

BAB V KESIMPULAN DAN SARAN

1. Metode DPSO telah berhasil dieksplorasi dengan membuat prototipe yang menciptakan jadwal kuliah, dimana pemenuhan kendala lemah (*soft constraints*) dicapai dalam level tertentu. Sedangkan kendala tegas (*hard constraints*) dijamin dipenuhi.
2. Metode DPSO telah dapat digunakan dalam pembuatan *timetable* perkuliahan. Berdasarkan hasil eksekusi prototipe, jumlah partikel dan epoch mempengaruhi waktu dan kualitas dari hasil *generate* jadwal. Dari 44 kendala dosen tipe satu, 23 kendala dosen tipe dua dan satu kendala mahasiswa, sejumlah kendala di antaranya terpenuhi. Pemenuhan kendala lemah ini dipengaruhi oleh jumlah epoch dan partikel. Pada pengujian dengan 100 epoch dan 100 partikel, 96% kendala dosen tipe satu terpenuhi, 43% kendala dosen tipe dua terpenuhi, dan 33% kendala mahasiswa terpenuhi.

DAFTAR PUSTAKA

1. Afaq, Hifza., Saini, Sanjay., On the Solutions to the Travelling Salesman Problem using Nature Inspired Computing Techniques, India, IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 4, No 2, July 2011.
2. Altay, Ayca., Kayakutlu, Gulgun., 2011, Animal Food Chain Based Particle Swarm Optimization, London U.K, Proceedings of the World Congress on Engineering 2011 Vol II WCE 2011, July 6 – 8.
3. Chen, Yin-Yann., 2010, A Production Planning Problem Solved by the Particle Swarm Optimization, Hong Kong, Proceedings of the International MultiConference of Engineers and Computer Scientists 2010 Vol III, IMECS 2010.
4. Chu, Shu-Chuan., Chen, Yi-Tin., Ho, Jiun-Huei., 2006, Timetable Scheduling Using Particle Swarm Optimization, Taiwan, First International Conference on Innovative Computing, Information and Control.
5. Devi, Suprava ., Panigrahi, Sunil Kumar., 2011, Intelligent Decision Making Using Particle Swarm Optimization for Optimizing Product-Mix Model, India, International Journal of Computer Science & Informatics, Volume-1, Issue-1.
6. Fan, Huilian., 2010, Discrete Particle Swarm Optimization for TSP based on neighborhood, Journal of Computational Information System 6:10.
7. Gani, Taufiq Abdul ., Khader, Ahamad Tajudin., Budiarto, Rahmat., 2004, Optimizing Examination Timetabling using a Hybrid Evolution Strategies, Malaysia, 2nd International Conference and Autonomous Robots and Agents.
8. Hsieh, Ling-Feng., Huang, Chao-Jung., Huang, Chien-Lin., 2007, Applying Particle Swarm Optimization to Schedule Order Picking Routes, in a Distribution Centre, Taiwan, Asian Journal of Management and Humanity Science, Vol. 1, No. 4, pp. 558-576.
9. Huang, Jui-Ching., Wu, Shan-Ying., Pan, Wen-Tsao., 2011, Using particle swarm optimization (PSO) to perform financial characteristic study for enterprises in Taiwan, China, African Journal of Business Management Vol. 5(30), pp. 12040-12049.
10. Kanthavel, K., Prasad, P.S.S., Vignesh K.P., 2012, A Study of Sweep Algorithm Based Candidate List Performance in Capacitated Vehicle Routing Problem using Nested Particle Swarm Optimization, European Journal of Scientific Research.
11. Kiruthiga G., Senthilkumar M., 2011, A Survey about Dynamic Tasks Scheduling in Heterogeneous Processors Using Hybrid Particle Swarm Optimization, India, IJCS Vol. 2, ISSue 3.
12. Kumar, D. Nagesh., Reddy, M. Janga., 2007, Multipurpose Reservoir Operation Using Particle Swarm Optimization, Journal Of Water Resource Planning and Management @ ASCE.

13. Kumar, Kuldeep., Sikander., Sharma, Ramesh., Mehta, Kaushal., 2012, Genetic Algorithm Approach to Automate University Timetable, International Journal of Technical Research.
14. Montero, Elizabeth ., Riff, Mar'ia-Cristina., Altamirano, Leopoldo., 2011, A PSO Algorithm to Solve a Real Course+Exam Timetabling Problem, France, International Conference on Swarm Intelligence.
15. Mozafaria, Marzieh., Tafazzolia, Sajedeh., Jolaib, Fariborz., 2011, A new IPSO-SA approach for cardinality constrained portfolio optimization, Iran, International Journal of Industrial Engineering Computations, 2 249–262.
16. Mushi, A.R., 2006, Tabu Search *Heuristic* for University Course Time Tabling Problem, African Journal of Science and Technology, Science and Engineering series Vol. 7, No. 1, pp. 34-40.
17. Niasar, N. Salmani., Shanbezade, J., Perdam, M. M., Mohajeri M., 2009, Discrete Fuzzy Particle Swarm Optimization for Solving Traveling Salesman Problem, IEEE, International Conference on Information and Financial Engineering.
18. Norberciak, Maciej., 2006, Universal Method for Timetable based on Evolutionary Approach, Polandia, International Journal of Applied Mathematics and Computer Science Volume 3 Number 3.
19. Pang, W., Wang, Kang-ping, Zhou Chun-guang, Dong, Long-jiang, 2004, Fuzzy Discrete Particle Swarm Optimization for Solving Traveling Salesman Problem, IEEE, Proceedings of the Fourth International Conference on Computer and Information Technology (CIT'04).
20. Park, Jong-Bae ., Lee, Ki-Song., Shin, Joong-Rin., Lee, Kwang Y., 2005, A Particle Swarm Optimization for Economic Dispatch With Nonsmooth Cost Functions, IEEE Transaction on Power Systems, VOL. 20, NO. 1.
21. Rawat, Sandeep Singh., Rajamani, Lakshmi.,, 2005-2010, A Timetable Prediction for Technical Educational System Using Genetic Algorithm, India, Journal of Theoretical and Applied Information Technology.
22. Shahzad, Waseem., Khan, Farrukh Aslam., and Siddiqui, Abdul Basit., 2010, Weighted Clustering using Comprehensive Learning Particle Swarm Optimization for Mobile Ad Hoc Networks, Pakistan, International Journal of Future Generation Communication and Networking Vol. 3, No. 1.
23. Shi, X.H., Liang, Y.C., Lee, H.P., Lu, C., Wang, Q.X., 2007, Particle swarm optimization-based algorithms for TSP and generalized TSP, Singapore, Elsevier.
24. Sun, Jingjing., Lei, Xiujuan., 2009, Geese-inspired Hybrid Particle Swarm Optimization Algorithm for Traveling Salesman Problem, IEEE, International Conference on Artificial Intelligence and Computational Intelligence.
25. Tariq, M.P., Mirza, M.Waqar, Akbar, R., 2010, Multi-agent Based University Time Table Scheduling System (MUTSS), International Journal of Multidiciplinery Science and Engineering, Vol 1, No 1,
26. Tasgetiren, M.Fatih., Suganthan, P.N, Pan, Quan-ke., 2007, A Discrete Particle Swarm Optimization Algorithm for The Generalized Travelling Salesman Problem, London, GECCO.

27. Venkatesan, S.R., Logendran, D., Chandramohan, D., 2011, Optimization of Capacitated Vehicle Routing Problem using PSO, India, IJEST International Journal of Engineering Science Issues , Vol 8, Issue 4, No 2.
28. Wahyu Triadi Gunawan, 2010, Optimasi Penjadwalan Mata Kuliah dengan Metode Particle Swam Optimization, Skripsi Mahasiswa UAJY 05-07-04492.
29. Xue-Fei, H., Yun-xia, L., 2008, Optimization of Operational Route in AS/RS Based on Particle Swarm Algorithm, China, Journal of Southwest Jiaolong University.
30. Zhang, Jiang-wei., Xiong, Wei., 2008, An Improved Particle Swarm Optimization Algorithm and its Application for Solving Travelling Salesman Problem, IEEE, World Congress on Computer Science and Information Engineering.
31. Zhen, Tong., Zhu, Yuhua., Zhang, Qiuwen., A, 2009, Particle Swarm Optimization Algorithm for the Open Vehicle Routing Problem, China, IEEE, International Conference on Environmental Science and Information Application Technology.

LAMPIRAN 1



SKPL

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

SISTEM PENGATURAN JADWAL KULIAH

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Rochim Widaryanto/ 105301536

Program Studi Magister Teknik Informatika

Universitas Atma Jaya Yogyakarta

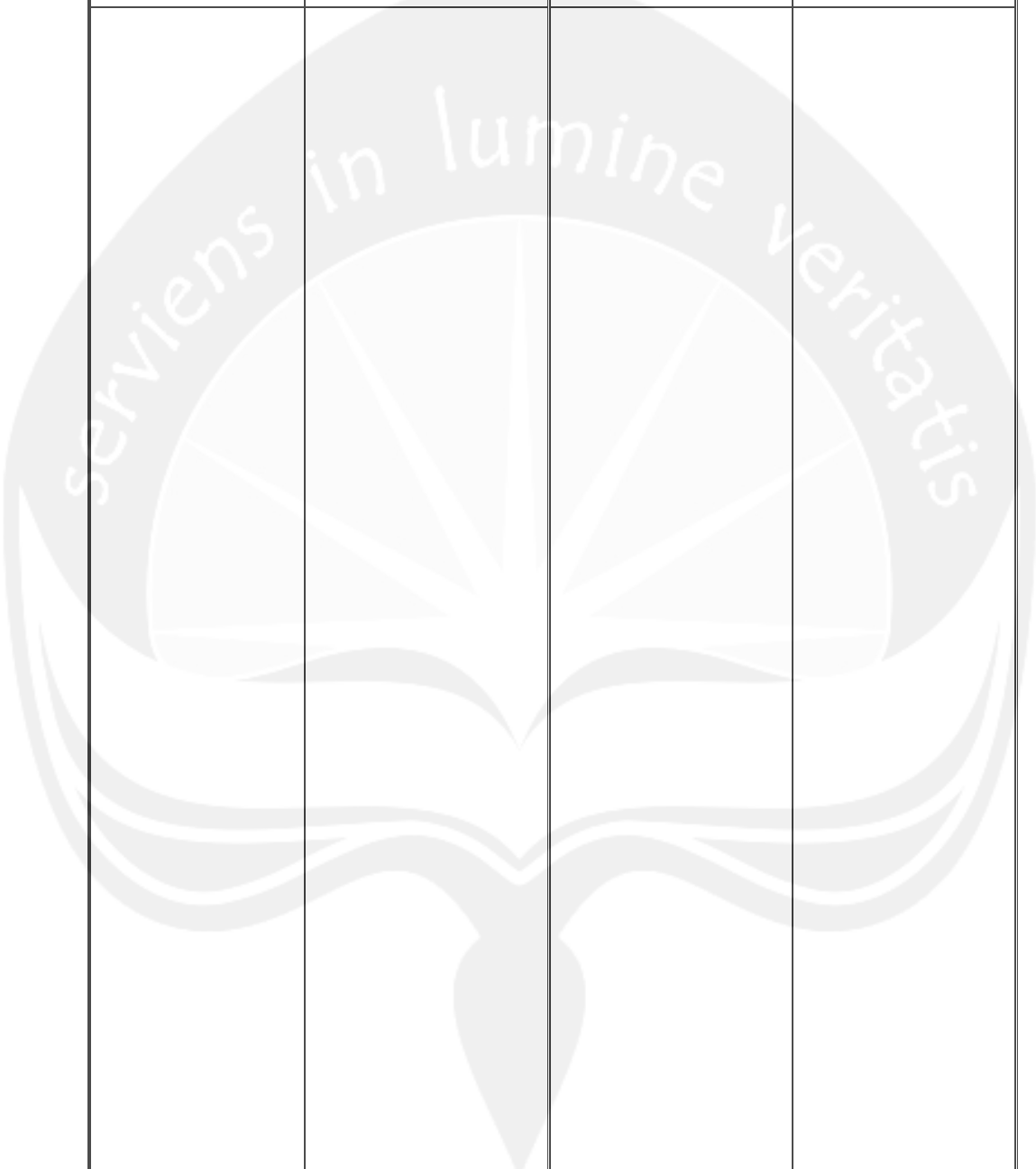
	Program Studi Magister Teknik Informatika	Nomor Dokumen		Halaman
		SKPL-SPJK		1/55
		Revisi		2013

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi
			

Daftar Isi

1	Pendahuluan	7
1.1	Tujuan	7
1.2	Lingkup Masalah.....	7
1.3	Definisi, Akronim dan Singkatan.....	7
1.4	Referensi	8
1.5	Deskripsi umum (Overview)	8
2	Deskripsi Kebutuhan	9
2.1	Perspektif produk.....	9
2.2	Kebutuhan Fungsionalitas Produk.....	10
2.3	Karakteristik Pengguna	17
2.4	Batasan-batasan.....	17
2.5	Asumsi dan Ketergantungan	18
3	Kebutuhan khusus	18
3.1	Kebutuhan antarmuka eksternal.....	18
3.1.1	Antarmuka pemakai.....	18
3.1.2	Antarmuka perangkat keras.....	18
3.1.3	Antarmuka perangkat lunak.....	18
3.2	Kebutuhan fungsionalitas Perangkat Lunak	20
3.2.1	Use Case Diagram.....	20
4	Spesifikasi Rinci Kebutuhan	20
4.1	Spesifikasi Kebutuhan Fungsionalitas.....	20
4.1.1	Use case Spesification : login.....	20
4.1.2	Use case Spesification : generateJadwal.....	20
4.1.3	Use case Spesification : manageDosen.....	20
4.1.4	Use case Spesification : manageKendalaDosen.....	20
4.1.5	Use case Spesification : manageKelasMataKuliah.....	20
4.1.6	Use case Spesification : manageKendalaMahasiswa.....	20
4.1.7	Use case Spesification : manageMataKuliah.....	20
4.1.8	Use case Spesification : manageRefSesi.....	20
4.1.9	Use case Spesification : manageRefRuang.....	20
4.1.10	Use case Spesification : manageRefHari.....	20
4.1.11	Use case Spesification : manageRefKelas.....	20
4.1.12	Use case Spesification : manageRefSemester.....	20

Program Studi Magister Teknik Informatika	SKPL – SPJK	4/ 55
---	-------------	-------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

4.1.13 Use case Spesification : manageUser.....	20
4.1.14 Use case Spesification : manageRefRole.....	20



Daftar Gambar

1. Arsitektur Perangkat Lunak SPJK	10
2. Use Case Diagram	20
3. ERD (Entity Relationship Diagram).....	55



1 Pendahuluan

1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak SPJK untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain perangkat lunak dan perangkat keras, dan pengguna) performansi (kemampuan perangkat lunak dari segi kecepatan, tempat penyimpanan yang dibutuhkan, serta keakuratan), dan atribut (*feature-feature* tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-SPJK ini juga mendefinisikan batasan perancangan perangkat lunak.

1.2 Lingkup Masalah

Untuk membantu kaprodi dalam menyelesaikan *timetable* perkuliahan, maka dilakukan eksplorasi metode DPSO yang dapat secara otomatis *men-generate timetable* dengan mempertimbangkan aturan kampus, preferensi dosen, dan preferensi mahasiswa.

1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-SPJK-XXX	Kode yang merepresentasikan kebutuhan pada SPJK dimana XXX merupakan nomor fungsi produk.

Program Studi Magister Teknik Informatika	SKPL – SPJK	7/ 55
---	-------------	-------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

SPJK	Program auto generate Timetable dengan menggunakan algoritma discrete particle swarm optimization.
ERD	Entity Relationship Diagram merupakan teknis grafis/diagram yang menggambarkan objek dan hubungan antar objek.

1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Aryo Wibisono, Spesifikasi Kebutuhan Perangkat Lunak Sistem Informasi Akademik Mahasiswa (SIATMA), Universitas Atma Jaya Yogyakarta.

1.5 Deskripsi umum (Overview)

Secara umum dokumen SKPL ini terbagi atas 3 bagian utama. Bagian utama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak tersebut, definisi, referensi dan deskripsi umum tentang dokumen SKPL ini.

Bagian kedua berisi penjelasan umum tentang perangkat lunak SPJK yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak SPJK tersebut.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak SPJK yang akan dikembangkan.

2 Deskripsi Kebutuhan

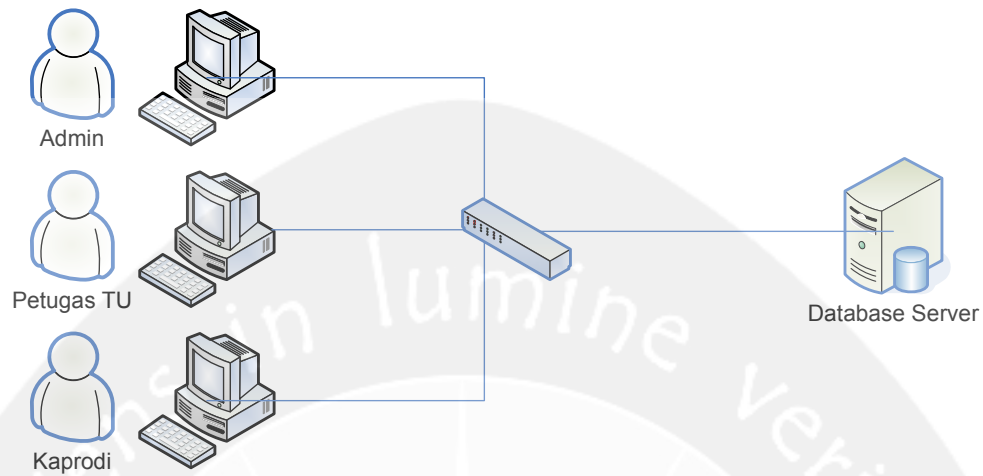
2.1 Perspektif produk

SPJK merupakan perangkat lunak yang dikembangkan untuk membantu Kaprodi dalam menciptakan suatu timetable. Sistem ini menangani pengelolaan data dan *generate* timetable yang merupakan bagian dari sistem yang digunakan untuk menyempurnakan pembelajaran.

Perangkat lunak SPJK ini berjalan pada platform berbasis windows, dan dibuat menggunakan bahasa pemrograman C#. Sedangkan untuk lingkungan pemrogramannya menggunakan Microsoft Visual Studio 2005.

Pengguna akan berinteraksi dengan sistem melalui antarmuka GUI (Graphical User Interface). Pada sistem ini, seperti terlihat pada gambar 1, arsitektur perangkat lunak yang digunakan berupa client server, di mana semua data disimpan di server. User dapat mengakses data yang ada di server tersebut.

Input data yang dimasukkan akan disimpan dalam database server, sehingga jika ada pencarian data, maka data yang diinginkan akan dicari ke database server yang selanjutnya dikirimkan ke client yang merequest melalui perangkat lunak.



Gambar 1. Arsitektur Perangkat lunak SPJK

2.2 Kebutuhan Fungsionalitas Produk

Fungsi produk pengembangan perangkat lunak SPJK adalah sebagai berikut :

a. User (Kaprodi, Petugas TU, Admin)

1. Fungsi login (SKPL SPJK-001)

Merupakan fungsi yang digunakan untuk masuk ke dalam sistem utama.

b. User (Kaprodi)

1. Fungsi manageDosen (SKPL SPJK-002)

Merupakan fungsi yang digunakan untuk mengelola data Dosen.

a) Fungsi add (SKPL SPJK-002-01)

Merupakan fungsi yang digunakan untuk menginput data dosen.

b) Fungsi edit (**SKPL SPJK-002-02**)

Merupakan fungsi yang digunakan untuk mengedit data dosen.

c) Fungsi delete (**SKPL SPJK-002-03**)

Merupakan fungsi yang digunakan untuk menghapus data dosen.

2. Fungsi manageKelasMataKuliah (**SKPL SPJK-003**)

Merupakan fungsi yang digunakan untuk mengelola data kelasMataKuliah.

a) add (**SKPL SPJK-003-01**)

Merupakan fungsi yang digunakan untuk menginput data kelasMataKuliah.

b) edit (**SKPL SPJK-003-02**)

Merupakan fungsi yang digunakan untuk mengedit data kelasMataKuliah.

c) delete (**SKPL SPJK-003-03**)

Merupakan fungsi yang digunakan untuk menghapus data kelasMataKuliah.

3. Fungsi manageKendalaDosen (**SKPL SPJK-004**)

Merupakan fungsi yang digunakan untuk mengelola data kendalaDosen.

a. add (**SKPL SPJK-004-01**)

Merupakan fungsi yang digunakan untuk menginput data kendalaDosen.

b. edit (**SKPL SPJK-004-02**)

Merupakan fungsi yang digunakan untuk mengedit data kendalaDosen.

c. delete (**SKPL SPJK-004-03**)

Merupakan fungsi yang digunakan untuk menghapus data kendalaDosen.

4. Fungsi manageKendalaMahasiswa (**SKPL SPJK-005**)

Merupakan fungsi yang digunakan untuk mengelola data kendalaMahasiswa.

a. add (**SKPL SPJK-005-01**)

Merupakan fungsi yang digunakan untuk menginput data kendalaMahasiswa.

b. edit (**SKPL SPJK-005-02**)

Merupakan fungsi yang digunakan untuk mengedit data kendalaMahasiswa.

c. delete (**SKPL SPJK-005-03**)

Merupakan fungsi yang digunakan untuk menghapus data kendalaMahasiswa.

5. Fungsi generateJadwal (**SKPL SPJK-006**)

Merupakan fungsi yang digunakan untuk mengelola data jadwal.

a. generateJadwal (**SKPL SPJK-006-01**)

Merupakan fungsi yang digunakan untuk melakukan generate jadwal.

b. showJadwal (**SKPL SPJK-006-02**)

Merupakan fungsi yang digunakan untuk menampilkan data jadwal.

c. report (**SKPL SPJK-006-03**)

Merupakan fungsi yang digunakan untuk report data jadwal.

c. User (Petugas TU)

1. Fungsi manageMataKuliah (**SKPL SPJK-007**)

Merupakan fungsi yang digunakan untuk mengelola data mataKuliah.

a) add (**SKPL SPJK-007-01**)

Merupakan fungsi yang digunakan untuk menginput data kelasMataKuliah.

b) edit (**SKPL SPJK-007-02**)

Merupakan fungsi yang digunakan untuk mengedit data kelasMataKuliah.

c) delete (**SKPL SPJK-007-03**)

Merupakan fungsi yang digunakan untuk menghapus data kelasMataKuliah.

2. Fungsi manageRefSesi (**SKPL SPJK-008**)

Merupakan fungsi yang digunakan untuk mengelola data refSesi.

a. add (**SKPL SPJK-008-01**)

Merupakan fungsi yang digunakan untuk menginput data refSesi.

b. edit (**SKPL SPJK-008-02**)

Merupakan fungsi yang digunakan untuk mengedit data refSesi.

c. delete (**SKPL SPJK-008-03**)

Merupakan fungsi yang digunakan untuk menghapus data refSesi.

3. Fungsi manageRefRuang (**SKPL SPJK-009**)

Merupakan fungsi yang digunakan untuk mengelola data refRuang.

a. add (**SKPL SPJK-009-01**)

Merupakan fungsi yang digunakan untuk menginput data refRuang.

b. edit (**SKPL SPJK-009-02**)

Merupakan fungsi yang digunakan untuk mengedit data refRuang.

c. delete (**SKPL SPJK-009-03**)

Merupakan fungsi yang digunakan untuk menghapus data refRuang.

4. Fungsi manageRefHari(**SKPL SPJK-010**)

Merupakan fungsi yang digunakan untuk mengelola data refHari.

a. add (**SKPL SPJK-010-01**)

Merupakan fungsi yang digunakan untuk menginput data refHari.

b. edit (**SKPL SPJK-010-02**)

Merupakan fungsi yang digunakan untuk mengedit data refHari.

c. delete (**SKPL SPJK-010-03**)

Merupakan fungsi yang digunakan untuk menghapus data refHari.

5. Fungsi manageRefKelas(**SKPL SPJK-011**)

Merupakan fungsi yang digunakan untuk mengelola data refHari.

a. add (**SKPL SPJK-011-01**)

Merupakan fungsi yang digunakan untuk menginput data refKelas.

b. edit (**SKPL SPJK-011-02**)

Merupakan fungsi yang digunakan untuk mengedit data refKelas.

c. delete (**SKPL SPJK-011-03**)

Merupakan fungsi yang digunakan untuk menghapus data refKelas.

6. Fungsi manageRefSemester(**SKPL SPJK-012**)

Merupakan fungsi yang digunakan untuk mengelola data refHari.

a. add (**SKPL SPJK-012-01**)

Merupakan fungsi yang digunakan untuk menginput data refSemester.

b. edit (**SKPL SPJK-012-02**)

Merupakan fungsi yang digunakan untuk mengedit data refSemester.

c. delete (**SKPL SPJK-012-03**)

Merupakan fungsi yang digunakan untuk menghapus data refSemester.

d. User (Administrator)

1. Fungsi manageUser (**SKPL SPJK-013**)

Merupakan fungsi yang digunakan untuk mengelola data user.

a. add (**SKPL SPJK-013-01**)

Merupakan fungsi yang digunakan untuk menginput data user.

b. edit (**SKPL SPJK-013-02**)

Merupakan fungsi yang digunakan untuk mengedit data user.

c. delete (**SKPL SPJK-013-03**)

Merupakan fungsi yang digunakan untuk menghapus data user

2. Fungsi manageRefRole (**SKPL SPJK-014**)

Merupakan fungsi yang digunakan untuk mengelola data refRole.

a. add (**SKPL SPJK-014-01**)

Merupakan fungsi yang digunakan untuk menginput data refRole.

b. edit (**SKPL SPJK-014-02**)

Program Studi Magister Teknik Informatika	SKPL – SPJK	16/ 55
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Merupakan fungsi yang digunakan untuk mengedit data refRole.

c. delete (**SKPL SPJK-014-03**)

Merupakan fungsi yang digunakan untuk menghapus data refRole

2.3 Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak SPJK adalah sebagai berikut :

1. Administrator

- a. Mengerti pengoperasian komputer dan internet.
- b. Mengerti proses pengelolaan sistem yang digunakan.

2. Kaprodi

- a. Mengerti pengoperasian komputer.
- b. Mengerti proses pengelolaan sistem yang digunakan

3. Petugas TU

- a. Mengerti pengoperasian komputer.
- b. Mengerti proses pengelolaan sistem yang digunakan.

2.4 Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak SPJK tersebut adalah :

- 1. Semua user mengakses SPJK dalam satu lingkungan jaringan berbasis local area network.

Program Studi Magister Teknik Informatika	SKPL – SPJK	17/ 55
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2.5 Asumsi dan Ketergantungan

Sistem ini dapat dijalankan pada perangkat PC dan Notebook.

3 Kebutuhan khusus

3.1 Kebutuhan antarmuka eksternal

Kebutuhan antar muka eksternal pada perangkat lunak SPJK meliputi kebutuhan antarmuka pemakai, antarmuka perangkat keras dan antarmuka perangkat lunak.

3.1.1 Antarmuka pemakai

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam layar komputer.

3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan dalam perangkat lunak SPJK adalah:

1. PC/Notebook
2. Keyboard
3. Mouse

3.1.3 Antarmuka perangkat lunak

Perangkat lunak yang dibutuhkan dalam mengoperasikan perangkat lunak SPJK adalah sebagai berikut:

a. Nama : SQL Server 2005

Sumber : Microsoft

Sebagai database yang dibutuhkan dalam mengoperasikan perangkat lunak SPJK.

