BAB II LANDASAN TEORI.....	6
2.1 Algoritma Genetika.....	6
2.1.1 Struktur Umum Algoritma Genetika	7
2.1.2 Komponen Algoritma Genetika.....	9
2.2 Penjadwalan Kuliah	12
2.3 Metode Pengembangan Sistem	14
2.4 <i>Agile Software Development</i>	15
2.5 <i>Unified Modeling Language (UML)</i>	18
2.5.1 <i>Activity Diagram</i>	21
2.5.2 <i>Use Case Diagram</i>	22
2.5.3 <i>Class diagram</i>	23
2.6 Penelitian Terkait.....	23
BAB III OBJEK DAN METODOLOGI PENELITIAN	26

3.1 Profil Universitas ARS	26
3.1.1 Gambaran umum Organisasi.....	26
3.1.2 Visi, Misi, Tujuan.....	27
3.1.3 Gambaran Fungsional Organisasi	28
3.1.4 Proses Penjadwalan Kuliah Universitas ARS.....	29
3.2 Metodologi Penelitian	31
BAB IV PEMBAHASAN.....	40
4.1 Mempersiapkan kebutuhan/ <i>Requirement</i>	40
4.2 Perencanaan/ <i>Planning</i>	41
4.3 Desain	65
4.4 Implementasi	83
BAB V KESIMPULAN DAN SARAN	112
5.1 Kesimpulan.....	112
5.2 Saran	112
DAFTAR PUSTAKA	101
Lampiran 1.....	103
Lampiran 2.....	104
Lampiran 3.....	105

DAFTAR GAMBAR

Gambar 1. 1. Jumlah Mahasiswa Program Studi Sistem Informasi	3
Gambar 2. 1 Gambar Basic Algoritma Genetika	6
Gambar 2. 2. Faktor-faktor yang mempengaruhi penjadwalan matakuliah	13
Gambar 2. 3. <i>Agile Lifecycle</i>	17
Gambar 2. 4 UML.....	21
Gambar 3. 1. Gambaran Organisasi Universitas ARS	28
Gambar 3. 2. Proses Penjadwalan mata kuliah.....	30
Gambar 3. 3. Gambaran Metodologi Penelitian.....	31
Gambar 3. 4. Faktor yang mempengaruhi penjadwalan matakuliah	33
Gambar 3. 5. Flowchart Operasi Algoritma Genetika	34
Gambar 4. 1. Use Case aplikasi penjadwalan kuliah	42
Gambar 4. 2. Activity Diagram <i>log in</i>	43
Gambar 4. 3. Activity Diagram input dosen	44
Gambar 4. 4. Activity Diagram edit dosen	45
Gambar 4. 5. Activity Diagram hapus dosen	46
Gambar 4. 6. Activity Diagram input mata kuliah.....	47
Gambar 4. 7. Activity Diagram edit mata kuliah.....	48
Gambar 4. 8. Activity Diagram hapus mata kuliah.....	49
Gambar 4. 9. Activity Diagram input pengampu	50
Gambar 4. 10. Activity Diagram edit pengampu	51
Gambar 4. 11. Activity Diagram hapus pengampu	52
Gambar 4. 12. Activity Diagram input ruang	53
Gambar 4. 13. Activity Diagram edit ruang	54
Gambar 4. 14. Activity Diagram hapus ruang	55
Gambar 4. 15. Activity Diagram input jam	56
Gambar 4. 16. Activity Diagram edit jam	57
Gambar 4. 17. Activity Diagram hapus jam	58

Gambar 4. 18. Activity Diagram input hari	59
Gambar 4. 19. Activity Diagram edit hari	60
Gambar 4. 20. Activity Diagram hapus hari	61
Gambar 4. 21. Activity Diagram <i>constraint</i>	62
Gambar 4. 22. Activity Diagram penjadwalan	63
Gambar 4. 23. Class diagram	64
Gambar 4. 24. Desain <i>log in</i>	65
Gambar 4. 25. Desain beranda	66
Gambar 4. 26. Desain menu dosen	66
Gambar 4. 27. Desain menu input dosen	67
Gambar 4. 28. Desain menu edit dosen	68
Gambar 4. 29. Desain menu hapus dosen	68
Gambar 4. 30. Desain menu mata kuliah	69
Gambar 4. 31. Desain input mata kuliah	70
Gambar 4. 32. Desain edit mata kuliah	70
Gambar 4. 33. Desain hapus mata kuliah	71
Gambar 4. 34. Desain menu pengampu	72
Gambar 4. 35. Desain input pengampu	72
Gambar 4. 36. Desain edit pengampu	73
Gambar 4. 37. Desain hapus pengampu	74
Gambar 4. 38. Desain menu ruang	74
Gambar 4. 39. Desain input ruang	75
Gambar 4. 40. Desain edit ruang	76
Gambar 4. 41. Desain hapus ruang	76
Gambar 4. 42. Desain menu jam	77
Gambar 4. 43. Desain input jam	78
Gambar 4. 44. Desain edit jam	78
Gambar 4. 45. Desain hapus jam	79
Gambar 4. 46. Desain menu hari	80

Gambar 4. 47. Desain input hari	80
Gambar 4. 48. Desain edit hari	81
Gambar 4. 49. Desain hapus hari	82
Gambar 4. 50. Desain menu constraint	82
Gambar 4. 51. Desain menu penjadwalan.....	83
Gambar 4. 52. Tampilan <i>log in</i>	84
Gambar 4. 53. Tampilan beranda	85
Gambar 4. 54. Tampilan menu dosen	85
Gambar 4. 55. Tampilan input dosen.....	86
Gambar 4. 56. Tampilan edit dosen.....	87
Gambar 4. 57. Tampilan hapus dosen.....	87
Gambar 4. 58. Tampilan menu matakuliah.....	88
Gambar 4. 59. Tampilan input matakuliah	89
Gambar 4. 60. Tampilan edit matakuliah	89
Gambar 4. 61. Tampilan hapus matakuliah	90
Gambar 4. 62. Tampilan menu pengampu	91
Gambar 4. 63. Tampilan input pengampu	91
Gambar 4. 64. Tampilan edit pengampu	92
Gambar 4. 65. Tampilan hapus pengampu	93
Gambar 4. 66. Tampilan menu ruang	93
Gambar 4. 67. Tampilan input ruang	94
Gambar 4. 68. Tampilan edit ruang	94
Gambar 4. 69. Tampilan hapus ruang	95
Gambar 4. 70. Tampilan menu jam	96
Gambar 4. 71. Tampilan input jam.....	96
Gambar 4. 72. Tampilan edit jam.....	97
Gambar 4. 73. Tampilan hapus jam.....	98
Gambar 4. 74. Desain menu hari.....	98
Gambar 4. 75. Desain input hari	99

Gambar 4. 76. Desain edit hari	99
Gambar 4. 77. Desain hapus hari	100
Gambar 4. 78. Desain menu <i>constraint</i>	101
Gambar 4. 79. Desain menu penjadwalan.....	101
Gambar 4. 80. Penjadwakan di ruang 401	106
Gambar 4. 81. Penjadwalan di ruang 402	107
Gambar 4. 82. Penjadwakan di ruang 403	108
Gambar 4. 83. Penjadwakan di ruang 404	109
Gambar 4. 84. Desain menu penjadwalan.....	110
Gambar 4. 85. Penjadwalan kuliah di laboratorium	111

DAFTAR TABEL

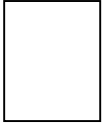
Tabel 2. 1 Penelitian Terkait	23
Tabel 3. 1 Perkuliahan	35
Tabel 3. 2 Ruang.....	35
Tabel 3. 3 Waktu	35
Tabel 3. 4 Populasi Awal.....	36
Tabel 3. 5 Pembobotan.....	37
Tabel 3. 6 Kromosom dalam <i>crossover</i>	38
Tabel 3. 7 Kromosom baru dalam <i>crossover</i>	38
Tabel 3. 8 Mutasi.....	39
Tabel 4. 1. Kebutuhan fungsional aplikasi	40
Tabel 4. 2. Kebutuhan non fungsional aplikasi	41
Tabel 4. 3.Tahapan <i>coding</i>	102

DAFTAR RUMUS

Rumus 1. 1. <i>Fitness</i>	10
Rumus 3. 1. <i>Fitness</i> Kromosom.....	37

DAFTAR SIMBOL

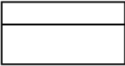
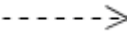
SIMBOL *USE CASE*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (<i>sinergi</i>).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

SIMBOL ACTIVITY DIAGRAM

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

SIMBOL CLASS DIAGRAM

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dizaman yang serba cepat saat ini, manusia dituntut untuk melakukan pekerjaan secepat dan setepat mungkin. Namun permasalahan selalu ada dalam setiap pekerjaan yang dilakukan manusia. Disamping itu, manusia juga mengembangkan teknologi yang kian hari semakin pesat perkembangannya. Hal ini dimaksudkan untuk membantu pekerjaan manusia. Semakin berkembangnya teknologi maka semakin banyak pula aplikasi yang bisa membantu manusia untuk mempermudah dan mempercepat pekerjaannya. Namun tidak semua manusia dapat mengaplikasikan hasil dari perkembangan ilmu pengetahuan yang sudah ada. Hal ini dibuktikan dengan masih banyaknya pekerjaan manusia yang tidak menggunakan aplikasi, meskipun aplikasi tersebut sudah ada dan berkembang. Salah satunya dalam mengerjakan sebuah penjadwalan. Dalam proses penyusunan jadwal perkuliahan terdapat beberapa hal yang rumit dan sering mengalami kesulitan dalam penyusunannya, Seperti pengalokasian mata kuliah, dosen dan ruangan yang akan digunakan untuk perkuliahan agar tidak terjadi bentrok dengan jadwal mata kuliah, dosen dan ruang yang lain dalam satu periode jadwal kuliah (Sayed, 2015).

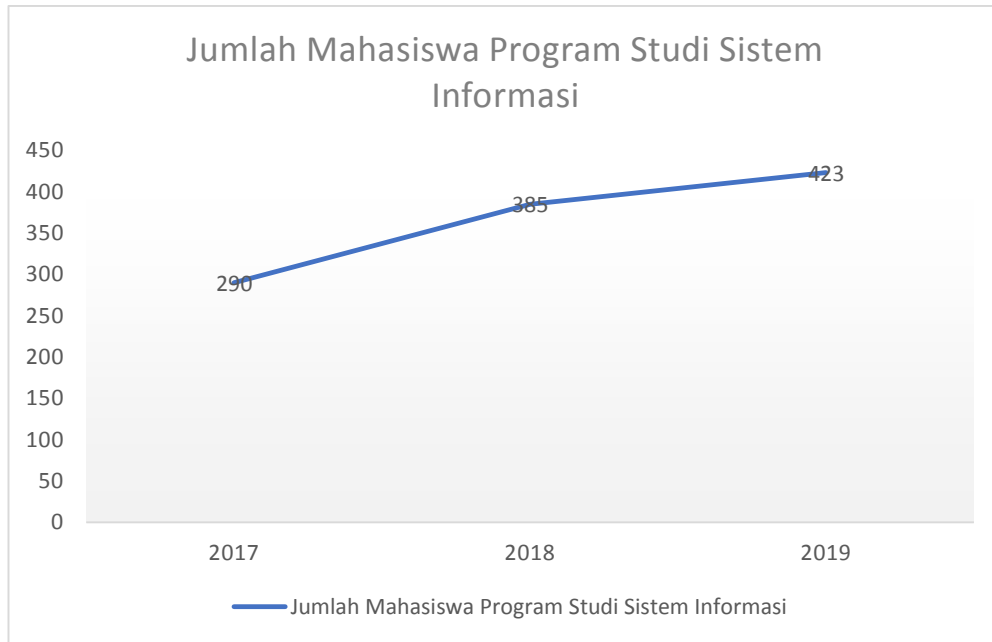
Menurut Pinedo (2012), penjadwalan dapat didefinisikan sebagai proses pengalokasian sumber daya untuk mengerjakan sekumpulan tugas dalam jangka waktu tertentu dengan 2 arti penting sebagai berikut.

1. Penjadwalan merupakan suatu fungsi pengambilan keputusan untuk membuat atau menentukan jadwal.
2. Penjadwalan merupakan suatu teori yang berisi sekumpulan prinsip dasar, model, teknik, dan kesimpulan logis dalam proses pengambilan keputusan yang memberikan pengertian dalam fungsi penjadwalan.

Penjadwalan mata kuliah dalam suatu kampus merupakan suatu tantangan pekerjaan yang cukup kompleks untuk diselesaikan. Kompleksitasnya bisa dipandang dari

sisi mahasiswa, dosen yang mengampu, mata kuliah yang diampu, waktu perkuliahan, waktu kesediaan dosen dan ruangan untuk menjalankan perkuliahan. Sebuah penjadwalan yang bagus ialah sebuah penjadwalan yang dijalankan oleh semua pihak yang berkaitan dalam aktivitas belajar mengajar tidak cuma diperhatikan dari sisi mahasiswa, juga patut dilihat dari sisi dosen, yakni kemungkinan-kemungkinan dosen yang akan mengajar lebih dari satu mata kuliah yang ada, sehingga penting untuk dipertimbangkan solusi supaya dosen tidak perlu mengajar dua mata kuliah yang berbeda di hari dan jam yang sama. Disamping itu, patut dipertimbangkan juga aspek-aspek kebijakan universitas sehingga kegiatan perkuliahan bisa dapat dilakukan dengan baik.

Proses pembuatan jadwal perkuliahan di Universitas ARS saat ini pengerjaan pembuatan jadwal masih secara manual dengan menggunakan Microsoft Office Excel, sehingga memerlukan waktu pengerjaan yang cukup lama dengan kisaran waktu sekitar 14 hari dari pelaksanaan rapat hingga proses pembuatan jadwal selesai dan terdapat kendala dalam pengalokasian mata kuliah, dosen, mahasiswa yang mengambil mata kuliah beserta ruangnya agar tidak terjadi kesalahan penjadwalan yang ada. Saat ini dikelola oleh staf dengan acuan kesediaan mengajar seluruh dosen yang mengajar dengan berbagai ketentuan dari masing-masing Kepala Program Studi. Teknis penjadwalannya adalah dengan menginputkan dosen, matakuliah, ruangan, kelas dan waktu dalam aplikasi Microsoft Excel, setelahnya disesuaikan dengan ketersediaan mengajar dosen, dengan demikian banyak jadwal yang berbentrok pada awal penginputan sehingga diperlukan penyesuaian-penyesuaian yang memakan cukup banyak waktu. Disamping itu, semakin meningkatnya jumlah mahasiswa di 3 tahun terakhir membuat urgensi dari aplikasi penunjang yang membantu tenaga kependidikan diperlukan.



Gambar 1. 1.
Jumlah Mahasiswa Program Studi Sistem Informasi

Kompleksitas penjadwalan matakuliah dapat dipecahkan dengan beberapa metode, salah satunya dengan mengaplikasikan algoritma pencarian heuristik. Untuk kasus yang sederhana bisa mengaplikasikan algoritma pencarian heuristik yang umum seperti *breadth-first search* and *depth-first search*. Sebagai masukan dan prasyarat yang kompleks misalnya pada kasus penjadwalan matakuliah, maka algoritma pencarian heuristik tidak bisa diaplikasikan untuk mendapatkan sebuah solusi yang terbaik. Dalam kasus penjadwalan matakuliah, diperlukan algoritma yang dapat menyelesaikan masalah multi-objektif dan multi-kriteria. Algoritma genetika Salah satu algoritma yang dapat digunakan untuk penyelesaiannya (Hanita, 2011).

1.2 Rumusan Masalah

Dalam penyajian Tesis ini penulis melakukan merumuskan masalah sebagai berikut:

1. Bagaimana menerapkan algoritma genetika dalam penjadwalan mata kuliah di Program Studi Sistem Informasi Universitas ARS?
2. Faktor-faktor apa saja yang berpengaruh dalam penjadwalan mata kuliah baik itu ketentuan atau kriteria khusus di Program Studi Sistem Informasi Universitas ARS?

1.3 Tujuan Penelitian

Dalam penyajian Tesis ini penulis mempunyai tujuan sebagai berikut:

1. Mengimplementasikan algoritma genetika dalam penjadwalan mata kuliah di Program Studi Sistem Informasi Univ ARS.
2. Menganalisis dan mengidentifikasi faktor-faktor yang berpengaruh dalam penjadwalan dalam mata kuliah di Program Studi Sistem Informasi Universitas ARS.

1.4 Batasan Masalah

Untuk membatasi masalah yang akan dibahas, maka perlu kiranya dibuat ruang lingkup sebagai berikut:

1. Penelitian tidak membahas jadwal di luar perkuliahan di Program Studi Sistem Informasi Universitas ARS.
2. Jadwal perkuliahan yang digunakan adalah jadwal tahun ajaran 2019-2020 semester genap Program Studi Sistem Informasi Universitas ARS.

1.5 Sistematika Penulisan

Dalam penyusunan tesis ini penulis mengambil sistematika penulisan sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi mengenai latar belakang penelitian, rumusan masalah, tujuan penelitian, ruang lingkup penelitian, metode penelitian dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini menjelaskan beberapa teori dan pengertian yang melandasi penelitian yang berhubungan dengan analisis yang dilakukan diantaranya adalah teori tentang algoritma gentika, penjadwalan kuliah, metodologi pengembangan sistem dan tentang *agile development methods*.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan gambaran umum Universitas ARS baik secara organisasi dan proses penjadwalan, selain itu di bab ini dijelaskan juga tentang metodologi penelitian yang akan dilakukan disertai dengan tahapan dari algoritma genetika.

BAB IV PEMBAHASAN

Bab ini membahas tentang penjadwalan yang akan menunjukkan hasil perbedaan yang dihasilkan oleh algoritma genetika, sehingga hasil tersebut bisa dijadikan acuan sebagai pengembangan sistem penjadwalan di Universitas ARS.

BAB V KESIMPULAN

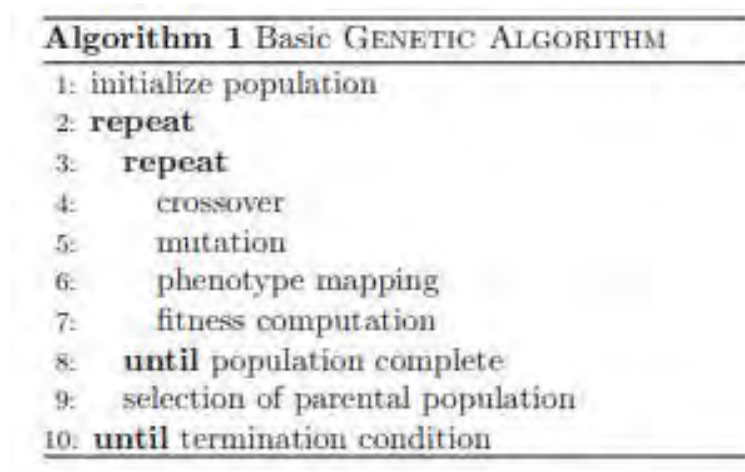
Bab ini berisi kesimpulan dari hasil keseluruhan penelitian yang telah dilakukan, isi dalam bab ini berupa analisis dari kajian pustaka dan juga interpretasi dari tema yang telah diangkat juga berisikan tentang saran yang bertujuan untuk usulan penelitian selanjutnya.

BAB II

LANDASAN TEORI

2.1 Algoritma Genetika

Algoritma Genetika merupakan pendekatan pencarian heuristik yang dapat diterapkan secara luas berbagai masalah pengoptimalan. Evolusi adalah dasar dari Algoritma Genetika. Algoritma genetika telah berkembang hingga struktur kompleks yang memungkinkan kelangsungan hidup di berbagai jenis lingkungan. Pendekatan yang diambil oleh algoritma ini adalah dengan menggabungkan secara acak berbagai pilihan solusi terbaik, di dalam suatu kumpulan untuk mendapatkan generasi solusi terbaik berikutnya yaitu pada suatu kondisi yang memaksimalkan kecocokannya. Algoritma Genetika klasik didasarkan pada sekumpulan solusi kandidat yang mewakili solusi untuk masalah pengoptimalan yang ingin kita selesaikan. Solusi adalah potensi kandidat untuk masalah pengoptimalan yang optimal (Kramer, 2017).



Gambar 2. 1
Gambar Basic Algoritma Genetika (Kramer, 2017)

Algoritma genetika pertama kali ditemukan pada tahun 1975 oleh John Holland melalui sebuah penelitian dari Universitas Michigan kemudian David Golberg yang merupakan murid dari John Holland mengembangkan Algoritma Genetika di tahun 1989. Awalnya tujuan Holland adalah hanya untuk mempelajari fenomena adaptasi seperti yang terjadi di alam sekaligus untuk mengembangkan cara-cara bagaimana mekanisme adaptasi alam dapat diimpor ke sebuah

sistem komputer, namun seiring berjalannya waktu algoritma ini mampu memecahkan masalah yang spesifik (Hanita, 2011). Algoritma genetika adalah algoritma optimasi pencarian yang memaksimalkan atau meminimalkan fungsi yang diberikan. Awalnya algoritma genetika memiliki prosedur heuristik, algoritma genetika tidak dijamin untuk menemukan yang optimal, tetapi pengalaman telah menunjukkan bahwa algoritma genetika dapat menemukan solusi yang sangat baik untuk berbagai masalah (Sohal, 2017).

Algoritma Genetika merupakan algoritma pencarian yang berdasar pada mekanisme seleksi secara alamiah dan genetika alami. Karena mengambil dari kaidah konsep genetika biologi, beberapa istilah dalam Algoritma Genetika juga mengambil konsep yang sama seperti dilansir dari buku berjudul “Algoritma Genetika dalam Matlab” oleh Suyanto (2005), yaitu:

1. Populasi adalah sekelompok individu yang akan dicari penyelesaiannya dalam Algoritma Genetika.
2. Kromosom atau Individu adalah penyelesaian yang merupakan komponen iterasi dan atau satu permasalahan.
3. Gen adalah yang bertugas menyusun kromosom menjadi individu utuh dan bagian dari kromosom yang memiliki nilai tertentu.
4. Nilai *fitness* adalah yang menentukan apakah individu tersebut berkualitas atau tidaknya derajat kebaikan suatu individu.
5. Generasi adalah jumlah tingkatan peranakan sebuah kelompok populasi yang akan dikembangkan menjadi populasi yang mutan, mengalami crossover dan mutasi beberapa kali sehingga menimbulkan generasi terbaik.

2.1.1 Struktur Umum Algoritma Genetika

Suatu solusi yang dalam bentuk masih seperti simbol, itu merupakan Kromosom. Populasi pertama dibangun secara acak, sementara populasi itu sendiri merupakan hasil evolusi daripada kromosom-kromosom melalui proses iterasi atau percobaan yang bisa disebut dengan istilah generasi dengan menggunakan alat ukur yang disebut dengan fungsi *fitness*. Proses evaluasi akan terjadi pada semua generasi kromosom dan kualitas dari kromosom dalam

populasi akan ditunjukkan dengan nilai *fitness* dari pada suatu kromosom (Charles L Karr, 1999). Struktur dari suatu algoritma genetik secara umum dapat dijabarkan dengan langkah-langkah sebagai berikut:

1. Membangkitkan populasi awal

Proses membangkitkan sejumlah individu secara acak atau melalui prosedur tertentu merupakan proses membangkitkan populasi. Populasi awal ini dibangkitkan secara random atau acak sehingga dihasilkan solusi pertama. Populasi tersebut terdiri atas sejumlah kromosom yang merepresentasikan dari solusi yang diinginkan. Pada setiap individunya syarat-syarat yang harus dipenuhi untuk menunjukkan suatu solusi harus benar-benar diperhatikan.

2. Membentuk generasi baru

Generasi baru ini dikenal dengan istilah anak (*offspring*). Operasi pada proses reproduksi atau seleksi, *crossover* dan mutasi digunakan untuk membentuk generasi baru. Proses ini dilakukan secara berulang-ulang sehingga dihasilkan jumlah kromosom yang cukup dalam membentuk generasi yang baru dimana generasi baru tersebut merupakan representasi daripada solusi yang baru.

3. Evaluasi solusi

Pada setiap generasi, kromosom akan mengalami proses evaluasi yang menggunakan alat ukur yang disebut dengan *fitness*. Nilai *fitness* daripada suatu kromosom dapat menggambarkan kualitas dari kromosom dalam populasi tersebut. Pada proses ini akan dilakukan proses evaluasi setiap populasi dengan cara menghitung nilai *fitness* dari setiap kromosom dan mengevaluasinya sampai kriteria henti terpenuhi. Jika kriteria henti belum terpenuhi maka akan dilakukan lagi pembentukan generasi baru dengan mengulangi langkah 2. Beberapa kriteria henti yang sering digunakan antara lain yaitu berhenti pada generasi tertentu, berhenti setelah beberapa generasi berturut-turut didapatkan nilai *fitness* tertinggi yang konsisten, berhenti pada generasi yang tidak mendapatkan nilai *fitness* yang lebih tinggi.

2.1.2 Komponen Algoritma Genetika

Terdapat enam komponen utama yang ada di dalam algoritma genetika, (Hanita, 2011) yaitu sebagai berikut:

1. Teknik Pengkodean

Teknik Pengkodean adalah cara bagaimana mengkodekan gen dari kromosom. Teknik ini untuk menyatakan populasi awal sebagai calon solusi dari suatu masalah ke dalam suatu kromosom sebagai kunci dari pokok masalah tersebut. Teknik ini meliputi pengkodean gen dan juga kromosom. Gen merupakan bagian daripada kromosom yang dapat direpresentasikan ke dalam bentuk *string bit*, *tree*, *array*, bilangan real, daftar aturan, elemen permutasi, elemen program, atau representasi lainnya yang dapat diimplementasikan pada operator genetika.

2. Prosedur Inisialisasi (*generate* populasi awal)

Prosedur ini merupakan proses secara acak (random) yang menghasilkan sejumlah individu. Banyaknya populasi tergantung dari pada masalah yang akan diselesaikan dan juga tergantung dengan jenis operator genetika yang akan diterapkan. Setelah menentukan jumlah populasi, maka proses selanjutnya adalah melakukan inisialisasi terhadap kromosom yang ada di dalam populasi tersebut. Inisialisasi kromosom tersebut dilakukan secara acak, namun tetap memperhatikan domain solusi dan kendala dari permasalahan yang ada.

3. Fungsi Evaluasi

Sebagai ukuran kinerja individu yang di evaluasi dan individu yang akan dipertahankan adalah individu dengan kromosom yang nilai fitnessnya tinggi, sedangkan yang akan diganti adalah individu yang kromosomnya bernilai *fitness* rendah. Fungsi *fitness* bergantung pada masalah tertentu dari representasi yang digunakan. Rumus *fitness* yang digunakan, (Lee dkk, 2001) adalah sebagai berikut:

$$Fitness = \frac{1}{1 + (F_1 B_1 + F_2 B_2 + \dots)}$$

Rumus 1. 1. *Fitness*

Keterangan: B_n = Bobot pelanggaran
 F_n = Banyaknya pelanggaran
 $N = 1, \dots, n$

4. Seleksi

Pada proses persilangan dan mutase, proses seleksi ini bertujuan untuk memilih individu-individu mana yang akan dipilih atau berfungsi untuk memilih calon induk yang baik. Keturunan yang baik tentunya berasal dari induk yang baik. Langkah pertama dalam proses seleksi adalah mencari nilai *fitness*. Pada wadah seleksi tersebut, masing-masing dari individu seleksi akan menerima peluang reproduksi yang bergantung pada nilai objektif dirinya sendiri terhadap nilai objektif dari semua individu. Dan pada tahap seleksi selanjutnya nilai *fitness* inilah yang nantinya akan digunakan. Terdapat beberapa metode untuk memilih kromosom yang akan digunakan diantaranya sebagai berikut:

a. *Roulette wheel selection*

Merupakan proses untuk menyeleksi *parent* dengan tujuan untuk mempertahankan nilai dari *fitness*-nya supaya memiliki kesempatan untuk dapat menyeleksi kromosom yang baik. Proses ini dapat di ibaratkan seperti permainan roda rolet (*roulette wheel*), yang mana semua kromosom ditempatkan dalam populasi yang besar sesuai dengan fungsi *fitness*. Kromosom dipilih berdasarkan dari nilai *fitness*, semakin besar nilai *fitness*-nya maka kromosom tersebut memiliki peluang untuk dipilih beberapa kali. Metode ini merupakan metode yang paling sederhana, yang juga biasa dikenal dengan nama *stochastic sampling with replacement*.

b. Seleksi *good fitness*

Seleksi ini dapat dilakukan dengan cara menghilangkan setengah dari jumlah populasi yang memiliki nilai *fitness* yang paling rendah, sehingga yang tersisa hanyalah kumpulan solusi yang terbaik. Sisa solusi hasil dari seleksi tersebut disebut sebagai populasi induk. Dikarenakan jumlah populasi harus tetap, maka diperlukan *generate* solusi baru yaitu

sebanyak setengah daripada jumlah populasi yang ada. Terdapat dua cara yang dapat digunakan untuk men-*generate* solusi baru, yaitu melalui reproduksi kromosom baru dan melalui mutasi dari solusi induk. Tujuannya adalah untuk menemukan solusi alternatif yang lebih baik dari solusi-solusi yang sudah dihasilkan sebelumnya.

5. Operator Genetika

Pemilihan operator juga sangat menentukan keberhasilan daripada algoritma genetika untuk menemukan solusi yang optimum pada suatu masalah yang diberikan, ini dikarenakan algoritma genetika merupakan proses pencarian yang acak dan juga heuristik. Untuk menghindari terjadinya konvergensi *premature* atau proses dimana belum waktunya untuk mendapatkan solusi optimum, dalam artian bahwa solusi yang diperoleh merupakan hasil optimum lokal yang harus diperhatikan. Terdapat dua operator genetika yaitu:

- a. *Crossover* (Persilangan) adalah proses dalam algoritma genetika yang berfungsi untuk menggabungkan dua buah kromosom parent menjadi kromosom baru (*offspring*) pada suatu waktu tertentu. Melalui proses *crossover* pada dua buah kromosom dapat diperoleh sebuah kromosom yang mengarah pada solusi yang baik. Cara yang cukup sederhana dalam proses *crossover* yaitu melalui proses pemilihan satu titik yang dipisahkan secara acak yang kemudian membentuk *offspring* dengan cara segmen dari satu induk ke sebelah kiri dari titik yang dipisahkan yang dikombinasikan dengan segmen dari induk yang lain ke sebelah kanan dari titik yang dipisahkan.
- b. *Mutation* (Mutasi) adalah proses untuk mengubah salah satu atau lebih beberapa gen dari suatu kromosom. Fungsi daripada proses ini adalah untuk menggantikan gen yang hilang dari populasi akibat proses seleksi yang mungkin menyebabkan munculnya kembali gen yang tidak muncul pada saat inisialisasi populasi.

6. Parameter Kontrol

Operator - operator seleksi perlu dikendalikan oleh parameter control genetika. Dalam memecahkan suatu masalah untuk menentukan penampilan kinerja algoritma genetika maka pemilihan parameter genetika turut berpengaruh (Desiani, 2006). Probabilitas *crossover* (P_c)

dan probabilitas mutasi (P_m) adalah parameter dasar dalam algoritma genetika yang akan dijelaskan sebagai berikut:

- a. Probabilitas *Crossover* (P_c) Probabilitas *crossover* dapat mengendalikan operator *crossover* pada setiap generasi dalam populasi yang mengalami *crossover*. Semakin besar nilai dari probabilitas *crossover* maka akan semakin cepat pula struktur individu baru yang terbentuk ke dalam populasi. Namun jika nilai probabilitas *crossover* terlalu besar, maka individu yang adalah kandidat solusi terbaik memiliki kemungkinan dapat hilang lebih cepat pada generasi selanjutnya. Nilai probabilitas *crossover* yang disarankan yaitu antara 80 % - 95 %.

- b. Probabilitas mutasi (P_m)

Probabilitas mutasi dapat mengendalikan operator mutasi pada setiap generasi dengan peluang mutasi yang digunakan lebih kecil daripada peluang *crossover*. Melalui seleksi alamiah, operator mutasi dalam algoritma genetika tidak sering terjadi ini dikarenakan mutasi jarang sekali muncul. Nilai probabilitas mutasi yang disarankan sangat kecil yaitu antara 0.5% - 1%.

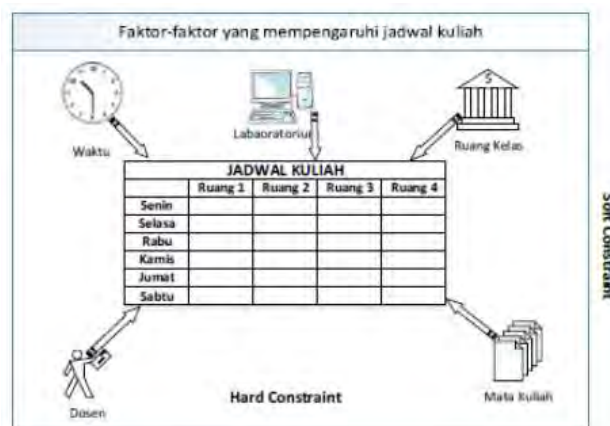
2.2 Penjadwalan Kuliah

Penjadwalan adalah kegiatan pengalokasian yang konsisten antara ruangan, dosen dan mahasiswa dengan pengaturan yang terbaik (Rudová, 2014). Definisi umumnya adalah merupakan serangkaian peristiwa (konferensi, kendaraan, perkuliahan, acara-acara publik, dll). Keterbatasan ini dapat dikategorikan sebagai *hard constraint* dan *soft constraint* dimana *hard constraint* prioritas yang lebih tinggi daripada *soft constraint*. Dalam penyusunan penjadwalan terdapat Batasan kaku (*hard constraintss*) dan batasan lunak (*soft constraintss*) (Suhartono, 2015) seperti berikut ini:

1. Batasan kaku (*hard constraintss*)
 - a. Ruangan tidak dapat digunakan dalam waktu yang bersamaan.
 - b. Dosen tidak dapat mengajar dalam waktu yang bersamaan.
 - c. Tidak ada waktu perkuliahan yang lewat dari jam 18.00

- d. Tidak ada proses belajar mengajar pada waktu 12.10 -13.00 karena waktu tersebut dialokasikan untuk istirahat, solat dan makan.
 - e. Jumlah mahasiswa yang mengambil mata kuliah tidak boleh lebih dari kapasitas ruangan yang digunakan.
 - f. Mata kuliah dengan semester dan kelas yang sama tidak dapat dilaksanakan diwaktu yang bersamaan.
 - g. Matakuliah yang bersifat non-praktikum tidak dapat menggunakan ruangan praktikum.
 - h. Mata kuliah yang bersifat praktikum tidak dapat menggunakan ruangan non-praktikum.
2. Batasan lunak (*soft constraintss*):
- a. Dosen dapat melakukan *request* waktu untuk mengajar yang terdiri dari pagi (08:00, 08:50, 09:40, 10:30, 11:20) dan siang (13:00, 13:50, 14:40, 15:30, 16:20, 17:10) setiap harinya.
 - b. Dosen dapat melakukan request waktu dengan catatan harus memilih waktu minimal sebanyak jumlah mata kuliah yang diampu atau diajarkan

Berdasarkan gambar 2.2 menjelaskan bahwa terdapat batasan-batasan yang harus dipenuhi serta tidak boleh dilanggar. Penjadwalan mata kuliah juga dipengaruhi oleh beberapa komponen diantaranya mata kuliah, ruang kelas, dosen, laboratorium komputer serta waktu dengan batasan-batasan tertentu. Batasan tersebut merupakan ukuran kualitas daripada penjadwalan mata kuliah, sehingga suatu jadwal mata kuliah yang optimal dapat terbentuk.



Gambar 2. 2.

Faktor-faktor yang mempengaruhi penjadwalan matakuliah (Hanita, 2011)

2.3 Metode Pengembangan Sistem

Pengembangan perangkat lunak mengacu pada pembuatan program komputer dan situs web. Program komputer adalah sekumpulan instruksi yang dijalankan (dilakukan) oleh komputer untuk tujuan tertentu atau serangkaian tujuan. Program juga disebut sebagai aplikasi atau perangkat lunak. Pengembangan perangkat lunak juga disebut *coding*, pemrograman komputer dan rekayasa perangkat lunak. Masing-masing istilah ini berarti pada dasarnya sama yaitu memberi tahu komputer apa yang harus dilakukan dengan menggunakan kode. Kode adalah kata dan simbol yang digunakan untuk mengkomunikasikan instruksi komputer. Kode ditulis dalam berbagai bahasa pemrograman. Bahasa pemrograman adalah sistem kata, frasa dan simbol yang memungkinkan anda membuat program. Ada banyak jenis bahasa pemrograman, yang masing-masing dibuat untuk memenuhi tujuan tertentu. Misalnya, beberapa bahasa lebih cocok untuk membuat *game* komputer, sementara yang lain lebih baik digunakan untuk membuat aplikasi telepon (Stanley, 2020).

Metode Pengembangan Sistem (Adi Nugroho, 2010) USDP (*Unified Software Development Process*) salah satu metode pengembangan sistem / perangkat lunak yang menggunakan UML (*Unified Modeling Language*) sebagai tool utamanya dengan tahapan yaitu:

1. Perencanaan (*Planning*)

Studi tentang kebutuhan pengguna (*user's specification*), studi-studi kelayakan (*feasibility study*) baik secara teknis maupun secara teknologi serta penjadwalan pengembangan suatu proyek sistem informasi dan atau perangkat lunak.

2. Analisis (*Analysis*)

Tahap menggali permasalahan yang muncul pada pengguna dengan mendekomposisi dan merealisasikan use case diagram lebih lanjut, mengenali komponen-komponen sistem, objek-objek, hubungan antar objek, dan sebagainya.

3. Perancangan (*Design*)

Mencari solusi permasalahan yang didapat dari tahap analisis, pada tahap ini dibagi menjadi dua yaitu:

- a. Tahap perancangan yang lebih menekankan pada platform apa hasil dari tahap analisis yang akan di implementasikan.
 - b. Tahap perancangan yang dimana melakukan penghalusan (refinement) kelas-kelas yang didapat pada tahap analisis serta menambahkan dan memodifikasi kelas-kelas yang akan lebih mengefisienkan serta mengefektifkan sistem/perangkat lunak yang akan dikembangkan.
4. Pengujian (*Testing*)
- Pada tahap ini digunakan untuk menentukan apakah sistem yang dibuat sudah sesuai dengan kebutuhan pengguna atau belum. Jika belum, proses selanjutnya adalah bersifat interaktif yaitu kembali ke tahap sebelumnya.

2.4 Agile Software Development

Agile Development Methods adalah sekelompok metodologi pengembangan perangkat lunak yang didasarkan pada pengembangan berulang, dimana persyaratan dan solusinya berkembang melalui kolaborasi antara tim lintas fungsi yang mengatur sendiri. Jadi, bisa disimpulkan jika *agile* merupakan model pengembangan jangka pendek yang memerlukan adaptasi cepat dan pengembangan terhadap perubahan dalam bentuk apapun. *Agile Software Development* memiliki tujuan sebagai berikut (Collier, 2011):

1. *High-value & working App system*

Dapat menghasilkan perangkat lunak yang mempunyai nilai jual yang tinggi, biaya pembuatan bisa di tekan dan perangkat lunak bisa berjalan dengan baik.

2. *Iterative, incremental, evolutionary*

Agile adalah metode pengembangan perangkat lunak yang iteratif, selalu mengalami perubahan, dan evolusioner.

3. *Cost control & value-driven development*

Menngembangkan perangkat lunak yang sesuai dengan kebutuhan pengguna dan tim bisa dengan cepat merespon kebutuhan yang diinginkan pengguna sehingga waktu dan biaya pembuatan perangkat lunak bisa dikontrol.

4. *High-quality production*

Kualitas dari perangkat lunak yang dibuat tetap terjaga, meskipun waktu dan biaya lebih sedikit.

5. *Flexible & risk management*

Pertemuan dengan klien untuk melakukan tes perangkat lunak dapat sering dilakukan sehingga fungsionalitas perangkat lunak mudah diubah dan akhirnya kegagalan perangkat lunakpun bisa diminimalisir.

6. *Collaboration*

Tim pengembang diharuskan sering bertemu untuk membahas perkembangan proyek dan feedback dari klien untuk ditambahkan dalam perangkat lunak, sehingga tim bisa berkolaborasi dengan maksimal.

7. *Self-organizing, self-managing teams*

8. Dengan *agile*, pengembang dapat memanajemen dirinya sendiri, sedangkan manajer tim hanya bertugas mengkolaborasikan pengembang perangkat lunak dengan klien. Sehingga terciptalah tim yang solid.

Metodologi manajemen proyek *agile* pertama kali diperkenalkan tahun 1980-an, ditandai dengan pendekatan non-linier. Metode ini tidak berakhir hanya pada langkah perencanaan, melainkan juga berlanjut ke persyaratan dan solusi berkembang di seluruh proses melalui kolaborasi antara pengorganisasian diri, lintas fungsi tim. Pelaksanaan pekerjaan proyek dilakukan terus menerus sepanjang umur proyek. Metode *agile* melibatkan perencanaan adaptif, evolusioner pengembangan, pengiriman awal, dan peningkatan berkelanjutan. *Agile* mendorong respon cepat dan fleksibel terhadap perubahan. Cara ini semakin populer dan sering digunakan untuk mengelola proses pembuatan perangkat lunak. Ini berfokus pada kerja tim dan menyelesaikan satu fitur perangkat lunak pada satu waktu (sepenuhnya) sebelum pindah ke berikutnya. Kami membahas beberapa data primer dan istilah yang terkait dengan proyek pengelolaan (Stanley, 2020).

Singkatnya, *agile* adalah metode manajemen proyek yang fokus tentang membagi tugas menjadi fase kerja yang pendek, dengan sering meninjau proyek dan adaptasi perencanaan

tengah pelaksanaan sesuai kebutuhan. Pendekatan *agile* ini fleksibel dan tidak sekaku beberapa metode proyek lainnya, pengelolaan nilai utama dalam *Development Agile* adalah memungkinkan tim untuk memberikan nilai lebih cepat, dengan kualitas dan prediksi yang lebih baik, dan bakat yang lebih besar untuk merespons perubahan. *Agile* secara harfiah berarti mampu bergerak dengan cepat dan mudah. Dalam proyek manajemen, *agile* adalah metodologi yang populer diindustri komputer (Stanley, 2020). *Agile Software Development* merupakan metodologi pengembangan perangkat lunak yang dilakukan secara iteratif. Iteratif yang dimaksud adalah di setiap iterasi, pengembangan perangkat lunak yang dilakukan harus menghasilkan suatu produk yang siap pakai namun belum memenuhi semua requirement atau yang biasa disebut *minimum viable product* (MVP) (Satria, 2019).



Gambar 2. 3.
Agile Lifecycle (Sohail, 2019)

Ada 6 tahap di dalam 1 iterasi *agile*, di antaranya adalah sebagai berikut :

1. *Requirements*

Pada tahap ini, tim *development* membahas *requirements* yang dibutuhkan bersama *Product Owner*. Biasanya, product owner menyediakan *Product Backlog Item* (PBI) beserta *flow* yang dibutuhkan

2. *Plan*

Pada tahap ini, tim *development* membahas apa saja yang akan dilaksanakan pada tahap *development* dalam 1 iterasi tersebut. Tim *development* membahas goal yang akan dicapai, PBI yang akan dieksekusi, *tasks* yang di-*breakdown* berdasarkan PBI tersebut, dan siapa saja yang akan mengerjakan task tersebut.

3. *Design*

Pada tahap ini, tim *development* membuat *prototype clickable* yang menggambarkan *flow* aplikasi berdasarkan dokumentasi yang diberikan

4. *Development*

Pada tahap ini, tim *development* mengimplementasikan kode berdasarkan *tasks* yang sudah di-*assign* pada mereka di tahap *plan*

5. *Release*

Pada tahap ini, produk yang sudah menjadi *minimum viable* akan di-*release* atau *deploy*

6. *Track & Monitor*

Pada tahap ini, produk yang sudah di-*release* akan kita *track* dan *monitor* untuk melihat apakah aplikasi yang kita *release* sudah bagus berdasarkan terjadi *bug* atau tidak dan *feedback* dari pengguna. Kalau pun sudah bagus, pada tahap ini kita bisa menemukan celah-celah untuk menyempurnakan.

2.5 Unified Modeling Language (UML)

UML digunakan sebagai notasi untuk sejumlah aktivitas seperti *use case modeling*, *target performance analysis*, serta desain arsitektural dan detail pada berbagai tingkat perincian. Artefak UML mewakili fondasi penting untuk perencanaan dan pengendalian proyek rekayasa perangkat lunak yang digerakkan oleh tonggak sejarah. Oleh karena itu, UML sebagian besar diimplementasikan dalam proyek-proyek berbasis rencana dengan dokumentasi yang relatif tinggi dan ketidak fleksibelan yang dihasilkan dari itu. Namun, UML lebih kompak, secara semantik lebih banyak, dan lebih cocok untuk menggambarkan fakta yang kompleks daripada

bahasa pemrograman. Itulah mengapa ia memiliki keuntungan penting untuk pemodelan kasus uji dan evolusi transformasional dari sistem perangkat lunak (Rumpe, 2016).

Menurut Satzinger (2012: 46), "*Unified Modeling Language (UML) is the standard set of model construct and notations defined by the Object Management Group (OMG), a standart organization for system development.*" Yang terjemahannya adalah *Unified Modeling Language (UML)* merupakan kumpulan model standar konstruksi dan notasi yang didefinisikan oleh *Object Management Group (OMG)*, yang adalah sebuah organisasi standart untuk pengembangan sistem.

Menurut Shelly dan Rosenblatt (2012: 151), "*The Unified Modeling Language (UML) is a widely used method of visualizing and documenting software systems design*". yang terjemahannya adalah *Unified Modeling Language (UML)* adalah metode yang digunakan secara luas memvisualisasikan dan mendokumentasikan desain sistem perangkat lunak. Berdasarkan pengertian diatas dapat disimpulkan bahwa *Unified Modeling Language (UML)* adalah metode yang diginkan untuk dapat mengembangkan sistem sesuai dengan keinginan *user*.

Merangkum tujuan UML dalam pemodelan, pengembangan, dan pemeliharaan sistem perangkat lunak, dalam hal ini (Unhelkar, 2018) menjelaskan ada lima cara UML digunakan di *Software Engineering*:

1. *Visualizing*

Visualisasi ini adalah tujuan utama UML karena notasi dan diagramnya menyediakan mekanisme standar industri untuk representasi bergambar persyaratan, proses, desain solusi, dan arsitektur. Visual ini dibuat menggunakan alat pemodelan CASE, yang juga memungkinkan berbagi pekerjaan pemodelan berbasis tim.

2. *Specifying*

UML memfasilitasi spesifikasi artefak pemodelan. Misalnya, spesifikasi untuk aktor, kasus penggunaan, kelas, atribut, dan operasi memberikan detail tambahan untuk notasi visual. Spesifikasi ini sangat membantu dalam meningkatkan kualitas solusi sejak ditinjau spesifikasi membantu menyelesaikan banyak kesalahpahaman antara pengguna dan pengembang.

3. *Contracting*

UML digunakan untuk konstruksi perangkat lunak karena memungkinkan pembuatan kode (misalnya, C ++, Java) tergantung pada alat *CASE* yang digunakan. Namun, fitur konstruksi UML ini memiliki aplikasi yang terbatas, terutama karena setelah kode dibuat, sebagian besar proyek praktis bekerja langsung pada modifikasi kode daripada desainnya. Perjalanan pulang pergi teknik dimaksudkan untuk membantu memodifikasi desain berdasarkan kode yang diperbarui, tetapi fitur ini dalam praktiknya tidak sepopuler pemikiran sebelumnya.

4. *Documenting*

Dengan bantuan UML, dokumentasi tambahan dan terperinci untuk persyaratan, arsitektur, desain, rencana proyek, pengujian, dan prototipe disediakan untuk meningkatkan spesifikasi dan representasi visual.

5. *Maintaining*

Model UML yang baik merupakan bantuan yang signifikan dalam pemeliharaan perangkat lunak yang berkelanjutan sistem. Model memungkinkan tampilan yang mudah dari sistem yang ada, arsitekturnya, dan desain TI. Ini memungkinkan pemrogram untuk mengidentifikasi tempat yang benar di dalam sistem untuk perubahan dan, lebih penting lagi, pahami efek perubahannya pada sistem lainnya.



Gambar 2. 4.
UML (Unhelkar, 2018)

2.5.1 Activity Diagram

Menurut Satzinger (2012: 57), *“an Activity Diagram describes user (or system) activities, the person who does each activity, and the sequential flow of these activities”* yang terjemahannya Activity Diagram yang menggambarkan beberapa aktivitas pengguna (atau sistem), orang yang melakukan setiap aktivitas, dan arus yang berurutan dari aktivitas. Berdasarkan pengertian diatas dapat disimpulkan bahwa *Activity Diagram* merupakan sebuah diagram yang menggambarkan setiap aktivitas yang terjadi yang dilakukan pengguna atau sistem secara berurutan.

Suatu *Activity Diagram* merupakan gambaran berbagai pengguna (atau sistem) kegiatan, orang yang melakukan setiap kegiatan, dan aliran sekuensial dari kegiatan tersebut. Simbol yang digunakan yaitu:

1. *Starting Activity (pseudo)*

Merupakan simbol untuk menandakan dimulainya aktivitas.

2. *Transition Arrow*

Merupakan garis penunjuk arah urutan aktivitas yang menggambarkan transisi dari suatu aktivitas.

3. *Activity*

Merupakan simbol yang menggambarkan aktivitas.

4. *Ending Activity (pseudo)*

Merupakan simbol untuk menandakan berakhirnya aktivitas.

5. *Swimlane*

Merupakan area persegi dalam *activity* diagram yang menunjukkan aktivitas diselesaikan *single agent*.

6. *Synchronization Bar*

Merupakan simbol yang digunakan untuk mengontrol pemisahan atau penyatuan dari jalur berurutan.

7. *Diamond*

Merupakan simbol poin keputusan dalam alur proses mengikuti satu jalur atau jalur lainnya.

2.5.2 *Use Case Diagram*

Menurut (Satzinger, 2012: 81), *use case diagram* menunjukkan sebuah tongkat sederhana yang digunakan untuk mewakili aktor (tangan ditunjukan langsung mengakses ke sistem langsung). Kasus penggunaan sendiri dilambangkan oleh oval dengan nama use case diagram didalamnya. Garis yang menghubungkan aktor dengan *use case diagram* menunjukkan bahwa aktor memanfaatkan penggunaan sistemnya. Pelaku juga dapat menggunakan sistem lain yang langsung menunjukkan antar muka dengan sistem yang sedang dikembangkan. Berikut adalah komponen – komponen dari suatu *Use Case Diagram*:

1. *System Boundary*

Menggambarkan batasan antara sistem dengan actor.

2. *Use Case*

Menggambarkan apa yang dilakukan actor di dalam sistem.

3. *Actors*

Menggambarkan user atau pengguna dari suatu sistem.

4. *Flow*

Menggambarkan hubungan antara use dengan *actor*.

2.5.3 *Class diagram*

Class diagram menjelaskan suatu deskripsi *class* serta *package* dan objek termasuk hubungan satu sama lain seperti asosiasi, pewarisan dan lain-lain. Pada *class diagram* terdapat 3 poin utama yaitu atribut, nama dan *method*. Diagram kelas adalah salah satu diagram paling populer di SE. Diagram kelas mewakili kuncinya entitas dalam bisnis serta domain teknis. Diagram kelas sangat struktural dan statis di alam, tanpa konten perilaku. Diagram kelas dapat menunjukkan kelas tingkat bisnis, sebagai serta kelas teknis yang diturunkan dari bahasa implementasi (misalnya, Java atau C ++-). Selain menunjukkan kelas, diagram kelas juga menunjukkan hubungan antar kelas. Seluruhnya deskripsi kelas (atau entitas, seperti yang mungkin disebut dalam ruang masalah) dan hubungan yang akan mereka miliki satu sama lain bersifat statis. Tidak ada ketergantungan yang ditunjukkan dalam diagram ini dan tidak ada konsep waktu (Unhelkar, 2018).

2.6 Penelitian Terkait

Penelitian terdahulu bertujuan untuk mendapatkan bahan perbandingan dan acuan. Dengan demikian penulis mencantumkan hasil-hasil penelitian terdahulu sebagai berikut:

Tabel 2. 1 Penelitian Terkait

NO	PENULIS	FOKUS PENELITIAN	JUDUL PENELITIAN	HASIL PENELITIAN
1.	Wiga Ayu Puspaningrum, Arif Djunaidy, Retno Aulia Vinarti	Penjadwalan Jurusan Sistem Informasi ITS.	<i>Penjadwalan Mata Kuliah Menggunakan Algoritma Genetika di Jurusan Sistem Informasi ITS</i>	Menghasilkan Aplikasi yang dibuat untuk membuat jadwal matakuliah dengan keadaan dimana data matakuliah yang ada akan berubah-ubah sesuai periode waktu penjadwalan dan matakuliah yang dibuka. Sehingga, aplikasi harus dapat

NO	PENULIS	FOKUS PENELITIAN	JUDUL PENELITIAN	HASIL PENELITIAN
				dijalankan dengan data masukan matakuliah yang berbedabeda sesuai dengan kebutuhan. Dari data yang didapat pada semester genap tahun ajaran 2011-2012 dan pada data semester ganjil tahun ajaran 2012-2013 Program Studi Sistem Informasi ITS.
2.	Herman, Lukman Syafie, Dirgahayu Lantara	Jadwal kuliah yang menyesuaikan dengan waktu shalat jumat.	<i>Lecture Scheduling Automation Using Genetic Algorithm</i>	Hasil penjadwalan kuliah menggunakan Algoritma Genetika dapat memberikan jadwal kuliah yang menyesuaikan antara jumlah kredit dengan rentang waktu dan kesesuaian antara waktu sholat Jumat dengan waktu kuliah serta waktu dosen tersedia untuk mengajar.
3.	Deeba Kannan, Kuntal Bajpayee, Samriddho Roy	Pengujian algoritma genetika dalam memecahkan permasalahan NP-hard.	<i>Solving Timetable Scheduling Problems Using Genetic Algorithm</i>	Penjadwalan tabel waktu adalah masalah NP-hard. Jadi, pengurangan kompleksitas waktu dan mengurangi tingkat tabrakan adalah masalah utama yang telah kami coba pecahkan dengan menggunakan Algoritma Genetika.
4.	Rahman Erama, Retantyo Wardoyo	Modifikasi Algoritma Genetika.	<i>Modifikasi Algoritma Genetika untuk Penyelesaian Permasalahan Penjadwalan Pelajaran Sekolah</i>	Modifikasi Algoritma Genetika pada penelitian ini dapat dijadikan sebagai alternatif menyelesaikan penjadwalan sekolah karena beberapa peneliti menilai bahwa proses crossover tidak menjamin dapat menghasilkan solusi yang lebih yang lebih baik dari solusi sebelumnya. Algoritma Genetika termodifikasi mencapai 2,30% lebih baik dibandingkan algoritma genetika sederhana untuk menghasilkan solusi.
5.	Entot Suhartono	Optmasi Penjadwalan Kuliah di AMIK JTC Semarang	<i>Optimasi Penjadwalan Mata Kuliah Dengan Algoritma Genetika (Studi Kasus</i>	Penggunaan metode <i>best fitness</i> Algoritma Genetika akan selalu menunjukkan kenaikan <i>fitness</i> atau dengan kata lain generasi selanjutnya lebih baik atau minimal sama dengan generasi sebelumnya.

NO	PENULIS	FOKUS PENELITIAN	JUDUL PENELITIAN	HASIL PENELITIAN
			<i>di AMIK JTC Semarang)</i>	

Perbedaan penelitian ini dengan penelitian terdahulu terletak dalam metode penelitian yang digunakan, penelitian ini menggunakan metode *agile iteraton*, yang sebelumnya belum pernah digunakan di penelitian terkait, dengan demikian proses peracancangan aplikasi di penelitian iini mempunyai keunikan tersendiri, selain itu penjadwalan yang berfokus pada tidak berbenturannya perkulahan dengan jam waktu solat menjadi fokus lainnya yang menjadikan penelitian ini berbeda dengan penelitian terdahulu.

BAB III

OBJEK DAN METODOLOGI PENELITIAN

3.1 Profil Universitas ARS

Objek penelitian dilakukan pada suatu perguruan tinggi swasta yang terletak di Kota Bandung yaitu Universitas Adhirajasa Reswara Sanjaya (ARS) yang memiliki 1 program pasca sarjana program studimanajemen, 4 fakultas yaitu fakultas teknologi dengan program studi teknik informatika dan sistem informasi, fakultas ekonomi dengan program studimanajemen dan akuntansi, fakultas komunikasi desain dengan program studi komunikasi dan desain komunikasi dan visual, fakultas ilmu keperawatan dengan program studi keperawatan dan pendidikan profesi ners, sekolah tinggi pariwisata dengan program studi perhotelan dan manajemen pariwisata.

3.1.1 Gambaran umum Organisasi

Universitas Adhirajasa Reswara Sanjaya merupakan sebuah perguruan tinggi swasta yang terdapat di Kota Bandung. Universitas ARS pertama kali resmi didirikan dengan nama Universitas ARS Internasional dengan Nomor SK 213/D/O/2000 pada tanggal 6 Oktober 2000. Kemudian pada tanggal 19 Januari 2010, Universitas ARS Internasional mengubah namanya menjadi Universitas BSI berdasarkan SK Menteri Pendidikan Nasional dengan nomor 04/D/O/2010. Saat ini, Universitas ARS kembali mandiri dengan sebuah nama baru Adhirajasa Reswara Sanjaya yang secara resmi mengubah nama Universitas BSI pada tanggal 9 Oktober 2019 berdasarkan SK Menteri Riset, Teknologi dan Pendidikan Tinggi dengan nomor 936/KPT/I/2019. Universitas Adhirajasa Reswara Sanjaya (Universitas ARS) memiliki arti universitas yang kuat, tangguh, unggul dan berjaya selama-lamanya. Universitas ARS berlokasi di Jalan Sekolah Internasional No. 1-2 Antapani Bandung 40282.

3.1.2 Visi, Misi, Tujuan

Setiap organisasi maupun perusahaan memiliki tujuan yang harus dicapai, baik untuk jangka waktu yang pendek, maupun untuk jangka waktu yang panjang. Visi dan Misi merupakan sebuah tolak ukur untuk mencapai kesuksesan dan kemajuan perusahaan, sehingga dalam praktiknya, perusahaan memiliki dasar dalam membuat kebijakan dan aturan untuk kepentingan perusahaan maupun karyawan.

Visi atau *Vision* bisa diartikan sebagai sebuah pandangan atau rencana dalam jangka panjang yang ingin dicapai oleh sebuah perusahaan. Visi sebuah perusahaan biasanya cenderung singkat, namun mampu mencakup keseluruhan cita-cita dan tujuan yang ingin dicapai oleh sebuah perusahaan. Sebagai sebuah organisasi yang bergerak di bidang pendidikan, Universitas ARS tentunya mempunyai mimpi dan harapan yang dituangkan dalam visi, misi dan tujuan organisasi. Berikut merupakan Visi, Misi dan Tujuan dari Universitas ARS:

1. Visi

Visi ARS University menjadi perguruan tinggi yang unggul berbasis teknologi informasi yang mampu menghasilkan tenaga profesional yang berjiwa kewirausahaan dalam menghadapi persaingan global pada tahun 2030.

2. Misi

Guna mendukung terwujudnya visi, perlu dukungan misi sebagai langkah dalam pencapaian visi. Misi dari Universitas ARS adalah sebagai berikut:

- a. Melaksanakan pendidikan tinggi berbasis ilmu pengetahuan teknologi dan desain
- b. Menyelenggarakan penelitian dan pengabdian masyarakat yang memberikan kontribusi positif terhadap peningkatan kualitas akademik dan masyarakat.
- c. Melaksanakan tata kelola universitas yang transparan, kredibel, bertanggung jawab dan adil.
- d. Mengembangkan kerjasama dan kemitraan dengan perguruan tinggi lain, instansi pemerintah dan swasta baik dalam negeri maupun luar negeri dalam menumbuhkan kompetensi dan jiwa kewirausahaan.

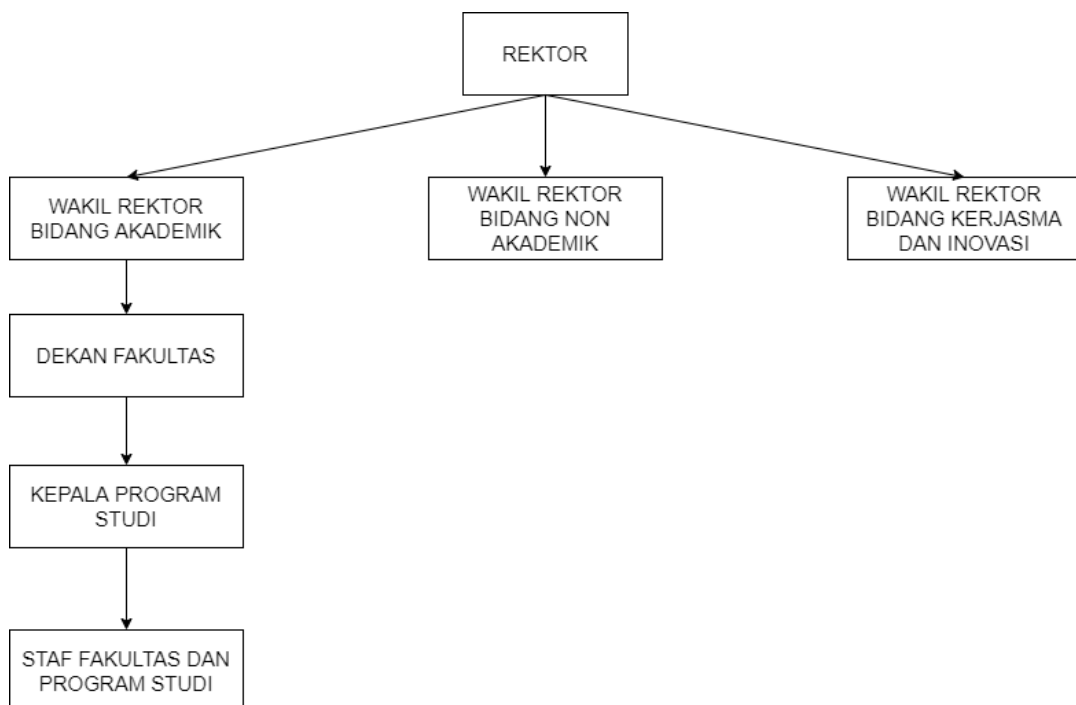
3. Tujuan

Sebagai perguruan tinggi yang andal Universitas ARS mempunyai beberapa tujuan, yakni:

- a. ARS University senantiasa adaptif terhadap perkembangan ilmu pengetahuan dan teknologi dalam menghasilkan lulusan yang memiliki kompetensi digital dan technopreneur.
- b. ARS University senantiasa membekali mahasiswa dengan sertifikasi kompetensi untuk menghasilkan lulusan yang mampu bersaing secara global sesuai kebutuhan industri terkini.