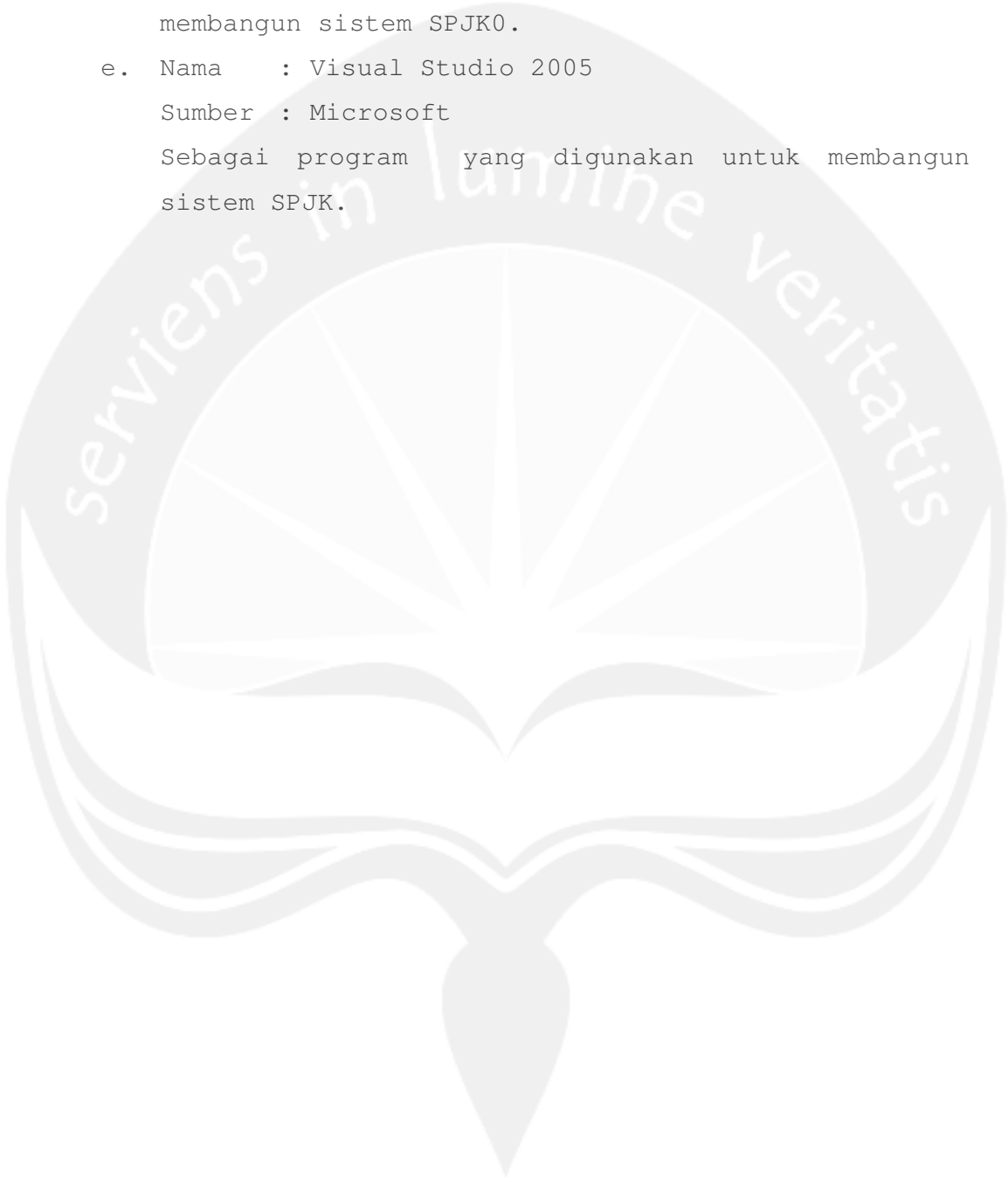
b. Nama : C#

Sebagai bahasa pemrograman yang digunakan untuk membangun sistem SPJK0.

e. Nama : Visual Studio 2005

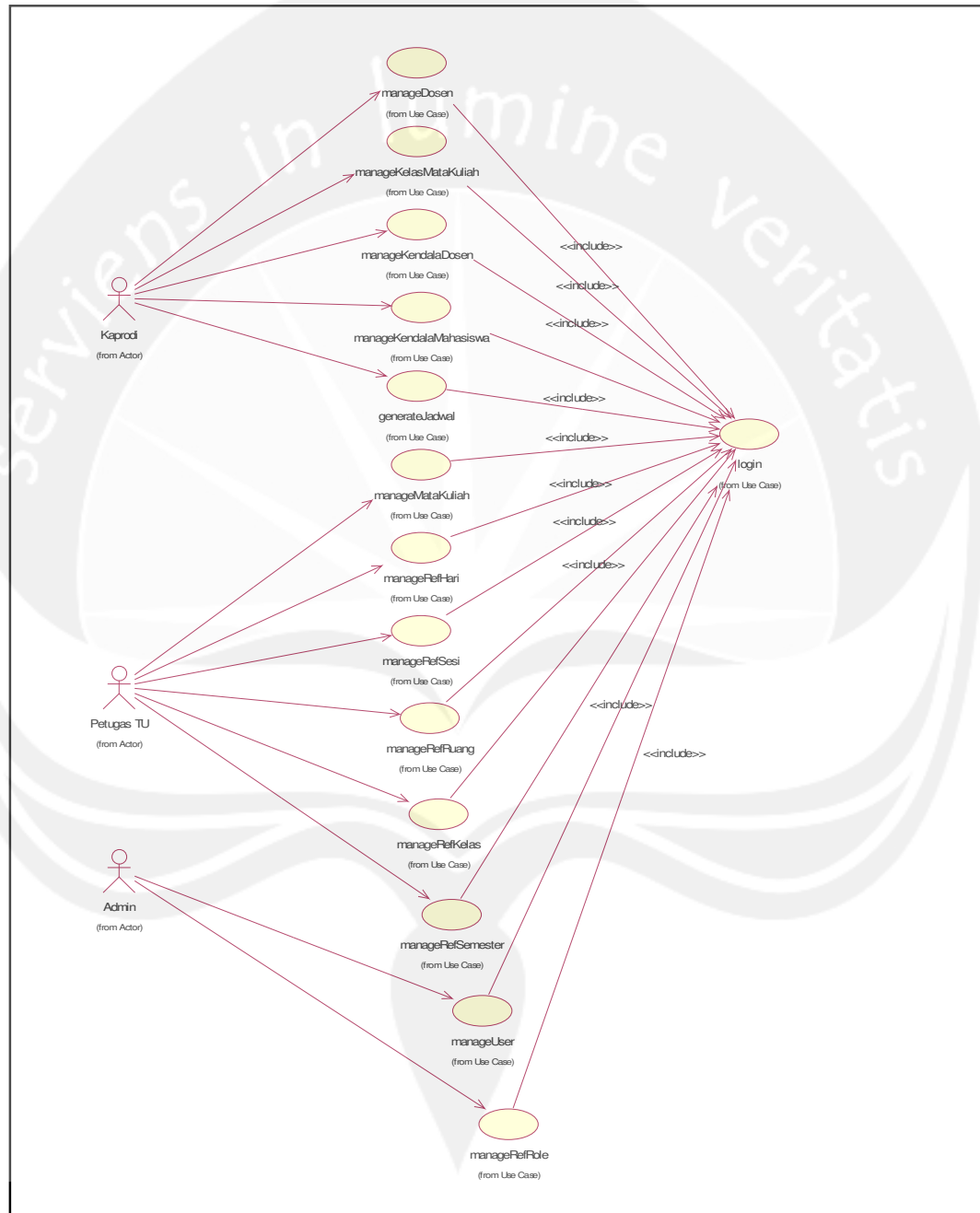
Sumber : Microsoft

Sebagai program yang digunakan untuk membangun sistem SPJK.



3.2 Kebutuhan fungsionalitas Perangkat Lunak

3.2.1 Use Case Diagram



Gambar 2. Use Case Diagram

4 Spesifikasi Rinci Kebutuhan

4.1 Spesifikasi Kebutuhan Fungsionalitas

4.1.1 Use case Spesification : login

1. Brief Description

Use Case ini digunakan oleh aktor untuk masuk ke dalam aplikasi SPJK.

2. Primary Actor

1. Kaprodi, Petugas TU, Admin.

3. Supporting Actor

None.

4. Basic Flow

1. Aktor mengisi nama dan password.
2. Aktor menekan *login*.
3. Sistem memeriksa nama dan password.
4. Sistem masuk ke dalam aplikasi utama.
5. Use Case ini selesai.

5. Alternative Flow

None.

6. Error Flow

E-1. Setelah Basic Flow path 3.

1. Sistem menampilkan pesan bahwa nama belum dimasukkan.
2. Kembali ke Basic Flow path 1.

E-2. Setelah Basic Flow path 3.

1. Sistem menampilkan pesan bahwa password belum dimasukkan.
2. Kembali ke Basic Flow path 1.

E-3. Setelah Basic Flow path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Basic Flow path 1.

7. PreConditions

Use Case ini dimulai ketika aktor akan melakukan login untuk masuk ke dalam aplikasi SPJK.

Program Studi Magister Teknik Informatika	SKPL – SPJK	21/ 55
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

8. PostConditions

1. User (Kaprod, Petugas TU, atau Admin) memasuki system.



4.1.2 Use case Spesification : generateJadwal

1. Brief Description

Use Case ini digunakan oleh aktor untuk menciptakan jadwal.

2. Primary Actor

1. Kaprodi.

3. Supporting Actor

None.

4. Basic Flow

1. Sistem melakukan *show* untuk menampilkan data.
2. Aktor menekan *generate*.
3. Sistem melakukan *show* untuk menampilkan data
4. Use Case ini selesai.

5. Alternative Flow

- A-1 Setelah Basic Flow path 1.
1. Aktor menekan *report*.
 2. Sistem menampilkan *report* data.
 3. kembali ke Basic Flow path 4.

6. Error Flow

None.

7. PreConditions

1. Use case ini dimulai setelah kaprodi memasuki generate jadwal dari aplikasi.

8. PostConditions

1. Jadwal sudah terbentuk.

4.1.3 Use case Spesification : manageDosen

1. Brief Description

Use Case ini digunakan untuk melakukan manage terhadap dosen.

2. Primary Actor

1. Kaprodi.

3. Supporting Actor

None.

4. Basic Flow

1. Sistem melakukan melakukan *show* untuk menampilkan data.
2. Aktor memilih *add*.
3. Sistem mengisi *Id*.
4. Aktor mengisi Nama, Nip, dan Status aktif.
5. Aktor menekan *Execute*.
6. Sistem melakukan *add*.
7. Use Case ini selesai.

5. Alternative Flow

A-1 Setelah Basic Flow path 1.

1. Aktor memilih *edit*.
2. Aktor mengisi *Id*.
3. Aktor menekan *cek id*.
4. Sistem mengisi Nama, Nip, dan Status aktif sesuai data yang ada.
5. Aktor mengubah Nama, Nip, dan Status aktif.
6. Aktor menekan *Execute*.
7. Sistem melakukan *edit*.
8. Kembali ke Basic Flow path 7.

A-2 Setelah Basic Flow path 1.

1. Aktor memilih *delete*.
2. Aktor mengisi *id*.
3. Aktor menekan *cek id*.

4. Sistem mengisi Nama, Nip, dan Status aktif sesuai data yang ada.
5. Aktor menekan *Execute*.
6. Sistem melakukan *delete*.
7. Kembali ke Basic Flow path 7.

6. Error Flow

E-1 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Nama belum diisi.
2. Kembali ke Basic Flow A-1 path 4.

E-2 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Nip belum diisi.
2. Kembali ke Basic Flow A-1 path 4.

E-3 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-4 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-1 path 2.

E-5 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-6 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Nama belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-7 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Nip belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-8 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

E-9 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-2 path 2.

E-10 Setelah Alternative Flow A-2 path 5.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

7. PreConditions

1. Use case ini dimulai setelah kaprodi memasuki manage dosen dari aplikasi.

8. PostConditions

1. dosen telah termanage.



4.1.4 Use case Spesification : manageKendalaDosen

1. Brief Description

Use Case ini digunakan untuk melakukan manage terhadap kendalaDosen.

2. Primary Actor

1. Kaprodi.

3. Supporting Actor

None.

4. Basic Flow

1. Sistem melakukan melakukan *show* untuk menampilkan data.
2. Aktor memilih *add*.
3. Sistem mengisi *Id*.
4. Aktor mengisi Dosen, Hari, Sesi, dan Nilai kendala.
5. Aktor menekan *execute*.
6. Sistem melakukan *add*.
7. Use Case ini selesai.

5. Alternative Flow

A-1 Setelah Basic Flow path 1.

1. Aktor memilih *edit*.
2. Aktor mengisi *Id*.
3. Aktor menekan *cek id*.
4. Sistem mengisi Dosen, Hari, Sesi, dan Nilai kendala sesuai data yang ada.
5. Aktor mengubah Dosen, Hari, Sesi, dan Nilai kendala.
6. Aktor menekan *execute*.
7. Sistem melakukan *edit*.
8. Kembali ke Basic Flow path 7.

A-2 Setelah Basic Flow path 1.

1. Aktor memilih *delete*.
2. Aktor mengisi *Id*.
3. aktor menekan *cek id*.

4. Sistem mengisi Dosen, Hari, Sesi, dan Nilai kendala sesuai data yang ada.
5. Aktor menekan *execute*.
6. Sistem melakukan *delete*.
7. Kembali ke Basic Flow path 7.

6. Error Flow

E-1 Setelah Basic Flow A-1 path 5.

1. Sistem menampilkan pesan bahwa Nilai kendala belum diisi.
2. Kembali ke Basic Flow path 4.

E-2 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Dosen belum diisi.
2. Kembali ke Basic Flow path 4.

Setelah Basic Flow A-1 path 5.

1. Sistem menampilkan pesan bahwa Hari belum diisi.
2. Kembali ke Basic Flow path 4.

E-4 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Sesi belum diisi.
2. Kembali ke Basic Flow path 7.

E-5 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-6 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-1 path 2.

E-7 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-8 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Nilai kendala belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-9 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Dosen belum diisi.

Program Studi Magister Teknik Informatika	SKPL – SPJK	28/ 55
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Kembali ke Alternative Flow A-1 path 5.

E-10 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Hari belum diisi.

2. Kembali ke Alternative Flow A-1 path 5.

E-11 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Sesi belum diisi.

2. Kembali ke Alternative Flow A-1 path 5.

E-12 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.

2. Kembali ke Alternative Flow A-2 path 2.

E-13 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.

2. Kembali ke Alternative Flow A-2 path 2.

E-14 Setelah Alternative Flow A-2 path 5.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.

2. Kembali ke Alternative Flow A-2 path 2.

7. PreConditions

1. Use case ini dimulai setelah kaprodi memasuki manage kendalaDosen dari aplikasi.

8. PostConditions

1. kendalaDosen telah termanage.

4.1.5 Use case Specification :
manageKelasMataKuliah

1. Brief Description

Use Case ini digunakan untuk melakukan manage terhadap kelasMataKuliah.

2. Primary Actor

1. Kaprodi.

3. Supporting Actor

None.

4. Basic Flow

1. Sistem melakukan melakukan *show* untuk menampilkan data.
2. Aktor memilih *add*.
3. Sistem mengisi *Id*.
4. Aktor mengisi *MK, Dosen dan Kelas*.
5. Aktor menekan *execute*.
6. Sistem melakukan *add*.
7. Use Case ini selesai.

5. Alternative Flow

A-1 Setelah Basic Flow path 1.

1. Aktor memilih *edit*.
2. Aktor mengisi *Id*.
3. Aktor menekan *cek id*.
4. Sistem mengisi *MK, Dosen, dan Kelas* sesuai data yang ada.
5. Aktor mengubah *MK, Dosen, dan Kelas*.
6. Aktor menekan *execute*.
7. Sistem melakukan *edit*.
8. Kembali ke Basic Flow path 7.

A-2 Setelah Basic Flow path 1.

1. Aktor memilih *delete*.
2. Aktor mengisi *Id*.
3. Aktor menekan *cek id*.

Program Studi Magister Teknik Informatika	SKPL – SPJK	30/ 55
---	-------------	--------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

4. Sistem mengisi MK, Dosen, dan Kelas sesuai data yang ada.
5. Aktor menekan *execute*.
6. Sistem melakukan *delete*.
7. Kembali ke Basic Flow path 7.

6. Error Flow

E-1 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa MK belum diisi.
2. Kembali ke Basic Flow path 4.

E-2 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Dosen belum diisi.
2. Kembali ke Basic Flow path 4.

E-3 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Kelas belum diisi.
2. Kembali ke Basic Flow path 4.

E-4 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-5 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-1 path 2.

E-6 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-7 Setelah alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa MK belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-8 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Dosen belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-9 Setelah alternative flow A-1 path 6.

1. Sistem menampilkan pesan bahwa kelas belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-10 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

E-11 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-2 path 2.

E-12 Setelah Alternative Flow A-2 path 5.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

7. PreConditions

1. Use case ini dimulai setelah kaprodi memasuki manage mataKuliah dari aplikasi.

8. PostConditions

1. kelasMataKuliah telah termanage.

4.1.6 Use case Specification :
manageKendalaMahasiswa

1. Brief Description

Use Case ini digunakan untuk melakukan manage terhadap kendalaMahasiswa.

2. Primary Actor

1. Kaprodi.

3. Supporting Actor

None.

4. Basic Flow

1. Sistem melakukan melakukan *show* untuk menampilkan data.
2. Aktor memilih *add*.
3. Sistem mengisi *Id*.
4. Aktor mengisi Nilai kendala, Hari, dan Sesi.
5. Aktor menekan *execute*.
6. Sistem melakukan *add*.
7. Use Case ini selesai.

5. Alternative Flow

A-1 Setelah Basic Flow path 1.

1. Aktor memilih *edit*.
2. Aktor mengisi *Id*.
3. Aktor menekan *cek id*.
4. Sistem mengisi Nilai kendala, Hari, dan Sesi sesuai data yang ada.
5. Aktor mengubah Nilai kendala, Hari, dan Sesi.
6. Aktor menekan *execute*.
7. Sistem melakukan *edit*.
8. Kembali ke Basic Flow path 7.

A-2 Setelah basic flow path 1.

1. Aktor memilih *delete*.
2. Aktor mengisi *id*.
3. Aktor menekan *cek id*.

Program Studi Magister Teknik Informatika	SKPL – SPJK	33/ 55
---	-------------	--------

4. Sistem mengisi Nilai kendala, Hari, dan Sesi sesuai data yang ada.
5. Aktor menekan *execute*.
6. Sistem melakukan *delete*.
7. Kembali ke Basic Flow path 7.

6. Error Flow

E-1 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Nilai kendala belum diisi.
2. Kembali ke Basic Flow path 4.

E-2 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Hari belum diisi.
2. Kembali ke Basic Flow path 4.

E-3 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Sesi belum diisi.
2. Kembali ke Basic Flow path 4.

E-4 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-5 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-1 path 2.

E-6 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-7 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Nilai Kendala belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-8 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Hari belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-9 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Sesi belum diisi.

2. Kembali ke Alternative Flow A-1 path 5.

E-10 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.

2. Kembali ke Alternative Flow A-2 path 2.

E-11 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.

2. Kembali ke Alternative Flow A-2 path 2.

E-12 Setelah Alternative Flow A-2 path 5.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.

2. Kembali ke Alternative Flow A-2 path 2.

7. PreConditions

1. Use case ini dimulai setelah kaprodi memasuki manage kendalaMahasiswa dari aplikasi.

8. PostConditions

1. kendalaMahasiswa telah termanage.

4.1.7 Use case Spesification : manageMataKuliah

1. Brief Description

Use Case ini digunakan untuk melakukan manage terhadap mataKuliah.

2. Primary Actor

1. Petugas TU.

3. Supporting Actor

None.

4. Basic Flow

1. Sistem melakukan melakukan *show* untuk menampilkan data.
2. Aktor memilih *Add*.
3. Sistem mengisi *id*.
4. Aktor mengisi *kodeMK*, *namaMK*, *semester* dan *jumlahSKS*.
5. Aktor menekan *execute*.
6. Sistem melakukan *add*.
7. Use Case ini selesai.

5. Alternative Flow

A-1 Setelah Basic Flow path 1.

1. Aktor memilih *edit*.
2. Aktor mengisi *id*.
3. Aktor menekan *cek id*.
4. Sistem mengisi *kodeMK*, *namaMK*, *semester* dan *jumlahSKS* sesuai data yang ada.
5. Aktor mengubah *kodeMK*, *namaMK*, *semester* dan *jumlahSKS*.
6. Aktor menekan *execute*.
7. Sistem melakukan *edit*.
8. Kembali ke Basic Flow path 7.

A-2 Setelah basic flow path 1.

1. Aktor memilih *delete*.
2. Aktor mengisi *id*.
3. Aktor menekan *cek id*.

4. Sistem mengisi kodeMK, namaMK, semester dan jumlahSKS sesuai data yang ada.
5. Aktor menekan *execute*.
6. Sistem melakukan *delete*.
7. Kembali ke Basic Flow path 7.

6. Error Flow

E-1 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa KodeMK belum diisi.
2. Kembali ke Basic Flow Flow path 4.

E-2 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa NamaMK belum diisi.
2. Kembali ke Basic Flow path 4.

E-3 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Semester belum diisi.
2. Kembali ke Basic Flow path 4.

E-4 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa JumlahSKS belum diisi.
2. Kembali ke Basic Flow path 4.

E-5 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-6 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-1 path 2.

E-7 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-8 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa KodeMK belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-9 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa NamaMK belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-10 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Semester belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-11 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa JumlahSKS belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-12 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

E-13 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-2 path 2.

E-14 Setelah Alternative Flow A-2 path 5.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

7. PreConditions

1. Use case ini dimulai setelah kaprodi memasuki manage mataKuliah dari aplikasi.

8. PostConditions

1. mataKuliah telah termanage.

4.1.8 Use case Spesification : manageRefSesi

1. Brief Description

Use Case ini digunakan untuk melakukan manage terhadap refSesi.

2. Primary Actor

1. Petugas TU.

3. Supporting Actor

None.

4. Basic Flow

1. Sistem melakukan melakukan *show* untuk menampilkan data.
2. Aktor memilih *add*.
3. Sistem mengisi *id*.
4. Aktor mengisi Jam Mulai dan Jam Selesai.
5. Aktor menekan *execute*.
6. Sistem melakukan *add*.
7. Use Case ini selesai.

5. Alternative Flow

A-1 Setelah Basic Flow path 1.

1. Aktor memilih *edit*.
2. Aktor mengisi *id*.
3. Aktor menekan *cek id*.
4. Sistem mengisi Jam Mulai, Jam Selesai dan *isDeleted* sesuai data yang ada.
5. Aktor mengubah Jam Mulai, Jam Selesai, dan *isDeleted*.
6. Aktor menekan *execute*.
7. Sistem melakukan *edit*.
8. Kembali ke Basic Flow path 7.

A-2 Setelah Basic Flow path 1.

1. Aktor memilih *delete*.
2. Aktor mengisi *id*.
3. Aktor menekan *cek id*.

4. Sistem mengisi Jam Mulai dan jam Selesai sesuai data yang ada.
5. Aktor menekan *execute*.
6. Sistem melakukan *delete*.
7. Kembali ke Basic Flow path 7.

6. Error Flow

- E-1 Setelah Basic Flow path 5.
1. Sistem menampilkan pesan bahwa jamMulai belum diisi.
 2. Kembali ke Basic Flow path 4.
- E-2 Setelah Basic flow path 5.
1. Sistem menampilkan pesan bahwa jamSelesai belum diisi.
 2. Kembali ke Basic Flow path 4.
- E-5 Setelah Alternative Flow A-1 path 3.
1. Sistem menampilkan pesan bahwa Id belum diisi.
 2. Kembali ke Alternative Flow A-1 path 2.
- E-6 Setelah Alternative Flow A-1 path 3.
1. Sistem menampilkan pesan bahwa data tidak ditemukan.
 2. Kembali ke Alternative Flow A-1 path 2.
- E-7 Setelah Alternative Flow A-1 path 6.
1. Sistem menampilkan pesan bahwa Id belum diisi.
 2. Kembali ke Alternative Flow A-1 path 2.
- E-8 Setelah alternative flow A-1 path 6.
1. Sistem menampilkan pesan bahwa jamMulai belum diisi.
 2. Kembali ke Alternative Flow A-1 path 5.
- E-9 Setelah alternative flow A-1 path 6.
1. Sistem menampilkan pesan bahwa jamSelesai belum diisi.
 2. Kembali ke Alternative Flow A-1 path 5.
- E-10 Setelah Alternative Flow A-2 path 3.
3. Sistem menampilkan pesan bahwa Id belum dimasukkan.
 4. Kembali ke Alternative Flow A-2 path 2.
- E-11 Setelah Alternative Flow A-2 path 3.
1. Sistem menampilkan pesan bahwa data tidak ditemukan.
 2. Kembali ke Alternative Flow A-2 path 2.

E-12 Setelah Alternative Flow A-2 path 5.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

7. PreConditions

1. Use case ini dimulai setelah petugas TU memasuki manage refSesi dari aplikasi.

8. PostConditions

1. refSesi telah termanage.



4.1.9 Use case Spesification : manageRefRuang

1. Brief Description

Use Case ini digunakan untuk melakukan manage terhadap refRuang.

2. Primary Actor

1. Petugas TU.

3. Supporting Actor

None.

4. Basic Flow

1. Sistem melakukan melakukan *show* untuk menampilkan data.
2. Aktor memilih *add*.
3. Sistem mengisi *id*.
4. Aktor mengisi Ruang.
5. Aktor menekan *execute*.
6. Sistem melakukan *add*.
7. Use Case ini selesai.

5. Alternative Flow

A-1 Setelah Basic Flow path 1.

1. Aktor memilih *edit*.
2. Aktor mengisi *id*.
3. Aktor menekan *cek id*.
4. Sistem mengisi Ruang dan *isDeleted*.
5. Aktor mengisi Ruang dan *isDeleted*.
6. Aktor menekan *execute*.
7. Sistem melakukan *edit*.
8. Kembali ke Basic Flow path 7.

A-2 Setelah basic flow path 1.

1. Aktor memilih *delete*.
2. Aktor mengisi *id*.
3. Aktor menekan *cek id*.
4. Sistem mengisi Ruang.
5. Aktor menekan *execute*.

Program Studi Magister Teknik Informatika	SKPL – SPJK	42/ 55
---	-------------	--------

6. Sistem melakukan *delete*.
7. Kembali ke Basic Flow path 7.

6. Error Flow

E-1 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Ruang belum diisi.
2. Kembali ke path 4.

E-2 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-3 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-1 path 2.

E-4 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-5 Setelah alternative flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Ruang belum diisi.
2. Kembali ke alternative flow A-1 path 5.

E-6 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

E-7 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-2 path 2.

E-8 Setelah Alternative Flow A-2 path 5.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

7. PreConditions

1. Use case ini dimulai setelah petugas TU memasuki manage refRuang dari aplikasi.

8. PostConditions

1. refRuang telah termanage.

4.1.10 Use case Spesification : manageRefHari

1. Brief Description

Use Case ini digunakan untuk melakukan manage terhadap refHari.

2. Primary Actor

1. Petugas TU.

3. Supporting Actor

None.

4. Basic Flow

1. Sistem melakukan melakukan *show* untuk menampilkan data.
2. Aktor memilih *add*.
3. Sistem mengisi *id*.
4. Aktor mengisi *hari*.
5. Aktor menekan *execute*.
6. Sistem melakukan *add*.
7. Use Case ini selesai.

5. Alternative Flow

A-1 Setelah basic flow path 1.

1. Aktor memilih *edit*.
2. Aktor mengisi *id*.
3. Aktor menekan *cek id*.
4. Sistem mengisi *hari* dan *isDeleted*.
5. Aktor mengubah *hari* dan *isDeleted*.
6. Aktor menekan *execute*.
7. Sistem melakukan *edit*.