3.1.3 Gambaran Fungsional Organisasi

Penelitian ini berfokus tentang penjadwalan mata kuliah yang dilakukan oleh fakultas masing-masing dengan tim pelaksana oleh staf fakultas dan program studi, berikut merupakan gambaran organisasi di Universitas ARS:

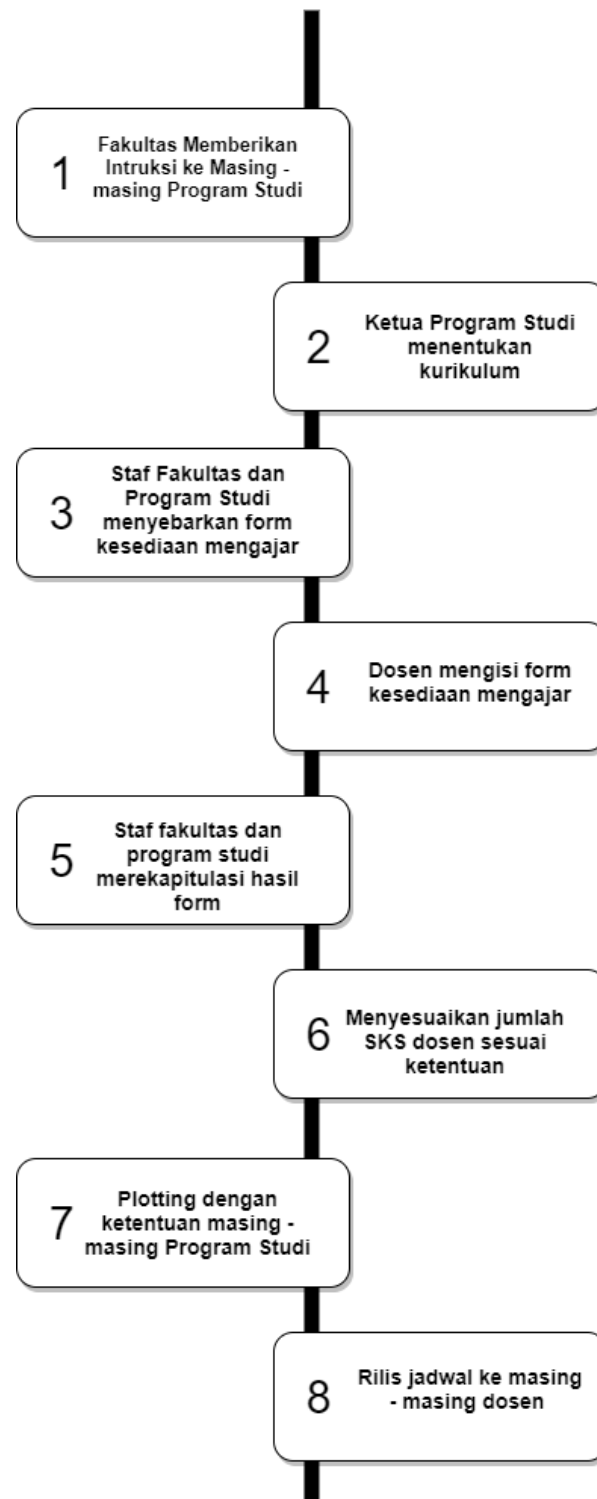


Gambar 3. 1.
Gambaran Organisasi Universitas ARS

3.1.4 Proses Penjadwalan Kuliah Universitas ARS

Proses penjadwalan di Universitas ARS dilakukan dengan tahapan-tahapan berikut ini:

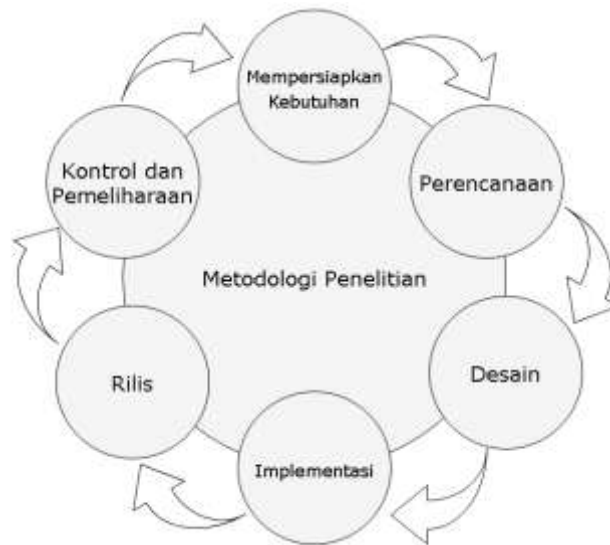
1. Fakultas memberikan intruksi kepada setiap program studi untuk menentukan kurikulum.
2. Masing – masing kepala program studi menentukan kurikulum yang akan dipakai.
3. Staf fakultas dan program studi menyebarkan formulir kesediaan mengajar kepada seluruh dosen, baik dosen tetap ataupun dosen luar biasa.
4. Dosen mengisi kesediaan mengajar yang diberikan oleh program studi masing-masing.
5. Staf fakultas dan program merekapitulasi hasil dari kesediaan mengajar dosen.
6. Staf fakultas dan program studi menyesuaikan jumlah SKS dosen, diantaranya:
 - a. Dosen yang menjabat sebagai pegawai di universitas memiliki batas maksimal 12 SKS
 - b. Dosen luar biasa mempunyai batas minimal 6 SKS mengajar dan tidak ada Batasan untuk maksimal SKS
7. Staf fakultas dan program studi melakukan plotting dengan Microsoft Excel.
 - a. Dosen tetap dan yang menjabat sebagai pegawai universitas akan diprioritaskan dalam penentuan jadwal sesuai dengan formulir kesediaan dosen.
 - b. Dosen luar biasa akan menyesuaikan dengan kekosongan jadwal yang masih tersedia.
8. Rilis jadwal ke masing – masing dosen.
9. Jika ada perubahan dari dosen aka nada perubahan yang dilakukan oleh staf yang bertugas dan akan mempengaruhi penjadwalan lain, sehingga terkadang menyulitkan dalam *plotting* jadwal.



Gambar 3. 2.

Proses Penjadwalan mata kuliah di Fakultas Teknologi Informasi Universitas ARS

3.2 Metodologi Penelitian



Gambar 3. 3.

Gambaran Metodologi Penelitian

Dalam penelitian ini, penulis menggunakan metode pengembangan Aplikasi dengan *Agile Software Development* yang di kombinasikan dengan Algoritma Genetika dalam perancangan aplikasi.

Tahapan yang digunakan pada penelitian ini meliputi:

1. Mempersiapkan kebutuhan

Pada tahap ini, tim *development* membahas *requirements* yang dibutuhkan bersama *Product Owner*. Biasanya, product owner menyediakan *Product Backlog Item* (PBI) dan/atau persona beserta *flow* yang dibutuhkan

2. Perencanaan

Pada tahap ini, tim *development* membahas apa saja yang akan dilaksanakan pada tahap *development* dalam 1 iterasi tersebut. Tim *development* membahas goal yang akan dicapai, PBI yang akan dieksekusi, tasks yang di-breakdown berdasarkan PBI tersebut, dan siapa saja yang akan mengerjakan task tersebut.

3. Desain

Pada tahap ini, tim *development* membuat *prototype clickable* yang menggambarkan flow aplikasi berdasarkan dokumentasi yang diberikan.

4. Implementasi

Pada tahap ini, tim *development* mengimplementasikan kode berdasarkan *tasks* yang sudah di-*assign* pada mereka di tahap *plan*.

5. Rilis

Pada tahap ini, produk yang sudah menjadi *minimum viable* akan di-*release* atau *deploy*.

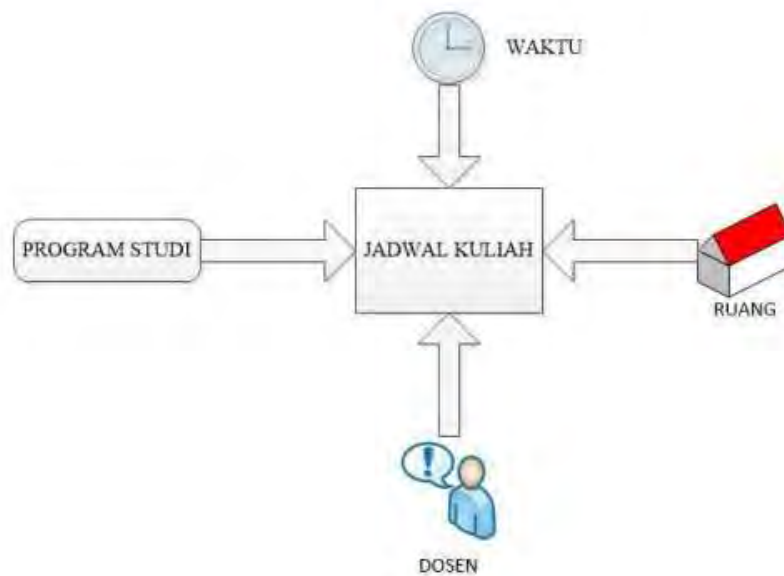
6. Kontrol dan Pemeliharaan

Pada tahap ini, produk yang sudah di-*release* akan kita track dan monitor untuk melihat apakah aplikasi yang kita release sudah bagus berdasarkan terjadi *bug* atau tidak dan *feedback* dari pengguna. Kalau pun sudah bagus, pada tahap ini kita bisa menemukan celah-celah untuk menyempurnakan.

Berdasarkan gambar 3.3 terdapat beberapa proses yang dapat dijelaskan sebagai berikut:

1. Perancangan Analisis Data

Berdasarkan Gambar 3.4, penjadwalan matakuliah dipengaruhi oleh beberapa komponen yang terdiri atas: dosen, ruang kelas, dan waktu dengan sejumlah constraint tertentu, dimana constraint tersebut ada yang harus dipenuhi atau tidak boleh dilanggar. constraint tersebut merupakan ukuran kualitas dari penjadwalan matakuliah, sehingga suatu jadwal matakuliah yang terbaik dapat terbentuk.



Gambar 3. 4.

Faktor yang mempengaruhi penjadwalan matakuliah

Berikut merupakan hal-hal yang mempengaruhi penjadwalan mata kuliah:

- Program Studi menampung data dosen, data ruang, data waktu, data matakuliah dan data mahasiswa.
- Dosen dapat meminta waktu mengajar tertentu yang diinginkan.
- Frekuensi pemakaian ruang.
- Frekuensi kelas dalam mengikuti perkuliahan.

Jika faktor-faktor diatas dapat terpenuhi berarti penjadwalan matakuliah sudah terbentuk dengan baik.

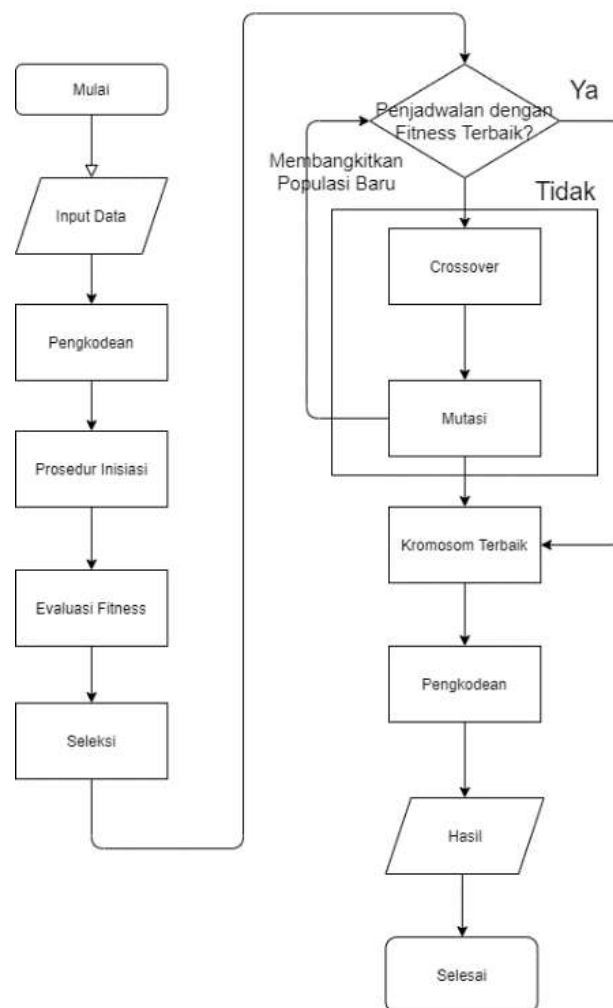
Perancangan analisis data dalam penelitian ini dilakukan dengan langkahlangkah sebagai berikut:

- Identifikasi Program Studi Sistem Informasi Universitas ARS.
- Identifikasi data dosen, data mahasiswa, data kelas, data matakuliah, data waktu dan data ruang.

- c. Identifikasi constraint-constraint yang ada di Program Studi Sistem Informasi Universitas ARS.

2. Perancangan Algoritma Genetika

Hal yang sangat penting dalam algoritma genetika adalah bagaimana untuk merepresentasikan solusi sedemikian rupa bahwa solusi tersebut layak untuk operasi genetik seperti *crossover* dan mutasi, bagaimana menentukan seberapa baik solusi bisa digunakan. Untuk itu harus dapat menghitung nilai fitness dari solusi tersebut.



Gambar 3. 5.
Flowchart Operasi Algoritma Genetika

3. Representasi Kromosom dan Populasi

Pembuatan representasi kromosom untuk penjadwalan perkuliahan bertujuan untuk memperoleh hasil penggabungan antara sks dan mata kuliah, dosen pengampu mata kuliah, semester dan kelas. Penentuan populasi awal adalah proses membangkitkan sejumlah kromosom secara acak (random). Kromosom menyatakan salah satu alternative solusi yang dimungkinkan. Kromosom dapat dikatakan sama dengan individu. Ukuran populasi tergantung pada masalah yang akan diselesaikan. Setelah ukuran populasi ditentukan, langkah selanjutnya adalah membangkitkan populasi awal dengan cara melakukan inisialisasi solusi yang dimungkinkan kedalam sejumlah kromosom.

Dalam tahap ini tabel dosen, tabel kelas dan tabel matakuliah lebih dulu dipasangkan, dari pasangan tabel tersebut akan dibangkitkan populasi awal dengan tabel ruang dan tabel waktu, sebagai berikut:

Tabel 3. 1 Perkuliahan

Id_Perkuliahan	Id_Dosen	Id_Kelas	Id_Matakuliah
1	1	1	2
2	2	1	1
3	3	1	3
4	4	2	5
.....			

Tabel 3. 2 Ruang

Id_Ruang	Namaruang
1	104
2	105
3	106
4	107
.....	

Tabel 3. 3 Waktu

Id_Waktu	Hari	Jamawal	Jamakhir
1	1	08.00	10.00
2	2	10.00	12.00
3	3	12.30	14.30
4	4	14.30	16.00
.....			

Tabel 3. 4 Populasi Awal
Gen

Populasi	Kromosom1	1	4	2	1	3	2	4	3	...	...
	Kromosom2	4	1	3	2	2	3	1	4	...	...



Keterangan: Merepresentasikan id tabel ruang

Merepresentasikan id tabel waktu

Kesimpulan kromosom1 pada tabel Populasi Awal diatas adalah:

- Id_perkuliahan 1 pada tabel perkuliahan akan dijadwalkan diruang 104 jam 14.30-16.00
- Id_perkuliahan 2 pada tabel perkuliahan akan dijadwalkan diruang 105 jam 08.00-10.00
- 3) Id_perkulian 3 pada tabel perkuliahan akan dijadwalkan diruang 106 jam 10.00-12.00
- Id_perkuliahan 4 pada tabel perkuliahan akan dijadwalkan diruang 107 jam 12.30-14.30

Kesimpulan kromosom2 sama halnya dengan kesimpulan kromosom1 hanya disesuaikan dengan gen nya masing-masing.

4. Fungsi Evaluasi

Suatu individu dievaluasi berdasarkan suatu fungsi tertentu sebagai ukuran performansinya. Didalam evolusi alam, individu yang bernilai fitness tinggi yang akan bertahan hidup. Sedangkan individu yang bernilai fitness rendah akan mati. Setiap constraint yang mempengaruhi nilai fitness memberi pengaruh yang berbeda terhadap nilai fitnessnya, tingkat pengaruh tersebut disebut dengan bobot. Semakin tinggi harga bobot suatu constraint akan mengurangi nilai fitness dari solusi tersebut. *Constraint* yang mempengaruhi evaluasi fitness terhadap solusi adalah sebagai berikut:

- Hard constraint* terdiri dari:

- (1) Dosen hanya dapat mengajar satu perkuliahan dalam satu waktu.

- (2) Ruang hanya dapat digunakan satu perkuliahan dalam satu waktu.
- (3) Kelas hanya dapat mengikuti satu perkuliahan dalam satu waktu.
- (4) Mata kuliah tidak boleh dijadwalkan melewati jam solat jum'at

Pada *hard constraint* tidak akan dihitung fungsi fitness karena jika ada gen yang melanggar *hard constraint* secara otomatis akan diganti dengan gen lain untuk mencari solusi lain.

b. *Soft constraint* terdiri dari:

- (1) Mata kuliah dengan ruangan khusus.
- (2) Dosen diberi tiga hari mengajar dalam seminggu.
- (3) Dosen hanya dapat mengajar tiga perkuliahan dalam sehari.
- (4) Kelas hanya dapat mengikuti tiga perkuliahan dalam sehari.

Berikut adalah tabel pembobotan *hard constrain* & *soft constraint* yang telah didiskusikan dengan kepala program studi:

Tabel 3. 5 Pembobotan

<i>Hard Consrtrain</i>	Kode Pembobotan	Nilai Pembobotan
Dosen hanya dapat mengajar satu perkuliahan dalam satu waktu.	B1	9
Ruang hanya dapat digunakan satu perkuliahan dalam satu waktu.	B2	9
Kelas hanya dapat mengikuti satu perkuliahan dalam satu waktu.	B3	7
Mata kuliah tidak boleh dijadwalkan melewati jam solat jum'at.	B4	9
<i>Soft Consrtrain</i>	Kode Pembobotan	Nilai Pembobotan
Mata kuliah dengan ruangan khusus	B5	3
Dosen diberi tiga hari mengajar dalam seminggu.	B6	4
Dosen hanya dapat mengajar tiga perkuliahan dalam sehari.	B7	4
Kelas hanya dapat mengikuti tiga perkuliahan dalam sehari.	B8	4

Rumus fitness yang digunakan adalah sebagai berikut:

$$\text{Fitness Kromosom1} = \frac{1}{1 + (\sum B_1 + \sum B_2 + \sum B_3 + \sum B_4 + \sum B_5 + \sum B_6 + \sum B_7 + \sum B_8)}$$

Rumus 3. 1. *Fitness Kromosom*

Keterangan: B1 sampai B8 adalah kode pembobotan.

5. Seleksi

Untuk mendapatkan solusi yang terbaik, harus dilakukan penyeleksian solusi yang memiliki nilai *fitness* yang tergolong rendah akan dihilangkan. Seleksi tersebut menggunakan metode *best fitness*. Solusi yang tersisa pada hasil seleksi ini dikenal dengan nama populasi induk.

6. Crossover

Suatu operasi *crossover* yang menggabungkan dua kromosom induk dan kemudian menciptakan kromosom baru yang dinamakan kromosom anak. Dengan kata lain, *crossover* merupakan suatu proses pertukaran nilai gen pada posisi gen yang sama dari kedua orang tua (induk). Berikut adalah contoh *crossover*:

Tabel 3. 6 Kromosom dalam *crossover*

		Gen									
Populasi	Induk1	1	4	2	1	3	2	4	3	...	...
	Induk2	4	1	3	2	2	3	1	4	...	...

Kromosom baru yang terbentuk:

Tabel 3. 7 Kromosom baru dalam *crossover*

		Gen									
Populasi	Anak1	1	4	3	2	3	2	1	4	...	...
	Anak2	4	1	2	1	2	3	4	3	...	...

7. Mutasi

Proses ini berperan untuk menggantikan gen yang hilang dari populasi akibat proses seleksi yang memungkinkan munculnya kembali gen yang tidak muncul pada inisialisasi populasi.

Berikut adalah contoh *crossover*:

Tabel 3. 8 Mutasi

Populasi	Gen									
	Sebelum	1	4	3	2	3	2	1	4	...
	Sesudah	1	2	3	2	3	4	1	4	...

8. Kondisi Selesai

Jika setelah beberapa generasi berturut-turut diperoleh nilai *fitness* tertinggi dari populasi maka proses algoritma akan dihentikan, sedangkan jika pada beberapa generasi berturut-turut nilai *fitness*nya belum mencapai kondisi tersebut, maka akan mengulang kembali proses/iterasi dari langkah evaluasi *fitness* terhadap populasi baru.

BAB IV

PEMBAHASAN

4.1 Mempersiapkan kebutuhan/*Requirement*

Dalam tahap ini iterasi dilakukan saat *user/client* memberikan penjelasan mengenai apa saja yang diperlukan, jika kebutuhan yang diperlukan tersebut mengalami perubahan, maka iterasi dalam tahap ini dilakukan. analisis Kebutuhan Fungsional Sistem yang dirancang memiliki kebutuhan fungsional seperti pada umumnya yaitu:

1. Dapat memberikan sebuah hasil yaitu penjadwalan mata pelajaran di sekolah Program Studi Sistem Informasi Universitas ARS.
2. Dapat memberikan kemudahan dalam pemeriksaan kurikulum/mata pelajaran di Program Studi Sistem Informasi Universitas ARS. Berikut merupakan tabel kebutuhan fungsional aplikasi penjadwalan mata kuliah di Program Studi Sistem Informasi Universitas ARS:

Tabel 4. 1. Kebutuhan fungsional aplikasi

No	Kebutuhan	Penjelasan
1	Input Dosen	Fungsi untuk melakukan input dosen.
2	Input Mata Kuliah	Fungsi untuk melakukan mata kuliah.
3	Input Data Pengampu	Fungsi untuk melakukan input Pengampu.
4	Input Data Ruang	Fungsi untuk melakukan input Ruang.
5	Input Data Jam	Fungsi untuk melakukan input Jam.
6	Input Data Hari	Fungsi untuk melakukan input hari.
7	Input Data Waktu Tidak Bersedia	Fungsi untuk melakukan input waktu yang tidak bersedia sesuai permintaan dosen.
8	Get Data Dosen	Fungsi untuk mengambil data dosen.
9	Get Data Mata Kuliah	Fungsi untuk mengambil data mata kuliah.
10	Proses Penjadwalan	Fungsi untuk proses penjadwalan menggunakan algoritma genetika.
11	Get Data Penjadwalan	Fungsi untuk <i>export</i> data penjadwalan.
12	<i>Log In</i>	Fungsi untuk log in petugas penjadwalan.

Berikut merupakan tabel kebutuhan non fungsional dalam perancangan aplikasi ini, diperlukan spesifikasi sebagai berikut:

Tabel 4. 2. Kebutuhan non fungsional aplikasi

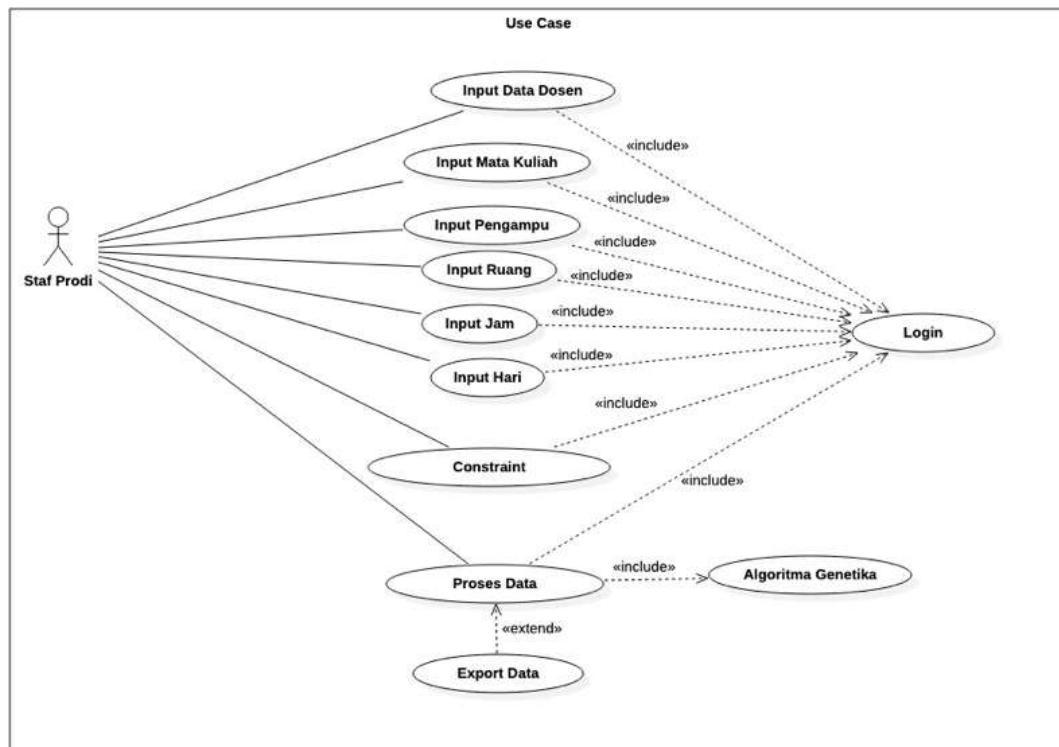
ID	Parameter	Kebutuhan
1	<i>Availability</i>	24 jam perhari, 7 hari perminggu
2	<i>Reliability</i>	Tidak pernah gagal
3	<i>Ergonomy</i>	<i>Interface</i> berbasis dekstop
4	<i>Memory</i>	Data yang disimpan tidak mudah hilang
5	<i>Response time</i>	Menampilkan hasil 5 detik
6	<i>Security</i>	Data tidak mudah hilang
7	Bahasa komunikasi	Semua tanya jawab harus dalam bahasa Indonesia

4.2 Perencanaan/*Planning*

Dalam tahap ini iterasi dilakukan ketika tahap sebelumnya ada perubahan, hal ini dikaksudkan untuk mengakomodasi kebutuhan yang sebelumnya diubah sehingga perencanaan yang sesuai dengan kebutuhan diperbaharui. Tahap perencanaan ini akan dijelaskan rancangan dalam aplikasi penjadwalan. Berdasarkan kebutuhan fungsional dan non fungsional yang telah dijelaskan di tabel 4.1 dan tabel 4,2, berikut merupakan gambaran perencanaan rancangan aplikasi yang akan dibangun, meliputi *Use Case*, *Activity Diagram*, *Sequence Diagram* dan *Class Diagram*:

1. *Use Case*

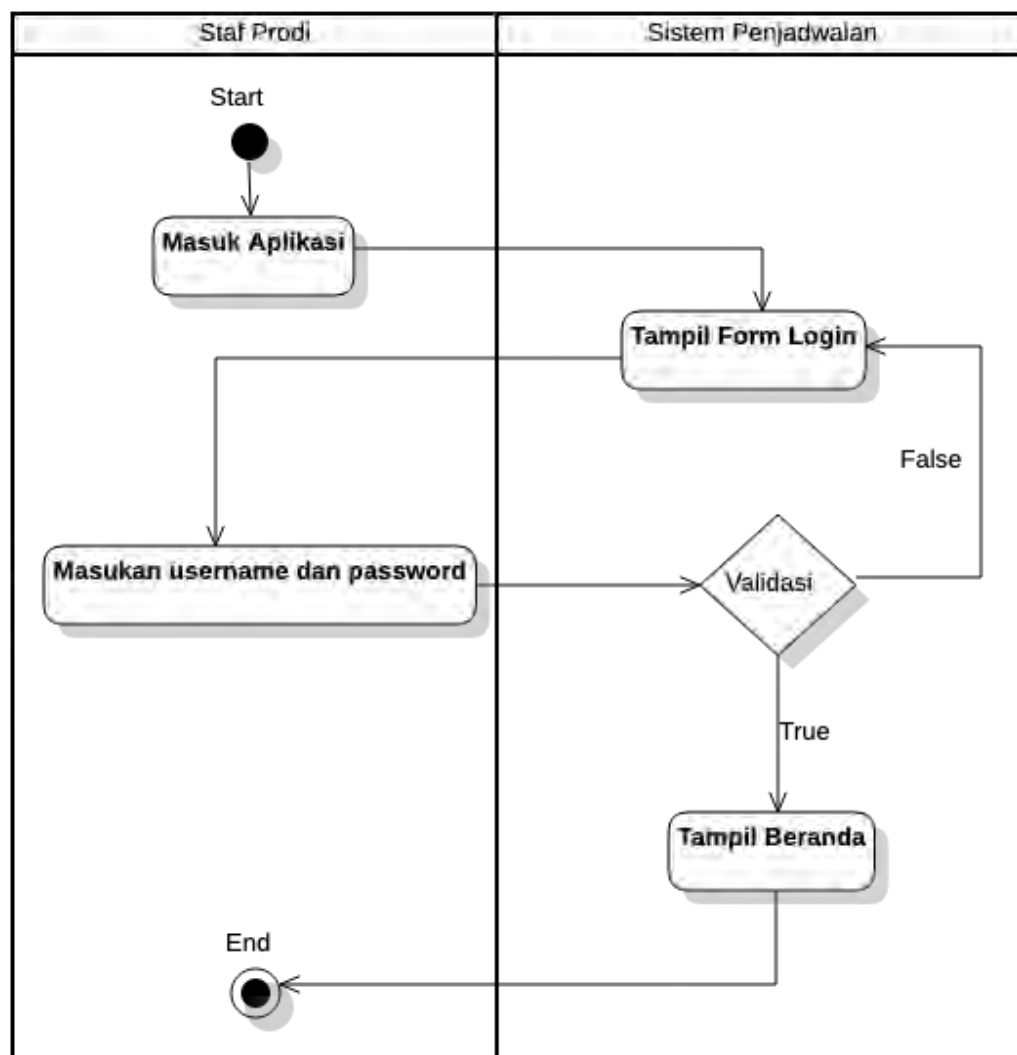
Use Case diagram menunjukan sebuah tongkat sederhana yang digunakan untuk mewakili aktor Berikut merupakan *Use Case* dengan menjelaskan menu apa saja yang terdapat dalam aplikasi penjadwalan sesuai dengan kebutuhan:



Gambar 4. 1. Use Case aplikasi penjadwalan kuliah

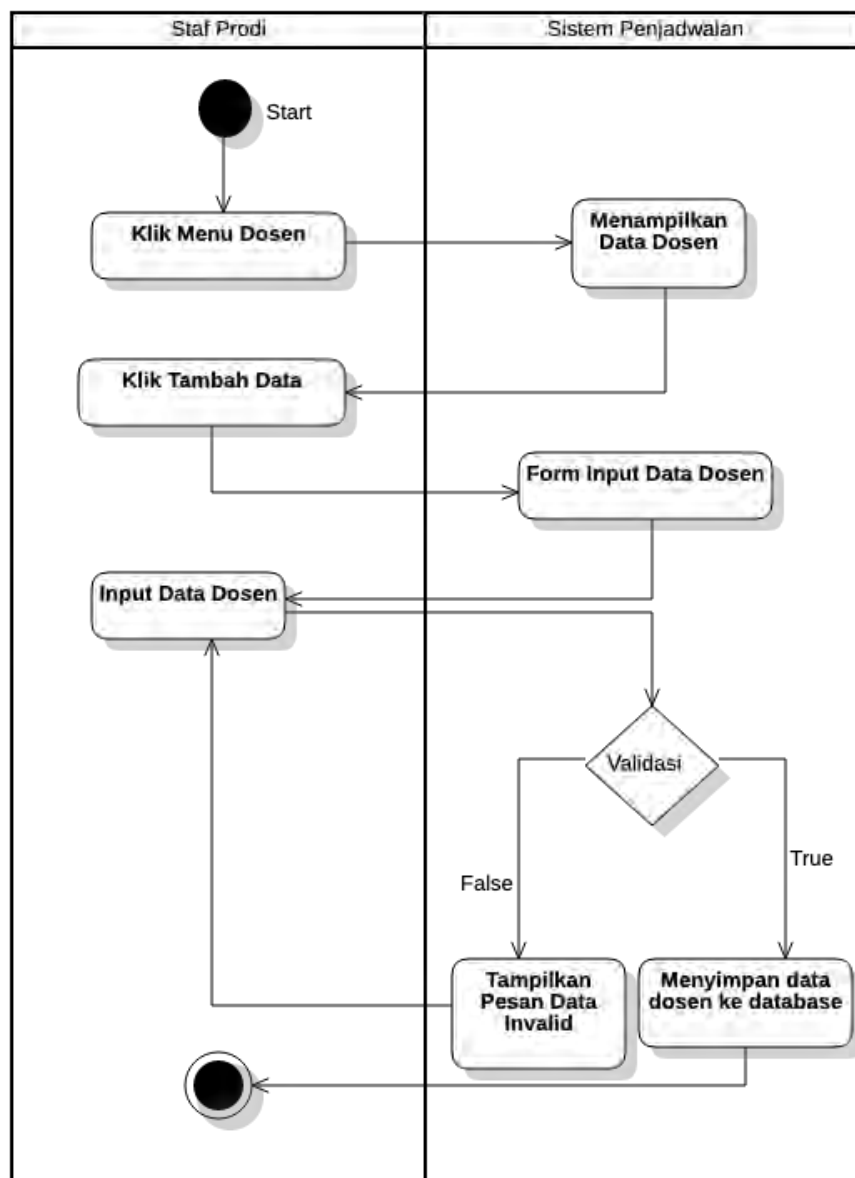
2. Activity Diagram

Activity Diagram merupakan yang gambaran beberapa aktivitas pengguna (atau sistem), orang yang melakukan setiap aktivitas, dan arus yang berurutan dari aktivitas. Gambar 4.2 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat *login*. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan *username* dan *password*. Jika *username* dan *password* sesuai, maka akan tampil menu beranda, sedangkan jika salah akan kembali ke form *login*.