8. Kembali ke Basic Flow path 7.

A-2 Setelah basic flow path 1.

1. Aktor memilih *delete*.
2. Aktor mengisi *id*.
3. Aktor menekan *cek id*.
4. Sistem mengisi *hari*.
5. Aktor menekan *execute*.

Program Studi Magister Teknik Informatika	SKPL – SPJK	44/ 55
---	-------------	--------

6. Sistem melakukan *delete*.

7. Kembali ke path 7.

6. Error Flow

E-1 Setelah Basic flow path 5.

1. Sistem menampilkan pesan bahwa namaHari belum diisi.

2. Kembali ke Alternative Flow path 4.

E-2 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa Id belum diisi.

2. Kembali ke Alternative Flow A-1 path 2.

E-3 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.

2. Kembali ke Alternative Flow A-1 path 2.

E-4 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Id belum diisi.

2. Kembali ke Alternative Flow A-1 path 2.

E-5 Setelah alternative flow A-1 path 6.

1. Sistem menampilkan pesan bahwa namaHari belum diisi.

2. Kembali ke Alternative Flow path 5.

E-6 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.

2. Kembali ke Alternative Flow A-2 path 2.

E-7 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.

2. Kembali ke Alternative Flow A-2 path 2.

E-8 Setelah Alternative Flow A-2 path 5.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.

2. Kembali ke Alternative Flow A-2 path 2.

7. PreConditions

1. Use case ini dimulai setelah petugas TU memasuki manage refHari dari aplikasi.

8. PostConditions

1. refHari telah termanage.

4.1.11 Use case Spesification : manageRefKelas

1. Brief Description

Use Case ini digunakan untuk melakukan manage terhadap refKelas.

2. Primary Actor

1. Petugas TU.

3. Supporting Actor

None.

4. Basic Flow

1. Sistem melakukan melakukan *show* untuk menampilkan data.
2. Aktor memilih *add*.
3. Aktor mengisi kelas.
4. Aktor menekan *execute*.
5. Sistem melakukan *add*.
6. Use Case ini selesai.

5. Alternative Flow

A-1 Setelah basic flow path 1.

1. Aktor memilih *edit*.
2. Aktor mengisi kelas untuk input.
3. Aktor menekan *cek id*.
4. Sistem mengisi kelas.
5. Aktor mengubah kelas.
6. Aktor menekan *execute*.
7. Sistem melakukan *edit*.
8. Kembali ke Basic Flow path 7.

A-2 Setelah basic flow path 1.

1. Aktor memilih *delete*.
2. Aktor mengisi kelas untuk input.
3. Aktor menekan *cek id*.
4. Sistem mengisi kelas.
5. Aktor menekan *execute*.
6. Sistem melakukan *delete*.

7. Kembali ke path 7.

6. Error Flow

E-1 Setelah Basic flow path 5.

1. Sistem menampilkan pesan bahwa namaHari belum diisi.
2. Kembali ke Alternative Flow path 4.

E-2 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-3 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-1 path 2.

E-4 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-5 Setelah alternative flow A-1 path 6.

1. Sistem menampilkan pesan bahwa namaHari belum diisi.
2. Kembali ke Alternative Flow path 5.

E-6 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

E-7 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-2 path 2.

E-8 Setelah Alternative Flow A-2 path 5.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

2. PreConditions

1. Use case ini dimulai setelah petugas TU memasuki manage refKelas dari aplikasi.

3. PostConditions

1. refKelas telah termanage.

4.1.12 Use case Spesification : manageRefSemester

1. Brief Description

Use Case ini digunakan untuk melakukan manage terhadap refSemester.

2. Primary Actor

1. Petugas TU.

3. Supporting Actor

None.

4. Basic Flow

1. Sistem melakukan melakukan *show* untuk menampilkan data.
2. Aktor memilih *add*.
3. Aktor mengisi semester.
4. Aktor menekan *execute*.
5. Sistem melakukan *add*.
6. Use Case ini selesai.

5. Alternative Flow

A-1 Setelah basic flow path 1.

1. Aktor memilih *edit*.
2. Aktor mengisi kelas untuk input.
3. Aktor menekan *cek id*.
4. Sistem mengisi semester dan *isDeleted*.
5. Aktor mengubah semester dan *isDeleted*.
6. Aktor menekan *execute*.
7. Sistem melakukan *edit*.
8. Kembali ke Basic Flow path 7.

A-2 Setelah basic flow path 1.

1. Aktor memilih *delete*.
2. Aktor mengisi kelas untuk input.
3. Aktor menekan *cek id*.
4. Sistem mengisi semester.
5. Aktor menekan *execute*.
6. Sistem melakukan *delete*.

7. Kembali ke path 7.

6. Error Flow

E-1 Setelah Basic flow path 5.

1. Sistem menampilkan pesan bahwa namaHari belum diisi.
2. Kembali ke Alternative Flow path 4.

E-2 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-3 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-1 path 2.

E-4 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-5 Setelah alternative flow A-1 path 6.

1. Sistem menampilkan pesan bahwa namaHari belum diisi.
2. Kembali ke Alternative Flow path 5.

E-6 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

E-7 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-2 path 2.

E-8 Setelah Alternative Flow A-2 path 5.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

2. PreConditions

1. Use case ini dimulai setelah petugas TU memasuki manage refKelas dari aplikasi.

3. PostConditions

1. refKelas telah termanage.

4.1.13 Use case Spesification : manageUser

1. Brief Description

Use Case ini digunakan untuk melakukan manage terhadap user.

2. Primary Actor

1. Admin.

3. Supporting Actor

None.

4. Basic Flow

1. Sistem melakukan melakukan *show* untuk menampilkan data.
2. Aktor memilih *add*.
3. Sistem mengisi *id*.
4. Aktor mengisi *Role, Nama, Password, Alamat, Telephone, Email*.
5. Aktor menekan *execute*.
6. Sistem melakukan *add*.
7. Use Case ini selesai.

5. Alternative Flow

A-1 Setelah Basic Flow path 1.

1. Aktor memilih *edit*.
2. Aktor mengisi *id*.
3. Aktor menekan *cek id*.
4. Sistem mengisi *Role, Nama, Password, Alamat, Telephone, Email*.
5. Aktor mengubah *Role, Nama, Password, Alamat, Telephone, Email*.
6. Aktor menekan *execute*.
7. Sistem melakukan *edit*.
8. Kembali ke Basic Flow path 7.

A-2 Setelah Basic Flow path 1.

1. Aktor memilih *delete*.
2. Aktor mengisi *id*.

Program Studi Magister Teknik Informatika	SKPL – SPJK	50/ 55
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

3. Aktor menekan *cek id*.
4. Sistem mengisi Role, Nama, Password, Alamat, Telephone, Email.
5. Aktor menekan *execute*.
6. Sistem melakukan *delete*.
7. Kembali ke Basic Flow path 7.

6. Error Flow

E-1 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Role belum diisi.
2. Kembali ke Basic Flow path 4.

E-2 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Nama belum diisi.
2. Kembali ke Basic Flow path 4.

E-3 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Password belum diisi.
2. Kembali ke Basic Flow path 4.

E-4 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Alamat belum diisi.
2. Kembali ke Basic Flow path 4.

E-5 Setelah Basic Flow path 5.

1. Sistem menampilkan pesan bahwa Telephone belum diisi.
2. Kembali ke Basic Flow path 4.

E-6 Setelah Basic Glow path 5.

1. Sistem menampilkan pesan bahwa Email belum diisi.
2. Kembali ke Basic Flow path 4.

E-7 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-8 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-1 path 2.

E-9 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

Program Studi Magister Teknik Informatika	SKPL – SPJK	51/ 55
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

E-10 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Role belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-11 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Nama belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-12 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Password belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-13 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Alamat belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-14 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Telephone belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-15 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Email belum diisi.
2. Kembali ke Alternative Flow A-1 path 5.

E-16 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

E-17 Setelah Alternative Flow A-2 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-2 path 2.

E-18 Setelah Alternative Flow A-2 path 5.

1. Sistem menampilkan pesan bahwa Id belum dimasukkan.
2. Kembali ke Alternative Flow A-2 path 2.

2. PreConditions

1. Use case ini dimulai setelah admin memasuki manage manageUser dari aplikasi.

3. PostConditions

1. user telah termanage.

4.1.14 Use case Spesification : manageRefRole

1. Brief Description

Use Case ini digunakan untuk melakukan manage terhadap refRole.

2. Primary Actor

1. Admin.

3. Supporting Actor

None.

4. Basic Flow

1. Sistem melakukan melakukan *show* untuk menampilkan data.
2. Aktor memilih *add*.
3. Sistem mengisi *id*.
4. Aktor mengisi *role*.
5. Aktor menekan *execute*.
6. Sistem melakukan *add*.
7. Use Case ini selesai.

5. Alternative Flow

A-1 Setelah Basic Flow path 1.

1. Aktor memilih *edit*.
2. Aktor memilih *id*.
3. Aktor menekan *cek id*.
4. Sistem mengisi *role* dan *isDeleted*.
5. Aktor mengubah *role* dan *isDeleted*.
6. Aktor menekan *execute*.
7. Sistem melakukan *edit*.

8. Kembali ke Basic Flow path 7.

A-2 Setelah Basic Flow path 1.

1. Aktor memilih *delete*.
2. Aktor memilih *id*.
3. Aktor menekan *cek id*.
4. Sistem mengisi *role*.
5. Aktor menekan *execute*.

Program Studi Magister Teknik Informatika	SKPL – SPJK	53/ 55
---	-------------	--------

6. Sistem melakukan *delete*.
7. Kembali ke Basic Flow path 7.

6. Error Flow

E-1 Setelah Basic Flow A-1 path 5.

1. Sistem menampilkan pesan bahwa Role belum diisi.
2. Kembali ke path 4.

E-2 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-3 Setelah Alternative Flow A-1 path 3.

1. Sistem menampilkan pesan bahwa data tidak ditemukan.
2. Kembali ke Alternative Flow A-1 path 2.

E-4 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Id belum diisi.
2. Kembali ke Alternative Flow A-1 path 2.

E-5 Setelah Alternative Flow A-1 path 6.

1. Sistem menampilkan pesan bahwa Role belum diisi.
2. Kembali ke Alternative Flow A-1 path 3.

E-6 Setelah Alternative Flow A-2 path 3.

3. Sistem menampilkan pesan bahwa Id belum dimasukkan.
4. Kembali ke Alternative Flow A-2 path 2.

E-7 Setelah Alternative Flow A-2 path 3.

3. Sistem menampilkan pesan bahwa data tidak ditemukan.
4. Kembali ke Alternative Flow A-2 path 2.

E-8 Setelah Alternative Flow A-2 path 5.

3. Sistem menampilkan pesan bahwa Id belum dimasukkan.
4. Kembali ke Alternative Flow A-2 path 2.

7. PreConditions

1. Use case ini dimulai setelah admin memasuki manage refRole dari aplikasi.

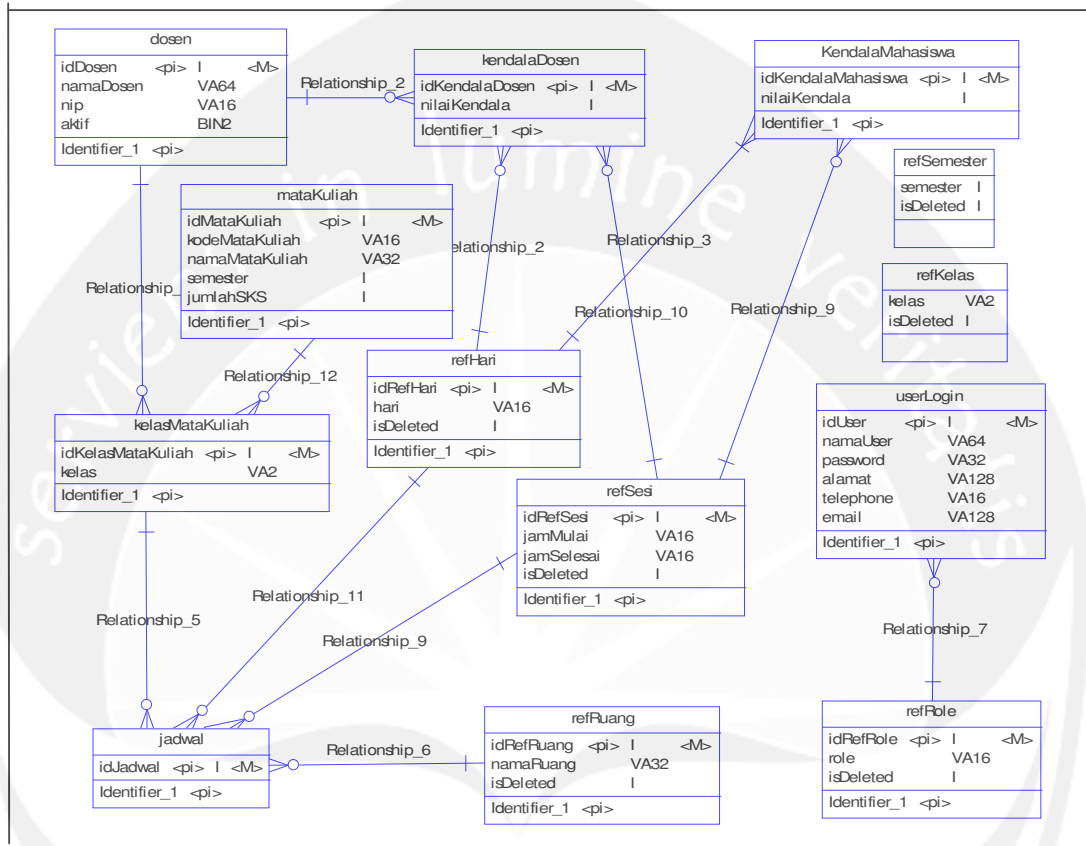
8. PostConditions

1. refRole telah termanage.

Program Studi Magister Teknik Informatika	SKPL – SPJK	54/ 55
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

ERD (Entity Relationship Diagram)

Diagram alir data digambarkan dalam bentuk entity relationship diagram berikut.



Gambar 3. ERD (Entity Relationship Diagram)

LAMPIRAN 2



DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

SISTEM PENGATURAN JADWAL KULIAH

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Rochim Widaryanto / 4362

Program Studi Teknik Informatika - Fakultas Teknologi
Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Magister Teknik Informatika	Nomor Dokumen		Halaman
		DPPL-SPJK		1/102
		Revisi		2013

Program Studi Teknik Informatika	DPPL- SPJK	1/ 102
----------------------------------	------------	--------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	
G	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiksa oleh								
Disetujui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

DAFTAR ISI

1. Pendahuluan.....	7
1.1 Tujuan.....	7
1.2 Ruang Lingkup.....	7
1.3 Definisi dan Akronim.....	7
1.4 Referensi	8
2. Perancangan Sistem (System Design)	9
2.1 Perancangan Arsitektur (Architectural Design)	9
2.2 Perancangan Rinci (Detail Design)	10
3. Perancangan Data (Data Design).....	87
3.1 Pyhsical Data Model.....	87
4. Perancangan Antarmuka (UI Design)	88
4.1 Antarmuka Login.....	88
4.2 Antarmuka Generate Jadwal	89
4.3 Antarmuka Report.....	90
4.4 Antarmuka Manage Dosen	91
4.5 Antarmuka Manage Kelas Mata Kuliah.....	92
4.6 Antarmuka Manage Kendala Mahasiswa.....	93
4.7 Antarmuka Manage Kendala Dosen	94
4.8 Antarmuka Manage Ref Hari	96
4.9 Antarmuka Manage Ref Sesi	97
4.10 Antarmuka Manage Ref Ruang.....	98
4.11 Antarmuka Manage Ref Kelas.....	99
4.12 Antarmuka Manage Ref Semester.....	100
4.13 Antarmuka Manage User.....	101
4.14 Antarmuka Manage Ref Role	102

DAFTAR GAMBAR

Gambar 1	Perancangan Arsitektur.....	9
Gambar 2	Sequence Diagram: Login.....	10
Gambar 3	Sequence Diagram: manageDosen.....	11
Gambar 4	Sequence Diagram: add.....	12
Gambar 5	Sequence Diagram: edit.....	13
Gambar 6	Sequence Diagram: delete.....	14
Gambar 7	Sequence Diagram: generateJadwal.....	15
Gambar 8	Sequence Diagram: generate.....	16
Gambar 9	Sequence Diagram: report.....	17
Gambar 10	Sequence Diagram: manageMataKuliah.....	18
Gambar 11	Sequence Diagram: add.....	19
Gambar 12	Sequence Diagram: edit.....	20
Gambar 13	Sequence Diagram: delete.....	21
Gambar 14	Sequence Diagram: manageKendalaDosen.....	22
Gambar 15	Sequence Diagram: add.....	23
Gambar 16	Sequence Diagram: edit.....	24
Gambar 17	Sequence Diagram: delete.....	25
Gambar 18	Sequence Diagram: manageKendalaMahasiswa.....	26
Gambar 19	Sequence Diagram: add.....	27
Gambar 20	Sequence Diagram: edit.....	28
Gambar 21	Sequence Diagram: delete.....	29
Gambar 22	Sequence Diagram: manageMataKuliah.....	30
Gambar 23	Sequence Diagram: add.....	31
Gambar 24	Sequence Diagram: edit.....	32
Gambar 25	Sequence Diagram: delete.....	33
Gambar 26	Sequence Diagram: manageRefHari.....	34
Gambar 27	Sequence Diagram: add.....	35
Gambar 28	Sequence Diagram: edit.....	36
Gambar 29	Sequence Diagram: delete.....	37
Gambar 30	Sequence Diagram: manageRefRuang.....	38
Gambar 31	Sequence Diagram: add.....	39
Gambar 32	Sequence Diagram: edit.....	40
Gambar 33	Sequence Diagram: delete.....	41
Gambar 34	Sequence Diagram: manageRefSesi.....	42
Gambar 35	Sequence Diagram: add.....	43
Gambar 36	Sequence Diagram: edit.....	44
Gambar 37	Sequence Diagram: delete.....	45
Gambar 38	Sequence Diagram: manageRefKelas.....	46
Gambar 39	Sequence Diagram: add.....	47
Gambar 40	Sequence Diagram: edit.....	48
Gambar 41	Sequence Diagram: delete.....	49
Gambar 42	Sequence Diagram: manageRefSemester.....	50
Gambar 43	Sequence Diagram: add.....	51
Gambar 44	Sequence Diagram: edit.....	52
Gambar 45	Sequence Diagram: delete.....	53
Gambar 46	Sequence Diagram: manageUser.....	54
Gambar 47	Sequence Diagram: add.....	55
Gambar 48	Sequence Diagram: edit.....	56
Gambar 49	Sequence Diagram: delete.....	57
Gambar 50	Sequence Diagram: manageUser.....	58
Gambar 51	Sequence Diagram: add.....	59

Gambar 52	Sequence Diagram: edit.....	60
Gambar 53	Sequence Diagram: delete.....	61
Gambar 54	Class Diagram.....	62
Gambar 55	Entity Relationship Diagram.....	87
Gambar 56	Rancangan Antarmuka Login.....	88
Gambar 57	Rancangan Antarmuka Generate Jadwal.....	89
Gambar 58	Rancangan Antarmuka Report.....	90
Gambar 59	Rancangan Antarmuka Manage Dosen.....	91
Gambar 60	Rancangan Antarmuka Manage Kelas Mata Kuliah.....	92
Gambar 61	Rancangan Antarmuka Manage Kendala Mahasiswa.....	93
Gambar 62	Rancangan Antarmuka Manage Kendala Dosen.....	94
Gambar 63	Rancangan Antarmuka Manage Mata Kuliah.....	95
Gambar 64	Rancangan Antarmuka Manage Ref Hari.....	96
Gambar 65	Rancangan Antarmuka Manage Ref Sesi.....	97
Gambar 66	Rancangan Antarmuka Manage Ref Ruang.....	98
Gambar 67	Rancangan Antarmuka Manage Ref Kelas.....	99
Gambar 68	Rancangan Antarmuka Manage Ref Semester.....	100
Gambar 69	Rancangan Antarmuka Manage User.....	101
Gambar 70	Rancangan Antarmuka Manage Ref Role.....	102

1. Pendahuluan

1.1 Tujuan

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) SPJK bertujuan untuk mendefinisikan perancangan perangkat lunak yang akan dikembangkan. Dokumen DPPL tersebut digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap selanjutnya.

1.2 Ruang Lingkup

Perangkat Lunak SPJK dibangun dengan tujuan untuk membantu kaprodi dalam menyelesaikan timetable perkuliahan. Permasalahan timetable akan diselesaikan dengan mengeksplorasi algoritma DPSO agar komputasi menjadi lebih optimal. SPJK berjalan pada platform berbasis windows. Dengan SPJK ini diharapkan kaprodi dapat menyelesaikan timetable dengan optimal.

1.3 Definisi dan Akronim

Tabel 1.1 Daftar definisi akronim dan singkatan

Keyword/Phrase	Definisi
DPPL	Deskripsi Perancangan Perangkat Lunak disebut juga Software Design Description (SDD) merupakan deskripsi dari perancangan produk/perangkat lunak yang akan dikembangkan.
SPJK	Program auto generate Timetable dengan menggunakan algoritma discrete particle swarm optimization.

1.4 Referensi

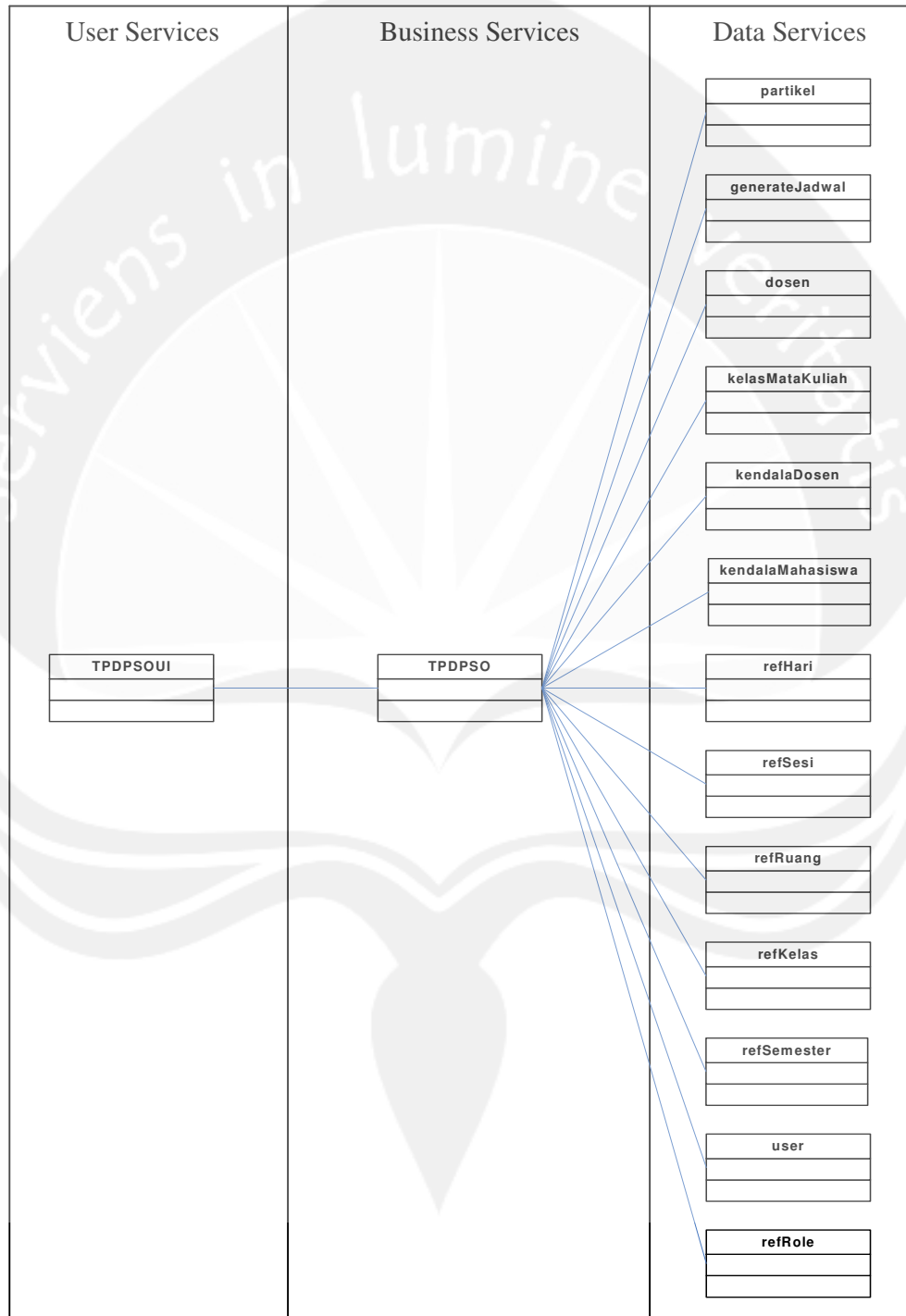
Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Aryo Wibisono, Deskripsi Perangkat Lunak Sistem Informasi Akademik Mahasiswa (SIATMA), Universitas Atma Jaya Yogyakarta.



2. Perancangan Sistem (System Design)

2.1 Perancangan Arsitektur (Architectural Design)

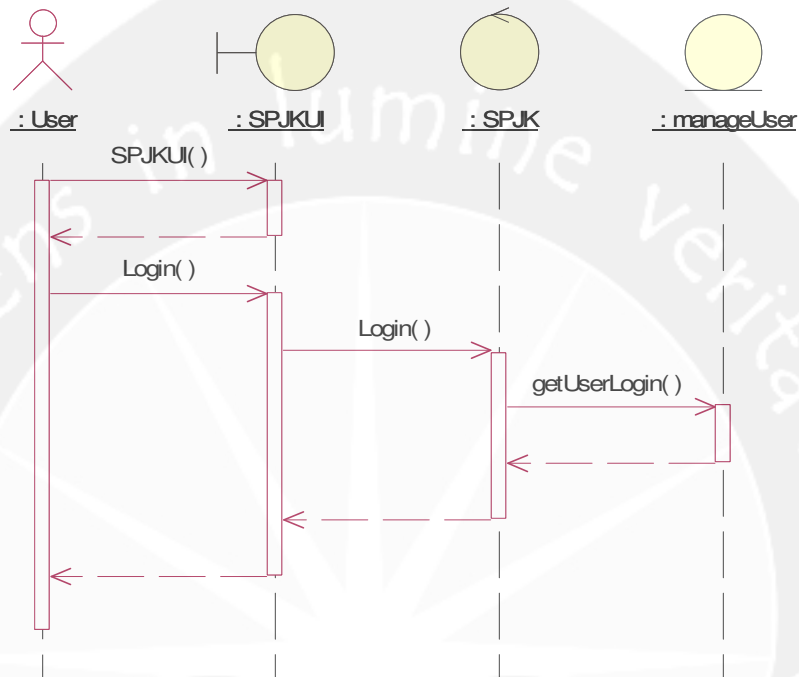


Gambar 1. Perancangan Arsitektur

2.2 Perancangan Rinci (Detail Design)

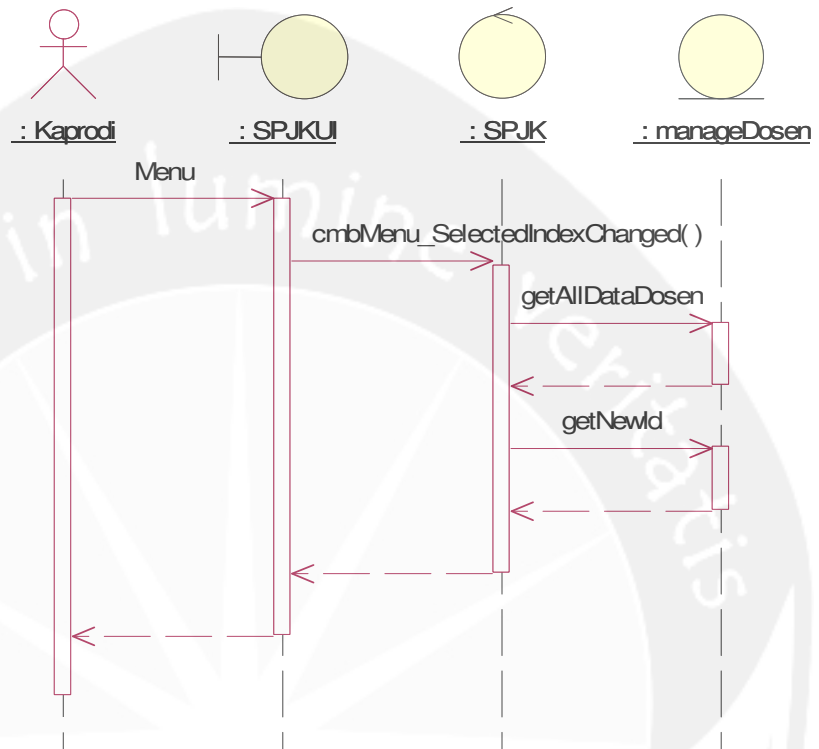
a. Sequence Diagram

1. login



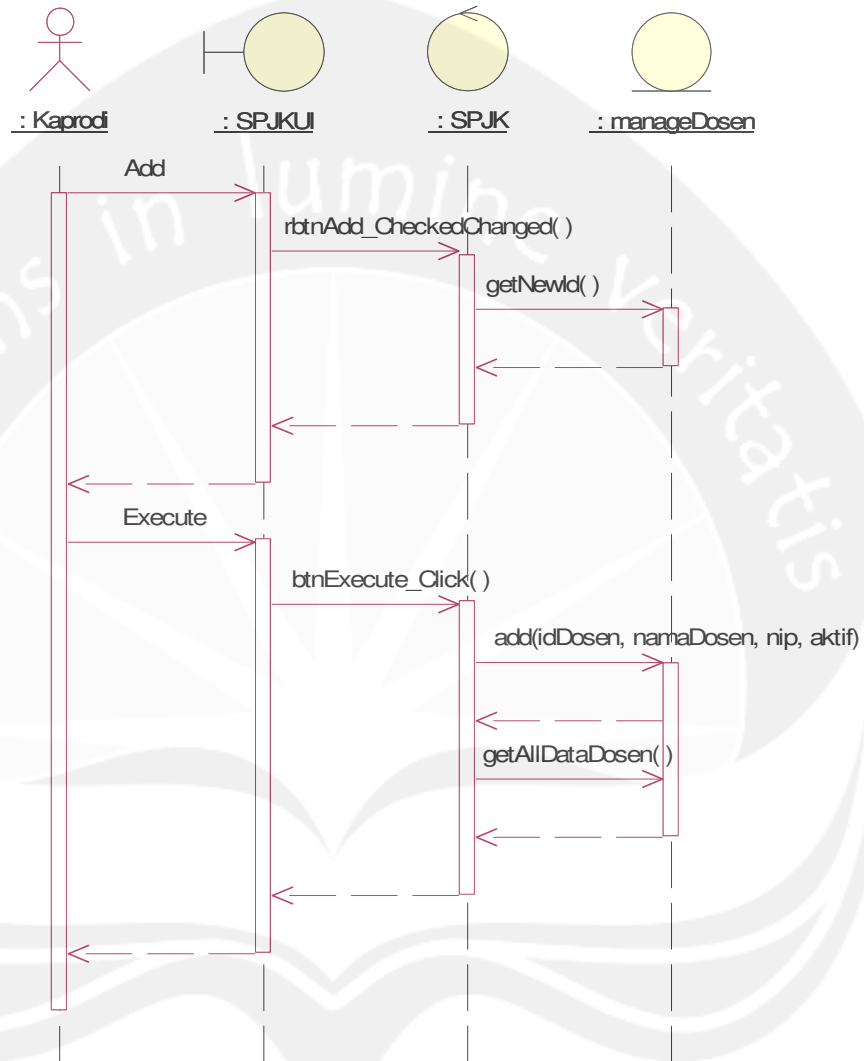
Gambar 2. Sequence Diagram: login

2. manageDosen



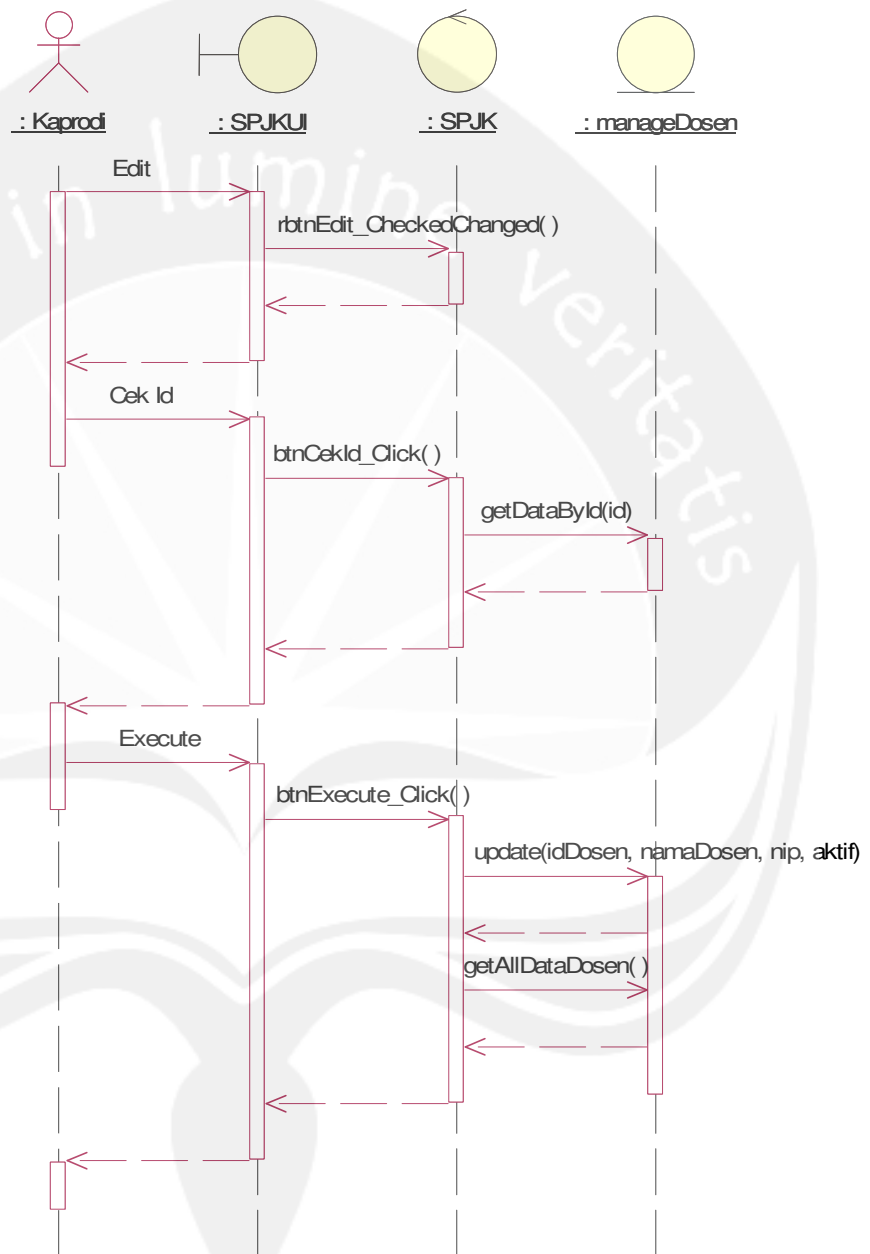
Gambar 3. Sequence Diagram: manageDosen

2.1. add



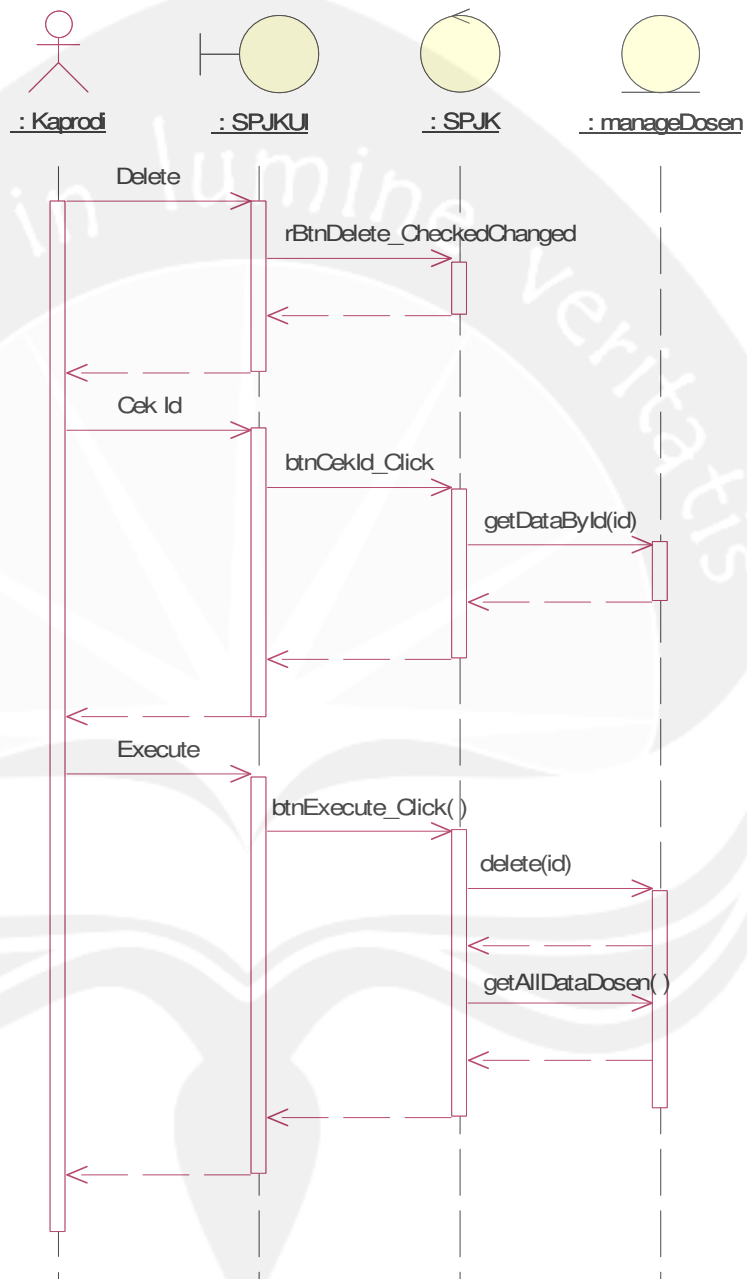
Gambar 4. Sequence Diagram: Add

2.2. edit



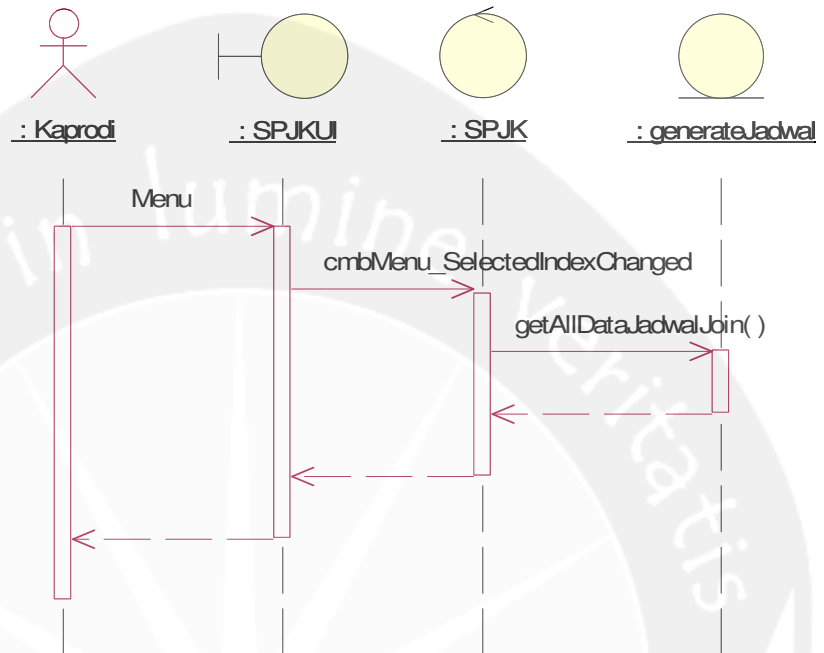
Gambar 5 Sequence Diagram: edit

2.3. delete



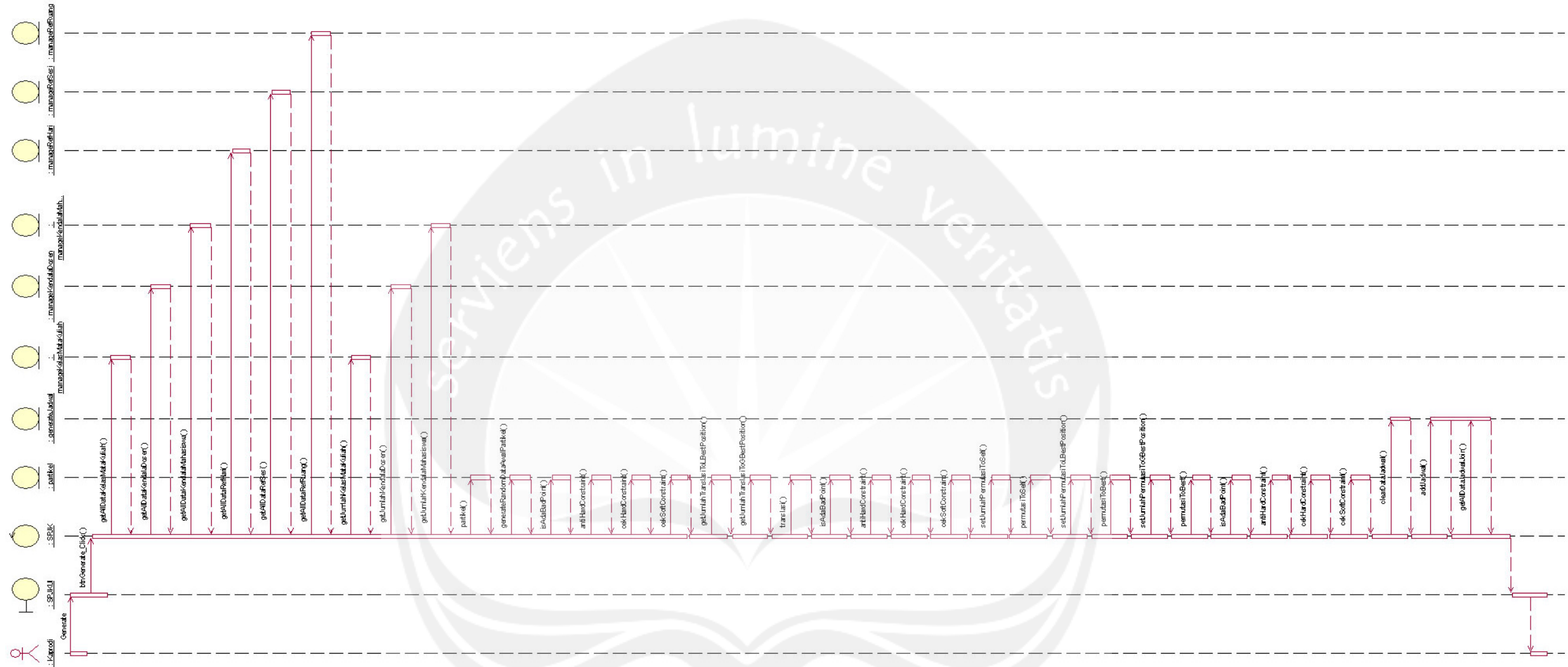
Gambar 6 Sequence Diagram: delete

3. generateJadwal



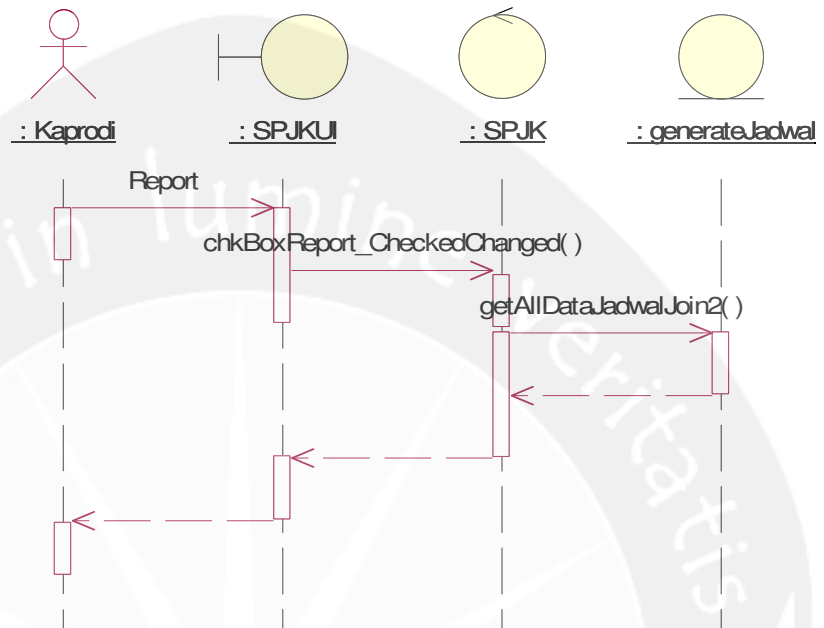
Gambar 7. Sequence Diagram: generateJadwal

3.1. generate



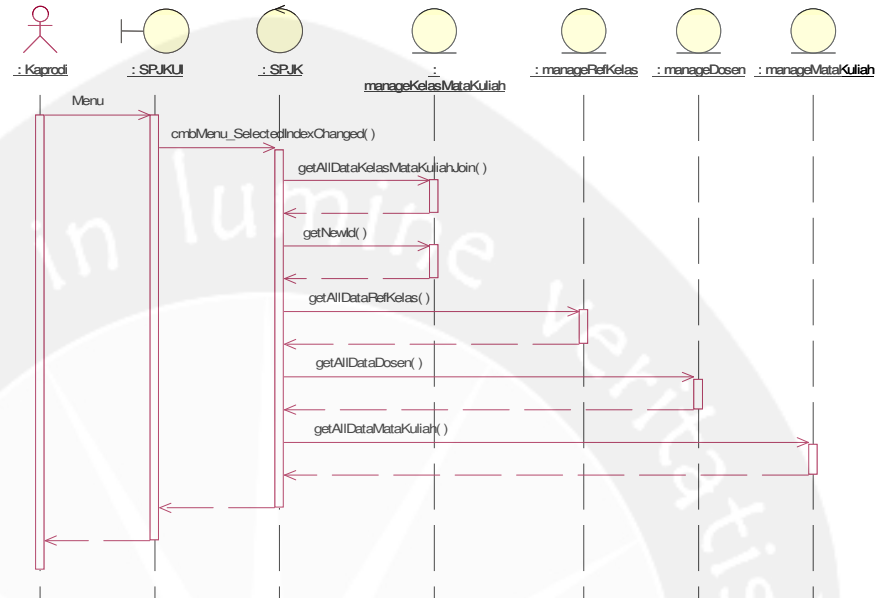
Gambar 8. Sequence Diagram: generate

3.2. Report



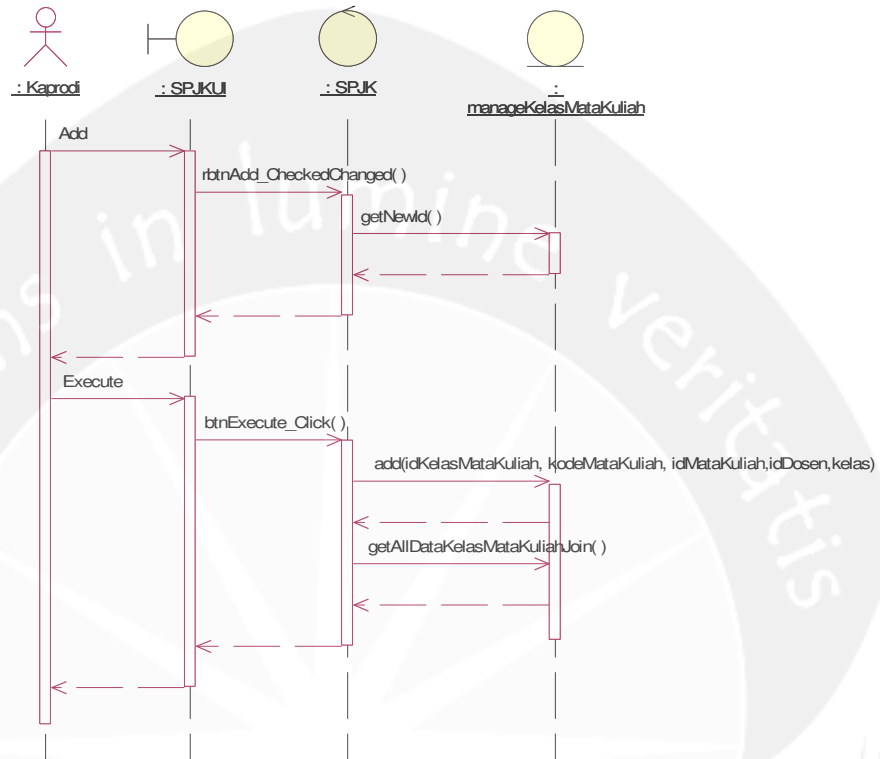
Gambar 9. Sequence Diagram: Report

4. manageKelasMataKuliah



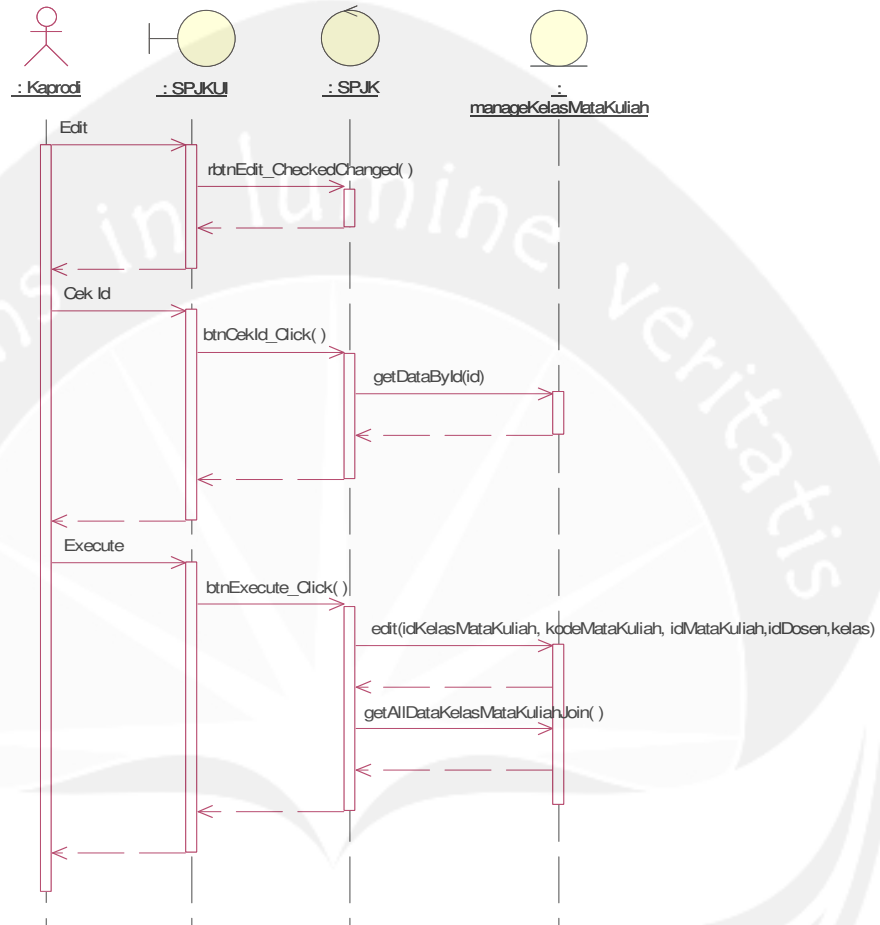
Gambar 10. Sequence Diagram: manageKelasMataKuliah