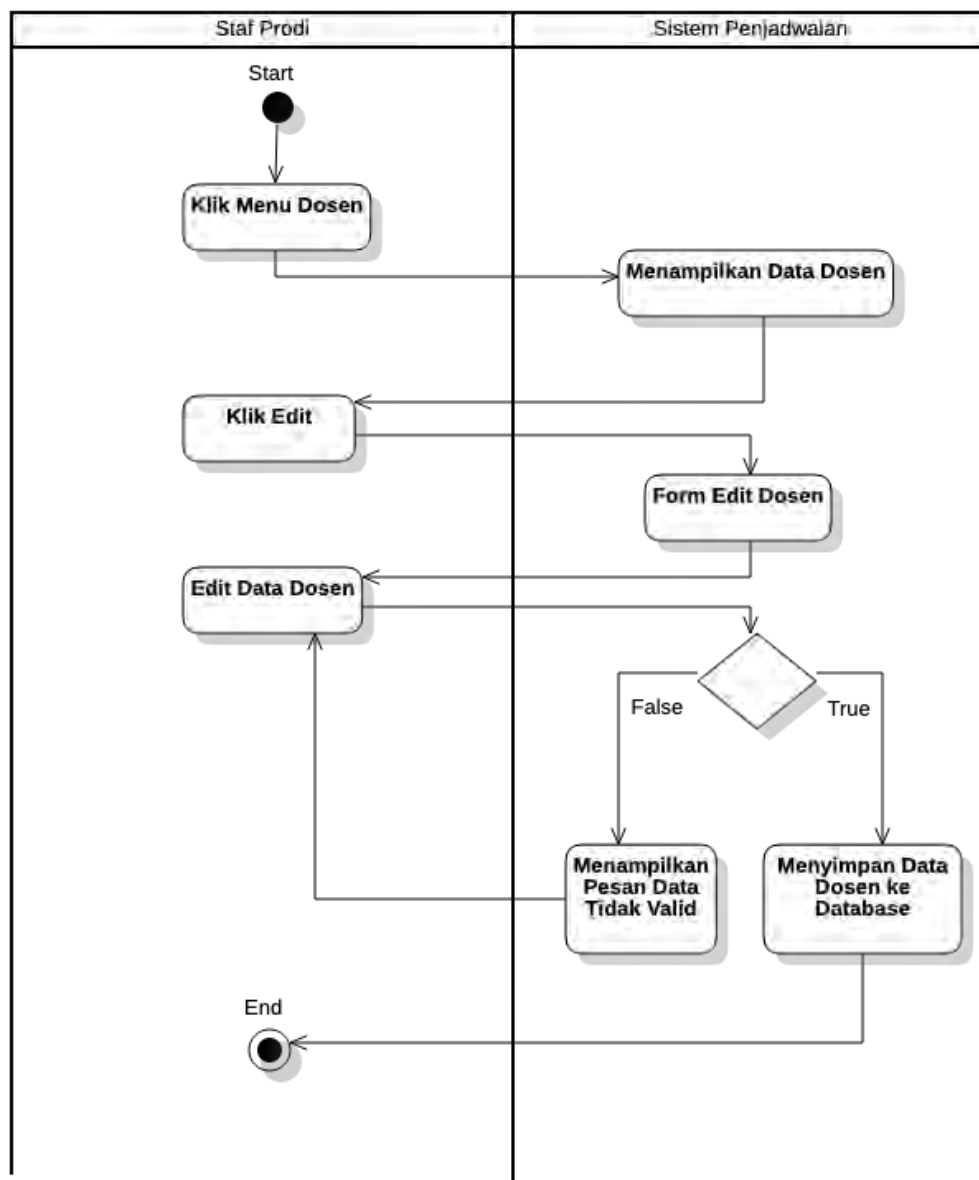
Gambar 4. 2. Activity Diagram *log in*

Gambar 4.3 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat input dosen. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukkan klik menu dosen, lalu akan muncul data dosen, dan akan ada *button* tambah data untuk menambah data dosen. Data yang diinputkan akan divalidasi, jika tidak valid maka sistem akan menampilkan pesan data tidak valid, sedangkan jika valid, sistem akan menyimpan data tersebut.



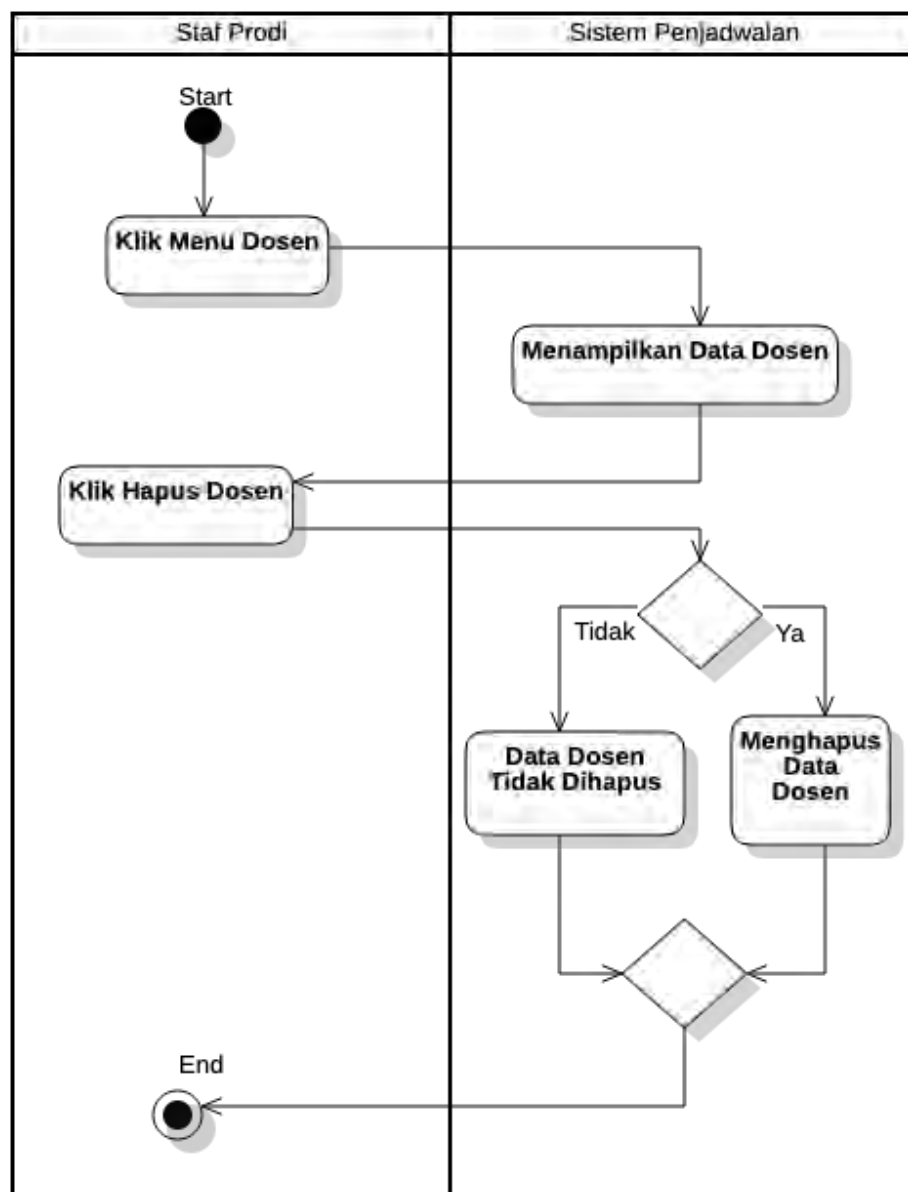
Gambar 4. 3. Activity Diagram input dosen

Gambar 4.4 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat edit dosen. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik menu dosen, lalu akan muncul data dosen, dan akan ada *button* edit untuk edit data dosen. Data yang diinputkan akan divalidasi, jika valid akan disimpan di database, sedangkan jika tidak akan kembali ke edit data dosen.



Gambar 4. 4. Activity Diagram edit dosen

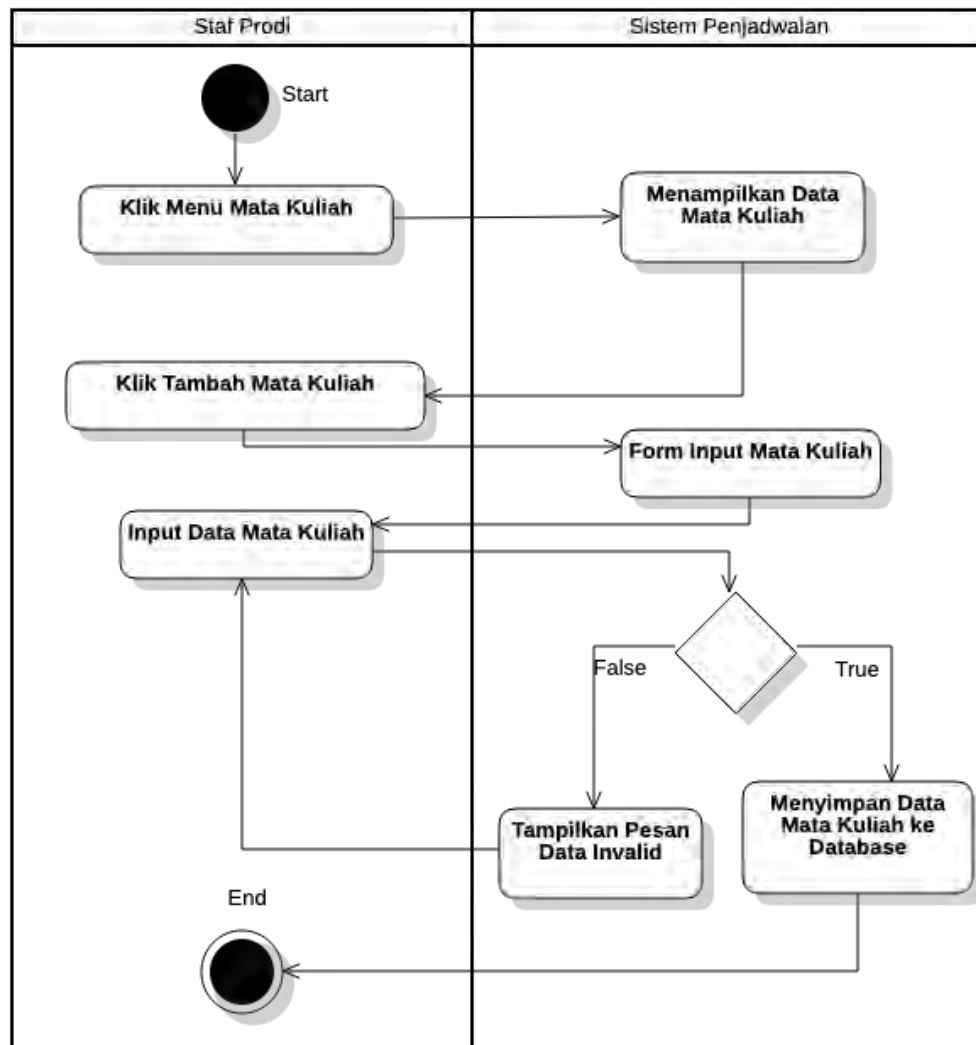
Gambar 4.5 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat hapus dosen. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik dosen, lalu akan muncul data dosen, dan akan ada *button* hapus untuk hapus data dosen. Akan ada notifikasi apakah yakin menghapus data tersebut atau tidak, jika ya, maka data akan terhapus.



Gambar 4. 5. Activity Diagram hapus dosen

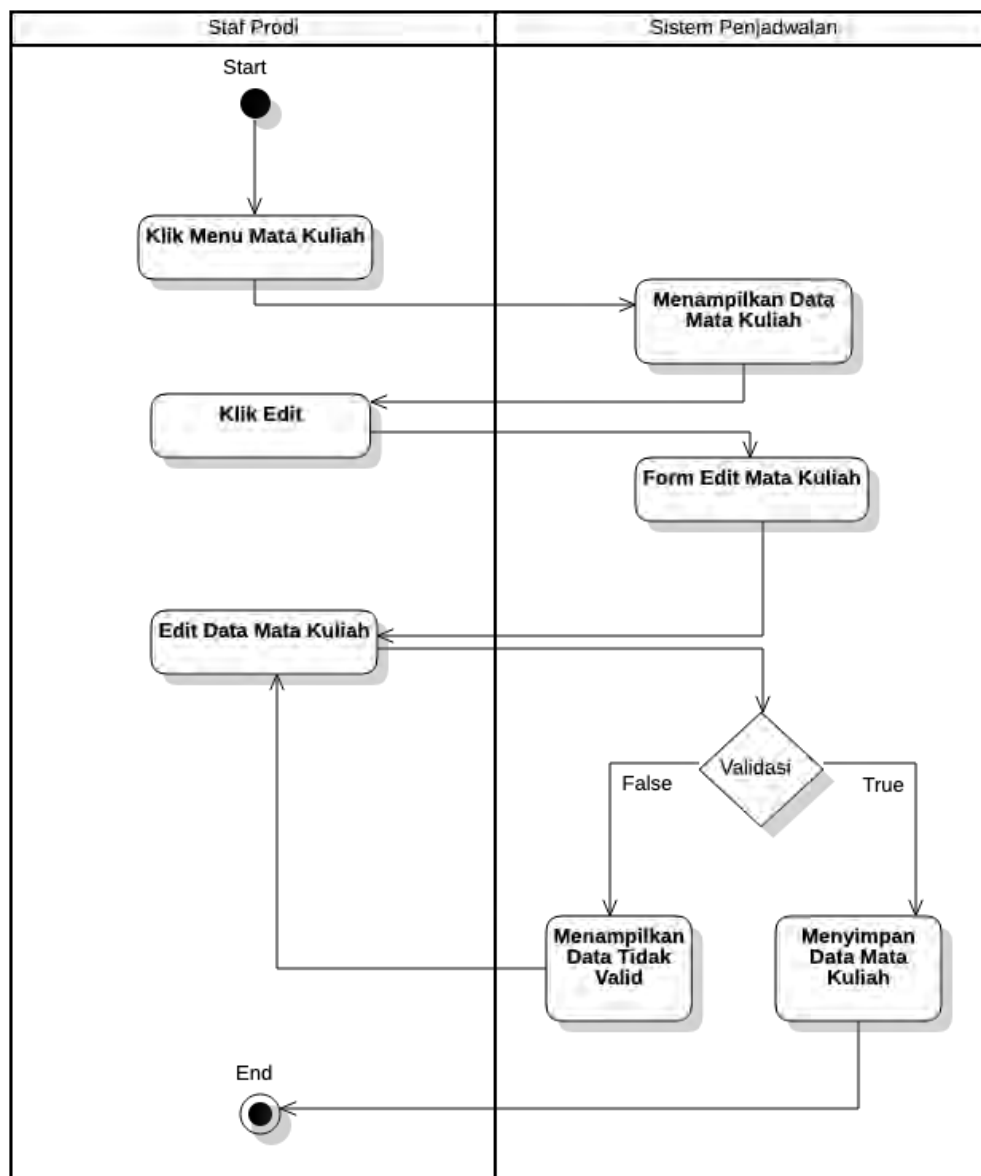
Gambar 4.6 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat input mata kuliah. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik menu mata kuliah, lalu akan muncul data mata kuliah, dan akan ada *button* tambah data untuk menambah data mata kuliah. Data yang diinputkan akan divalidasi, jika tidak valid maka

sistem akan menampilkan pesan data tidak valid, sedangkan jika valid, sistem akan menyimpan data tersebut.



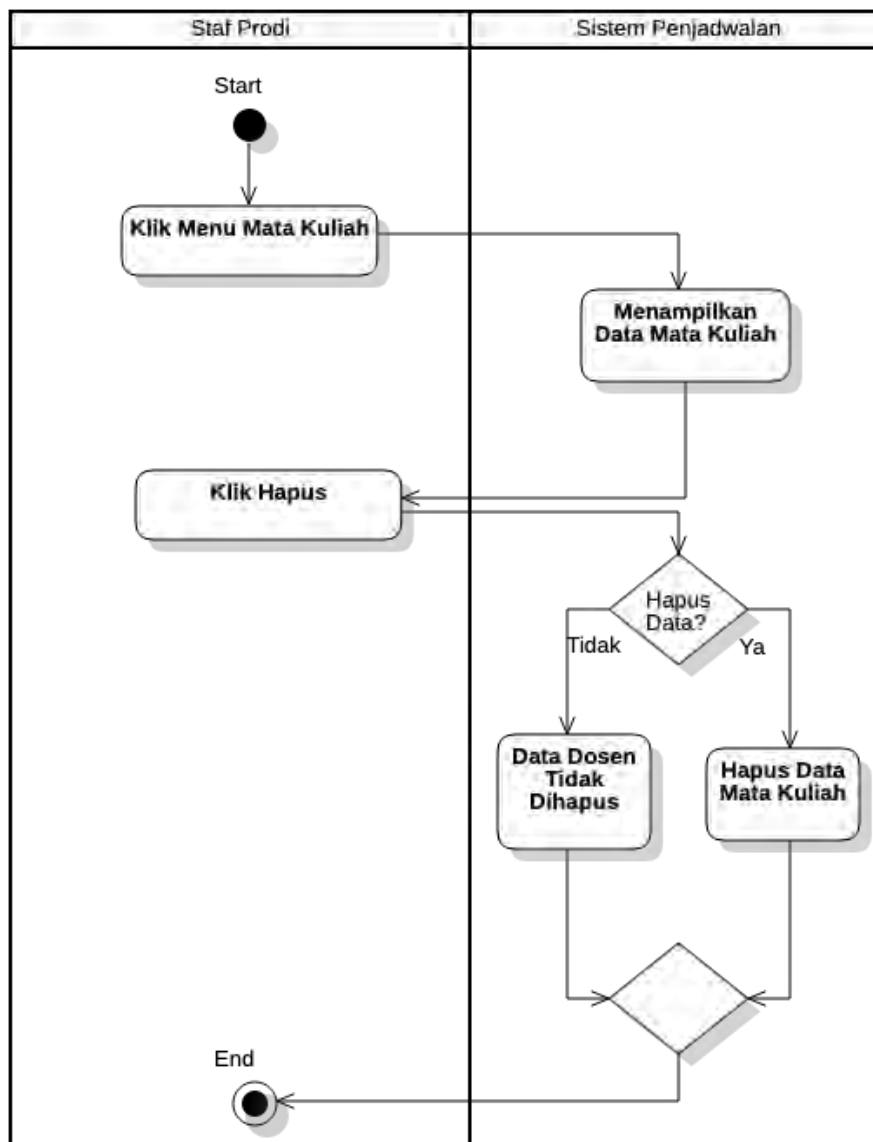
Gambar 4. 6. Activity Diagram input mata kuliah

Gambar 4.7 merupakan *activity* diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat edit mata kuliah. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik mata kuliah, lalu akan muncul data mata kuliah, dan akan ada *button* edit untuk edit data mata kuliah. Data yang diinputkan akan divalidasi, jika valid akan disimpan di database, sedangkan jika tidak akan kembali ke edit data mata kuliah.



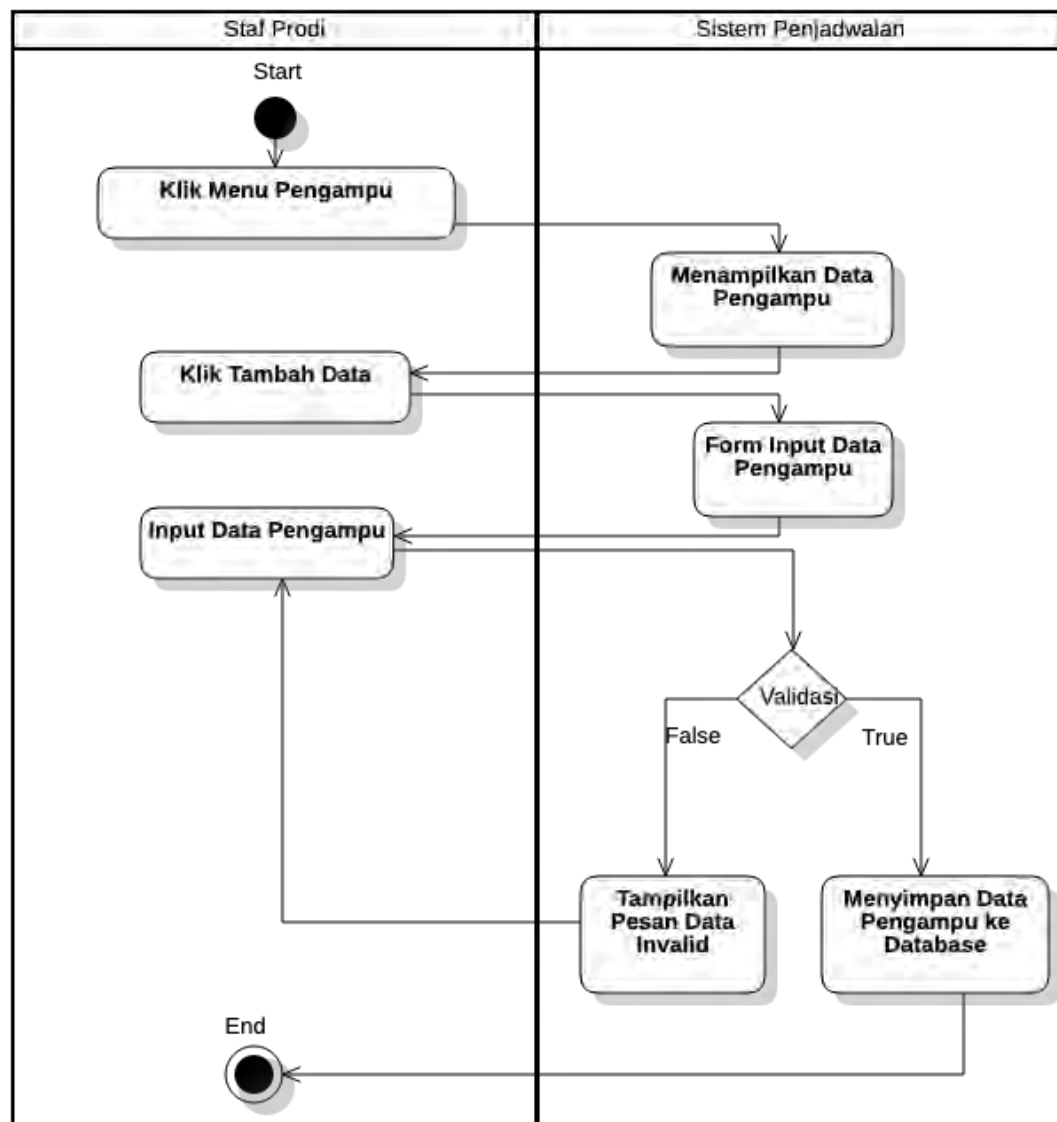
Gambar 4. 7. Activity Diagram edit mata kuliah

Gambar 4.8 merupakan *activity* diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat hapus mata kuliah. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik mata kuliah, lalu akan muncul data mata kuliah, dan akan ada *button* hapus untuk hapus data mata kuliah. Akan ada notifikasi apakah yakin menghapus data tersebut atau tidak, jika ya, maka data akan terhapus.



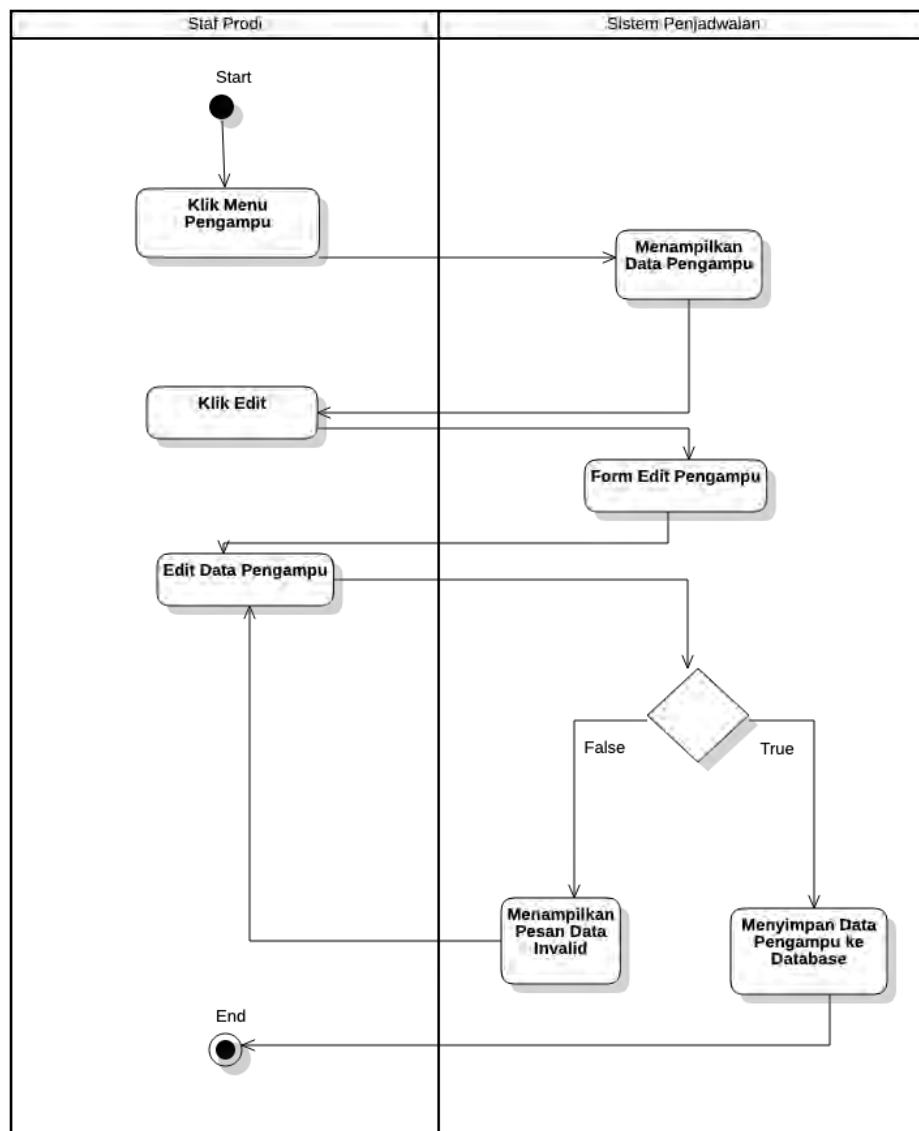
Gambar 4. 8. Activity Diagram hapus mata kuliah

Gambar 4.9 merupakan *activity* diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat input pengampu. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik menu mata pengampu, lalu akan muncul data pengampu, dan akan ada *button* tambah data untuk menambah data pengampu. Data yang diinputkan akan divalidasi, jika tidak valid maka sistem akan menampilkan pesan data tidak valid, sedangkan jika valid, sistem akan menyimpan data tersebut.



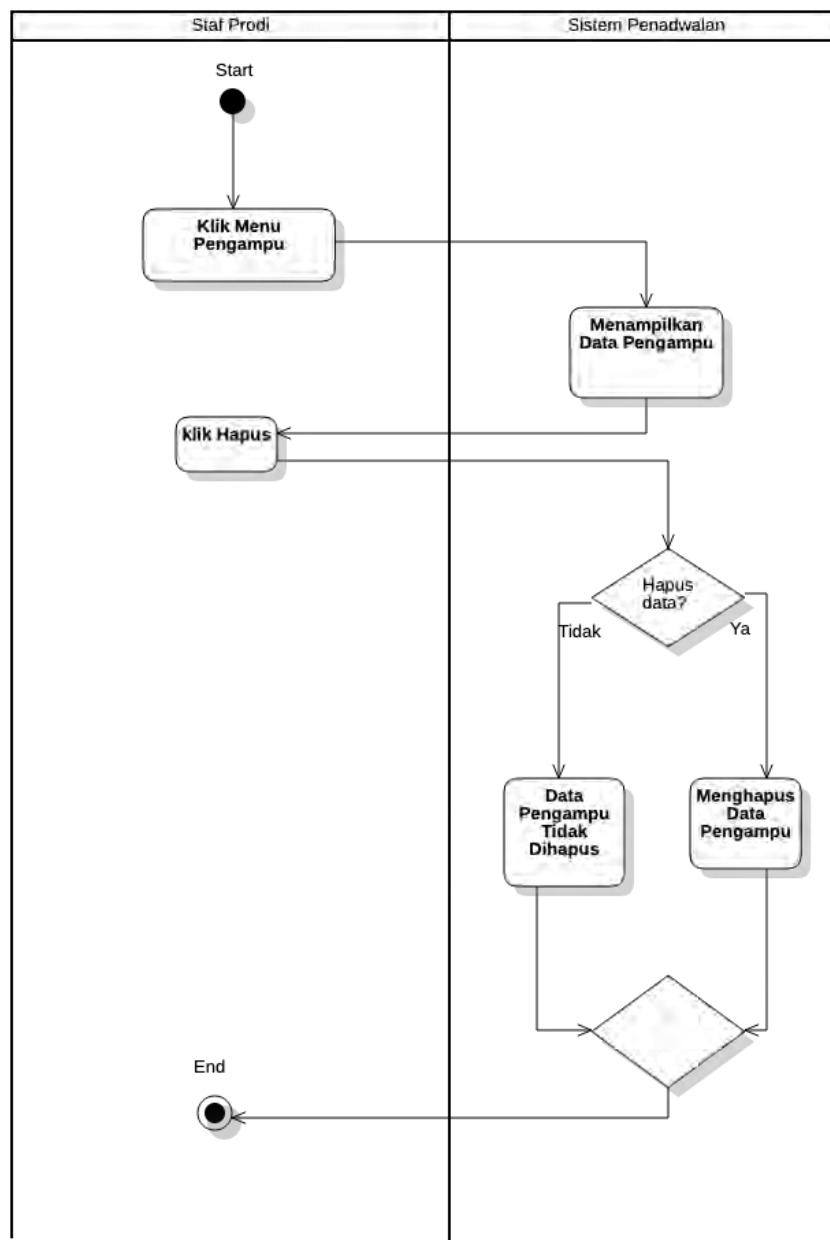
Gambar 4. 9. Activity Diagram input pengampu

Gambar 4.10 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat edit pengampu. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukkan klik pengampu, lalu akan muncul data pengampu, dan akan ada *button* edit untuk edit data pengampu. Data yang diinputkan akan divalidasi, jika valid akan disimpan di database, sedangkan jika tidak akan kembali ke edit data pengampu.