4.1.1. add



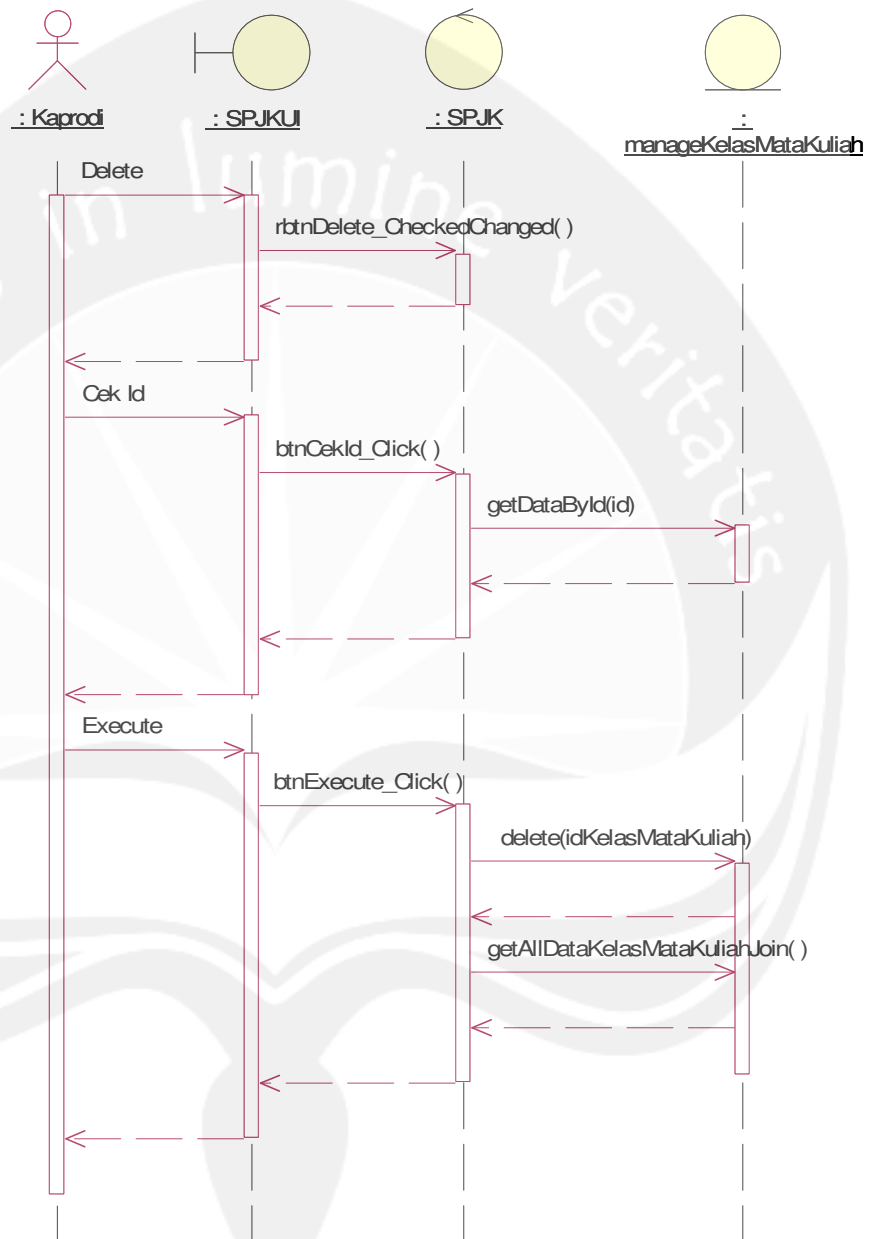
Gambar 11. Sequence Diagram: add

4.2. edit



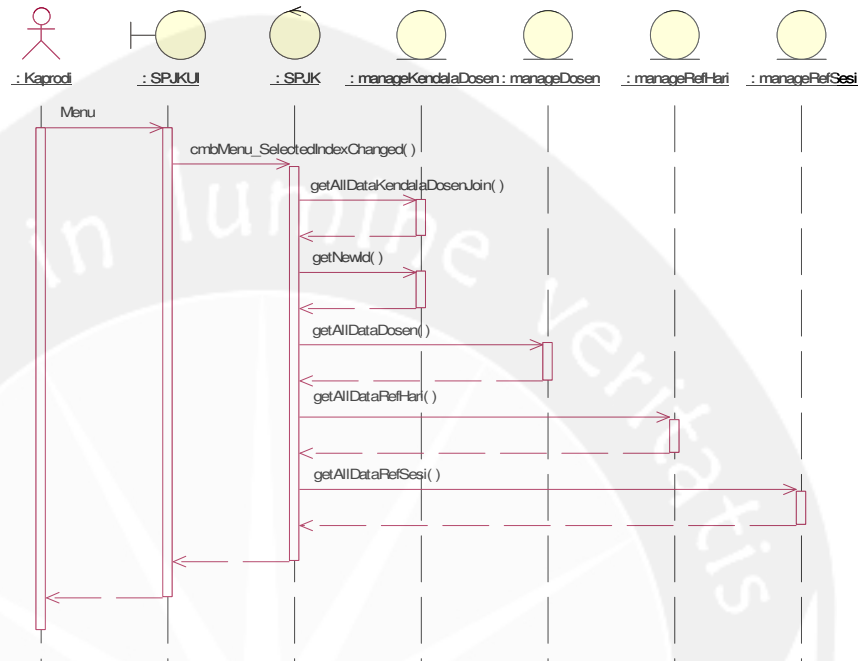
Gambar 12. Sequence Diagram: edit

4.3. delete



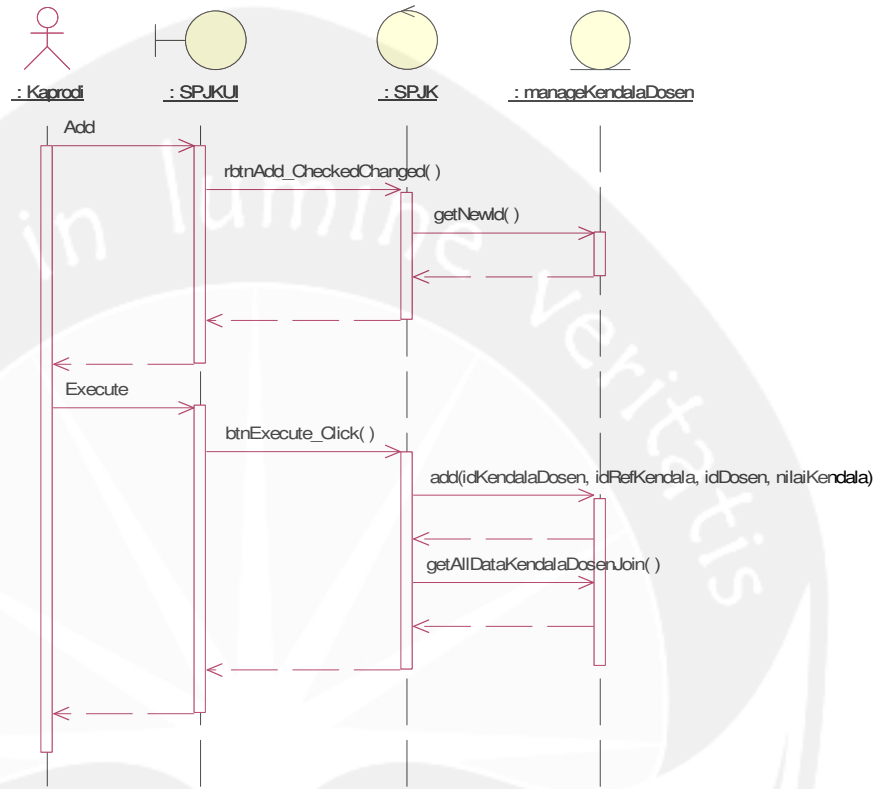
Gambar 13. Sequence Diagram: delete

5. manageKendalaDosen



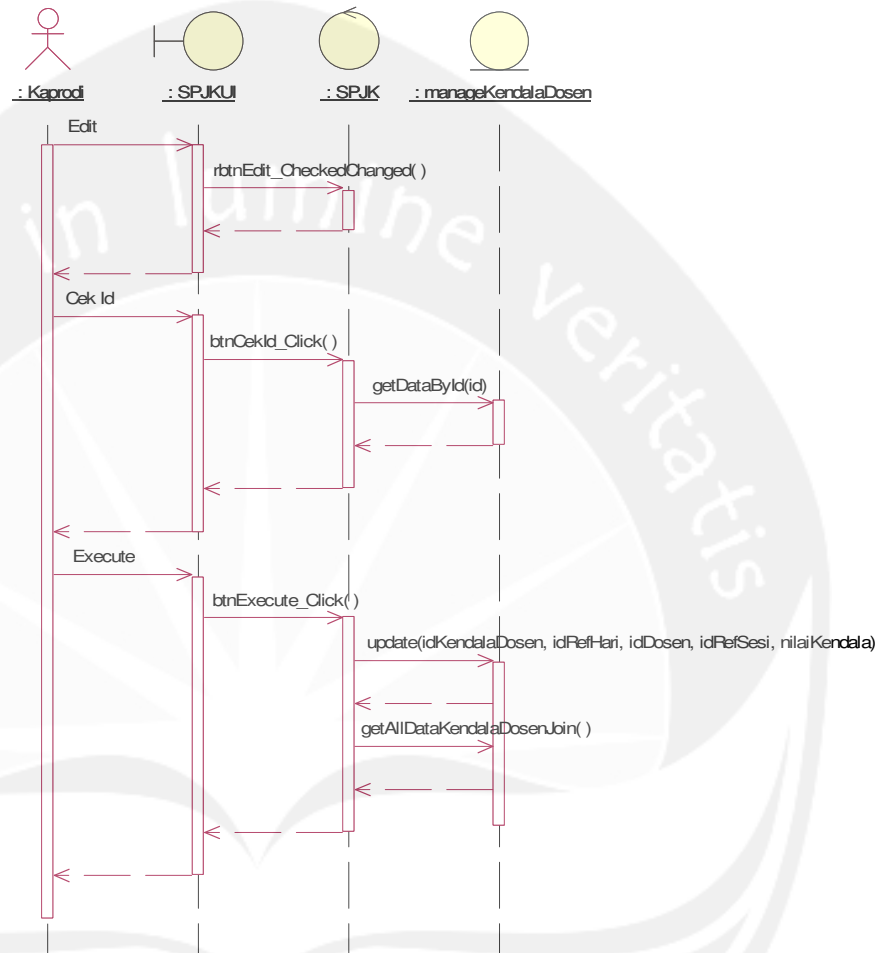
Gambar 14. Sequence Diagram: manageKendalaDosen

5.1. add



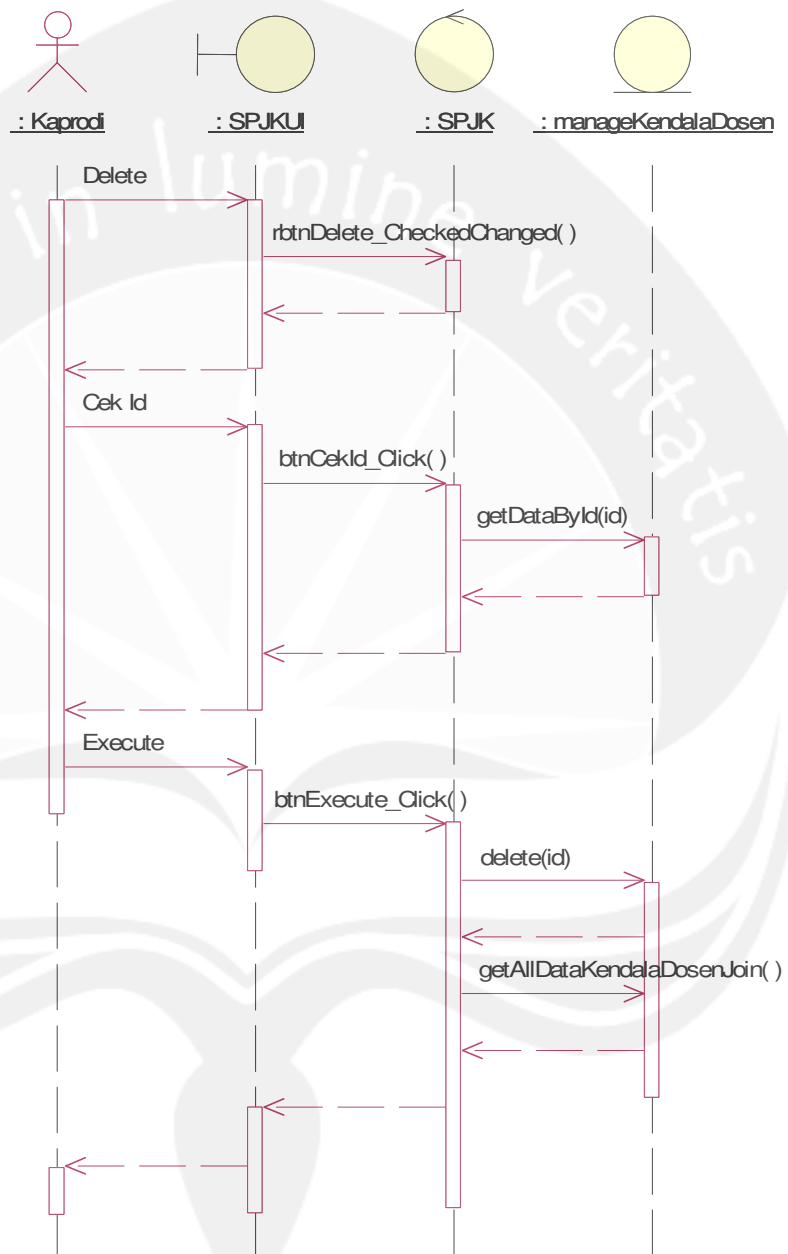
Gambar 15. Sequence Diagram: add

5.2. edit



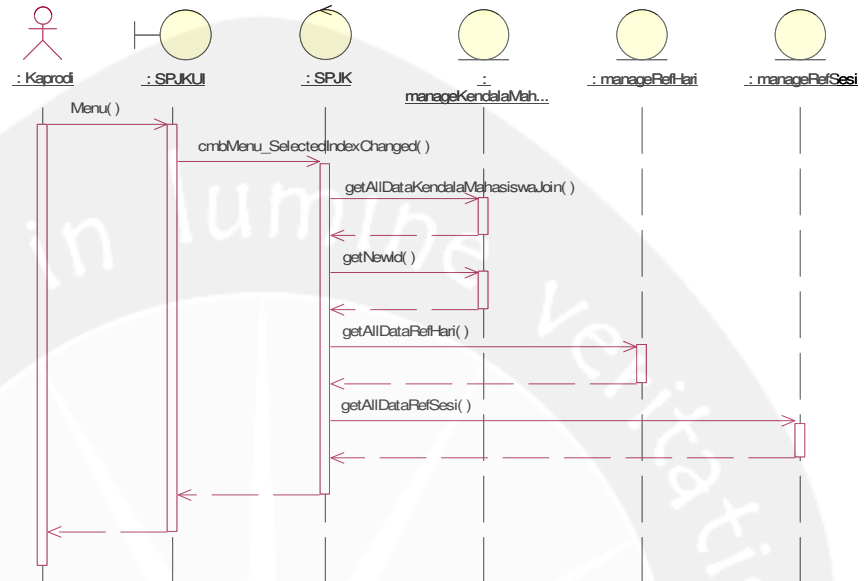
Gambar 16. Sequence Diagram: edit

5.3. delete



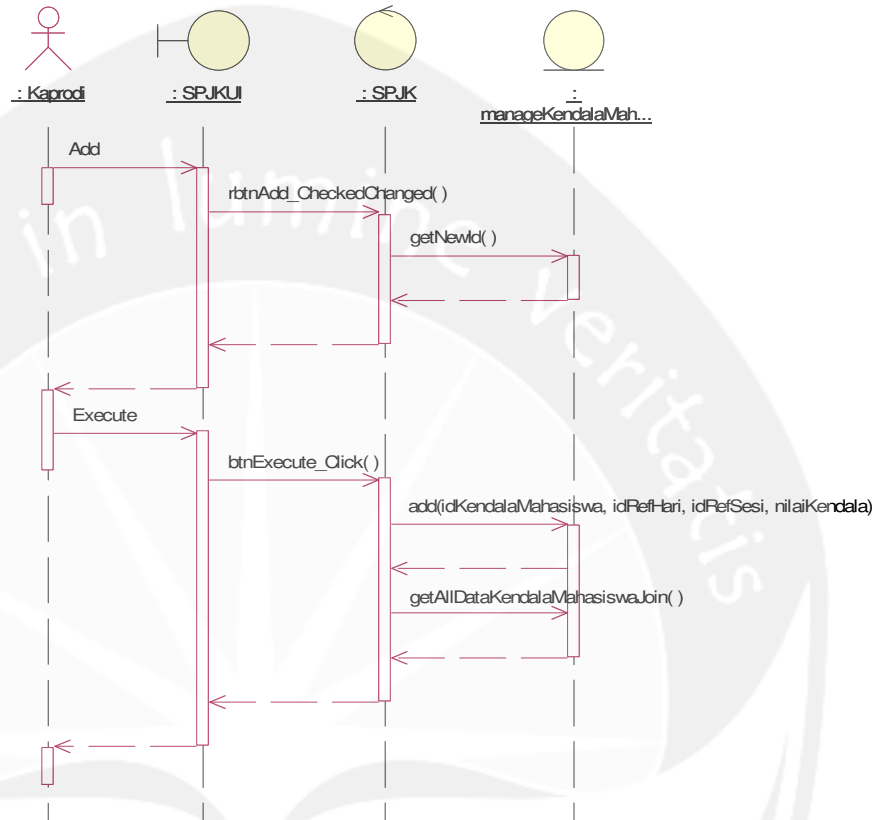
Gambar 17. Sequence Diagram: delete

6. manageKendalaMahasiswa



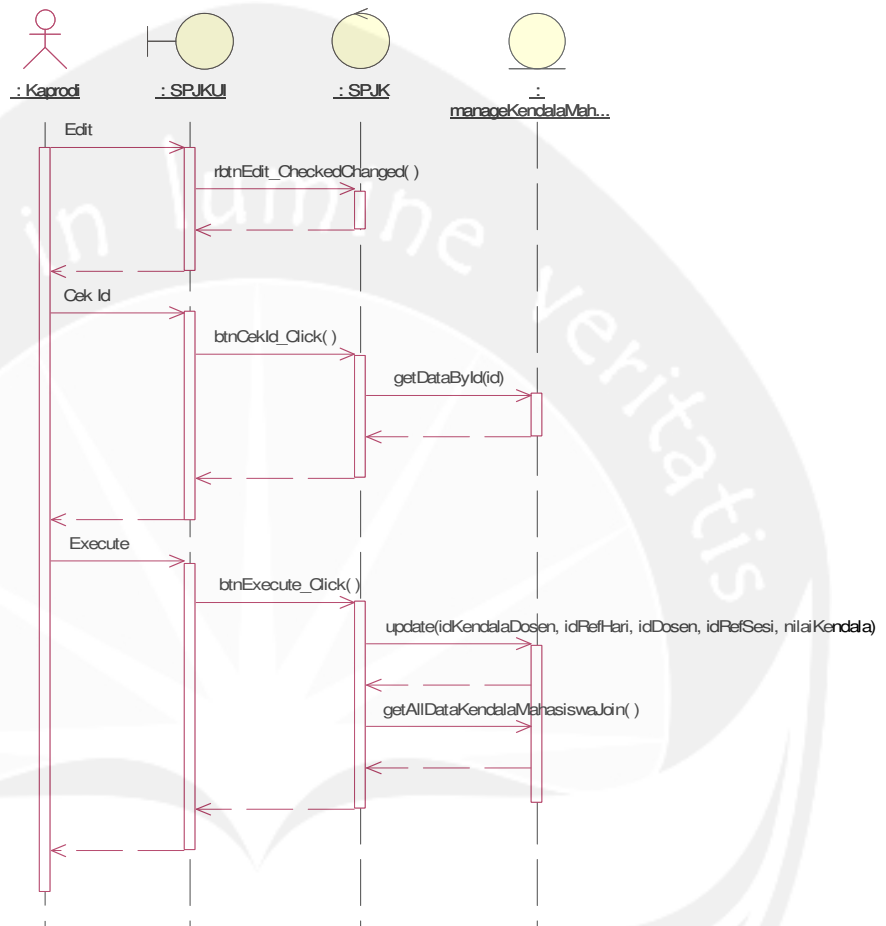
Gambar 18. Sequence Diagram: manageKendalaMahasiswa

6.1. add



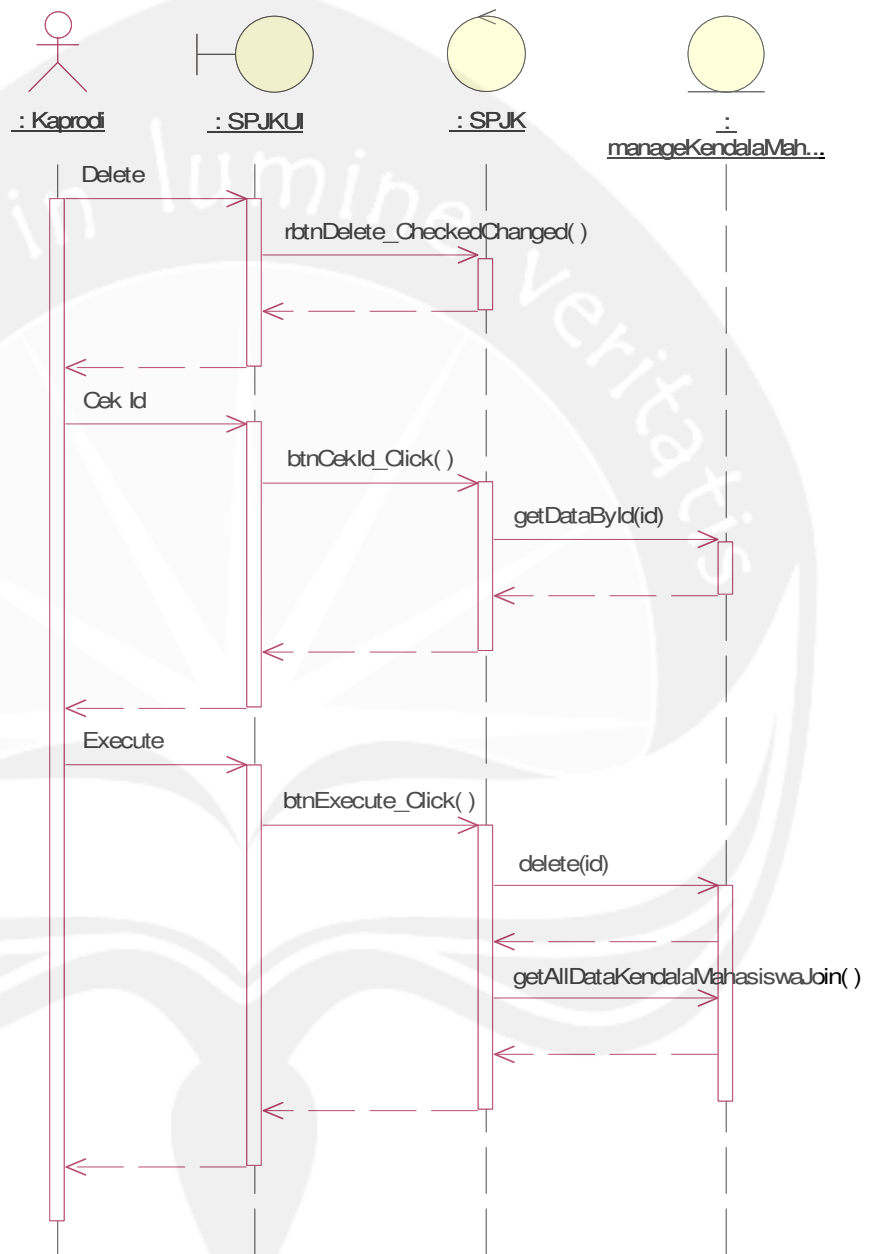
Gambar 19. Sequence Diagram: add

6.2. edit



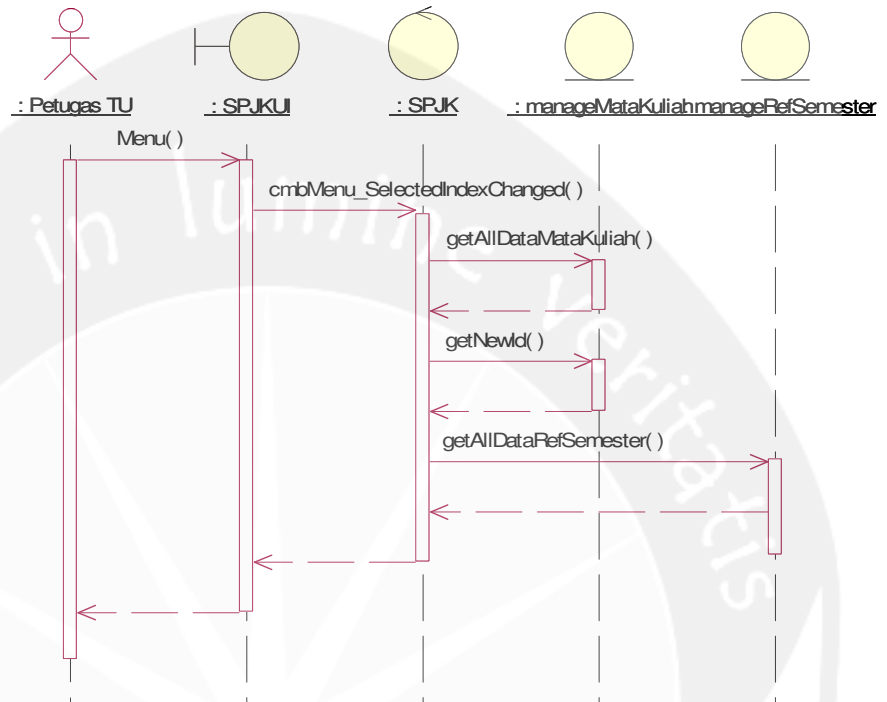
Gambar 20. Sequence Diagram: edit

6.3. delete



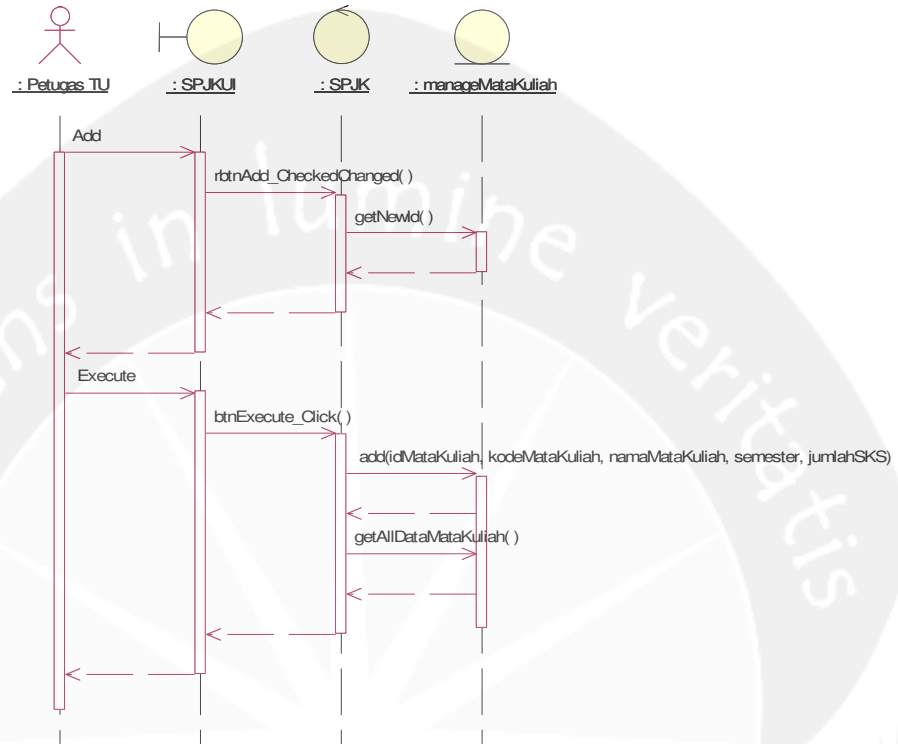
Gambar 21. Sequence Diagram: delete

7. manageMataKuliah



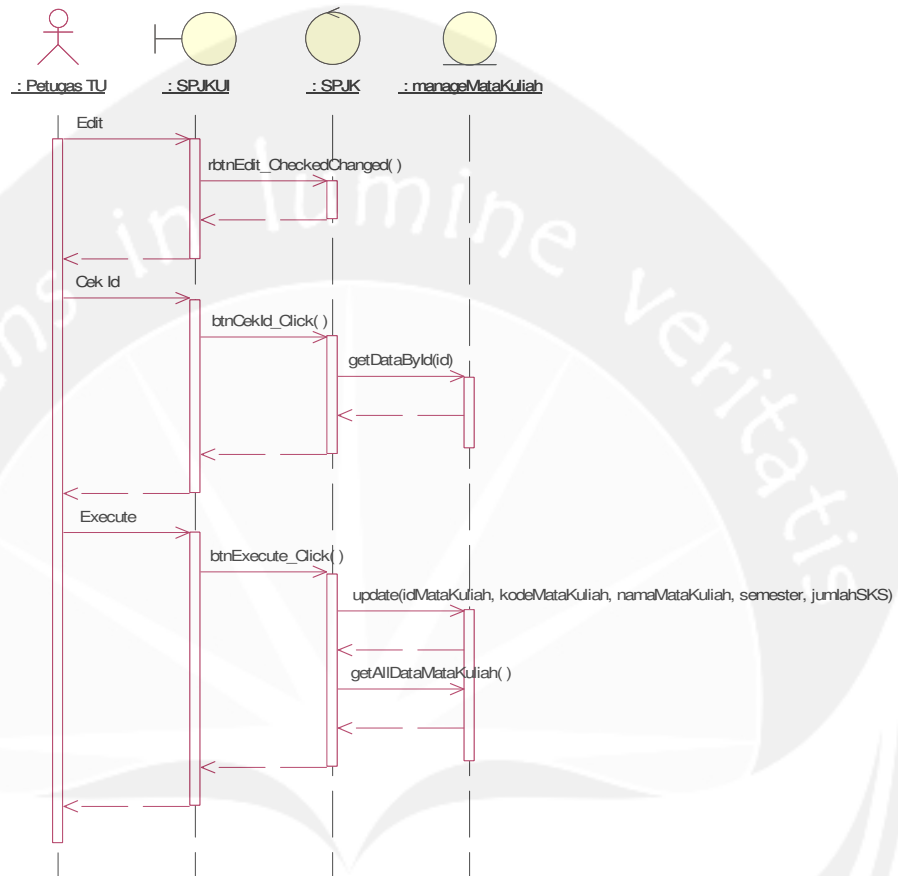
Gambar 22. Sequence Diagram: manageMataKuliah