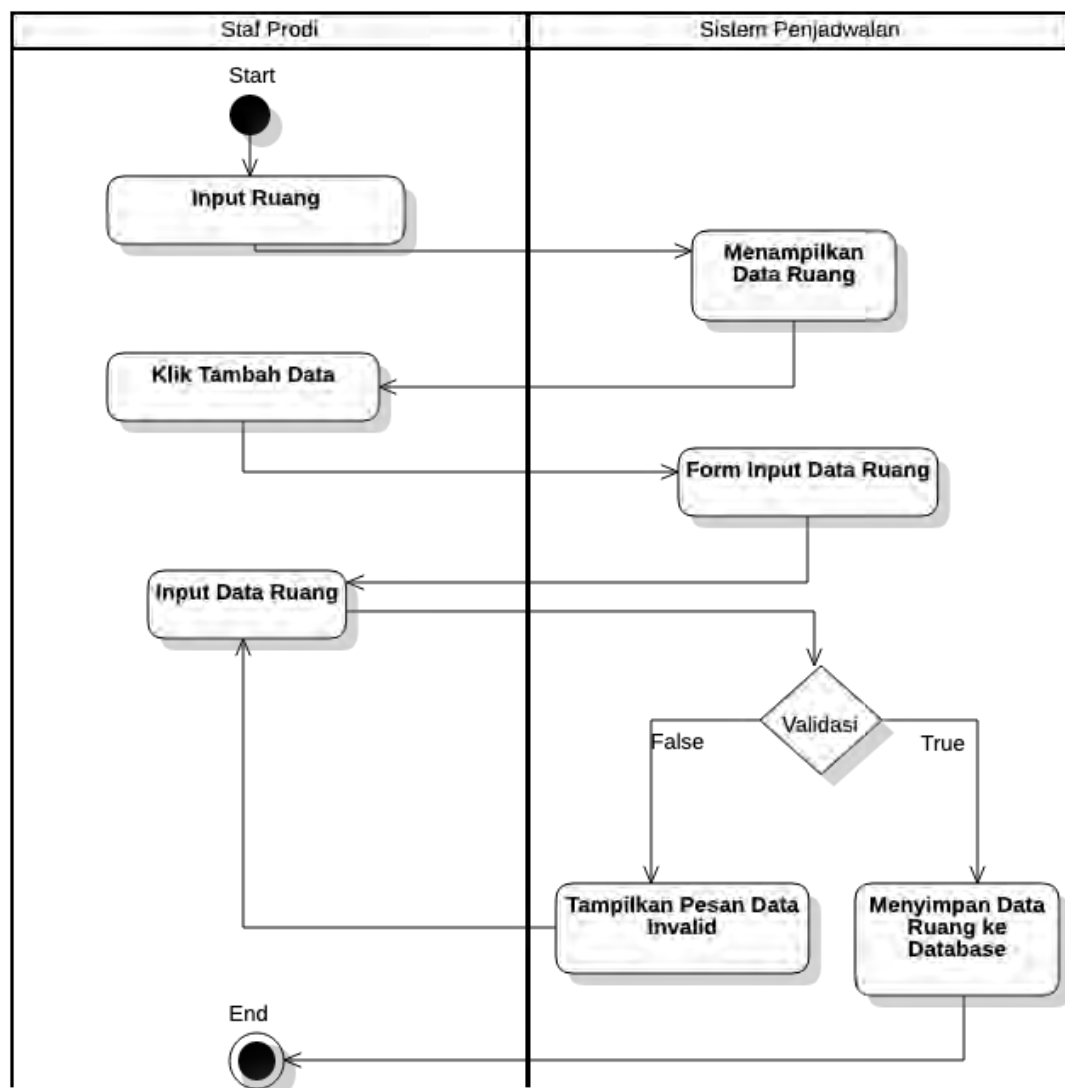
Gambar 4. 10. Activity Diagram edit pengampu

Gambar 4.11 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat hapus pengampu. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik pengampu, lalu akan muncul data pengampu, dan akan ada *button* hapus untuk hapus data pengampu. Akan ada notifikasi apakah yakin menghapus data tersebut atau tidak, jika ya, maka data akan terhapus.



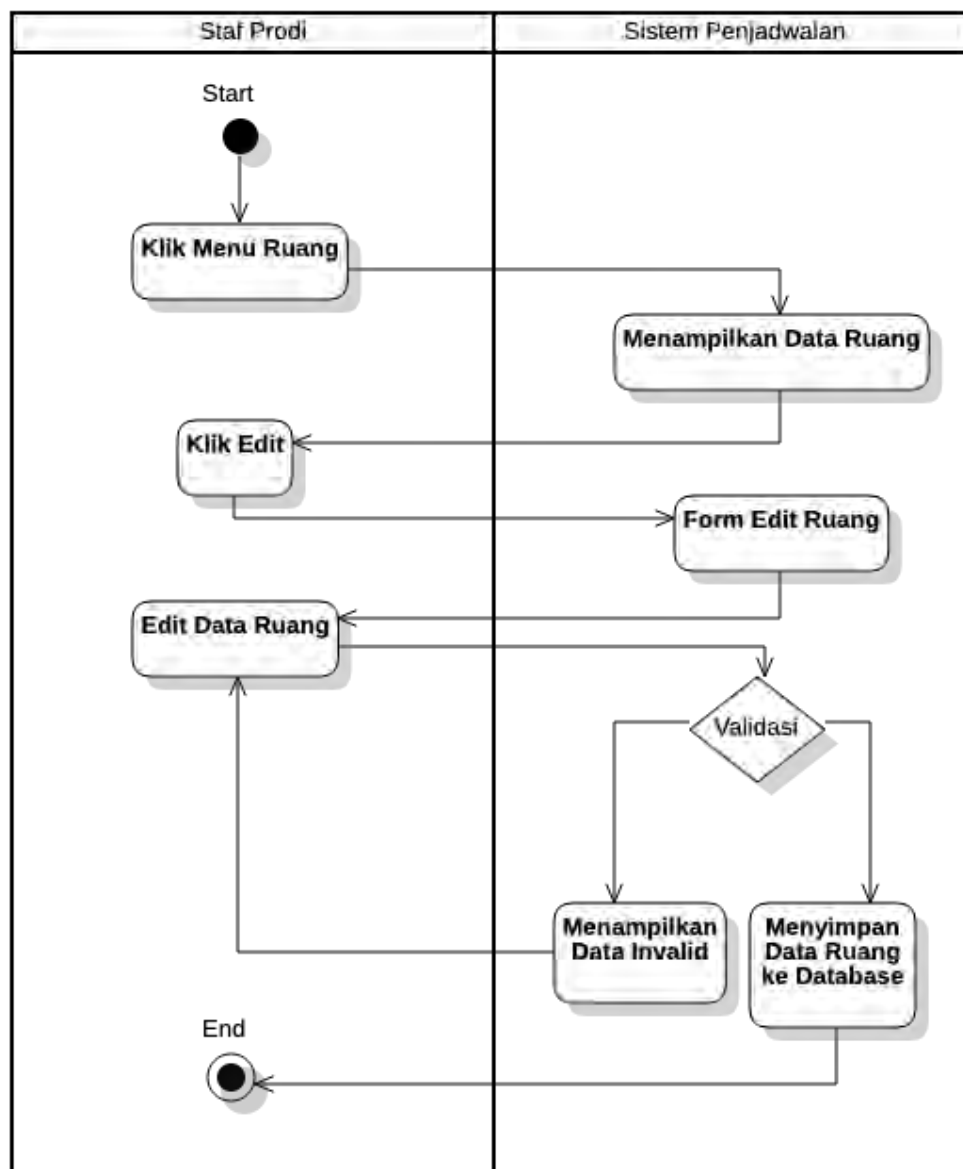
Gambar 4. 11. Activity Diagram hapus pengampu

Gambar 4.12 merupakan *activity* diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat input ruang. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik menu ruang, lalu akan muncul data ruang, dan akan ada *button* tambah data untuk menambah data ruang. Data yang diinputkan akan divalidasi, jika tidak valid maka sistem akan menampilkan pesan data tidak valid, sedangkan jika valid, sistem akan menyimpan data tersebut.



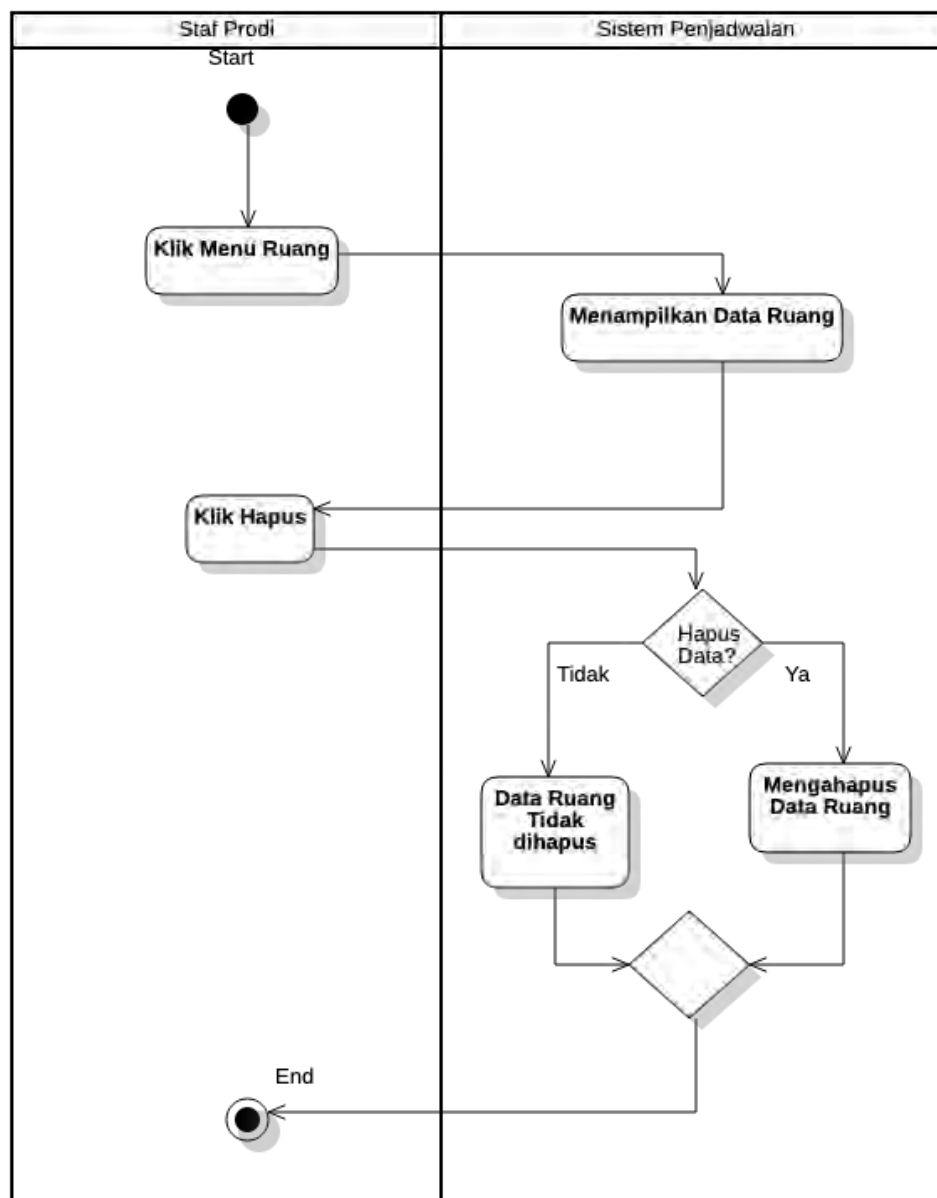
Gambar 4. 12. Activity Diagram input ruang

Gambar 4.13 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat proses edit ruang. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukkan klik ruang, lalu akan muncul data ruang, dan akan ada *button* edit untuk edit data ruang. Data yang diinputkan akan divalidasi, jika valid akan disimpan di database, sedangkan jika tidak akan kembali ke edit data ruang.



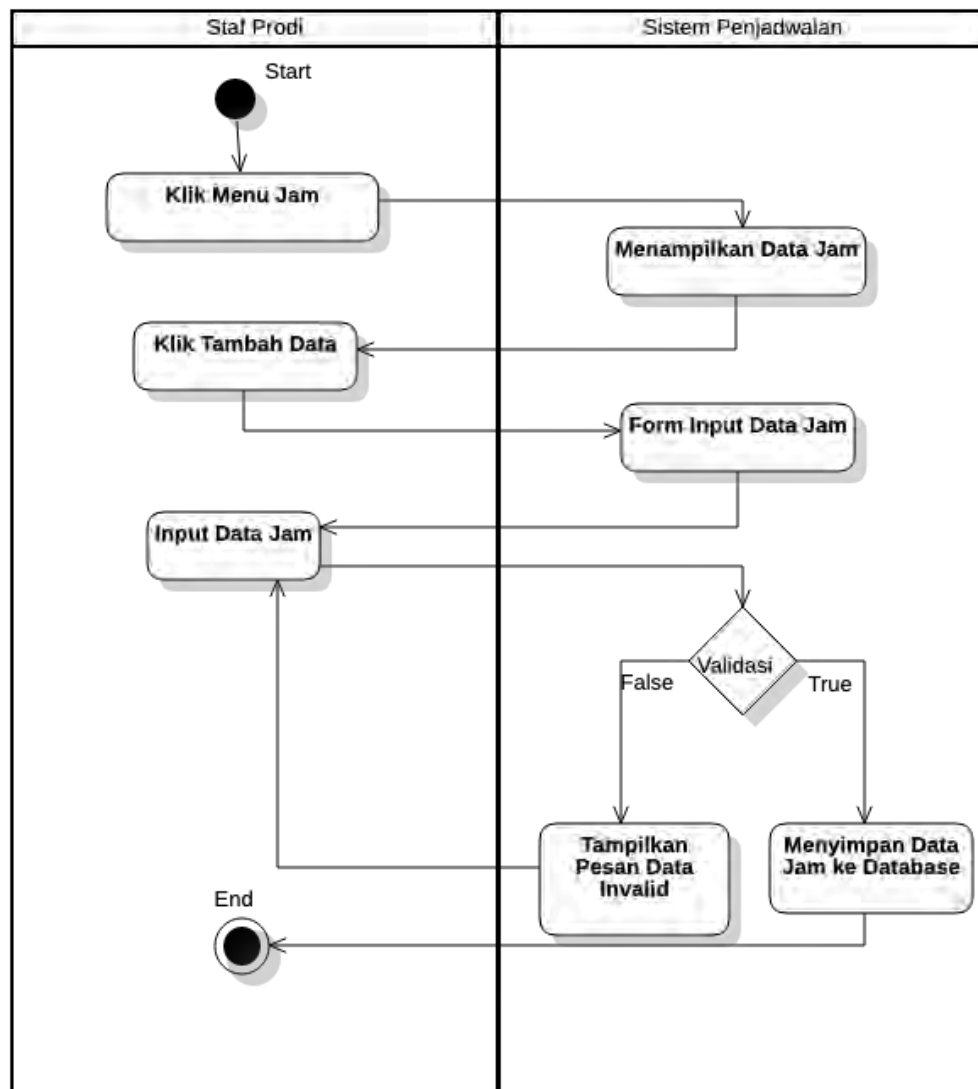
Gambar 4. 13. Activity Diagram edit ruang

Gambar 4.14 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat proses hapus ruang. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik ruang, lalu akan muncul data ruang, dan akan ada *button* hapus untuk hapus data ruang. Akan ada notifikasi apakah yakin menghapus data tersebut atau tidak, jika ya, maka data akan terhapus.



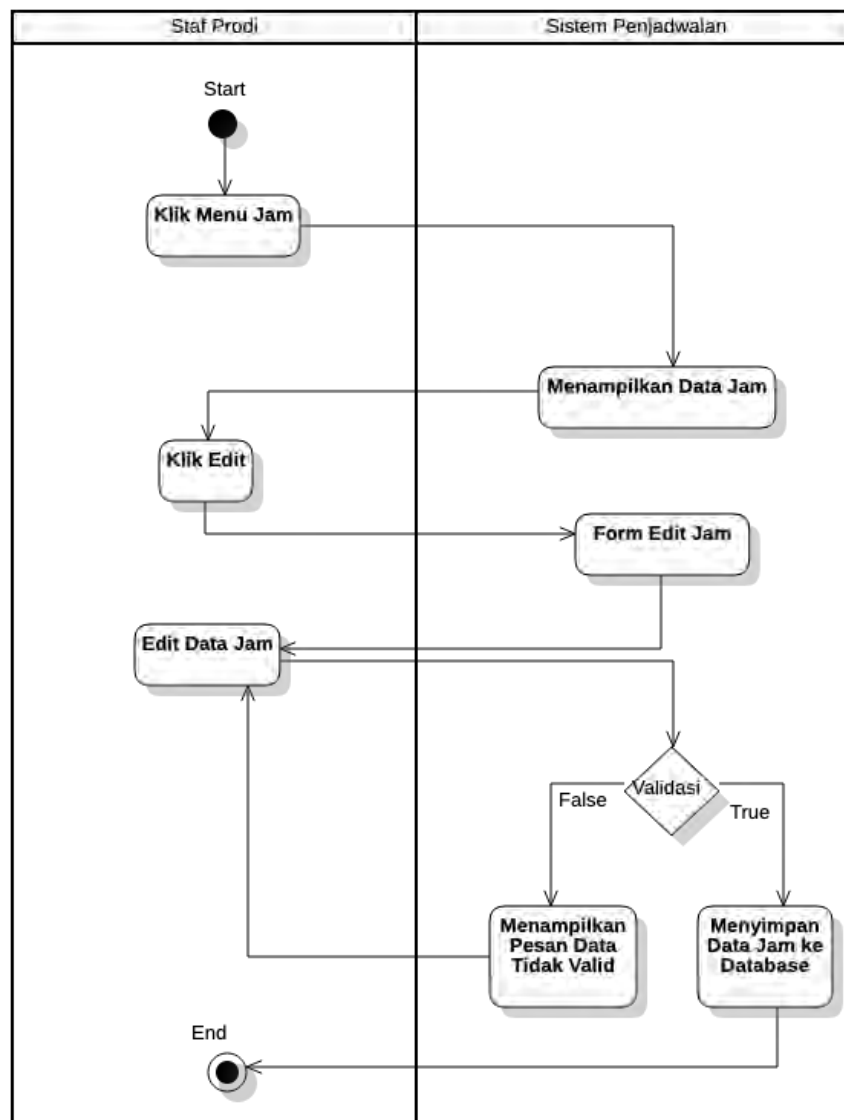
Gambar 4. 14. Activity Diagram hapus ruang

Gambar 4.15 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat proses input jam. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik menu jam, lalu akan muncul data jam, dan akan ada *button* tambah data untuk menambah data jam. Data yang diinputkan akan divalidasi, jika tidak valid maka sistem akan menampilkan pesan data tidak valid, sedangkan jika valid, sistem akan menyimpan data tersebut.



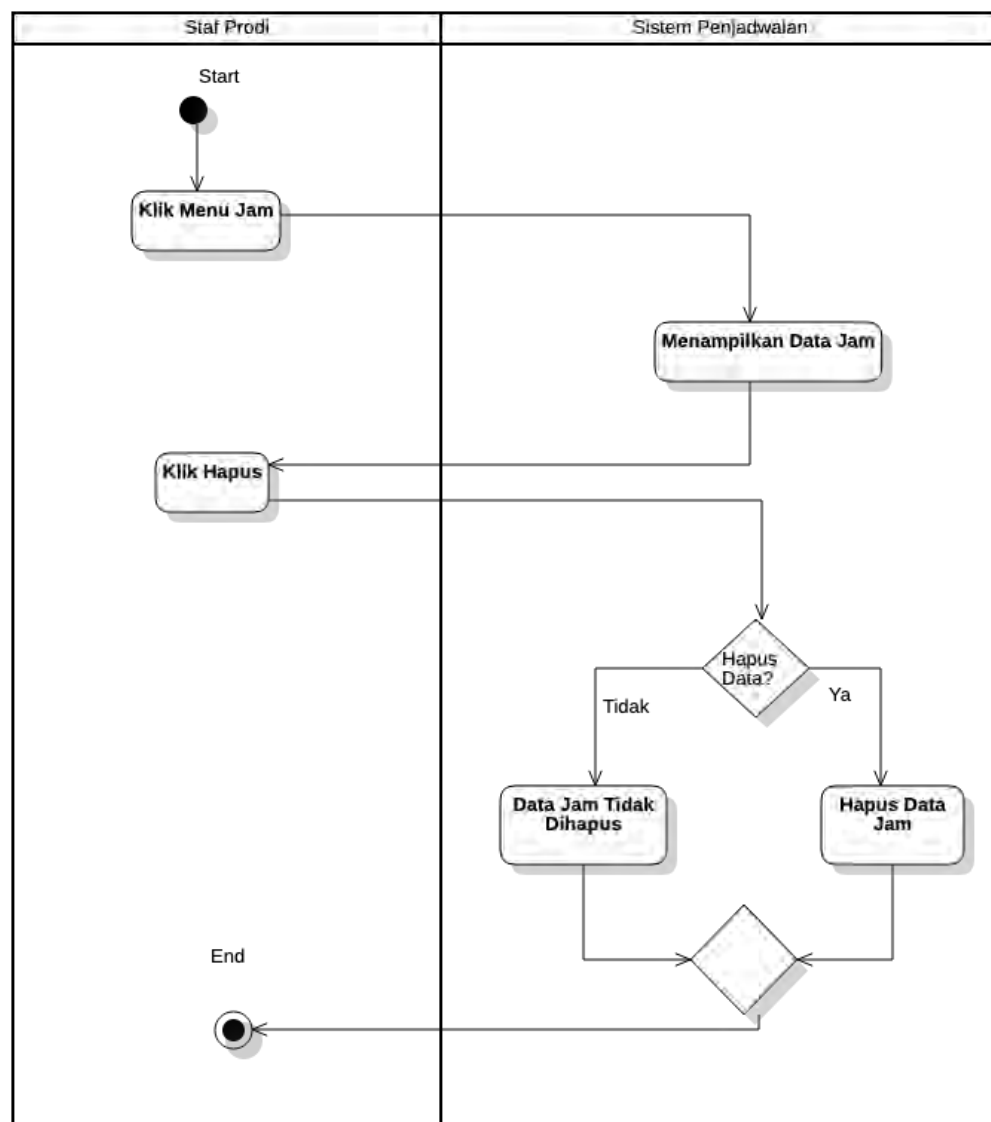
Gambar 4. 15. Activity Diagram input jam

Gambar 4.16 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat proses edit jam. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik jam, lalu akan muncul data jam, dan akan ada *button* edit untuk edit data jam. Data yang diinputkan akan divalidasi, jika valid akan disimpan di database, sedangkan jika tidak akan kembali ke edit data jam.



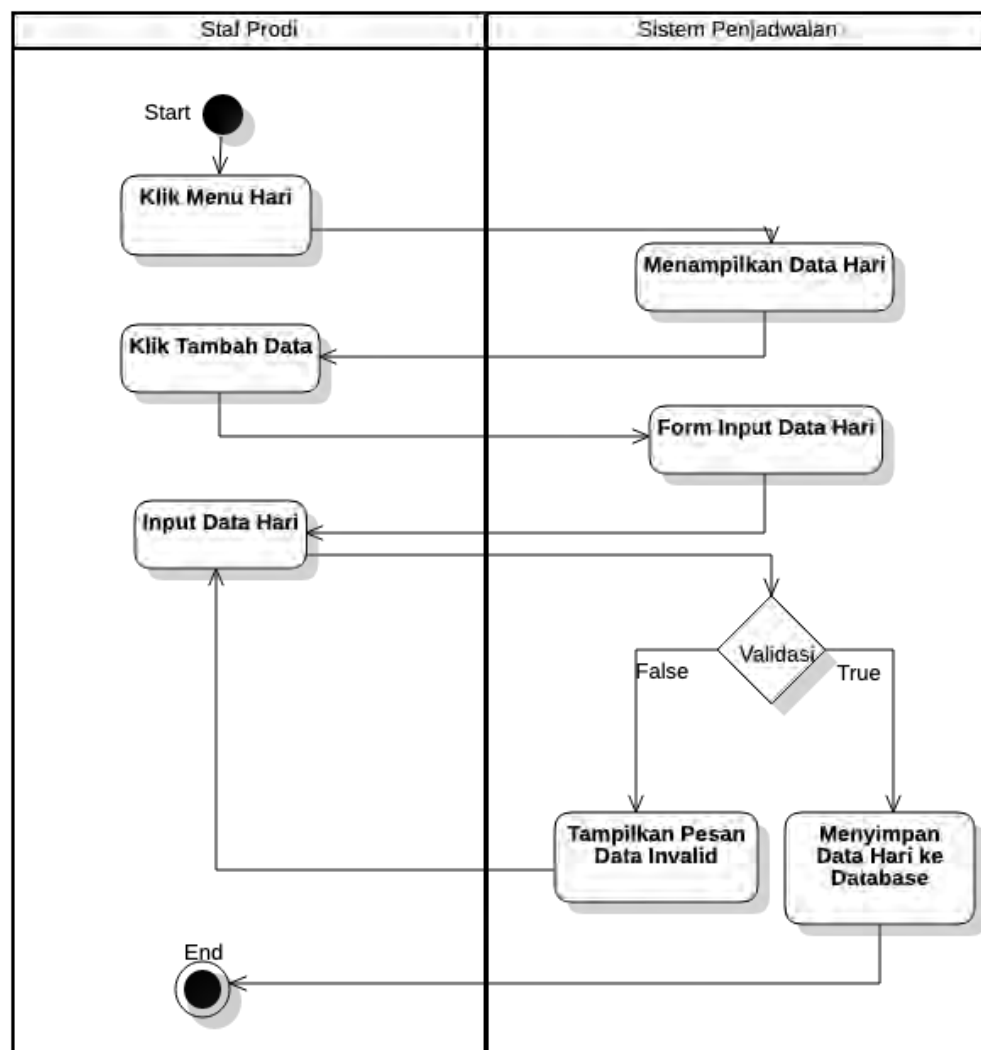
Gambar 4. 16. Activity Diagram edit jam

Gambar 4.17 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat proses hapus jam. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik jam, lalu akan muncul data jam, dan akan ada *button* hapus untuk hapus data jam. Akan ada notifikasi apakah yakin menghapus data tersebut atau tidak, jika ya, maka data akan terhapus.



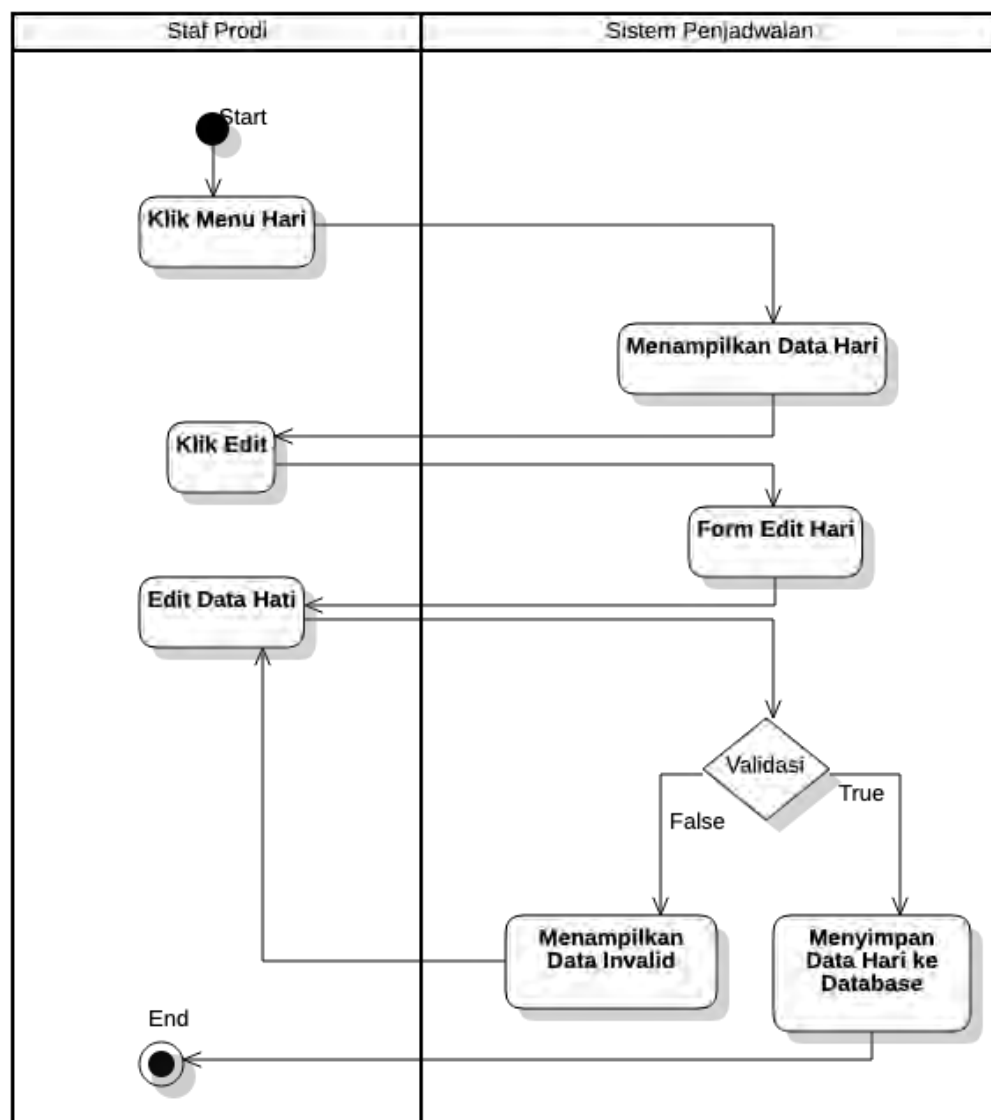
Gambar 4. 17. Activity Diagram hapus jam

Gambar 4.18 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat proses input hari. Aktor yang terlibat dalam *activity diagram* ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukkan klik menu hari, lalu akan muncul data hari, dan akan ada *button* tambah data untuk menambah data hari. Data yang diinputkan akan divalidasi, jika tidak valid maka sistem akan menampilkan pesan data tidak valid, sedangkan jika valid, sistem akan menyimpan data tersebut.



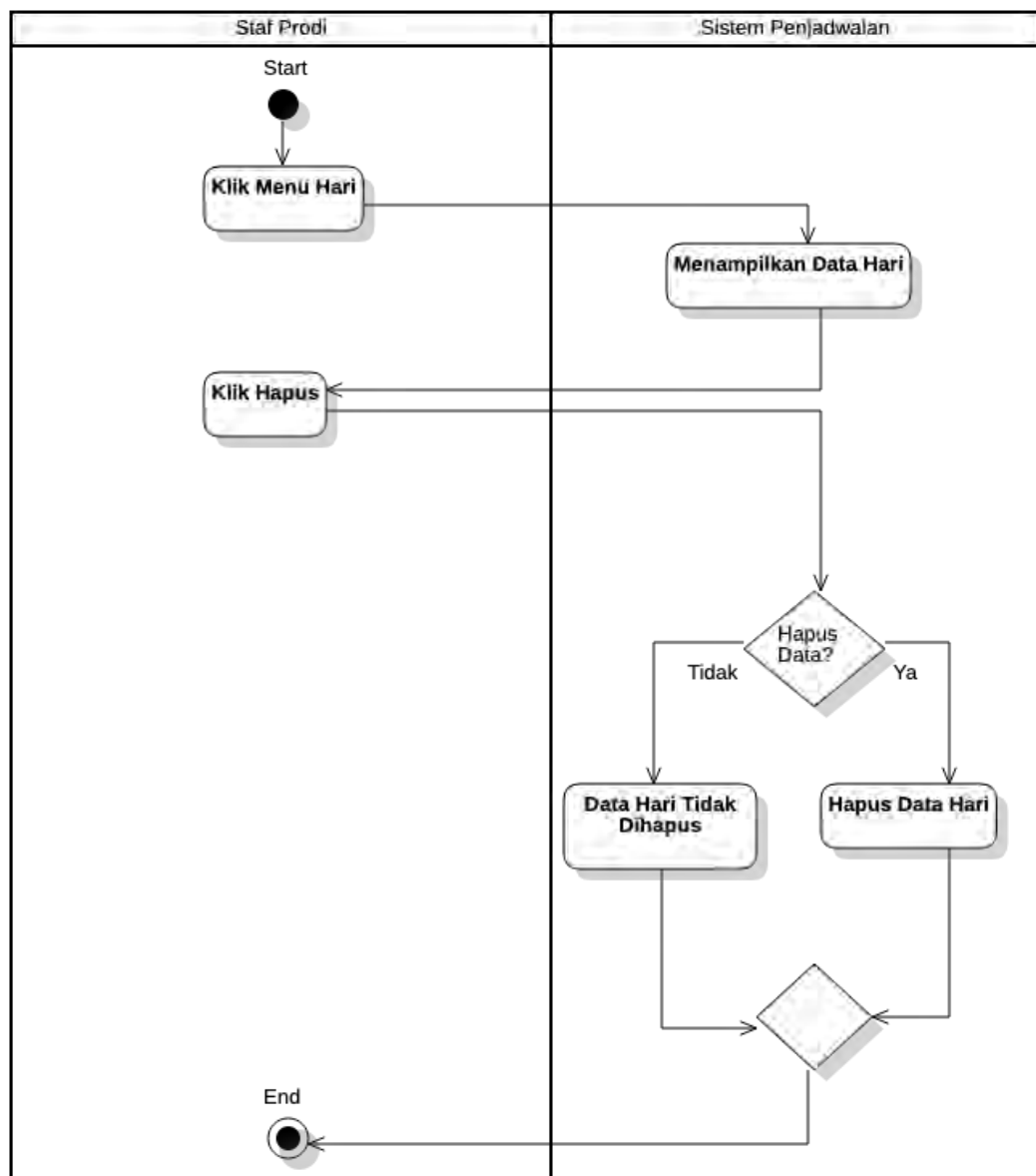
Gambar 4. 18. Activity Diagram input hari

Gambar 4.19 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat proses edit hari. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik hari, lalu akan muncul data hari, dan akan ada *button* edit untuk edit data hari. Data yang diinputkan akan divalidasi, jika valid akan disimpan di database, sedangkan jika tidak akan kembali ke edit data hari.