7.1. add



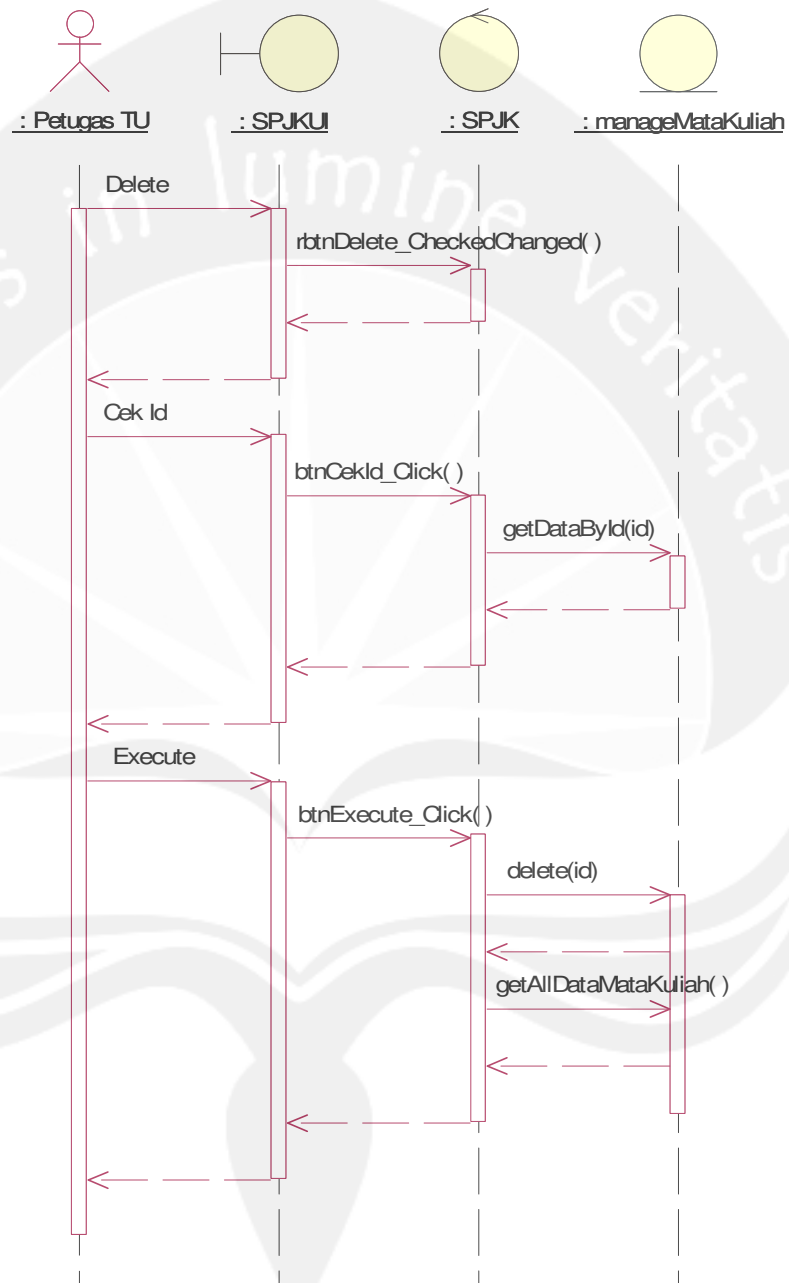
Gambar 23. Sequence Diagram: add

7.2. edit



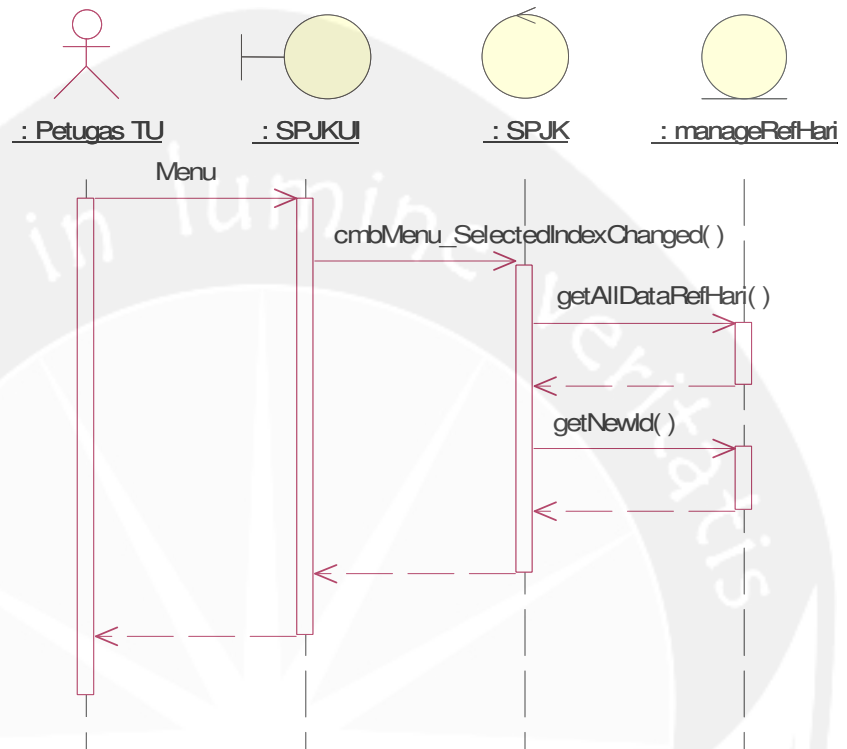
Gambar 24. Sequence Diagram: edit

7.3. delete



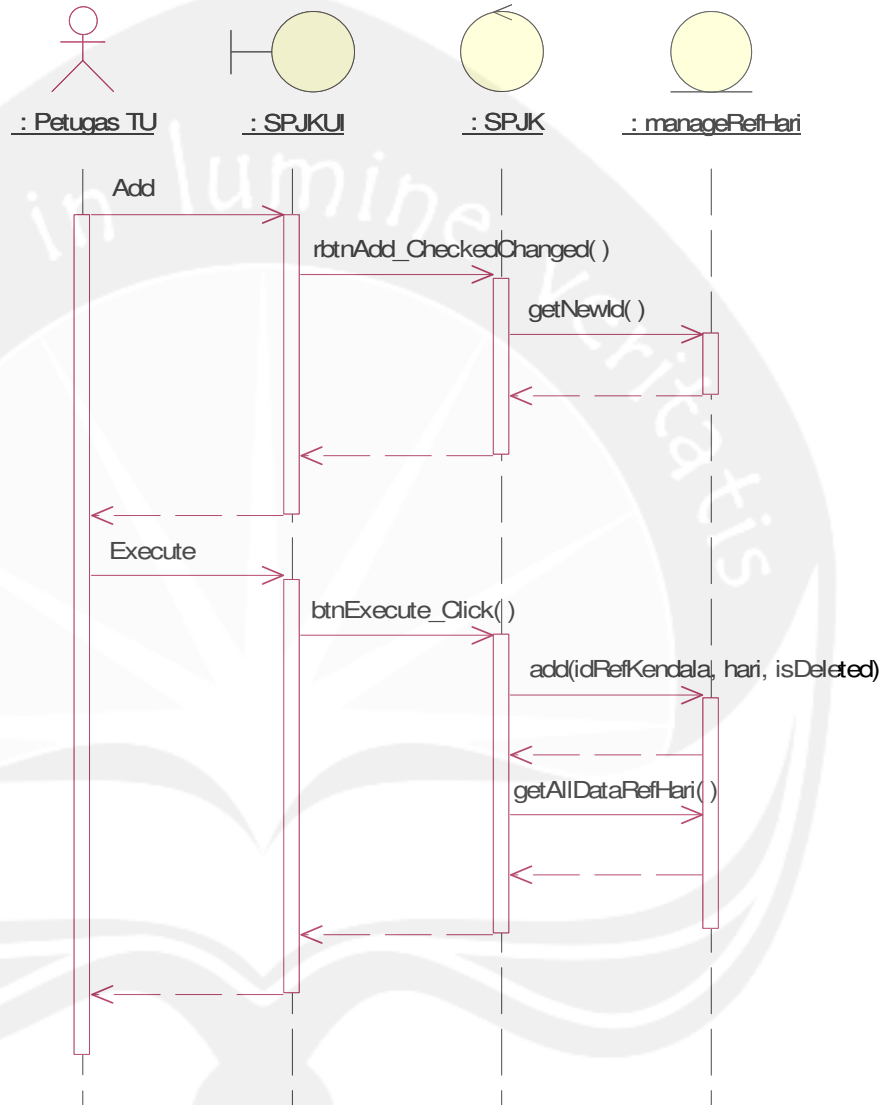
Gambar 25. Sequence Diagram: delete

8. manageRefHari



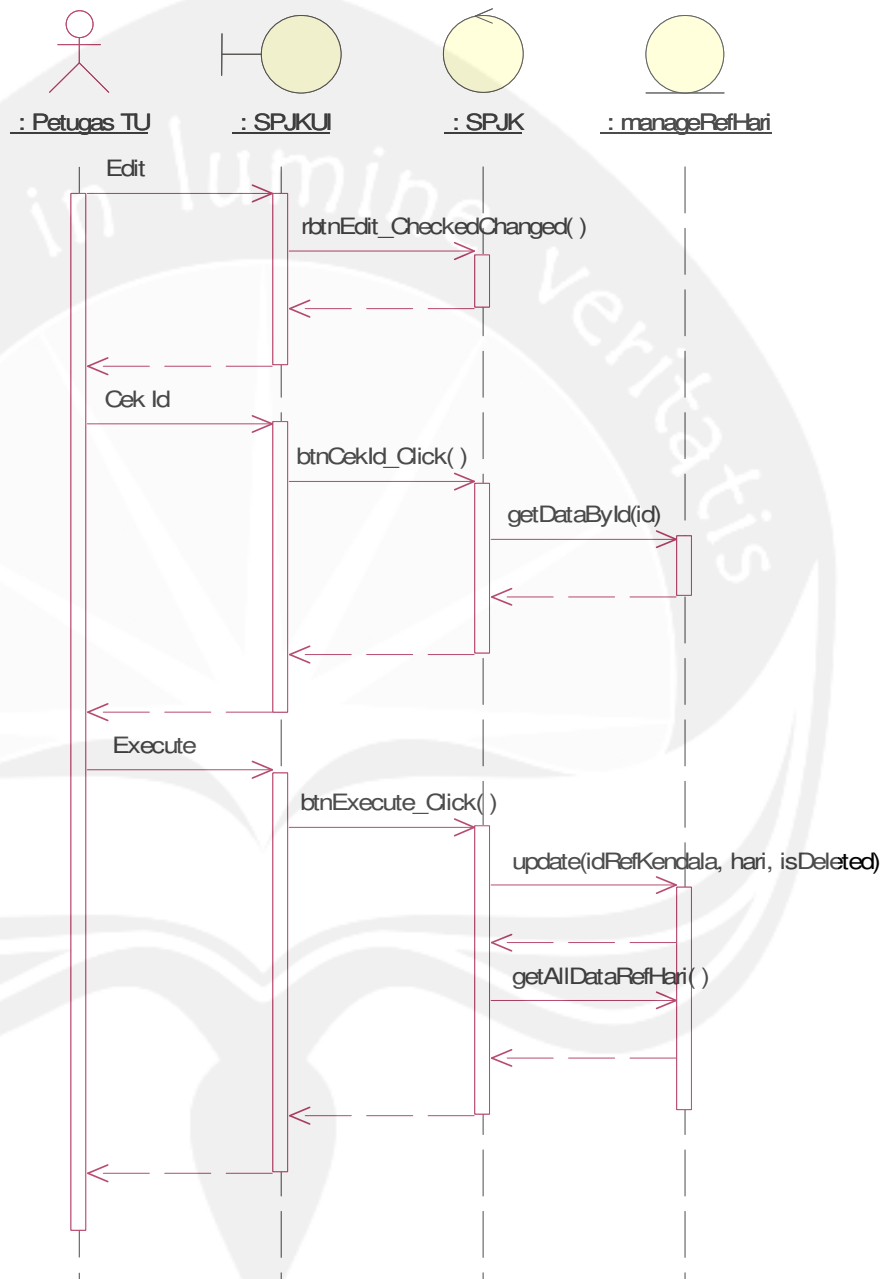
Gambar 26. Sequence Diagram: manageRefHari

8.1. add



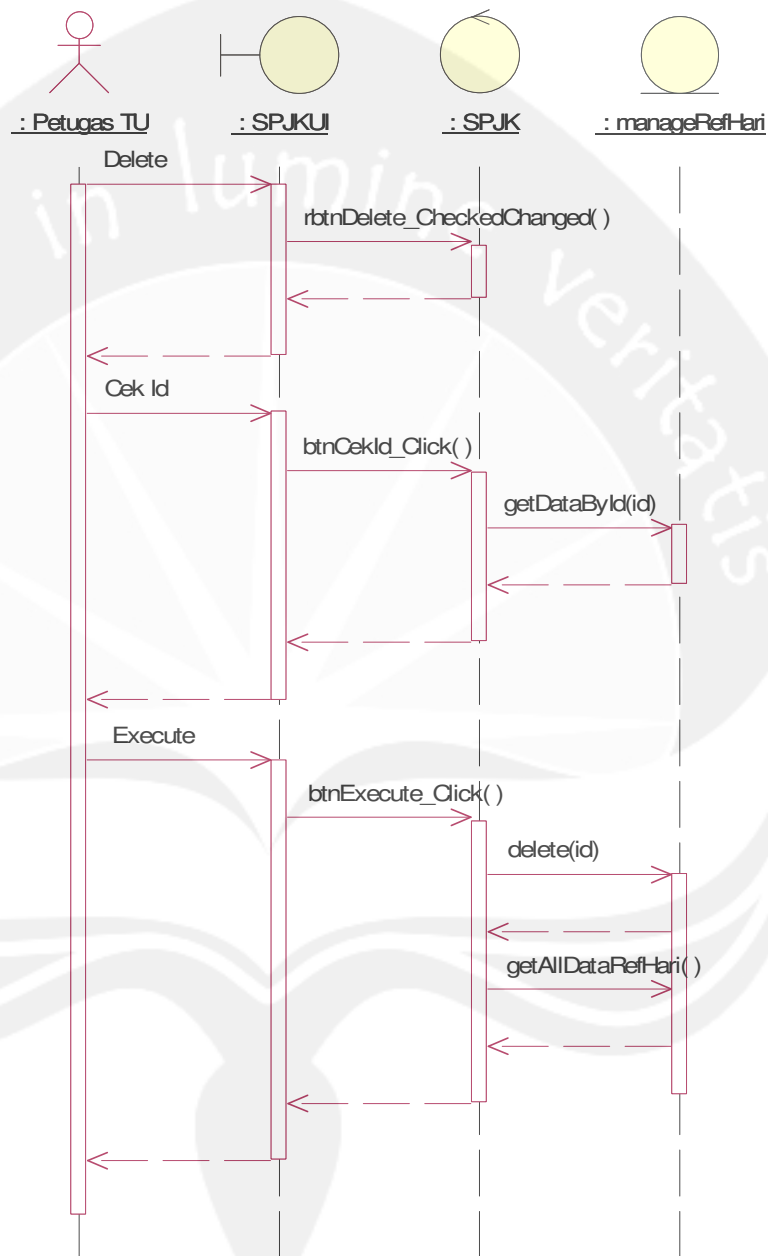
Gambar 27. Sequence Diagram: add

8.2. edit



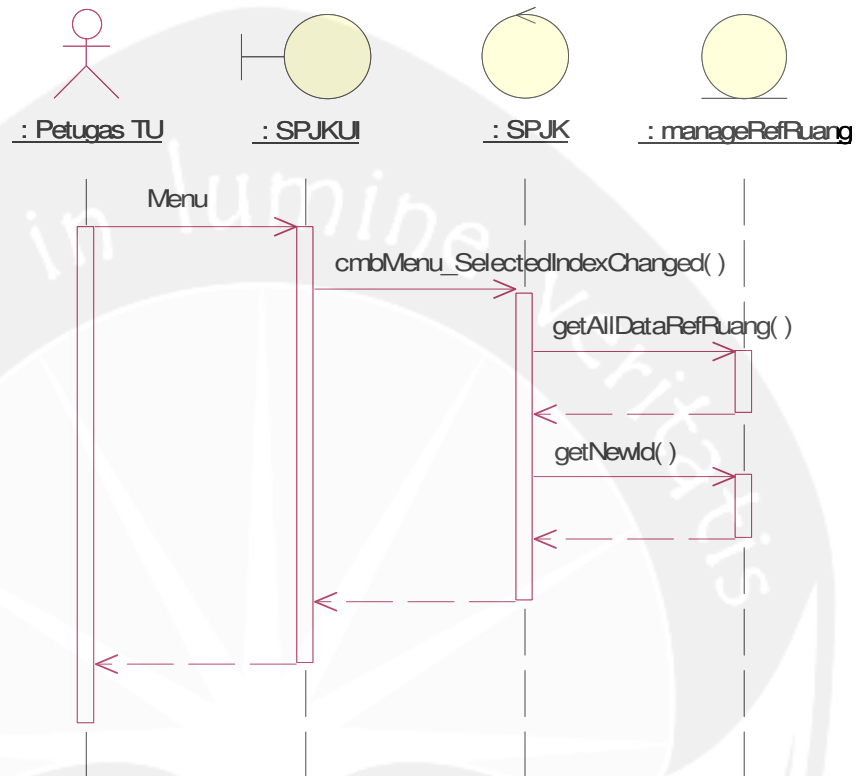
Gambar 28. Sequence Diagram: edit

8.3. delete



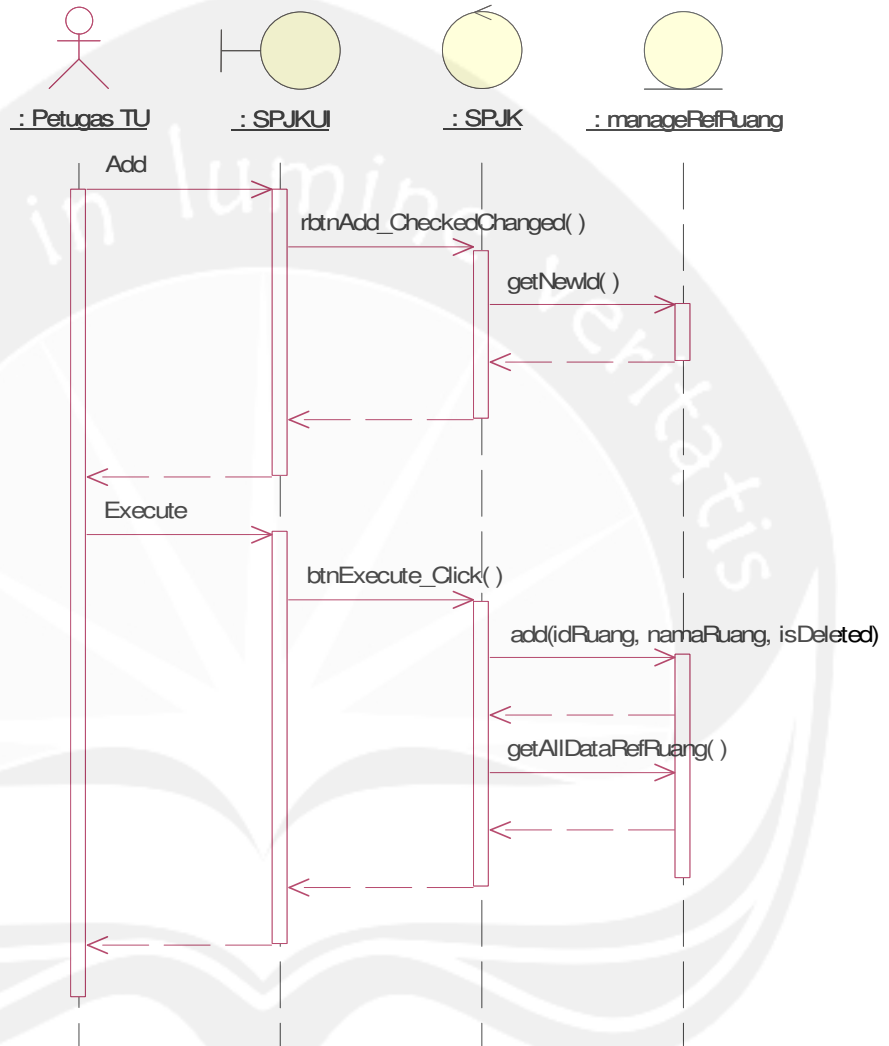
Gambar 29. Sequence Diagram: delete

9. manageRefRuang



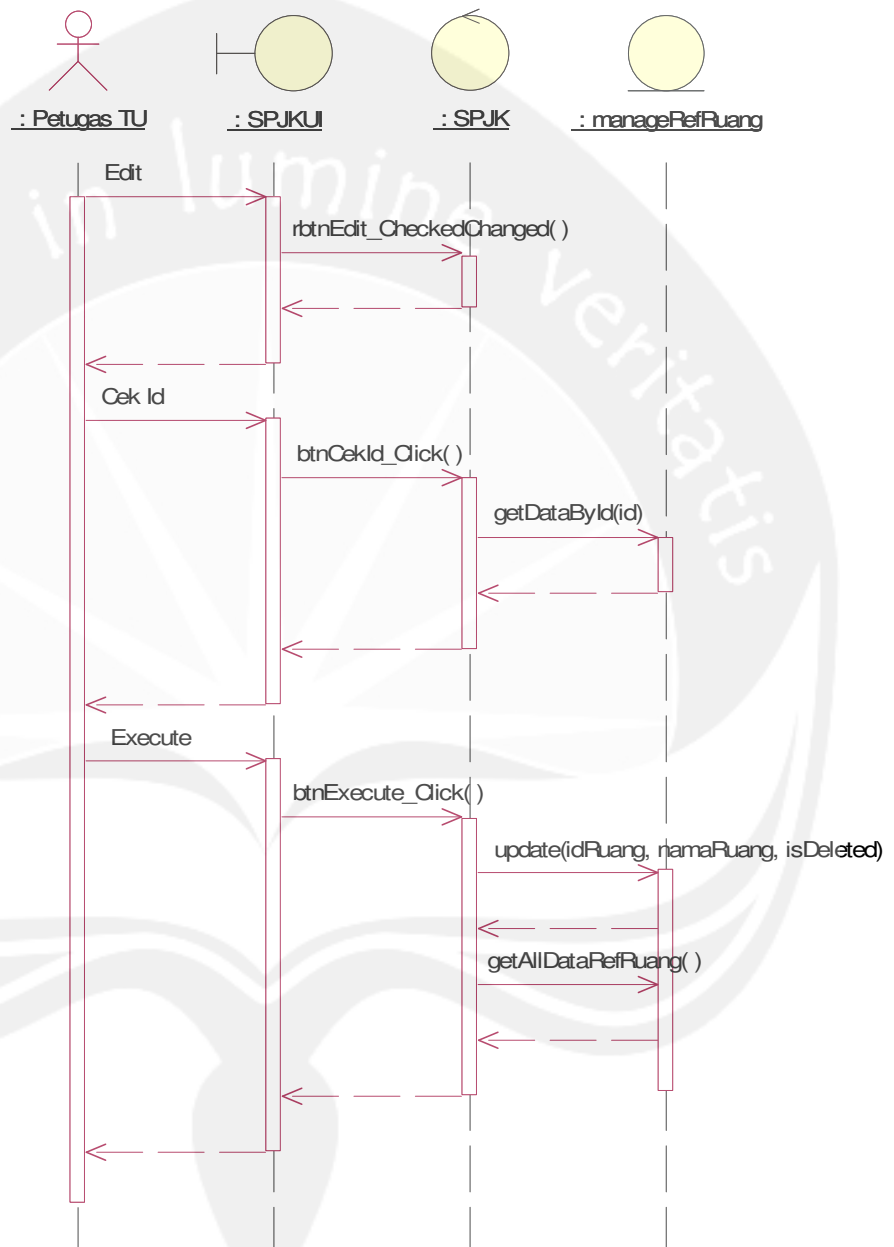
Gambar 30. Sequence Diagram: manageRefRuang

9.1. add



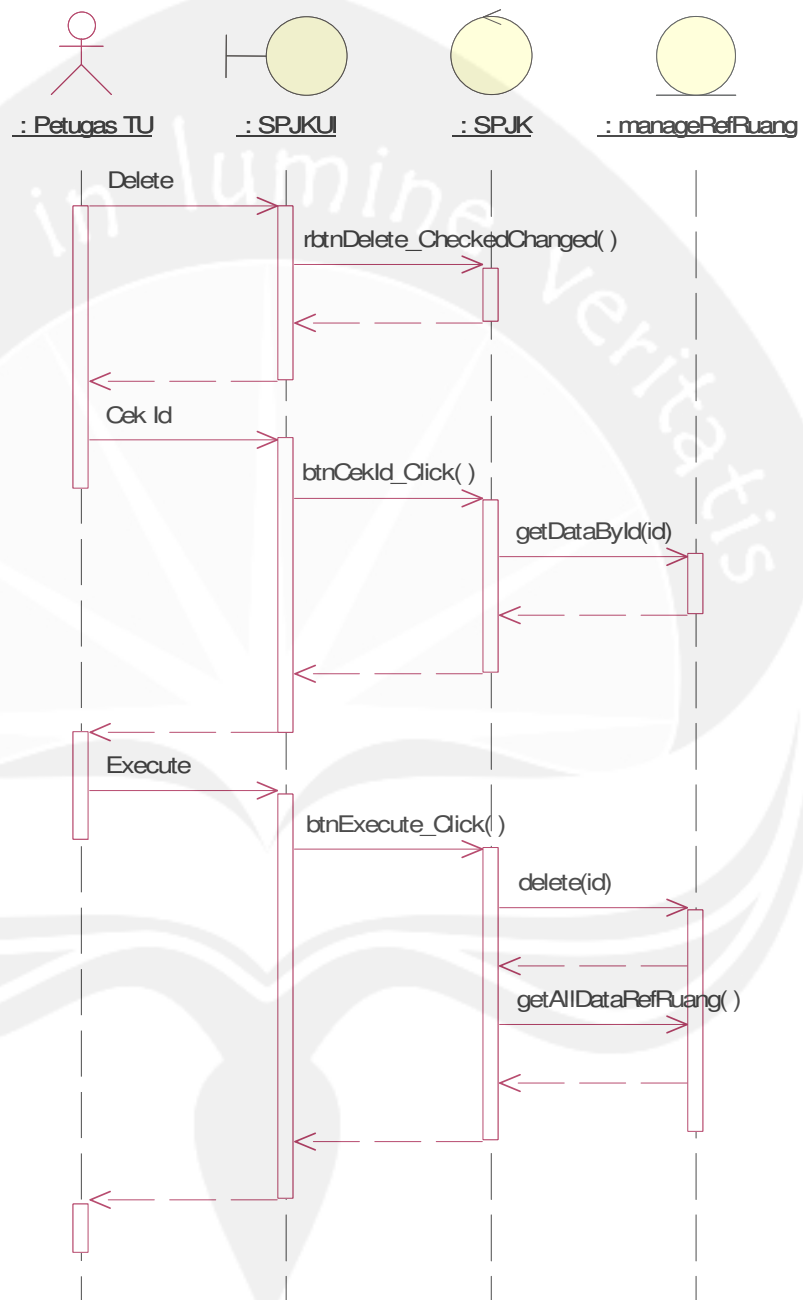
Gambar 31. Sequence Diagram: add

9.2. edit



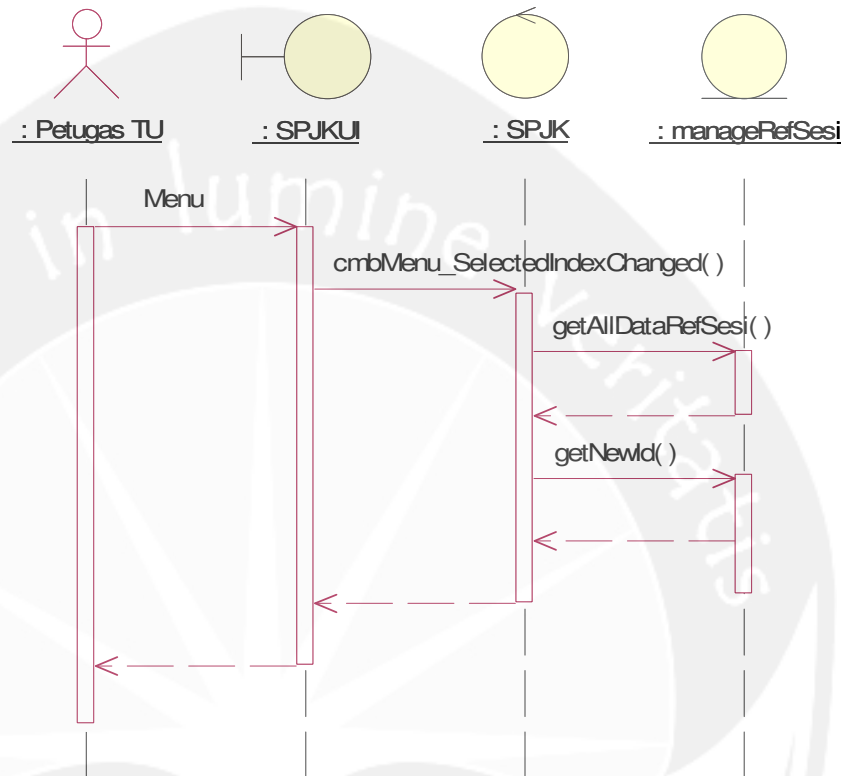
Gambar 32. Sequence Diagram: edit

9.3. delete



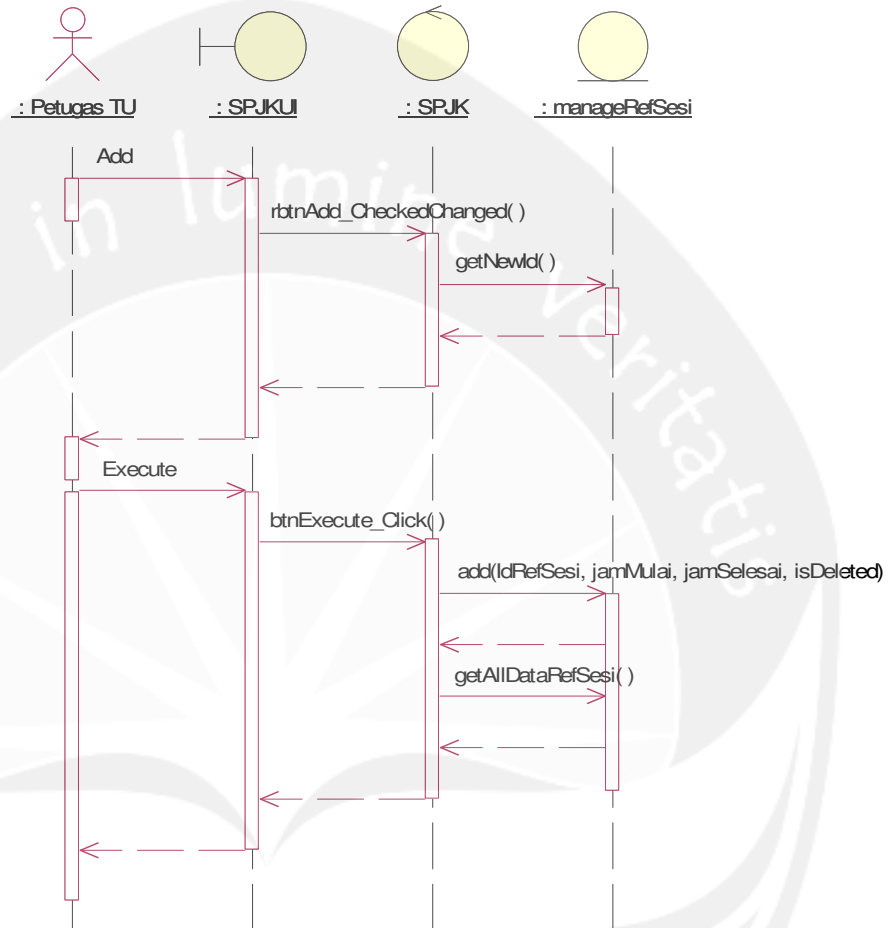
Gambar 33. Sequence Diagram: delete

10 . manageRefSesi



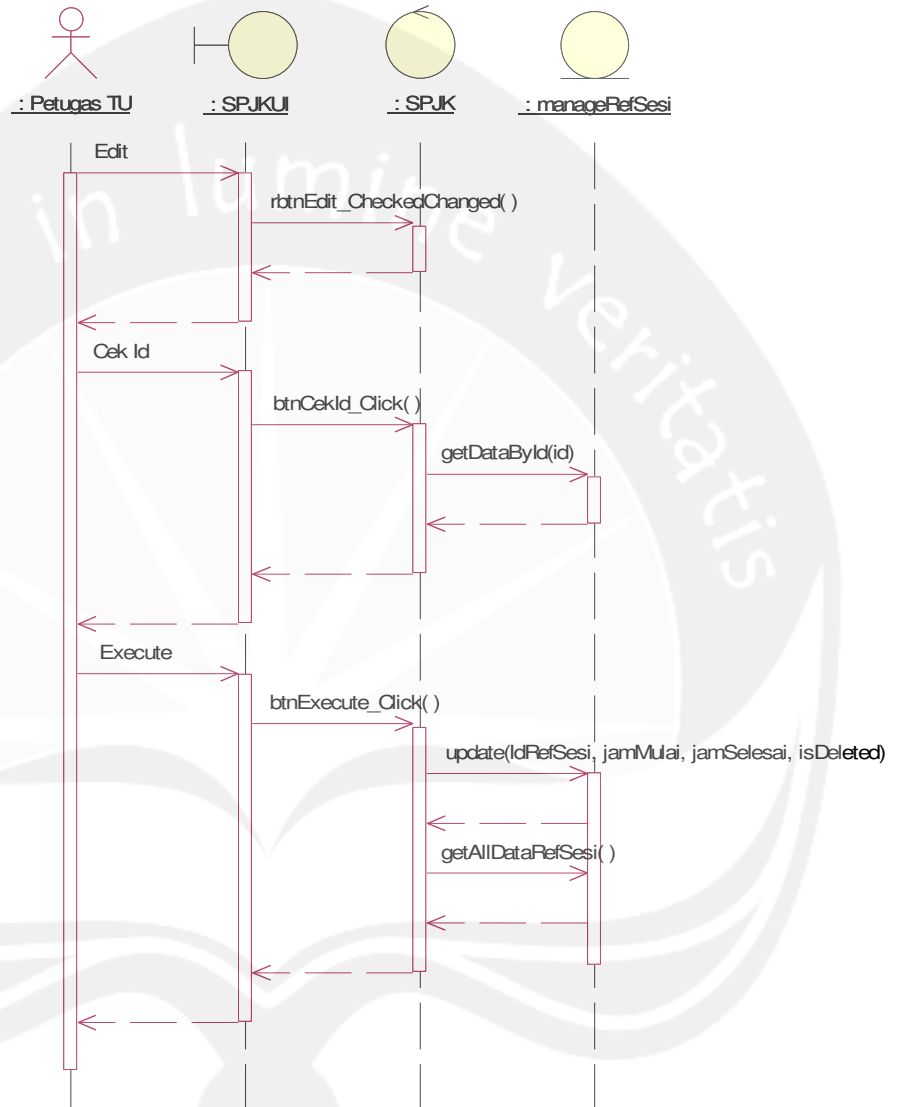
Gambar 34. Sequence Diagram: manageRefSesi

10.1. add



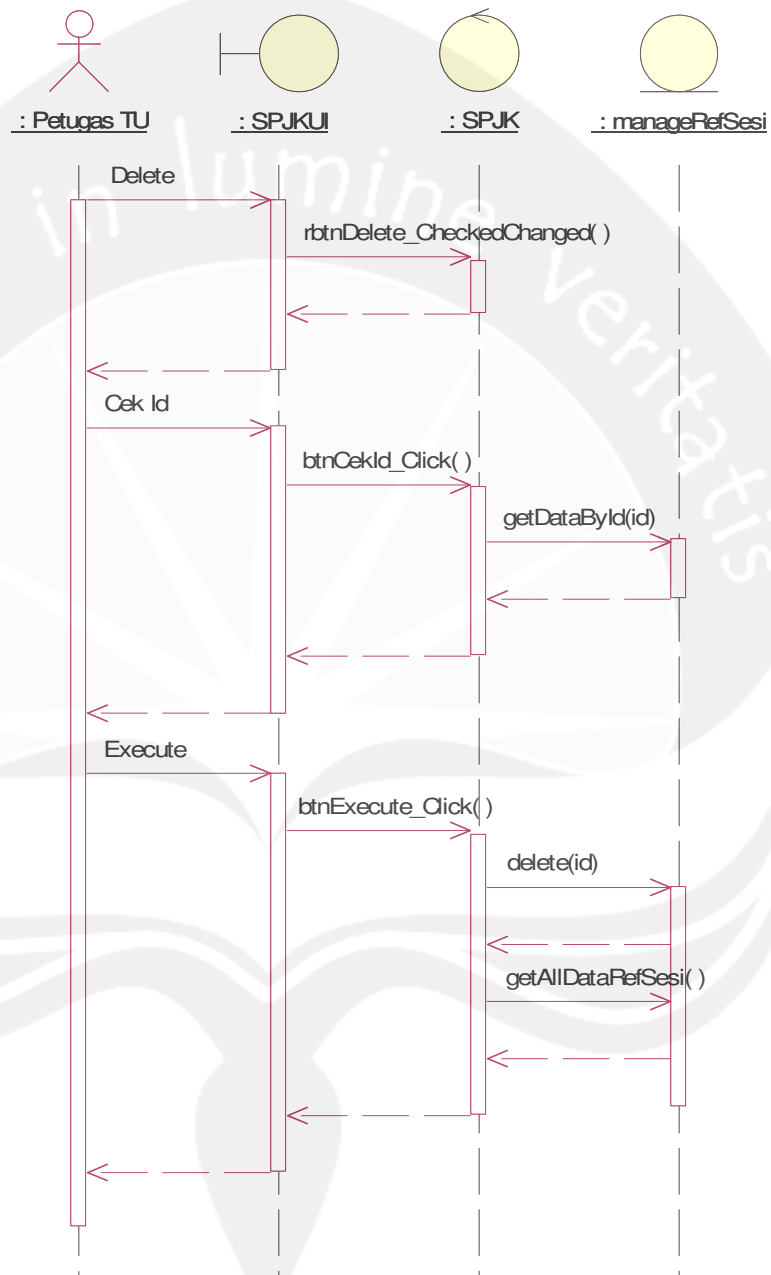
Gambar 35. Sequence Diagram: add

10.2. edit



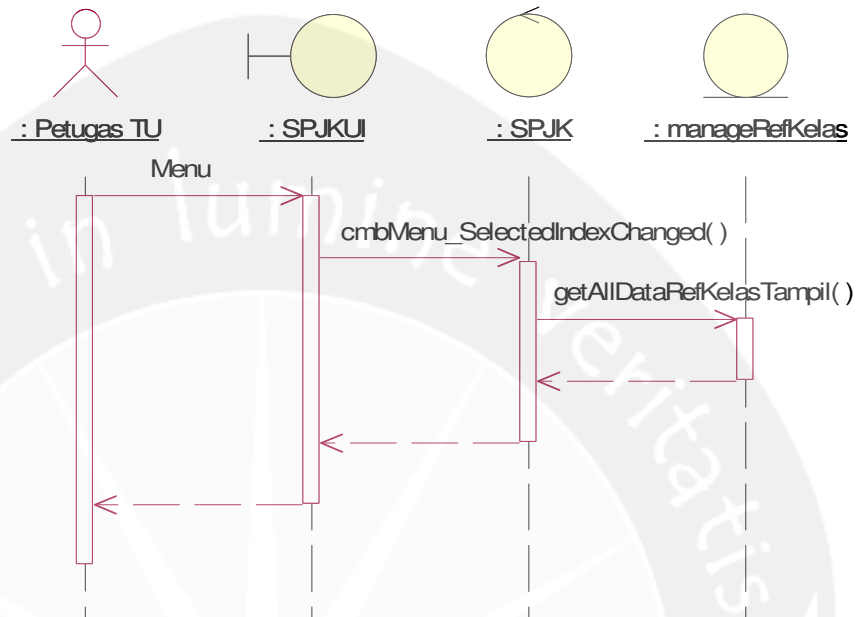
Gambar 36. Sequence Diagram: edit

10.3. delete



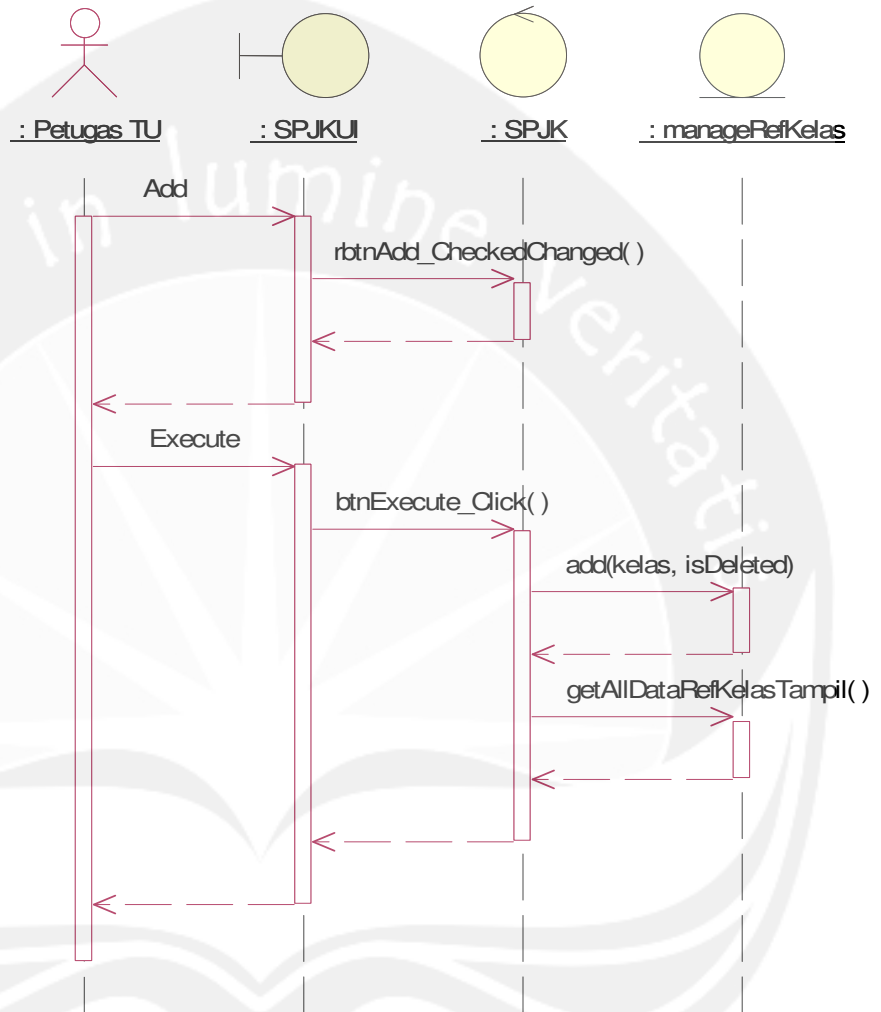
Gambar 37. Sequence Diagram: delete

11. manageRefKelas



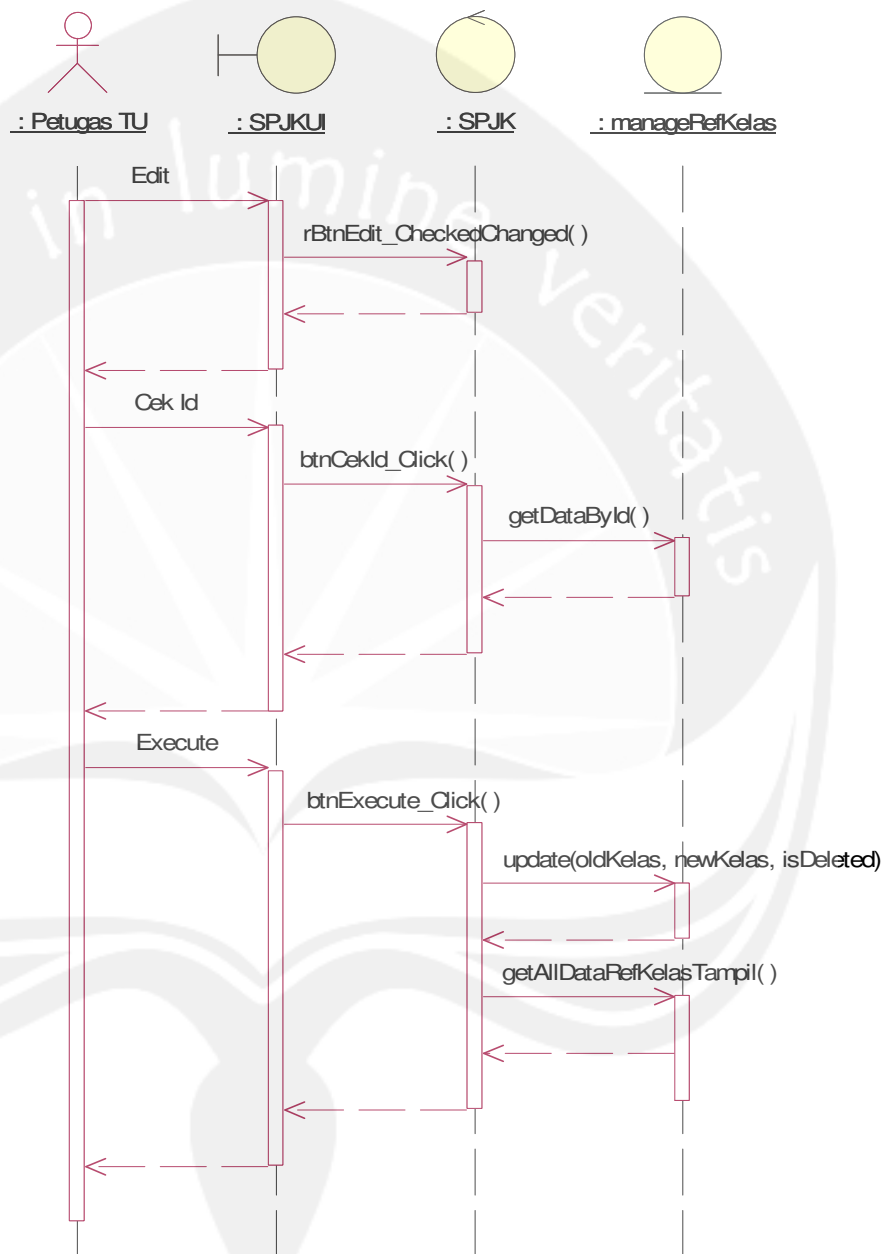
Gambar 38. Sequence Diagram: manageRefKelas

11.1. add



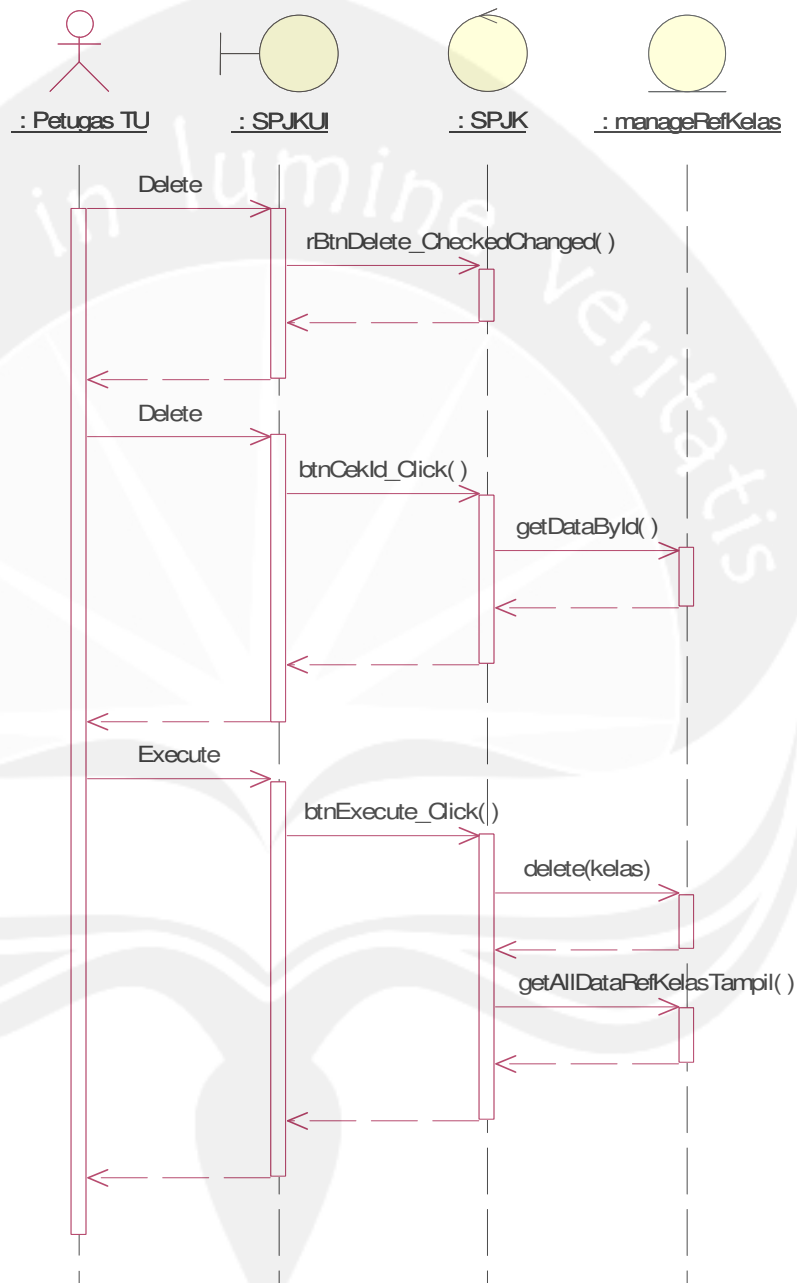
Gambar 39. Sequence Diagram: add

11.2. edit



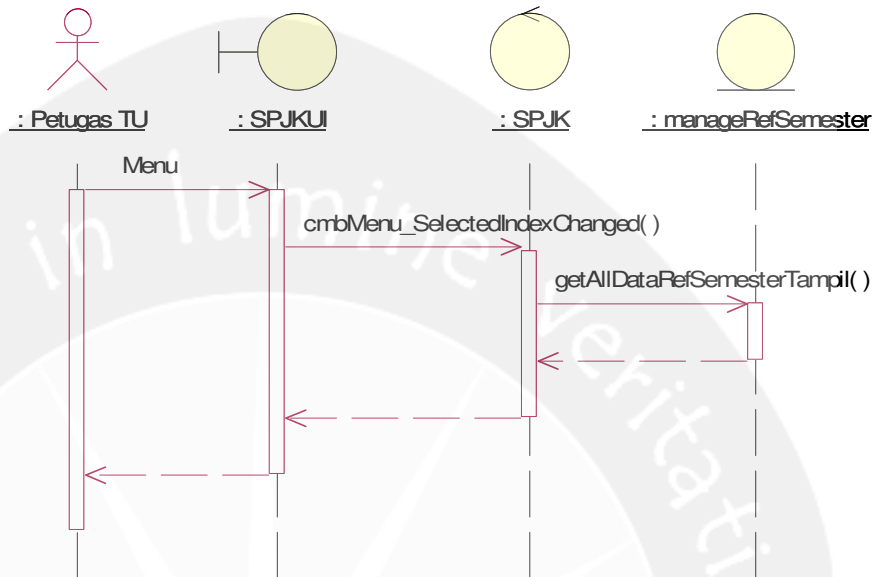
Gambar 40. Sequence Diagram: edit

11.3. delete



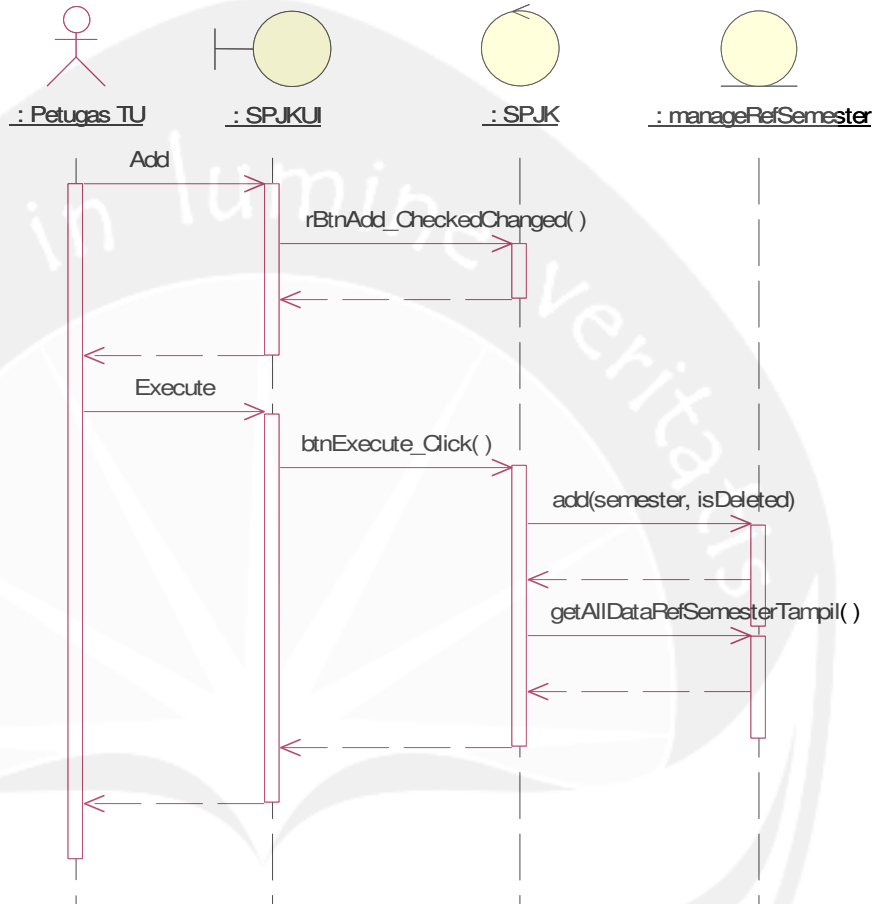
Gambar 41. Sequence Diagram: delete

12 . manageRefSemester



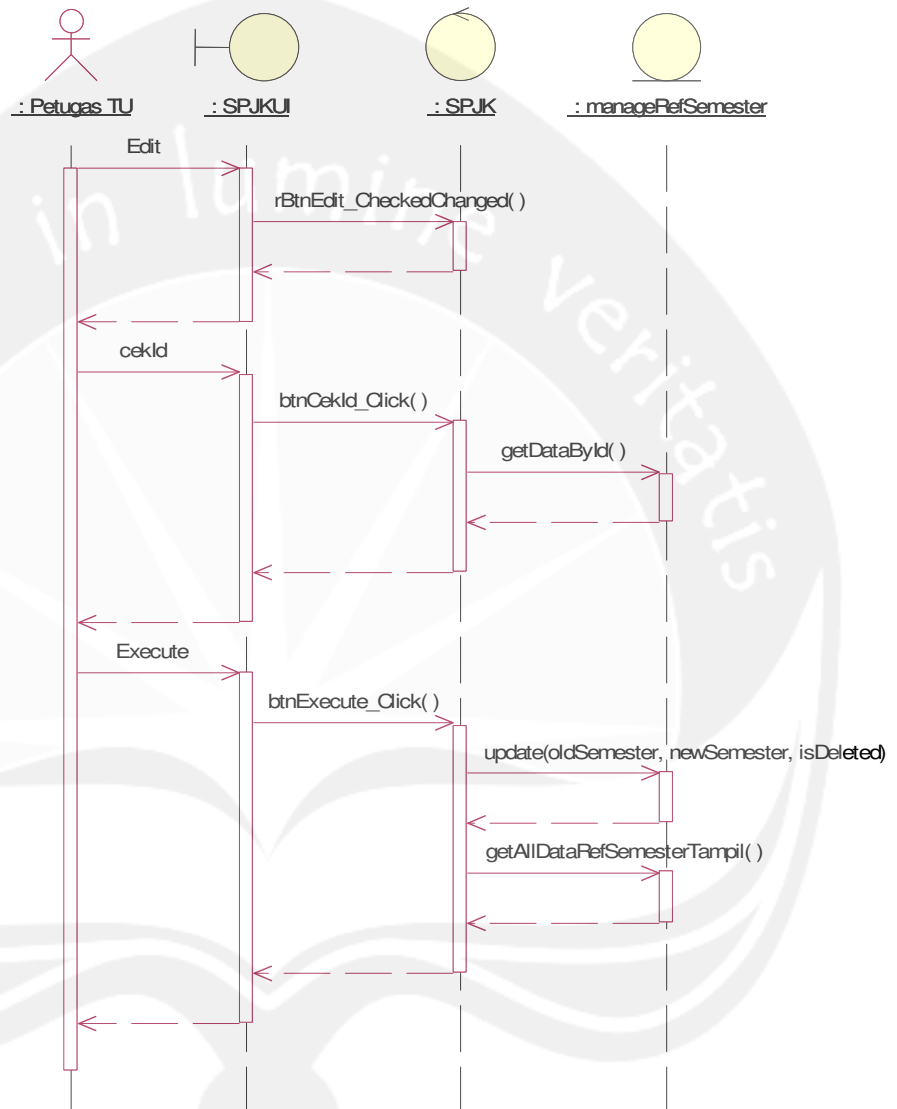
Gambar 42. Sequence Diagram: manageRefSemester