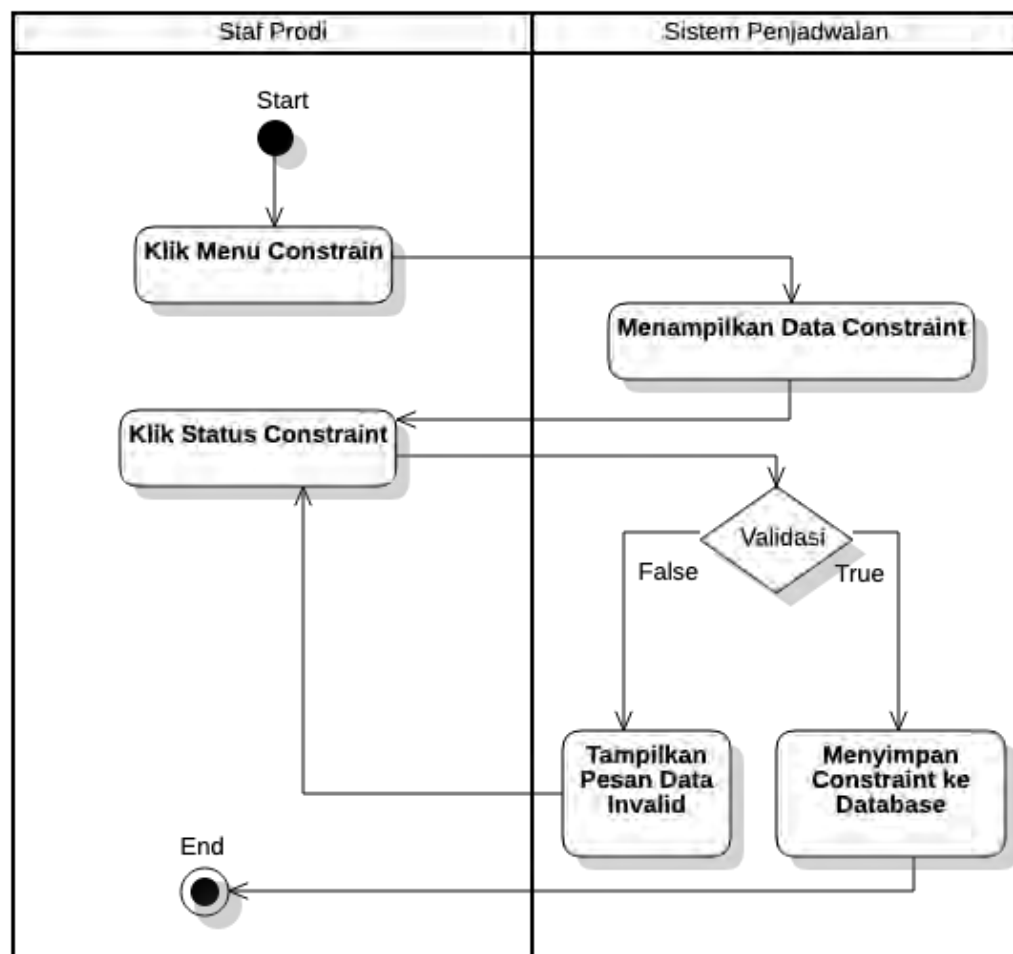
Gambar 4. 19. Activity Diagram edit hari

Gambar 4.20 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat proses hapus hari. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukan klik hari, lalu akan muncul data hari, dan akan ada *button* hapus untuk hapus data hari. Akan ada notifikasi apakah yakin menghapus data tersebut atau tidak, jika ya, maka data akan terhapus.



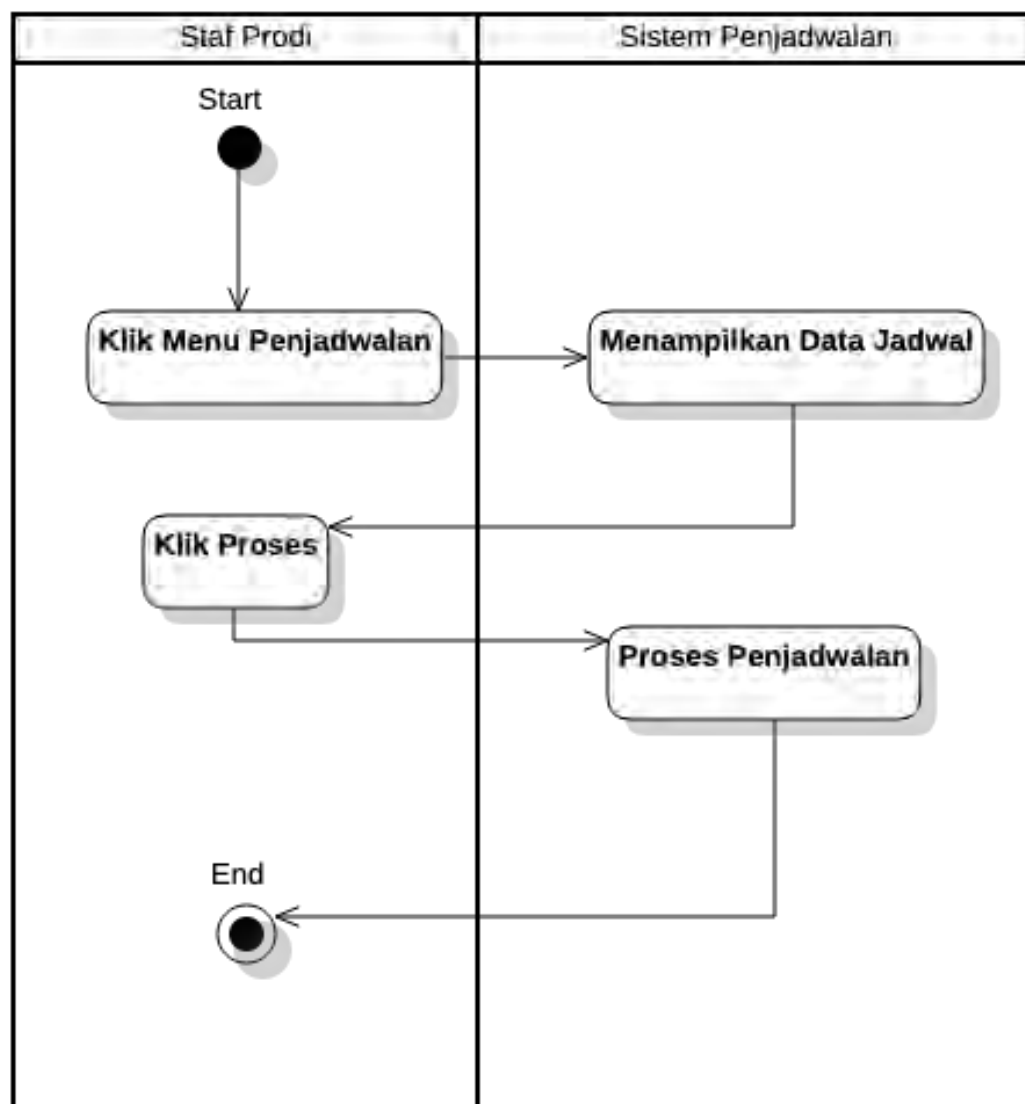
Gambar 4. 20. Activity Diagram hapus hari

Gambar 4.21 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat proses input *constraint*. Aktor yang terlibat dalam *activity diagram* ini adalah staf prodi dimulai dengan memasuki aplikasi dan klik menu *constraint*, lalu akan muncul data *constraint*, aktor akan memilih *constraint* masing dosen dan ceklis sesuai jam yang tertera.



Gambar 4. 21. Activity Diagram constraint

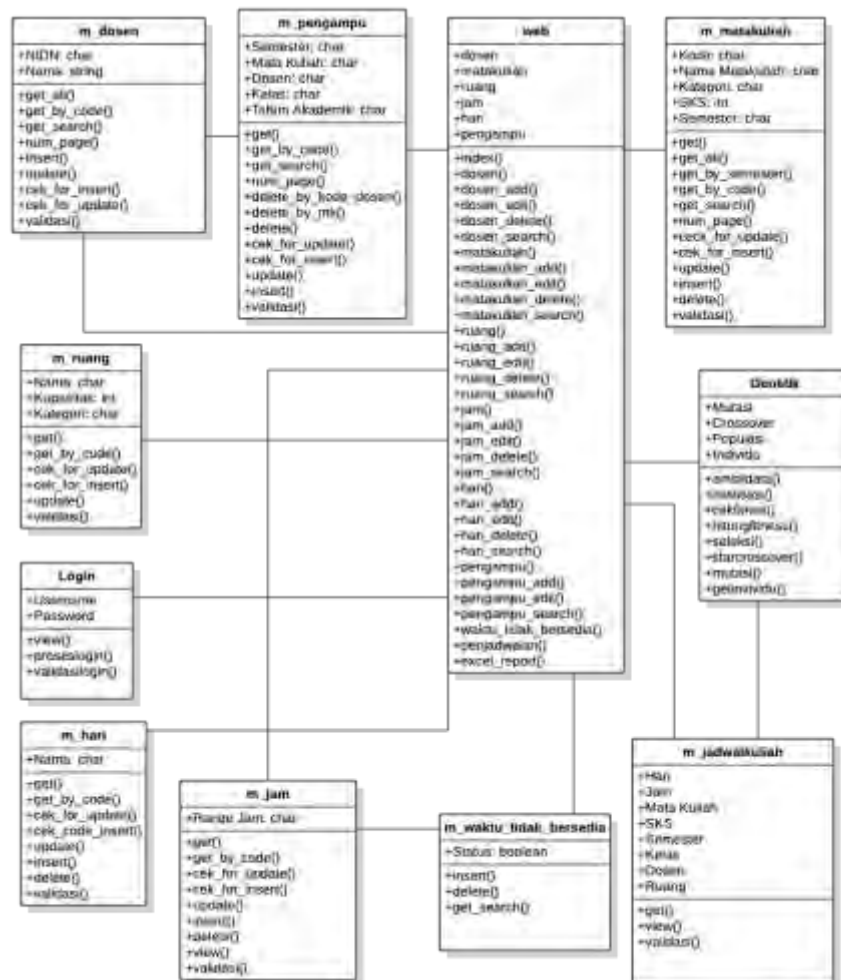
Gambar 4.22 merupakan activity diagram dengan memperlihatkan urutan aktifitas proses pada sistem saat proses penjadwalan. Aktor yang terlibat dalam *activity* diagram ini adalah staf prodi dimulai dengan memasuki aplikasi dan memasukkan klik penjadwalan, lalu akan muncul data jadwal, dan akan ada *button* proses untuk proses penjadwalan.



Gambar 4. 22. Activity Diagram penjadwalan

3. Class diagram

Use Case diagram menunjukkan sebuah tongkat sederhana yang digunakan untuk mewakili aktor gambar 4.23 merupakan Use Case dengan menjelaskan menu apa saja yang terdapat dalam aplikasi penjadwalan sesuai dengan kebutuhan:



Gambar 4. 23. Class diagram

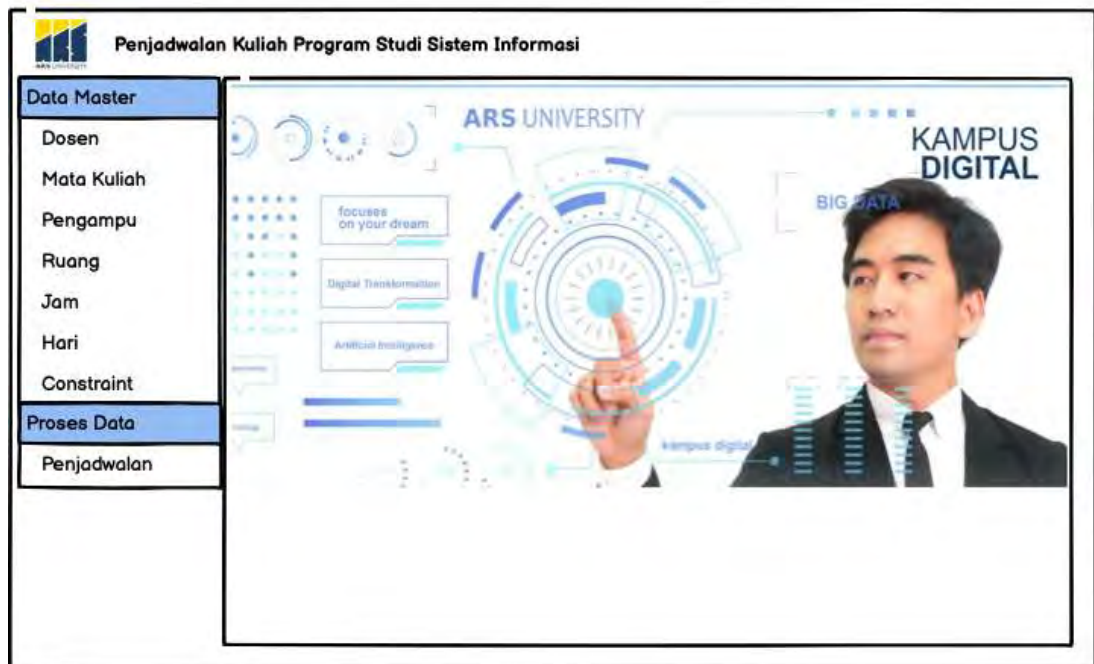
4.3 Desain

Iterasi dalam tahap ini dilakukan saat kebutuhan dan perencanaan mengalami perubahan, perubahan desain akan menyesuaikan dengan tahapan kebutuhan dan perencanaan. Dalam tahap ini akan ditampilkan desain aplikasi yang akan digunakan dalam penjadwalan algoritma genetika di program studi sistem informasi di Universitas ARS. Gambar 4.24 merupakan desain *log in* dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berguna untuk validasi *user*, sehingga pengguna diwajibkan untuk mengisi username dan password.

The image shows a login form centered on a white background. At the top center is the ARS University logo, which consists of a stylized 'ARS' in blue and yellow, with 'ARS UNIVERSITY' written in small black letters below it. Below the logo is a rectangular box containing three input fields: a 'username' field, a 'Password' field, and a 'Login' button. The 'Login' button is a small rectangle with the word 'Login' in black text.

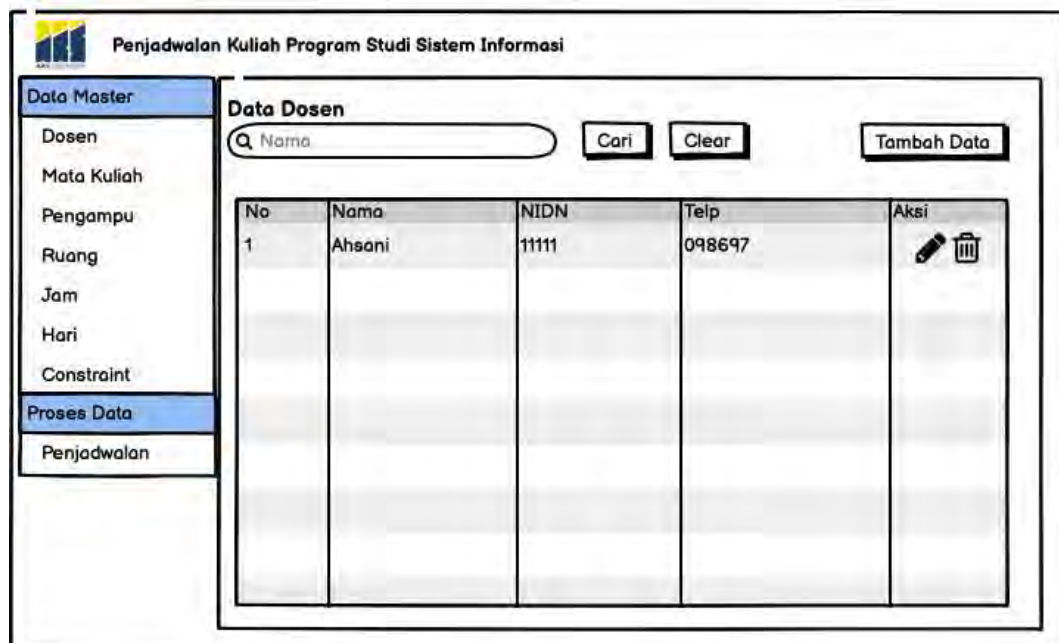
Gambar 4. 24. Desain *log in*

Gambar 4.25 merupakan desain beranda dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini merupakan beranda dari aplikasi penjadwalan, terlihat disebelah kiri merupakan menu yang tersedia dalam aplikasi penjadwalan. Halaman ini akan terlihat pertama kali setelah pengguna melakukan *login*.



Gambar 4. 25. Desain beranda

Gambar 4.26 merupakan desain menu dosen dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk mencari dan menambah data dosen, pengguna bisa mencari data yang sudah diinputkan yang terlihat dalam tabel yang ada di halaman menu dosen.



Gambar 4. 26. Desain menu dosen

Gambar 4.27 merupakan tampilan *input* dosen dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk menambah data dosen, pengguna bisa menambahkan data dosen dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

The screenshot displays a web application interface for course scheduling. On the left, a vertical menu contains the following items: 'Data Master' (highlighted in blue), 'Dosen', 'Mata Kuliah', 'Pengampu', 'Ruang', 'Jam', 'Hari', 'Constraint', 'Proses Data' (highlighted in blue), and 'Penjadwalan'. The main content area is titled 'Data Dosen' and contains a form with four text input fields labeled 'NIDN', 'NAMA', 'ALAMAT', and 'NOMOR TELEPON'. At the bottom of the form are two buttons: 'Save' and 'Cancel'.

Gambar 4. 27. Desain menu input dosen

Gambar 4.28 merupakan tampilan edit dosen dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk edit data dosen, pengguna bisa edit data dosen dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Dosen
- Mata Kuliah
- Pengampu
- Ruang
- Jam
- Hari
- Constraint

Proses Data

- Penjadwalan

Modul Edit Dosen

NIDN
1111

NAMA
Ahsani

ALAMAT
Bandung

NOMOR TELEPON
1234

Save Cancel

Gambar 4. 28. Desain menu edit dosen

Gambar 4.29 merupakan tampilan hapus dosen dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk menghapus data dosen, pengguna bisa menghapus data dosen klik *icon delete*. Akan ada notifikasi yang memberikan pesan apakah yakin ingin menghapus data ini, jika ya maka data akan terhapus.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Dosen
- Mata Kuliah
- Pengampu
- Ruang
- Jam
- Hari
- Constraint

Proses Data

- Penjadwalan

Data Dosen

Q Nama Cari Clear Tambah Data

No	Nama	NIDN	Telp	Aksi
1	Ahsani	1111	1234	

Anda yakin ingin menghapus data ini?

Cancel OK

Gambar 4. 29. Desain menu hapus dosen

Gambar 4.30 merupakan tampilan menu mata kuliah dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk mencari mata kuliah yang sudah diinputkan, pengguna bisa mencari data yang sudah diinputkan yang terlihat dalam tabel yang ada di halaman menu mata kuliah.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Dosen
- Mata Kuliah
- Pengampu
- Ruang
- Jam
- Hari
- Constraint

Proses Data

- Penjadwalan

Data Matakuliah

Q Nama

No	Kode MK	Nama	SKS	Jenis	Aksi
1	SI01	Bahasa Indonesia	2	TEORI	

Gambar 4. 30. Desain menu mata kuliah

Gambar 4.31 merupakan tampilan input mata kuliah dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk menambah data mata kuliah, pengguna bisa menambahkan data mata kuliah dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Dosen
- Mata Kuliah
- Pengampu
- Ruang
- Jam
- Hari
- Constraint

Proses Data

- Penjadwalan

Tambah Data Mata Kuliah

Kode Mata Kuliah

NAMA

Kategori

SKS

Semester

Save Cancel

Gambar 4. 31. Desain input mata kuliah

Gambar 4.32 merupakan tampilan edit mata kuliah dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk edit data mata kuliah, pengguna bisa edit data mata kuliah dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Dosen
- Mata Kuliah
- Pengampu
- Ruang
- Jam
- Hari
- Constraint

Proses Data

- Penjadwalan

Modul Edit Data Mata Kuliah

Kode Mata Kuliah

NAMA

Kategori

SKS

Semester

Save Cancel

Gambar 4. 32. Desain edit mata kuliah

Gambar 4.33 merupakan tampilan hapus mata kuliah dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk menghapus data mata kuliah, pengguna bisa menghapus data mata kuliah klik *icon delete*. Akan ada notifikasi yang memberikan pesan apakah yakin ingin menghapus data ini, jika ya maka data akan terhapus.

No	Kode MK	Nama	SKS	Jenis	Aksi
1	SI01	Bahasa Indonesia	2	TEORI	 

Gambar 4. 33. Desain hapus mata kuliah

Gambar 4.34 merupakan tampilan menu pengampu dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk mencari dan menambah data pengampu, pengguna bisa mencari data yang sudah diinputkan yang terlihat dalam tabel yang ada di halaman menu pengampu.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Dosen
- Mata Kuliah
- Pengampu
- Ruang
- Jam
- Hari
- Constraint
- Proses Data**
- Penjadwalan

Data Pengampu

Semester:

Tahun Akademik:

Dosen / Mata Kuliah:

No	Mata Kuliah	Dosen	Kelas	Tahun Akademik	Aksi
1	Ahsani	11111	098697		

Gambar 4. 34. Desain menu pengampu

Gambar 4.35 merupakan tampilan input pengampu dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk menambah data pengampu, pengguna bisa menambahkan data pengampu dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Dosen
- Mata Kuliah
- Pengampu
- Ruang
- Jam
- Hari
- Constraint
- Proses Data**
- Penjadwalan

Data Pengampu

NIDN:

Semester:

Mata Kuliah:

Dosen:

Kelas:

Tahun Akademik:

Gambar 4. 35. Desain input pengampu

Gambar 4.36 merupakan tampilan edit pengampu dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk edit data pengampu, pengguna bisa edit data pengampu dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol *save*.

The screenshot shows a web application interface for editing lecturer data. On the left is a sidebar menu with three main sections: 'Data Master' (containing Dosen, Mata Kuliah, Pengampu, Ruang, Jam, Hari, and Constraint), 'Proses Data', and 'Penjadwalan'. The 'Data Master' section is currently selected. The main area is titled 'Modul Edit Pengampu' and contains several input fields: 'NIDN' with a dropdown menu showing 'Genap', 'Mata Kuliah' with a dropdown menu showing 'Bahasa Indonesia', 'Dosen' with a text input field containing 'Ahsani', 'Kelas' with a text input field containing '16.A', and 'Tahun Akademik' with a dropdown menu showing '2019-2020'. At the bottom of the form are two buttons: 'Save' and 'Cancel'.

Gambar 4. 36. Desain edit pengampu

Gambar 4.37 merupakan tampilan hapus pengampu dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk menghapus data pegampu, pengguna bisa menghapus data pengampu klik *icon delete*. Akan ada notifikasi yang memberikan pesan apakah yakin ingin menghapus data ini, jika ya maka data akan terhapus.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master
Dosen
Mata Kuliah
Pengampu
Ruang
Jam
Hari
Constraint
Proses Data
Penjadwalan

Data Pengampu
Semester
Genap
Tahun Akademik
2019-2020
Dosen / Mata Kuliah

Cari Clear
Tambah Data
Anda yakin ingin menghapus data ini?
Cancel OK

No	Mata Kuliah	Dosen	Kelas	Tahun Akademik	Aksi
1	Ahsani	11111	098697		 

Gambar 4. 37. Desain hapus pengampu

Gambar 4.38 merupakan tampilan menu ruang dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk mencari dan menambah data ruang, pengguna bisa mencari data yang sudah diinputkan yang terlihat dalam tabel yang ada di halaman menu ruang.

Penjadwalan Kuliah Program Studi Sistem Informasi

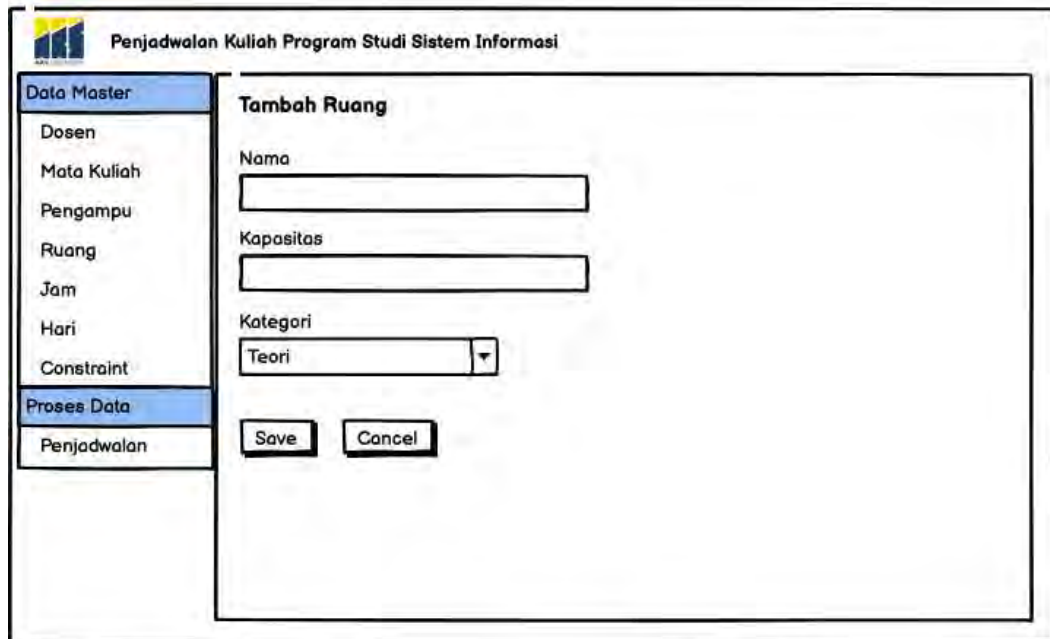
Data Master
Dosen
Mata Kuliah
Pengampu
Ruang
Jam
Hari
Constraint
Proses Data
Penjadwalan

Data Ruang
 Cari Clear
Tambah Data

No	Nama	Kapasitas	Jenis	Aksi
1	R.401	30	Teori	 

Gambar 4. 38. Desain menu ruang

Gambar 4.39 merupakan tampilan input ruang dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk menambah data ruang, pengguna bisa menambahkan data ruang dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.



The image shows a web application interface for managing lecture rooms. The title bar reads "Penjadwalan Kuliah Program Studi Sistem Informasi". On the left is a sidebar menu with two main sections: "Data Master" and "Proses Data". Under "Data Master", there are links for "Dosen", "Mata Kuliah", "Pengampu", "Ruang", "Jam", "Hari", and "Constraint". Under "Proses Data", there is a link for "Penjadwalan". The "Ruang" link is currently selected. The main content area is titled "Tambah Ruang" and contains a form with three input fields: "Nama" (Name), "Kapasitas" (Capacity), and "Kategori" (Category). The "Kategori" field is a dropdown menu with "Teori" selected. At the bottom of the form are two buttons: "Save" and "Cancel".

Gambar 4. 39. Desain input ruang

Gambar 4.40 merupakan tampilan edit ruang dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk edit data ruang, pengguna bisa edit data ruang dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

Gambar 4. 40. Desain edit ruang

Gambar 4.41 merupakan tampilan hapus ruang dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk menghapus data ruang, pengguna bisa menghapus data ruang klik *icon delete*. Akan ada notifikasi yang memberikan pesan apakah yakin ingin menghapus data ini, jika ya maka data akan terhapus.

No	Nama	Kapasitas	Jenis	Aksi
1	R.401	30	Teori	 

Gambar 4. 41. Desain hapus ruang

Gambar 4.42 merupakan tampilan menu jam dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk mencari dan menambah data jam, pengguna bisa mencari data yang sudah diinputkan yang terlihat dalam tabel yang ada di halaman menu dosen.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master	Data Jam		
Dosen	Nama Cari Clear Tambah Data		
Mata Kuliah	No	Range Jam	Aksi
Pengampu	1	08.00-09.00	30  
Ruang			
Jam			
Hari			
Constraint			
Proses Data			
Penjadwalan			

Gambar 4. 42. Desain menu jam

Gambar 4.43 merupakan tampilan input jam dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk menambah data jam, pengguna bisa menambahkan data jam dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Dosen
- Mata Kuliah
- Pengampu
- Ruang
- Jam
- Hari
- Constraint

Proses Data

Penjadwalan

Tambah Range Jam

Range Jam

Save Cancel

Gambar 4. 43. Desain input jam

Gambar 4.44 merupakan tampilan edit jam dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk edit data jam, pengguna bisa edit jam dosen dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Dosen
- Mata Kuliah
- Pengampu
- Ruang
- Jam
- Hari
- Constraint

Proses Data

Penjadwalan

Modul Edit Jam

Range Jam

Save Cancel

Gambar 4. 44. Desain edit jam

Gambar 4.45 merupakan tampilan hapus jam dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk menghapus data jam, pengguna bisa menghapus data jam klik *icon delete*. Akan ada notifikasi yang memberikan pesan apakah yakin ingin menghapus data ini, jika ya maka data akan terhapus.

No	Range Jam	Aksi
1	08.00-09.00	30  

Gambar 4. 45. Desain hapus jam

Gambar 4.46 merupakan tampilan menu hari dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk mencari dan menambah data hari, pengguna bisa mencari data yang sudah diinputkan yang terlihat dalam tabel yang ada di halaman menu dosen.

The screenshot shows a web application interface for scheduling. On the left is a sidebar menu with categories: 'Data Master' (containing Dosen, Mata Kuliah, Pengampu, Ruang, Jam, Hari, and Constraint) and 'Proses Data' (containing Penjadwalan). The main area is titled 'Data Hari' and contains a table with three columns: 'No', 'Nama', and 'Aksi'. The first row of the table has the values '1' and 'Senin'. In the 'Aksi' column for the first row, there are icons for editing (a pencil) and deleting (a trash can). A 'Tambah Data' button is located in the top right corner of the main area.

No	Nama	Aksi
1	Senin	

Gambar 4. 46. Desain menu hari

Gambar 4.47 merupakan tampilan input hari dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk menambah data hari, pengguna bisa menambahkan data hari dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

The screenshot shows the 'Tambah Hari' (Add Day) form within the same application. The sidebar menu is identical to the previous image. The main area is titled 'Tambah Hari' and contains a single text input field labeled 'Nama Hari'. Below the input field are two buttons: 'Save' and 'Cancel'.

Gambar 4. 47. Desain input hari

Gambar 4.48 merupakan tampilan edit hari dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk edit data hari, pengguna bisa edit data hari dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Dosen
- Mata Kuliah
- Pengampu
- Ruang
- Jam
- Hari
- Constraint

Proses Data

- Penjadwalan

Modul Edit Hari

Nama Hari

Senin

Save Cancel

Gambar 4. 48. Desain edit hari

Gambar 4.49 merupakan tampilan hapus hari dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk menghapus data hari, pengguna bisa menghapus data hari klik *icon delete*. Akan ada notifikasi yang memberikan pesan apakah yakin ingin menghapus data ini, jika ya maka data akan terhapus.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Dosen
- Mata Kuliah
- Pengampu
- Ruang
- Jam
- Hari
- Constraint

Proses Data

- Penjadwalan

Data Hari

Tambah Data

No	Nama	Aksi
1	Senin	 

Anda yakin ingin menghapus data ini?

Cancel OK

Gambar 4. 49. Desain hapus hari

Gambar 4.50 merupakan tampilan menu *constraint* dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk memilih *constraint*, pengguna bisa memilih waktu dosen yang tidak bersedia sesuai hari yang tertera. Klik tombol simpan untuk menyimpan data *constraint*

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Dosen
- Mata Kuliah
- Pengampu
- Ruang
- Jam
- Hari
- Constraint

Proses Data

- Penjadwalan

Constraint

Dosen

Ahsani Takwim

Simpan

Hari	Jam	Status
Senin	08.00-09.00	☐ Dosen Tidak bersedia

Gambar 4. 50. Desain menu constraint

Gambar 4.51 merupakan tampilan menu penjadwalan dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Desain ini direncanakan berfungsi untuk melakukan proses penjadwalan. Pengguna bisa memilih semester yang diinginkan dan klik proses, setelahnya akan muncul data penjadwalan dalam tabel yang tersedia, pengguna bisa *export* data dengan klik tombol *export*.

No	Hari	Sesi	Jam	Mata Kuliah	Semester	Kelas	Dosen	Ruang
1	Senin	(1-3)	08.00-09.00	Bahasa Indonesia	2	1	16A	Ahsani

Gambar 4. 51. Desain menu penjadwalan

4.4 Implementasi

Dalam tahap ini iterasi dilakukan menyesuaikan dengan perubahan di tahap desain, dikarenakan tahap ini merupakan implementasi dari desain yang telah dibuat, dengan demikian jika ada perubahan dalam tahap sebelumnya, tahapan implementasi akan menyesuaikan dengan perubahan yang ada. Tahap Implementasi juga merupakan tahap penerapan dari hasil analisis dan desain sistem yang telah dibuat sebelumnya. Aplikasi ini berbasis dekstop yang nantinya akan digunakan oleh staf prodi dalam proses penjadwalan, dalam tahap ini dipaparkan juga kebutuhan data apa saja yang diperlukan dalam penjadwalan dan hasil dari penjadwalan yang dilakukan oleh sistem.

1. Implementasi antar muka

Gambar 4.52 merupakan tampilan *log in* dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Pengguna bisa mengakses aplikasi penjadwalan dimulai dengan memasuki form *login* dan memasukan *username* dan *password*. Jika *username* dan *password* sesuai, maka akan tampil menu beranda, sedangkan jika salah akan kembali ke form *login*.



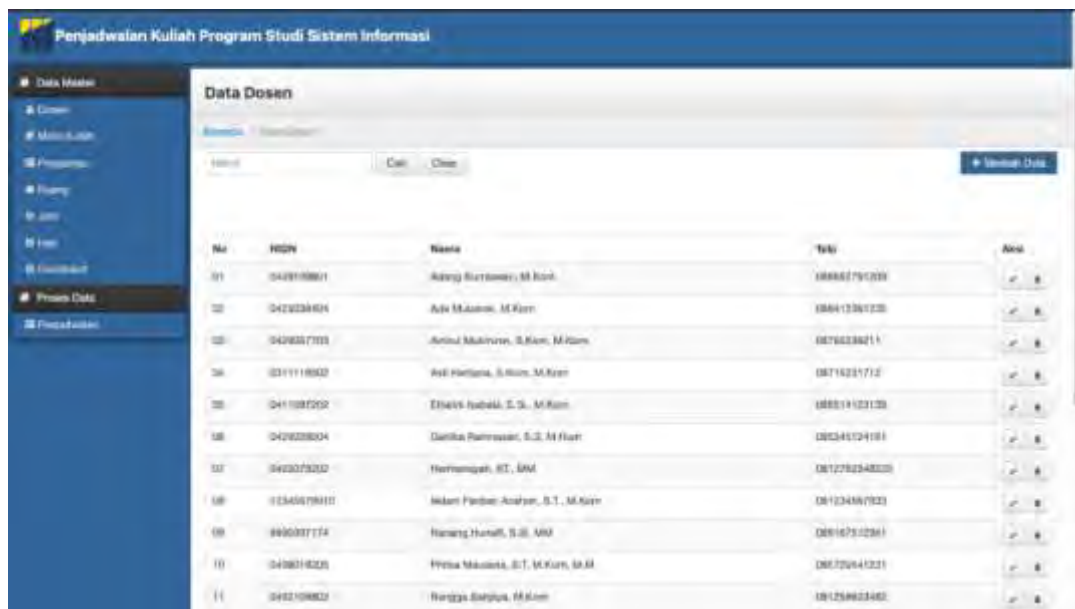
Gambar 4. 52. Tampilan *log in*

Gambar 4.53 merupakan tampilan beranda dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini merupakan beranda dari aplikasi penjadwalan, terlihat disebelah kiri merupakan menu yang tersedia dalam aplikasi penjadwalan. Halaman ini akan terlihat pertama kali setelah pengguna melakukan *login*.



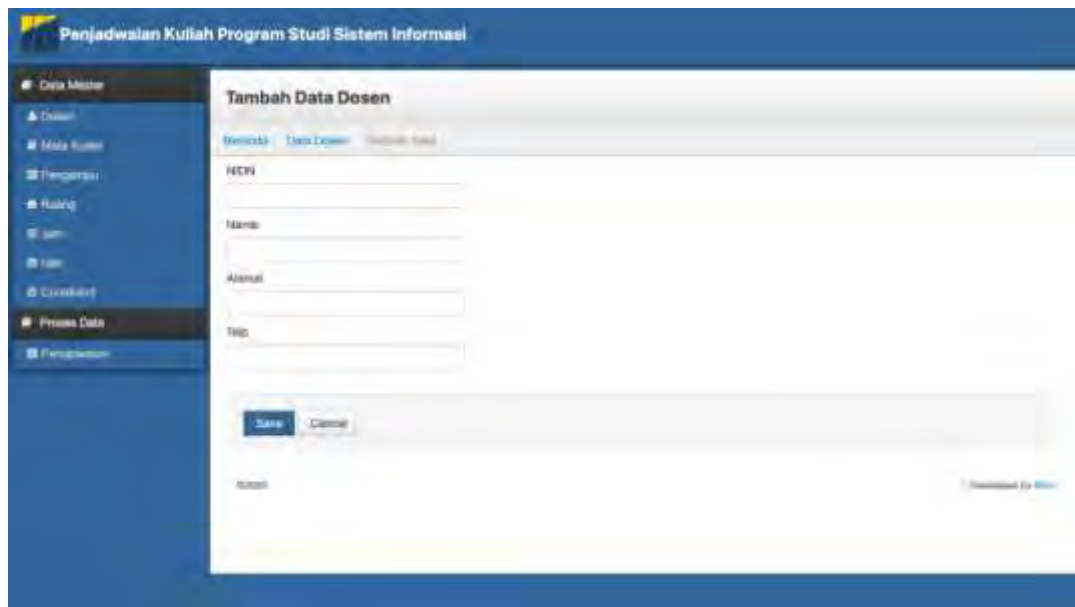
Gambar 4. 53. Tampilan beranda

Gambar 4.54 merupakan tampilan menu dosen dosen dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk mencari dan menambah data dosen, pengguna bisa mencari data yang sudah diinputkan yang terlihat dalam tabel yang ada di halaman menu dosen.



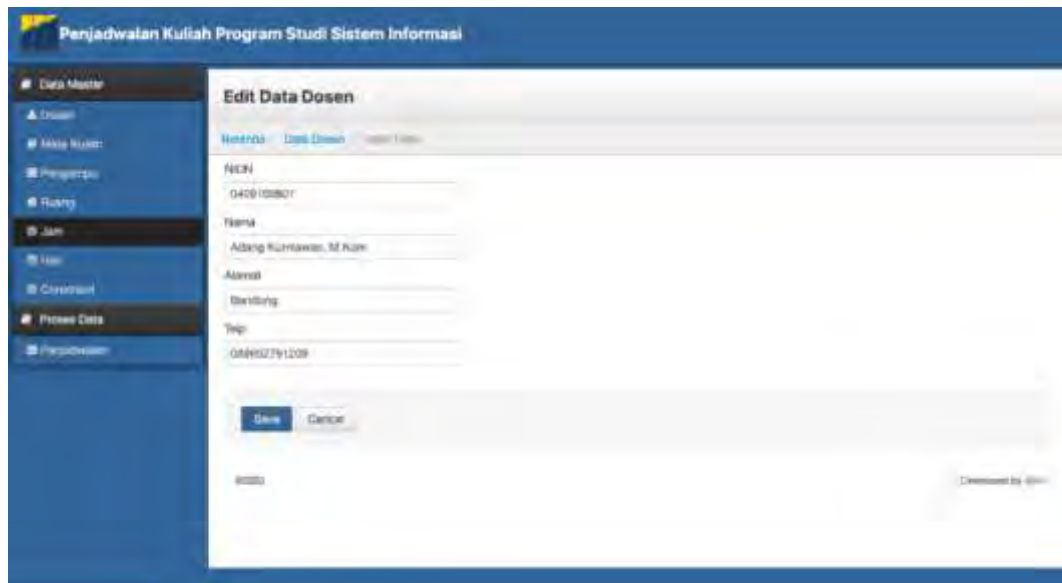
Gambar 4. 54. Tampilan menu dosen

Gambar 4.55 merupakan tampilan input dosen dosen dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk menambah data dosen, pengguna bisa menambahkan data dosen dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

The image shows a web application interface for managing lecturer data. The title bar at the top reads 'Penjadwalan Kuliah Program Studi Sistem Informasi'. On the left is a dark blue sidebar with a menu containing 'Data Master', 'Mata Kuliah', 'Pengajaran', 'Hubung', 'User', 'User', 'Cetak/Print', 'Print Data', and 'Pengaturan'. The main content area is titled 'Tambah Data Dosen' and contains a form with four input fields: 'NIDN', 'Nama', 'Alamat', and 'Telp'. Below the form are two buttons, 'Simpan' (Save) and 'Cancel'. At the bottom right of the form area, there is a small text 'Download By' followed by a logo.

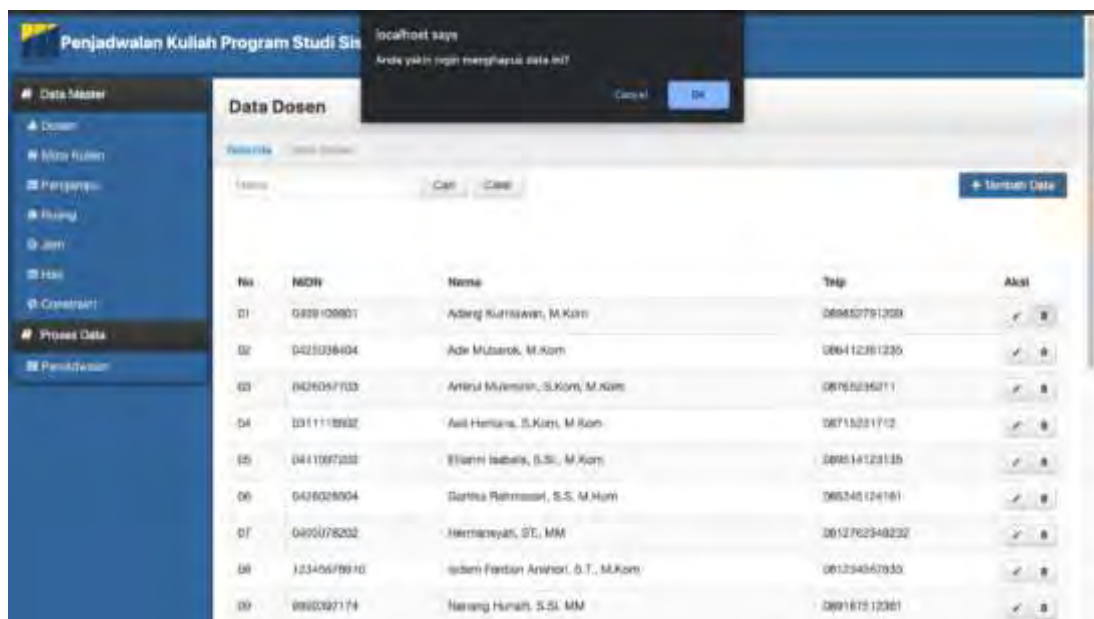
Gambar 4. 55. Tampilan input dosen

Gambar 4.56 merupakan tampilan edit dosen dosen dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk edit data dosen, pengguna bisa edit data dosen dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.



Gambar 4. 56. Tampilan edit dosen

Gambar 4.57 merupakan tampilan hapus dosen dosen dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk menghapus data dosen, pengguna bisa menghapus data dosen klik *icon delete*. Akan ada notifikasi yang memberikan pesan apakah yakin ingin menghapus data ini, jika ya maka data akan terhapus.



Gambar 4. 57. Tampilan hapus dosen

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Tambah
- Mata Kuliah
- Pengajar
- Ruang
- Jam
- Hari
- Courseware
- Proses Data
- Pengumuman

Tambah Data Mata Kuliah

Beranda / Data Mata Kuliah / Tambah Data

Kode Mata Kuliah:

Nama:

Category:

SKS:

Semester:

NOTES:

Download By:

Gambar 4. 59. Tampilan input matakuliah

Gambar 4.60 merupakan tampilan edit mata kuliah dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk edit data mata kuliah, pengguna bisa edit data mata kuliah dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Master

- Tambah
- Mata Kuliah
- Pengajar
- Ruang
- Jam
- Hari
- Courseware
- Proses Data
- Pengumuman

Edit Data Mata Kuliah

Beranda / Data Mata Kuliah / Edit Data

Kode Mata Kuliah:

Nama:

Category:

SKS:

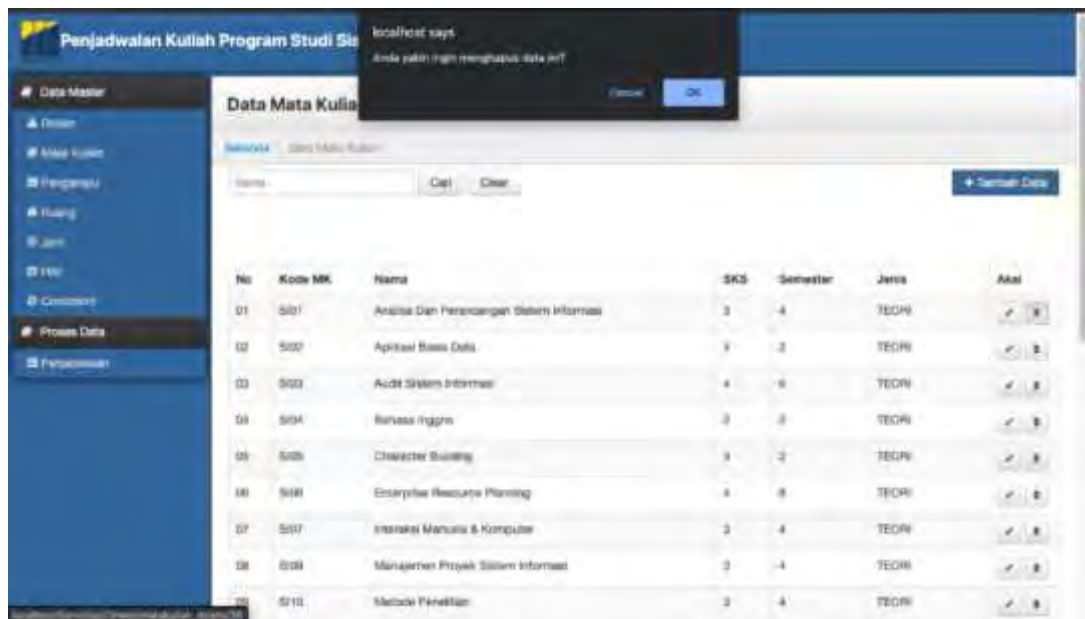
Semester:

NOTES:

Download By:

Gambar 4. 60. Tampilan edit matakuliah

Gambar 4.61 merupakan tampilan hapus mata kuliah dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk menghapus data mata kuliah, pengguna bisa menghapus data mata kuliah klik *icon delete*. Akan ada notifikasi yang memberikan pesan apakah yakin ingin menghapus data ini, jika ya maka data akan terhapus.



Gambar 4. 61. Tampilan hapus matakuliah

Gambar 4.62 merupakan tampilan menu pengampu dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk mencari dan menambah data pengampu, pengguna bisa mencari data yang sudah diinputkan yang terlihat dalam tabel yang ada di halaman menu pengampu.

Penjadwalan Kuliah Program Studi Sistem Informasi

Data Pengampu

Semester: 2019/2020
Tahun Akademik: 2019-2020
Dosen / Mata Kuliah: [Dropdown]
[Simpan] [Batal]

No	Mata Kuliah	Dosen	Kelas	Tahun Akademik	Aksi
001	Analisa Dan Perancangan Sistem Informasi	Andi Nurrobbil, S.Kom, M.Kom	IS.0A	2019-2020	[Edit] [Hapus]
002	Algoritma Basis Data	San Supren, M.Kom	IS.2A	2019-2020	[Edit] [Hapus]
003	Audit Sistem Informasi	Wahy Purnamasari, ST, M.Kom	IS.8A	2019-2020	[Edit] [Hapus]
004	Audit Sistem Informasi	Wahy Purnamasari, ST, M.Kom	IS.8D	2019-2020	[Edit] [Hapus]
005	Audit Sistem Informasi	Rusi, S.Bi, NIM	IS.8A	2019-2020	[Edit] [Hapus]
006	Audit Sistem Informasi	Adang Kurniawan, M.Kom	IS.8C	2019-2020	[Edit] [Hapus]
007	Bahasa Inggris	Gartha Hekrisnan, S.S, M.Hum	IS.2B	2019-2020	[Edit] [Hapus]

Gambar 4. 62. Tampilan menu pengampu

Gambar 4.63 merupakan tampilan input pengampu dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk menambah data pengampu, pengguna bisa menambahkan data pengampu dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

Penjadwalan Kuliah Program Studi Sistem Informasi

Tambah Data Pengampu

Semester: [Dropdown]
Mata Kuliah: [Dropdown]
Dosen: Adang Kurniawan, M.Kom
Kelas: [Dropdown]
Tahun Akademik: 2019-2020
[Simpan] [Batal]

Gambar 4. 63. Tampilan input pengampu

Gambar 4.64 merupakan tampilan edit pengampu dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk edit data pengampu, pengguna bisa edit data pengampu dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

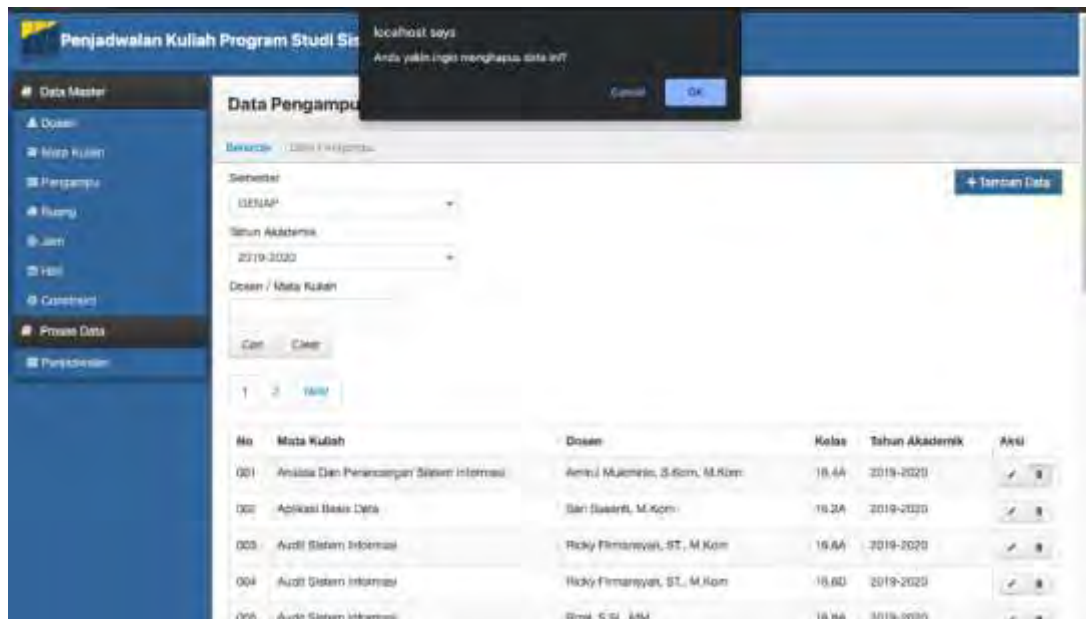
The screenshot shows a web application interface for editing lecturer data. The title bar reads 'Penjadwalan Kuliah Program Studi Sistem Informasi'. On the left is a sidebar menu with options like 'Data Master', 'Data Pengampu', and 'Proses Data'. The main content area is titled 'Edit Data Pengampu' and contains the following form fields:

- Mata Kuliah:** A dropdown menu with the selected value 'Analisa Dan Perancangan Sistem Info'.
- Dosen:** A dropdown menu with the selected value 'Areny-Iskumono, S.Sos, M.Kom'.
- Nama:** A text input field containing 'B.Ah'.
- Tahun Akademik:** A text input field containing '2019-2020'.

At the bottom of the form are two buttons: 'Save' (in blue) and 'Cancel' (in grey). There is also a 'Back' link at the bottom left and a 'Developer by' link at the bottom right.

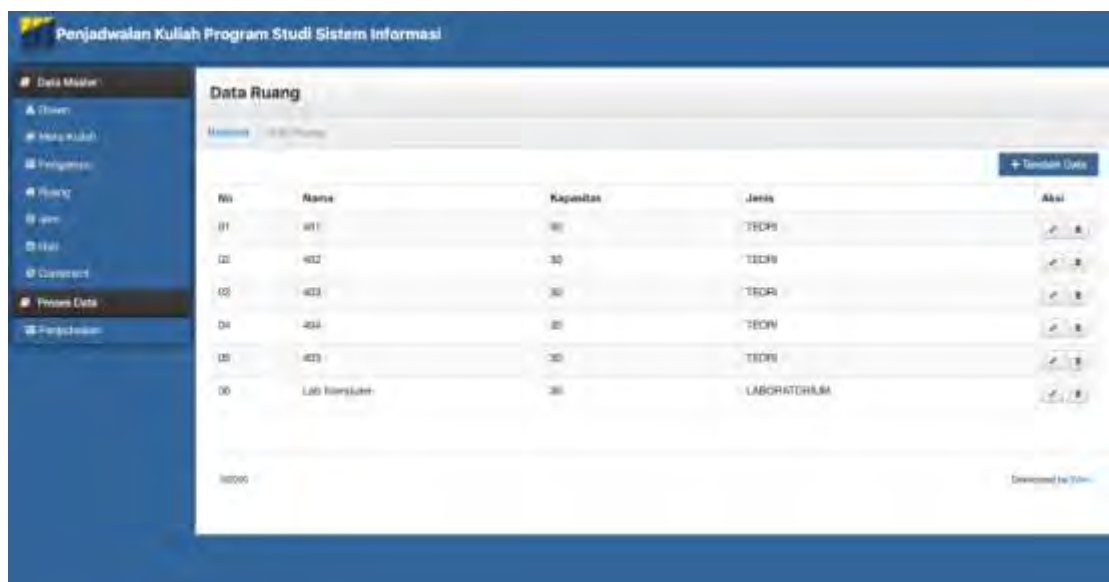
Gambar 4. 64. Tampilan edit pengampu

Gambar 4.65 merupakan tampilan hapus pengampu dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk menghapus data pegampu, pengguna bisa menghapus data pengampu klik *icon delete*. Akan ada notifikasi yang memberikan pesan apakah yakin ingin menghapus data ini, jika ya maka data akan terhapus.



Gambar 4. 65. Tampilan hapus pengampu

Gambar 4.66 merupakan tampilan menu ruang dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk mencari dan menambah data ruang, pengguna bisa mencari data yang sudah diinputkan yang terlihat dalam tabel yang ada di halaman menu ruang.



Gambar 4. 66. Tampilan menu ruang

Gambar 4.67 merupakan tampilan *input* ruang dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk menambah data ruang, pengguna bisa menambahkan data ruang dengan mengisi form

yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

The screenshot displays the 'Modul Tambah Ruang' (Add Room Module) within the 'Penjadwalan Kuliah Program Studi Sistem Informasi' application. The sidebar on the left contains a 'Data Master' section with options like 'Tromol', 'Mata Kuliah', 'Pengajar', 'Ruang', 'Jam', 'Hari', 'Concept', and 'Proses Data', and a 'Pengaturan' section. The main content area has a title bar with 'Beranda', 'Data Ruang', and 'Tambah Ruang'. Below this, there are input fields for 'Nama', 'Kapasitas', and a dropdown for 'Category' (currently showing 'TEORI'). At the bottom of the form are 'Save' and 'Cancel' buttons. A 'Logout' link is visible in the bottom right corner.

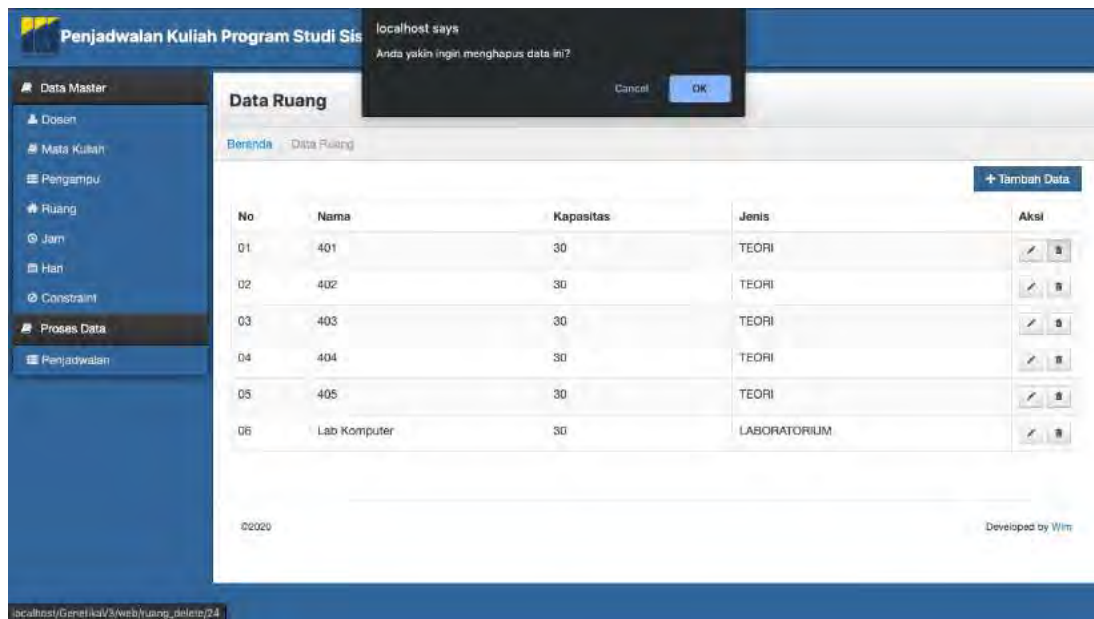
Gambar 4. 67. Tampilan input ruang

Gambar 4.68 merupakan tampilan edit ruang dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk edit data ruang, pengguna bisa edit data ruang dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

The screenshot displays the 'Modul Edit Ruang' (Edit Room Module) within the 'Penjadwalan Kuliah Program Studi Sistem Informasi' application. The sidebar on the left is identical to the previous image. The main content area has a title bar with 'Beranda', 'Data Ruang', and 'Edit Ruang'. Below this, there are input fields for 'Nama' (containing '401'), 'Kapasitas' (containing '30'), and a dropdown for 'Category' (currently showing 'TEORI'). At the bottom of the form are 'Save' and 'Cancel' buttons. A 'Logout' link is visible in the bottom right corner.

Gambar 4. 68. Tampilan edit ruang

Gambar 4.69 merupakan tampilan hapus ruang dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk menghapus data ruang, pengguna bisa menghapus data ruang klik *icon delete*. Akan ada notifikasi yang memberikan pesan apakah yakin ingin menghapus data ini, jika ya maka data akan terhapus.



Gambar 4. 69. Tampilan hapus ruang

Gambar 4.70 merupakan tampilan menu jam kuliah dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk mencari dan menambah data jam, pengguna bisa mencari data yang sudah diinputkan yang terlihat dalam tabel yang ada di halaman menu dosen.

No	Range Jam	Aksi
01	08.00-08.30	[Edit] [Delete]
02	08.30-09.00	[Edit] [Delete]
03	09.00-10.30	[Edit] [Delete]
04	10.30-11.00	[Edit] [Delete]
05	12.10-13.00	[Edit] [Delete]
06	13.00-13.30	[Edit] [Delete]
07	13.30-14.00	[Edit] [Delete]
08	14.00-14.30	[Edit] [Delete]
09	16.00-17.15	[Edit] [Delete]

Gambar 4. 70. Tampilan menu jam

Gambar 4.71 merupakan tampilan input jam kuliah dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. berfungsi untuk menambah data jam, pengguna bisa menambahkan data jam dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

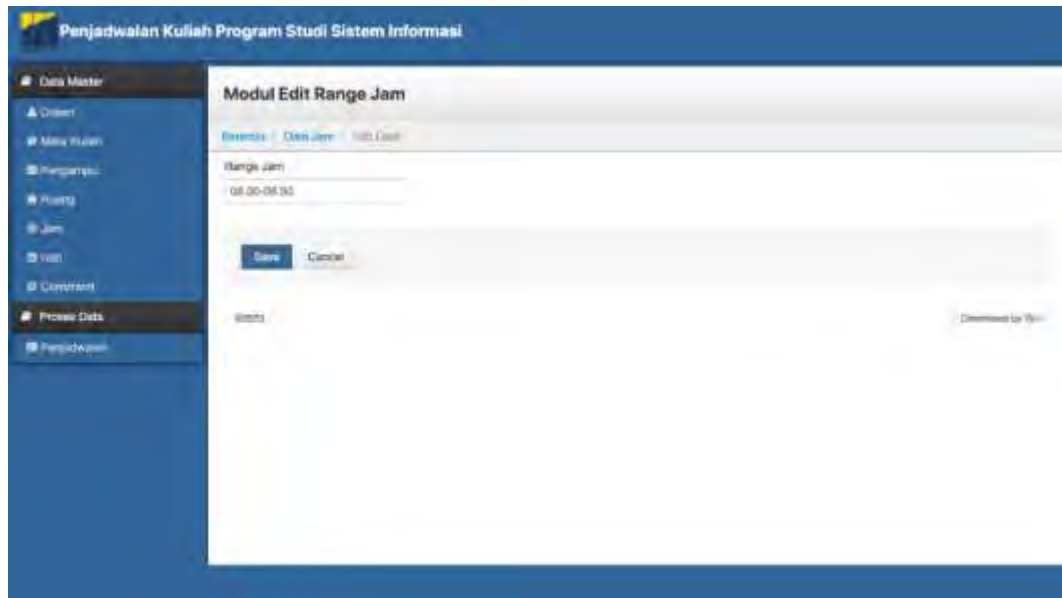
Modul Tambah Range Jam

Range Jam:

Validasi

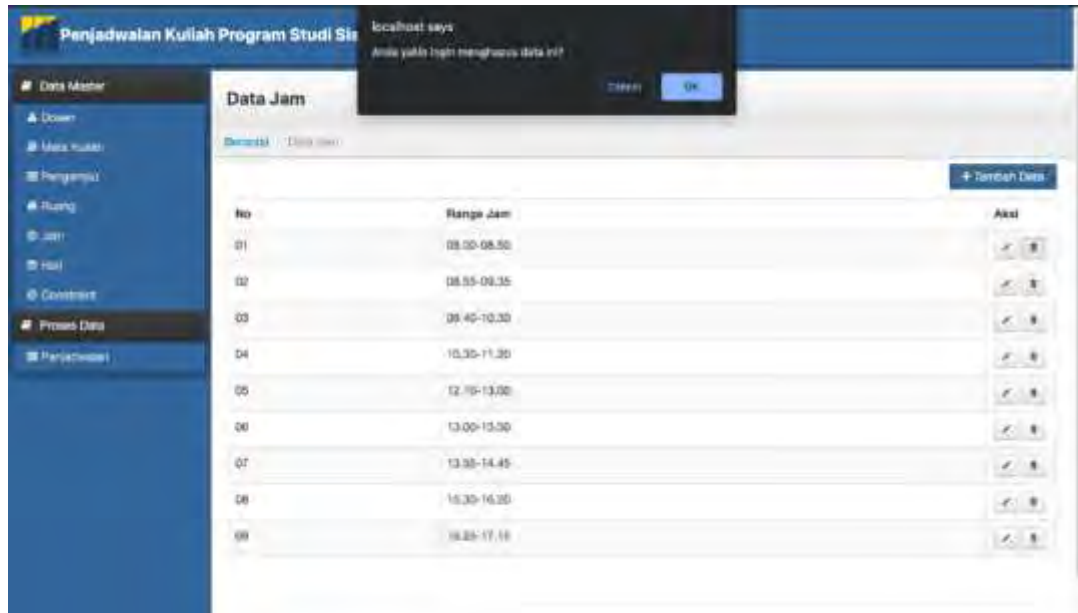
Gambar 4. 71. Tampilan input jam

Gambar 4.72 merupakan tampilan edit jam kuliah dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. berfungsi untuk edit data jam, pengguna bisa edit jam dosen dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.



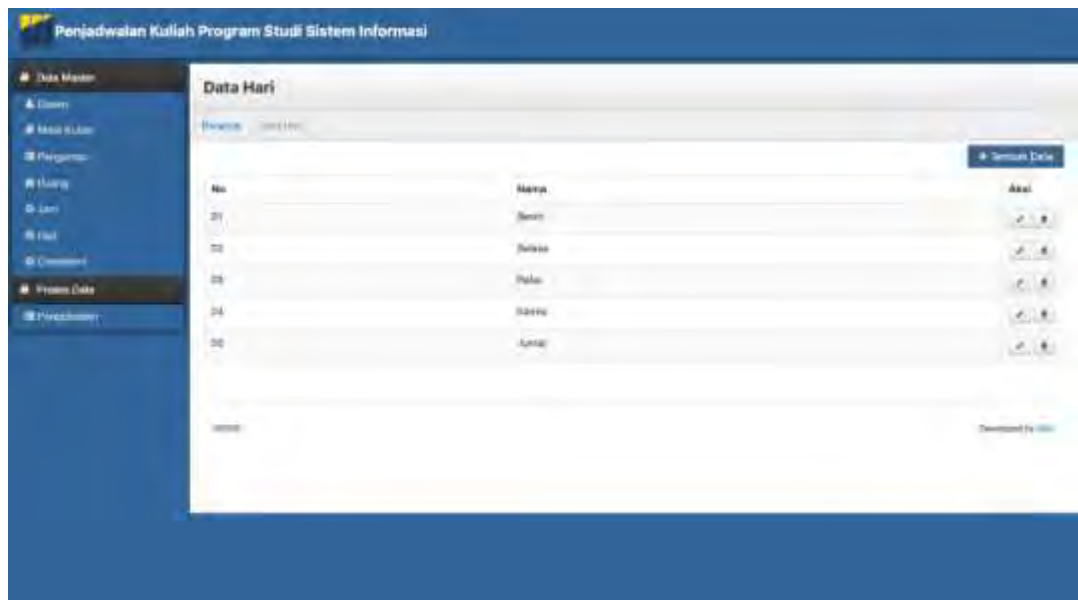
Gambar 4. 72. Tampilan edit jam

Gambar 4.73 merupakan tampilan hapus jam kuliah dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. berfungsi untuk menghapus data jam, pengguna bisa menghapus data jam klik *icon delete*. Akan ada notifikasi yang memberikan pesan apakah yakin ingin menghapus data ini, jika ya maka data akan terhapus.



Gambar 4. 73. Tampilan hapus jam

Gambar 4.74 merupakan tampilan menu hari dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. berfungsi untuk edit, hapus dan menambah data hari, pengguna bisa mencari data yang sudah diinputkan yang terlihat dalam tabel yang ada di halaman menu dosen.



Gambar 4. 74. Desain menu hari

Gambar 4.75 merupakan tampilan input hari dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk menambah

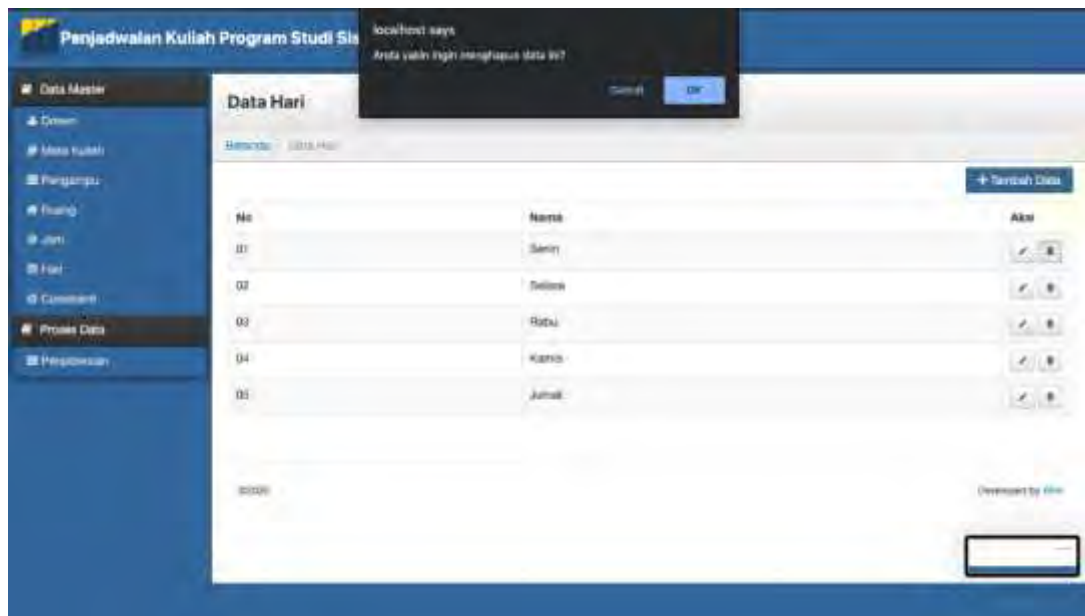
data hari, pengguna bisa menambahkan data hari dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

Gambar 4. 75. Desain input hari

Gambar 4.76 merupakan tampilan edit hari dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk edit data hari, pengguna bisa edit data hari dengan mengisi form yang tersedia, data akan tersimpan ketika pengguna mengisi data yang valid dan klik tombol save.

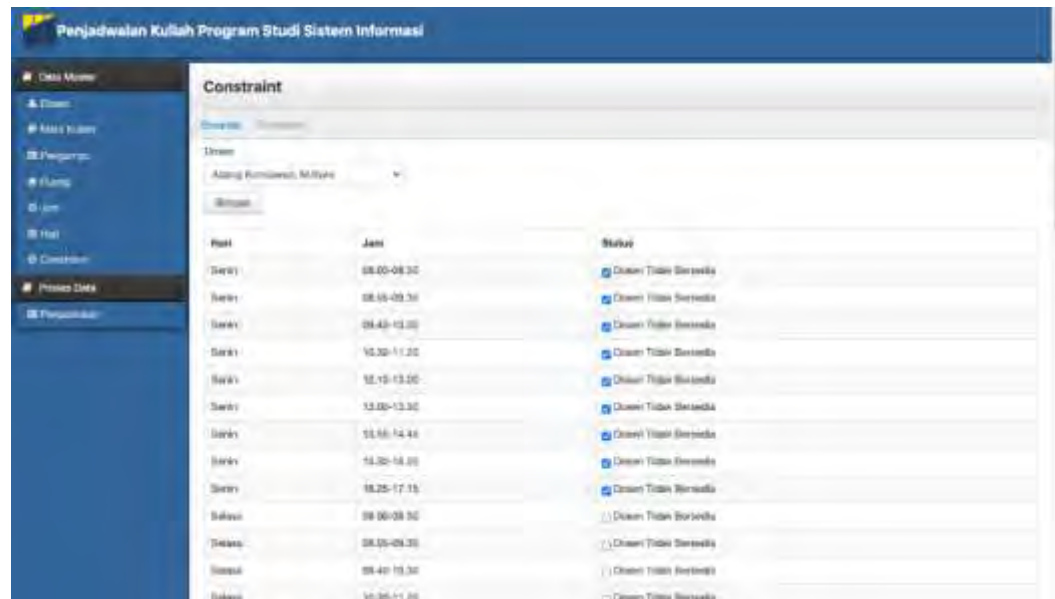
Gambar 4. 76. Desain edit hari

Gambar 4.77 merupakan tampilan hapus hari dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk menghapus data hari, pengguna bisa menghapus data hari klik *icon delete*. Akan ada notifikasi yang memberikan pesan apakah yakin ingin menghapus data ini, jika ya maka data akan terhapus.

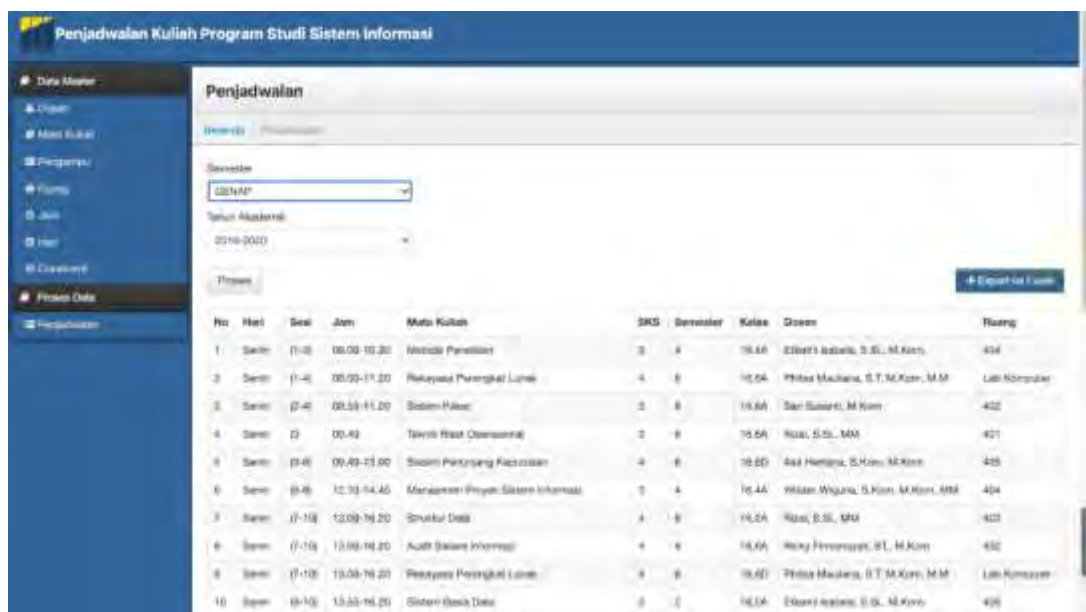


Gambar 4. 77. Desain hapus hari

Gambar 4.78 merupakan tampilan menu *constraint* dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk memilih *constraint*, pengguna bisa memilih waktu dosen yang tidak bersedia sesuai hari yang tertera. Klik tombol simpan untuk menyimpan data *constraint*.

Gambar 4. 78. Desain menu *constraint*

Gambar 4.79 merupakan tampilan penjadwalan mata kuliah dari aplikasi penjadwalan mata kuliah program Studi Sistem Informasi Universitas ARS. Halaman ini berfungsi untuk melakukan proses penjadwalan. Pengguna bisa memilih semester yang diinginkan dan klik proses, setelahnya akan muncul data penjadwalan dalam tabel yang tersedia, pengguna bisa *export* data dengan klik tombol *export*.



Gambar 4. 79. Desain menu penjadwalan

2. Implementasi tahapan *coding*

Dalam tahapan ini tabel 4.3 menunjukkan nama kelas, nama file bersama syntax yang berjalan dalam tahapan *coding*. Dapat dilihat bahwa jumlah kelas yang ada adalah sebanyak 10 kelas dilengkapi dengan *syntax* sebagai fungsi dari masing masing kelas.

Tabel 4. 3.Tahapan *coding*

No	Nama Kelas	Nama File	Syntax
1	Web	web.php	function index() function dosen() function dosen_add() function dosen_edit() function dosen_delete() function dosen_search() function matakuliah() function matakuliah_add() function matakuliah_edit() function matakuliah_delete() function matakuliah_search() function ruang() function ruang_add() function ruang_edit() function ruang_delete() function ruang_search() function jam() function jam_add() function jam_edit() function jam_delete() function jam_search() function hari() function hari_add() function hari_edit() function hari_delete() function hari_search() function pengampu() function pengampu_add() function pengampu_edit() function pengampu_search() function waktu_tidak_bersedia() function penjadwalan() function excel_report()
2	Genetik	genetik.php	public function AmbilData() public function Inisialisai() private function CekFitness()

No	Nama Kelas	Nama File	Syntax
			public function HitungFitness() public function Seleksi() public function StartCrossOver() public function Mutasi() public function GetIndividu()
3	m_dosen	m_dosen.php	function get_all() function get_by_kode() function get_search() function num_page() function insert() function update() function cek_for_insert() function cek_for_update()
4	m_hari	m_hari.php	function get() function get_by_kode() function cek_for_update() function cek_for_insert() function update() function insert() function delete() function validasi()
5	m_jadwalkuliah	m_jadwalkuliah.php	function get()
6	m_jam	m_jam.php	function get() function get_by_kode() function cek_for_update() function cek_for_insert() function update() function insert() function delete()

No	Nama Kelas	Nama File	Syntax
			function validasi()
7	m_matakuliah	m_matakuliah.php	function get() function get_all() function get_by_semester() function get_by_kode() function get_search() function num_page() function cek_for_update() function cek_for_insert() function update() function insert() function delete() function validasi()
8	m_pengampu	m_pengampu.php	function get() function get_by_kode() function get_search() function num_page() function delete_by_kode_dosen() function delete_by_mk() function delete() function cek_for_update() function cek_for_insert() function update() function insert() function validasi()
9	m_ruang	m_ruang.php	function get() function get_by_kode() function cek_for_update() function cek_for_insert() function update()

No	Nama Kelas	Nama File	Syntax
			function validasi()
10	m_waktu_tidak_bers edia	m_waktu_tidak_bers edia.php	function insert() function delete() function get_search()
11	login	Login.php	View() Proseslogin() Validasilogin()

3. Hasil penjadwalan

Hasil proses penjadwalan menggunakan algoritma genetika pada program studi sistem informasi universitas ARS tahun ajaran 2019-2020 genap berhasil menjadwalkan 19 perkuliahan sesuai dengan data yang ada. Maka dari itu akan diketahui apakah perkuliahan tersebut sudah memenuhi ketentuan yang sudah ditentukan. Gambar 4.80 menggambarkan penggunaan ruang 401 pada semester genap, menunjukkan ada 5 perkuliahan perminggunya dan frekuensi penggunaan kelas tersebut tidak terlalu tinggi, kemudian memenuhi *contstraint* dengan tidak menggunakan jam adzan saat perkuliahan berlangsung.

HARI/WAKTU	Senin	Selasa	Rabu	Kamis	Jumat
07.30-08.20	Enterprise Resource Planning 16.8A Nanang Hunaifi, S.Si. MM Enterprise Resource Planning 16.8C Nanang Hunaifi, S.Si. MM				Audit Sistem Informasi 16.6D Ricky Firmansyah, ST., M.Kom
08.25-09.15					
09.20-10.10					
10.15-11.05					
Adzan Dzohur 11.45					
11.50-12.40	Analisa Dan Perancangan Sistem Informasi 16.4A Amirul Mukminin, S.Kom, M.Kom Audit Sistem Informasi 16.8A Rizal, S.Si., MM				
13.40-14.10					
14.10-15.00					
Adzan Ashar 15.05					
15.15-16.05					Aplikasi Basis Data 16.A Sari Susanti, M.Kom
16.10-17.00					
17.05-17.55					

Gambar 4. 80. Penjadwakan di ruang 401

Gambar 4.81 menggambarkan penggunaan ruang 402 pada semester genap, menunjukkan ada 5 perkuliahan perminggunya dan frekuensi penggunaan kelas tersebut tidak terlalu tinggi, kemudian memenuhi *contstraint* dengan tidak menggunakan jam adzan saat perkuliahan berlangsung.

HARI/WAKTU	Senin	Selasa	Rabu	Kamis	Jumat
07.30-08.20	Struktur Data 16.2A Rizal, S.Si., MM		Sistem Basis Data 16.2A Elliarini Isabela, S.Si., M.Kom.		
08.25-09.15					
09.20-10.10					
10.15-11.05					
Adzan Dzhur 11.45					
11.50-12.40	Sistem Penunjang Keputusan 16.6D Asti Herliana, S.Kom, M.Kom				
14.10-15.00					
Adzan Ashar 15.05					
15.15-16.05	Manajemen Proyek Sistem Informasi 16.4A Wildan Wiguna, S.Kom, M.Kom, MM				
16.10-17.00					
17.05-17.55					

Gambar 4. 81. Penjadwalan di ruang 402

Gambar 4.82 menggambarkan penggunaan ruang 403 pada semester genap, menunjukkan ada 4 perkuliahan perminggunya dan frekuensi penggunaan kelas tersebut tidak terlalu tinggi, kemudian memenuhi *contstraint* dengan tidak menggunakan jam adzan saat perkuliahan berlangsung.

Waktu/Hari	Senin	Selasa	Rabu	Kamis	Jumat
07.30-08.20					
08.25-09.15					
09.20-10.10					
10.15-11.05					
Adzan Dzohur 11.45					
11.50-12.40				Teknik Riset Operasional 16.6A Rizal, S.Si., MM	
12.45-13.35					
13.40-14.10				Bahasa Inggris 2 16.2A Gartika Rahmasari, S.S, M.Hum	Sistem Penunjang Keputusan 16.6A Asti Herliana, S.Kom, M.Kom
14.10-15.00					
Adzan Ashar 15.05					
15.15-16.05					
16.10-17.00					
17.05-17.55					

Gambar 4. 82. Penjadwakan di ruang 403

Gambar 4.83 menggambarkan penggunaan ruang 404 pada semester genap, menunjukkan ada 5 perkuliahan perminggunya dan frekuensi penggunaan kelas tersebut tidak terlalu tinggi, kemudian memenuhi *contstraint* dengan tidak menggunakan jam adzan saat perkuliahan berlangsung.

Waktu/Hari	Senin	Selasa	Rabu	Kamis	Jumat
07.30-08.20	Sistem Pakar 16.8A Sari Susanti, M.Kom	Audit Sistem Informasi 16.8C Adang Kurniawan, M.Kom	Sistem Pakar 16.8C Sari Susanti, M.Kom		
08.25-09.15					
09.20-10.10					
10.15-11.05					
Adzan Dzbur 11.45					
11.50-12.40				Teknik Riset Operasional 16.6D Rizal, S.Si., MM	
12.45-13.35					
13.40-14.10					
14.10-15.00					
Adzan Ashar 15.05					
15.15-16.05	Metode Penelitian 16.4A Elliarini Isabela, S.Si., M.Kom.				
16.10-17.00					
17.05-17.55					

Gambar 4. 83. Penjadwakan di ruang 404

Gambar 4.84 menggambarkan penggunaan ruang 405 pada semester genap, menunjukkan ada 4 perkuliahan perminggunya dan frekuensi penggunaan kelas tersebut tidak terlalu tinggi, kemudian memenuhi *contstaint* dengan tidak menggunakan jam adzan saat perkuliahan berlangsung.

Waktu/Hari	Senin	Selasa	Rabu	Kamis	Jumat	
07.30-08.20	Sistem Basis Data 16.2A Elliarini Isabela, S.Si., M.Kom.					
08.25-09.15						
09.20-10.10						Multimedia 16.4A Ade Mubarak, M.Kom
10.15-11.05						
Adzan Dzhur 11.45						
11.50-12.40	Character Building 16.2A Hermansyah, ST., MM		Audit Sistem Informasi 16.6A Ricky Firmansyah, ST., M.Kom			
12.45-13.35						
13.40-14.10						
14.10-15.00						
Adzan Ashar 15.05						
15.15-16.05						
16.10-17.00						
17.05-17.55						

Gambar 4. 84. Desain menu penjadwalan

Gambar 4.85 menggambarkan penggunaan laboratorium pada semester genap, menunjukkan ada 4 perkuliahan perminggunya dan frekuensi penggunaan kelas tersebut tidak terlalu tinggi, kemudian memenuhi *contstraint* dengan tidak menggunakan jam adzan saat perkuliahan berlangsung.

Waktu/Hari	Senin	Selasa	Rabu	Kamis	Jumat
07.30-08.20		Rekayasa Perangkat Lunak 16.6D Phitsa Mauliana, S.T, M.Kom, M.M	Rekayasa Perangkat Lunak 16.6A Phitsa Mauliana, S.T, M.Kom, M.M		
08.25-09.15					
09.20-10.10					
10.15-11.05					
Adzan Dzuhur 11.45					
11.50-12.40					
12.45-13.35					
13.40-14.10				Web Programming III 16.4A Rangga Sanjaya, M.Kom	Web Programmin g I 16.2A Rizki Tri Prasetio, S.T., M.Kom
14.10-15.00					
Adzan Ashar 15.05					
15.15-16.05					
16.10-17.00					
17.05-17.55					

Gambar 4. 85. Penjadwalan kuliah di laboratorium

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat ditarik kesimpulan bahwa penjadwalan mata kuliah dengan menggunakan algoritma genetika dengan metodologi *agile iteration* adalah sebagai berikut:

1. Dengan mempersiapkan kebutuhan fungsional dan non fungsional, membuat perencanaan, desain dan implementasi, aplikasi penjadwalan dengan algoritma genetika dapat dilakukan di Program Studi Sistem Informasi Universitas ARS.
2. Ketersediaan ruangan dan waktu dosen yang kosong merupakan faktor yang mempengaruhi penjadwalan.
3. Metode *agile iteration* memberikan dampak yang cukup signifikan dalam perancangan aplikasi, karena secara dinamis bisa memberikan dampak yang signifikan terhadap pengguna maupun pengembang.

5.2 Saran

Pengembangan aplikasi yang bisa diakses oleh bukan hanya oleh admin/staf prodi, sehingga dosen dan juga *stakeholder* yang terlibat bisa melakukan input data secara *online* dan *realtime* agar monitoring bisa lebih efektif dan menghasilkan aplikasi yang lebih efektif. Kombinasikan metode dan algoritma yang bisa dilakukan sebagai optimalisasi metode dan algoritma lainnya. Menambahkan fitur sistem rekomendasi yang bisa mengakomodir perubahan jadwal secara online.

DAFTAR PUSTAKA

- Al-Jarrag, M.A., AL-Sawalqah, A.A. And AL-Hamdan, S.F., 2017. Developing A Course Timetable System for Academic Departments Using Genetic Algorithm. 3(1), pp.25–36
- Agus Saputra, 2011, Trik dan Solusi Jitu Pemrograman PHP, PT. Elex Media Komputindo.
- Anita Desiani & Muhammad Arhami, 2006, Konsep Kecerdasan Buatan. Andi Offset. Yogyakarta.
- Alif Ahsanil Satria, 2019. Agile Software Development, <https://medium.com>. (akses pada 14 Oktober 2020).
- Asad Sohail, 2019. Agile Project Management — Part 1: Understanding and Benefits <https://medium.com>. (akses pada 14 Oktober 2020).
- Bernhard Rumpe, 2016 Modeling with UML Language Concepts, Methods: Translation from the German language edition: Modellierung mit UML – Sprache, Konzepte und Methodik. Springer International Publishing Switzerland.
- Burke and Sanja Petrovic, 2002, Burke Recent research directions in automated timetabling, *European Journal of Operational Research* 140.
- Bhuvan Unhelkar, 2018, Software Engineering with UML. CRC Press Taylor & Francis Group, Danvers, USA.
- Charles L. Karr, L. Michael Freeman, Industrial Applications of Genetic Algorithms, Series III: United State of America: CRC Press, LLC 1999.
- Deeba Kannan Solving. 2019. Timetable Scheduling Problems Using Genetic Algorithm. *International Journal of Engineering and Advanced Technology (IJEAT)*. 8 (5C). 168 – 170.
- Entot Suhartono. 2015. Optimasi Penjadwalan Mata Kuliah Dengan Algoritma Genetika (Studi Kasus di AMIK JTC Semarang). *INFOKAM*. 11 (5). 132-146.
- ER, P, SAYED, S. AND AHMED, A., 2015. Automated Timetable Generator. 1(11), pp.118–121
- Gunawan, H. 2003. Aplikasi Algoritma Genetik Untuk Optimasi Masalah Penjadwalan Flow-Shop. Skripsi. Fakultas Teknologi Pertanian. Institut Pertanian Bogor, Bogor.
- Herman Lecture Scheduling. 2019. Automation Using Genetic Algorithm. *International Journal of Engineering and Advanced Technology (IJEAT)*. 8 (5C). 1444-1447.
- Jack C. Stanley and Erik D. Gross. (2020) ROJECT MANAGEMENT HANDBOOK Simplified Agile, Scrum and DevOps for Beginners, Firts edition. 2020
- Jain et al (2010). "Formulation of Genetic Algorithm to Generate Good Quality Course Timetable". *International Journal of Innovation, Management Technology* vol. 1 no.3 2010.

- J. W. Satzinger, R. B. Jackson and S. D. Burd, *Systems Analysis and Design in a Changing World*, Sixth ed., 2011.
- Lee Spector et al, editor, *GECCO 2001: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 211–218, San Francisco, CA, 2001. Morgan Kaufmann.
- Maulana, Gun Gun. 2017. Pembelajaran Dasar Algoritma Dan Pemrograman Menggunakan El-Goritma Berbasis Web. *Jurnal Teknik Mesin (JTM)*, Vol. 06, Edisi Spesial.
- Mery Hanita, "Penerapan Algoritma Genetika Pada Penjadwalan Mata Kuliah", Universitas Bengkulu, 2011.
- Oliver Kramer (2017). *Genetic Algorithm Essentials*, Studies in Computational Intelligence. Cham, Switzerland: Springer International Publishing AG.
- Pinedo, M. L. (2012). *Scheduling: Theory, Algorithms, and Systems* 4th Edition. New Jersey: Springer.
- Pressman, R.S. (2010), *Software Engineering: a practitioner's approach*, McGraw-Hill, New York.
- Rahman Erama. (2014). Modifikasi Algoritma Genetika untuk Penyelesaian Permasalahan Penjadwalan Pelajaran Sekolah. *Indonesian Journal of Computing and Cybernetics Systems*. 8 (2): pp.111-120.
- Rudová, H., 2014. *University Course Timetabling: From Theory to Practice*.
- Satzinger, J. W., Jackson, R. B., Burd, S. D. (2012). *System Analysis and Design in A Changing World*.
- SUHARTONO, E., 2015. Optimasi Penjadwalan Dengan Algoritma Genetika (Studi Kasus di AMIK JTC Semarang). pp.132–146
- Suyanto (2005). *Algoritma Genetika dalam Matlab*. Penerbit Andi. Yogyakarta.
- Wiga Ayu Puspaningrum. 2013. Penjadwalan Mata Kuliah Menggunakan Algoritma Genetika di Jurusan Sistem Informasi ITS. *JURNAL TEKNIK POMITS*. 2 (1): A-127 – A131
- YADAV, S.L. AND SOHAL, A., 2017. Comparative Study of Different Selection Techniques in Genetic Algorithm. *International Journal of Science and Mathematics Education*, 6(3), pp.251–254.

Lampiran 1

	STMIK LIKMI Sekolah Tinggi Manajemen Informatika dan Komputer	Jl. Ir. H. Juanda No.96, Bandung - 40132 Telp. (022) 250 2121 Fax (022) 250 5151 ✉ info@likmi.ac.id 📠 PMB LIKMI 📱 stmiklikmi 📧 @stmiklikmi 🌐 www.likmi.ac.id 📞 +62.81 721 9966 📺 stmiklikmi
---	---	---

Nomor : 024/TS/STK/VIII/2020	Bandung, 28 Agustus 2020
Perihal : <i>Permohonan ijin penelitian</i>	

Kepada Yth
Dekan Fakultas Teknologi Informasi
Universitas Adhirajasa Reswara Sanjaya
Jl. Sekolah Internasional No. 1-2
Antapani, Bandung

Dengan hormat,

Melalui surat ini perkenankan kami dari Program Pascasarjana STMIK LIKMI, mengajukan ijin penelitian/kunjunganbagimahasiswa kami yaitu :

Nama	: Ahsani Takwim
NPM	: 2019210045
Jenjang Pendidikan	: PascaSarjana (S2)
Bidang Minat	: Rekayasa Sistem Informasi
Judul Tesis	: Penjadwalan Kuliah dengan Metode Algoritma Genetika

Sehubungan dengan hal itu, kami mohon kiranya Bapak/Ibu dapat membantu mahasiswa yang bersangkutan dalam penelitian, yang diperlukan untuk penulisan Tesis.

Atas bantuan dan kerjasama Bapak/Ibu kami ucapkan terima kasih.

Hormat Kami,


Dr. Dingsulma Tjahjadi, S.E., M.T.
Wakil Ketua I

Lampiran 2



SUSUNAN JABATAN STRUKTURAL UNIVERSITAS ADHIRAJASA RESWARA SANJAYA (U-ARS)

TIM AKADEMIK

Warak Akademik : Dr. A. Rohendi, S.H, M.H, M.M

NO	NIP	NAMA	JABATAN
PASCASARJANA			
1	201303123	Dr. A. Rohendi, S.H, M.H, M.M	Direktur Pascasarjana
2	201411374	Rian Andriani, S.Pd, M.M	Kaprodi Magister Manajemen
3	201903143	Asep Dedy, S.Si, MM	Staf Pascasarjana
FAKULTAS EKONOMI			
1	201303124	Yani Restiani Widjaja, S.E, M.M	Dekan Fakultas Ekonomi
2	201510361	Deden Edwar Yokeu Bernardin, S.E, M.M	Kaprodi Akuntansi
3	201510355	Sahidillah Nurdin, S.E, M.M	Kaprodi Manajemen
4	201909003	Silfia Rachim, S.E	Staf Fakultas
FAKULTAS KOMUNIKASI & DESAIN			
1	201310559	Dr. Dasrun Hidayat, S.Sos, M.I.Kom	Dekan Fakultas Komunikasi & Desain
2	201803054	Sukatno Krisbianto, M.M	Kaprodi Ilmu Komunikasi
3	201303125	Hendi Yuliansyah, M.Sn, M.M	Kaprodi DKV
4	201703076	Reza Rizkina Taufik, S.I.Kom, M.M	Staf Fakultas
5	201703094	Agus Triyadi, S.Sn, M.M	Staf Fakultas
FAKULTAS TEKNOLOGI INFORMASI			
1	201609510	Rangga Sanjaya, S.T, M.Kom	Dekan Fakultas Teknologi Informasi
2	201509304	Yudi Ramdhani, S.T, M.Kom	Kaprodi Teknik Informatika
3	201509303	Rizki Tri Prasetyo, S.T, M.Kom	Kaprodi Sistem Informasi
4	201803063	Iedam Fardian Anshori, S.T, M.Kom	Staf Fakultas
5	202001003	Rissa Nurfitriana Handayani, S.Kom	
FAKULTAS ILMU KEPERAWATAN			
1	200905474	Sri Hayati, M.Kep	Dekan Ilmu Keperawatan
2	201801002	Mery Tania, S.Kep, Ners, M.Kep	Kaprodi Ilmu Keperawatan
3	201411372	Erna Irawan, S.Kep, S.T, M.Kom, M.Kep	Kaprodi Profesi Ners
4	201702017	Nurul Iklima, S.Kep, Ners, M.Kep	Staf Lab/ Fakultas
5	202002017	Umi Khasanah, S.Kep, Ners	
6	201904001	Anggi Saputra, S.Kep, Ners	



Jl. Sekolah Internasional No. 1-2
Antapani, Bandung - Indonesia
T: 022-7100124

<http://www.ars.ac.id>



Lampiran 3