12.1. add



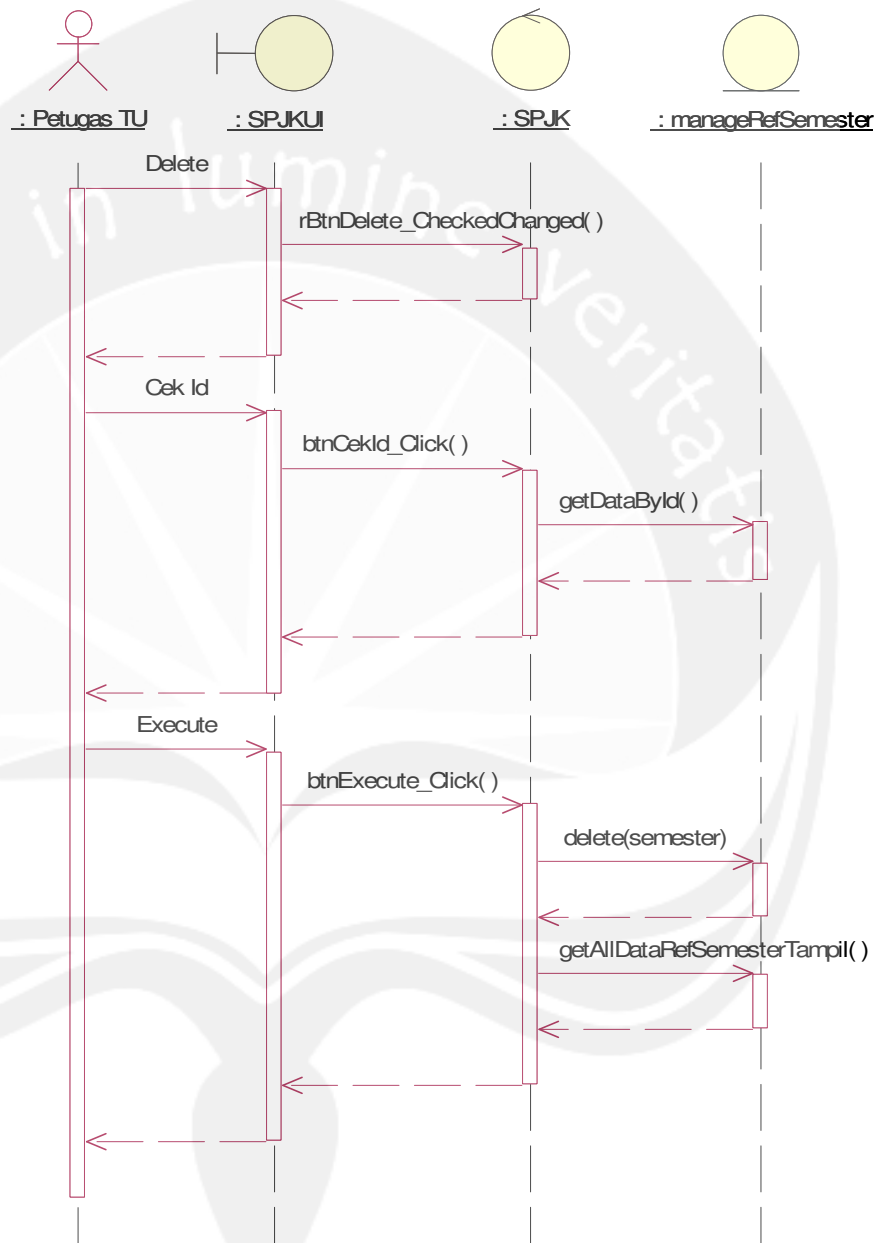
Gambar 43. Sequence Diagram: add

12.2. edit



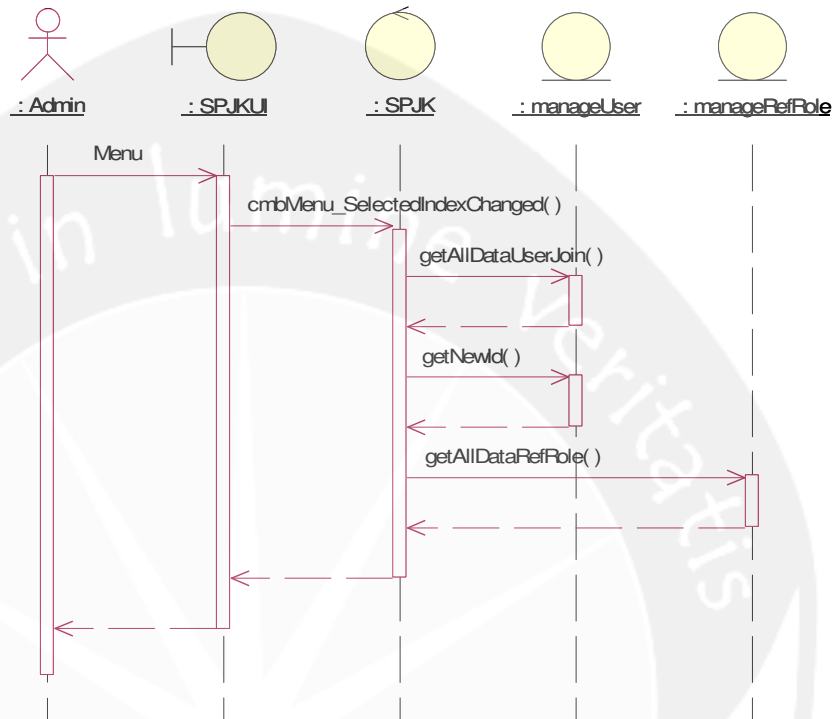
Gambar 44. Sequence Diagram: edit

12.3. delete



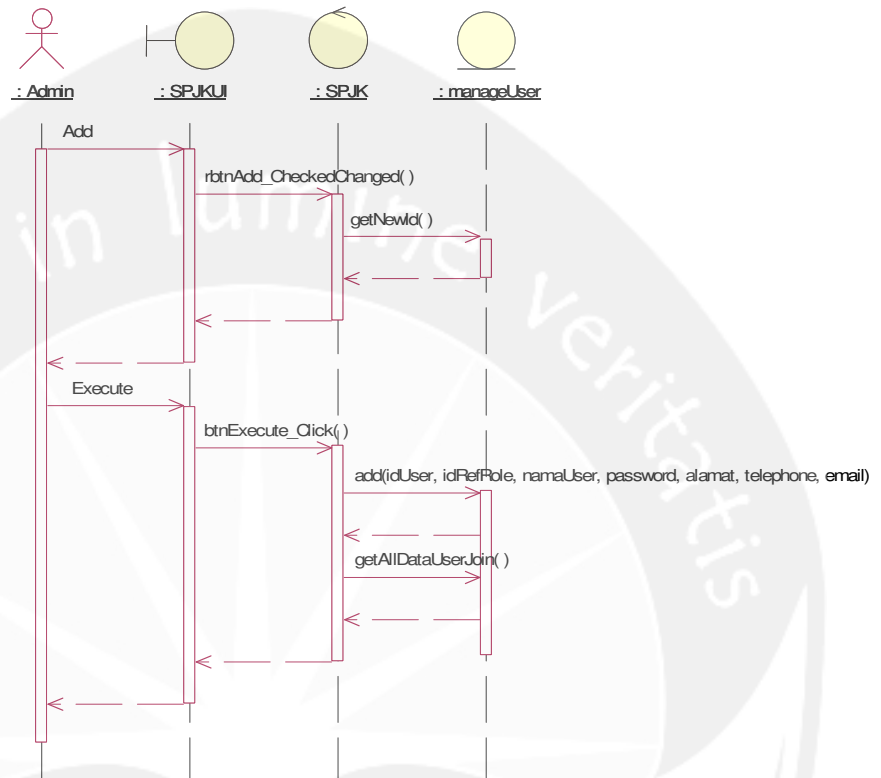
Gambar 45. Sequence Diagram: delete

13. manageUser



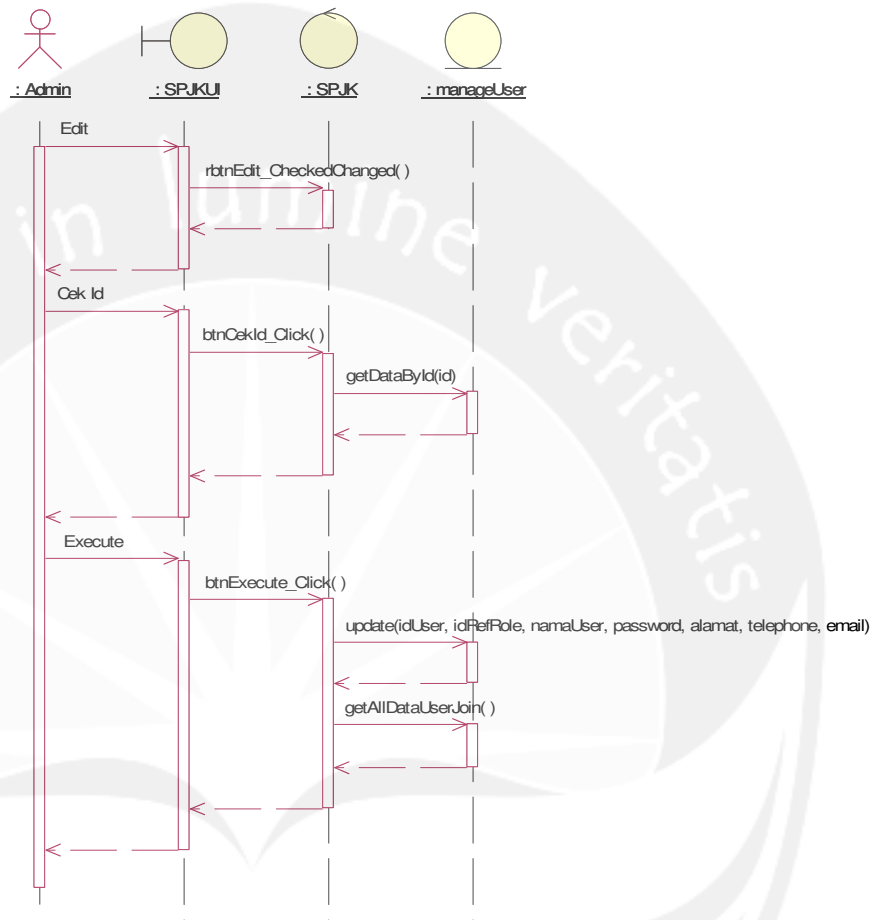
Gambar 46. Sequence Diagram: manageUser

13.1. add



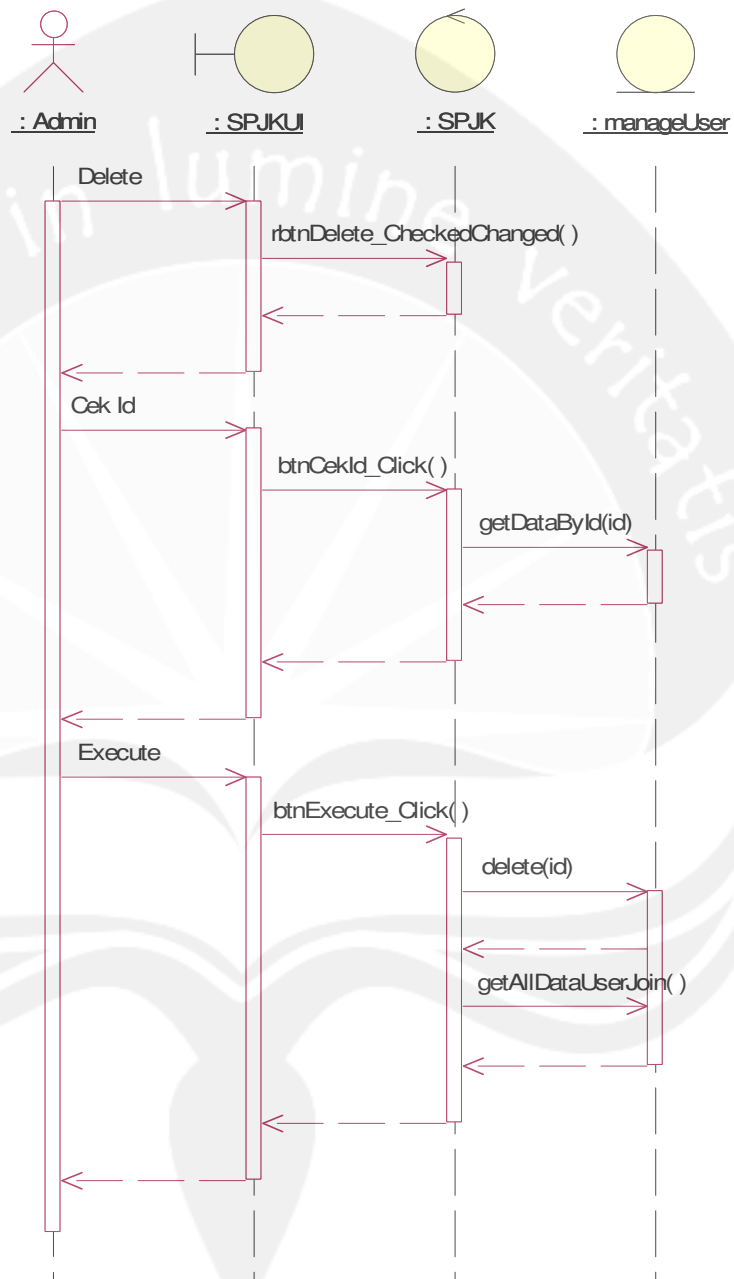
Gambar 47. Sequence Diagram: add

13.2. edit



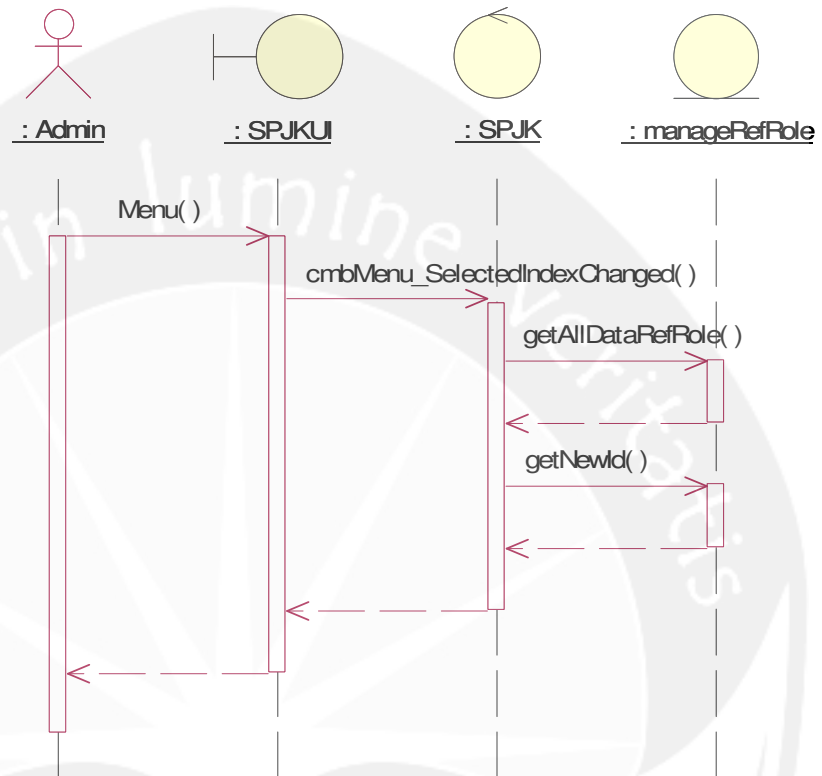
Gambar 48. Sequence Diagram: edit

13.3. delete



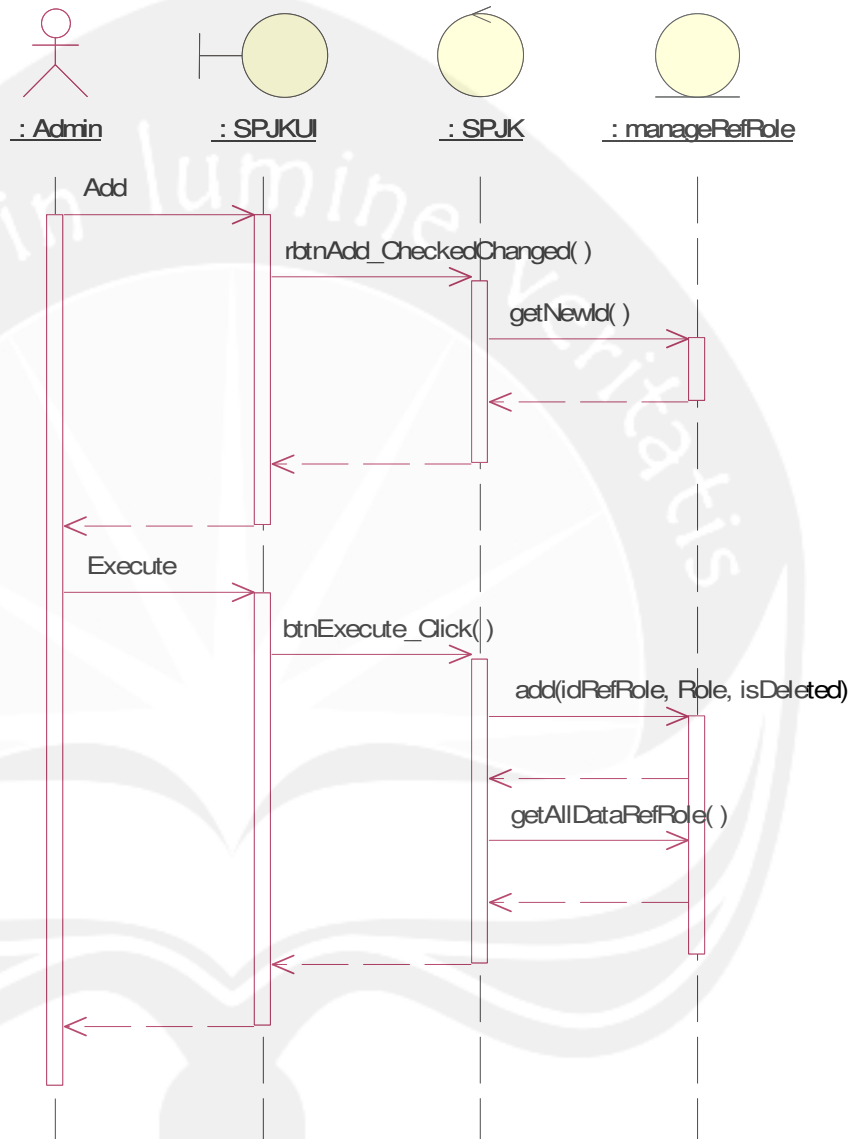
Gambar 49. Sequence Diagram: delete

14. manageRefRole



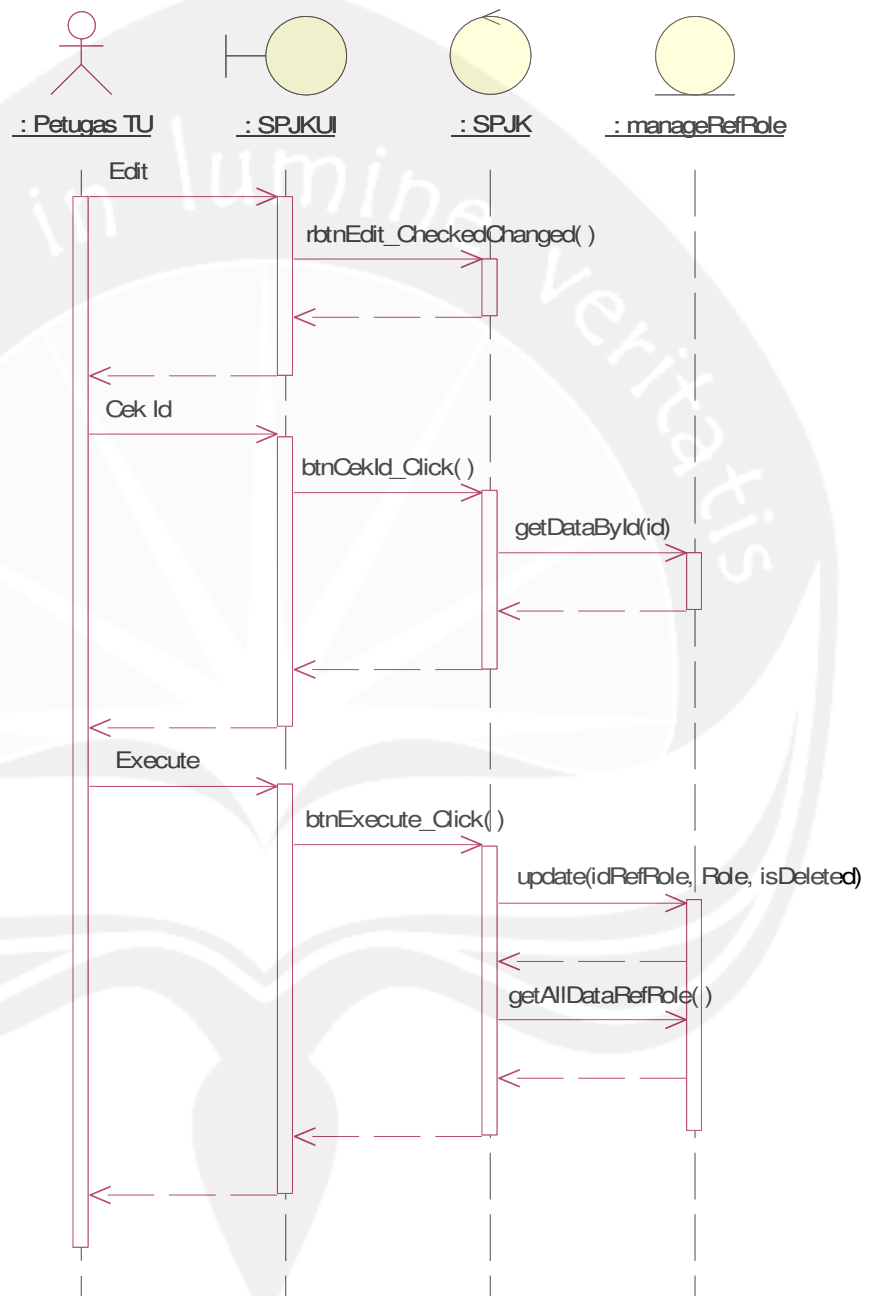
Gambar 50. Sequence Diagram: manageRefRole

14.1. add



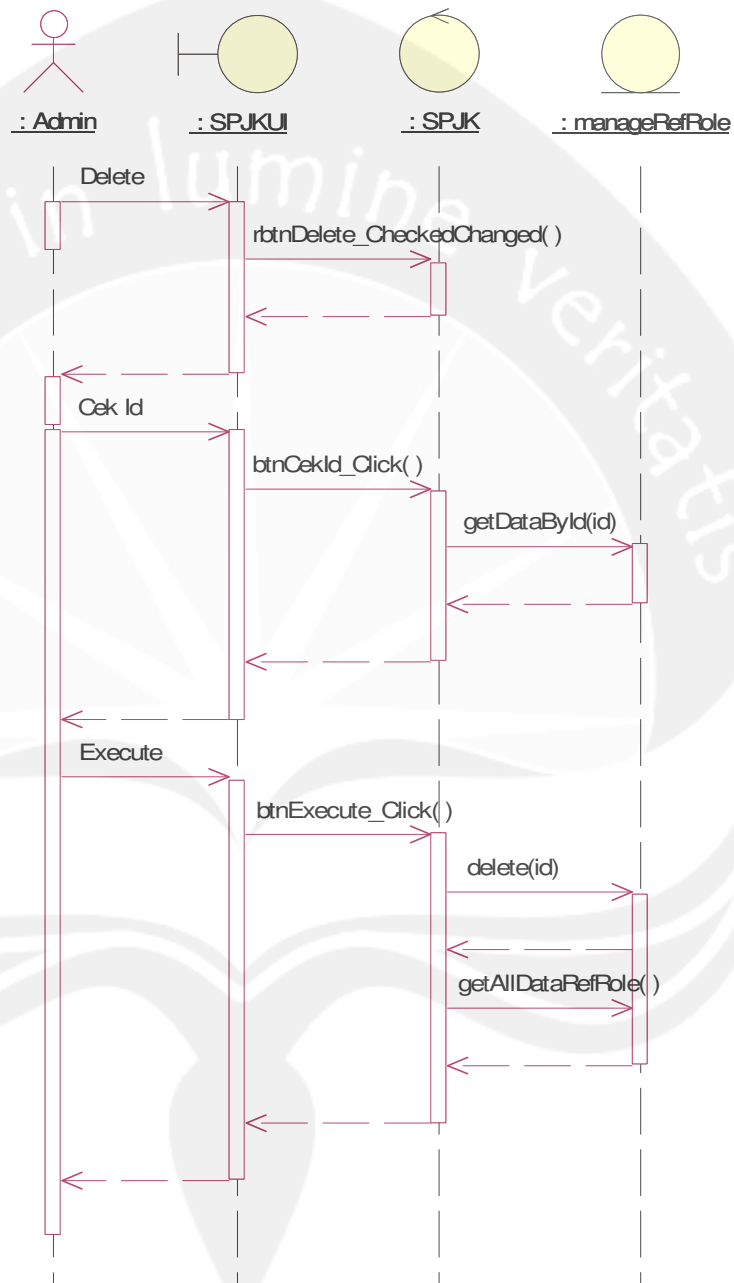
Gambar 51. Sequence Diagram: add

14.2. edit



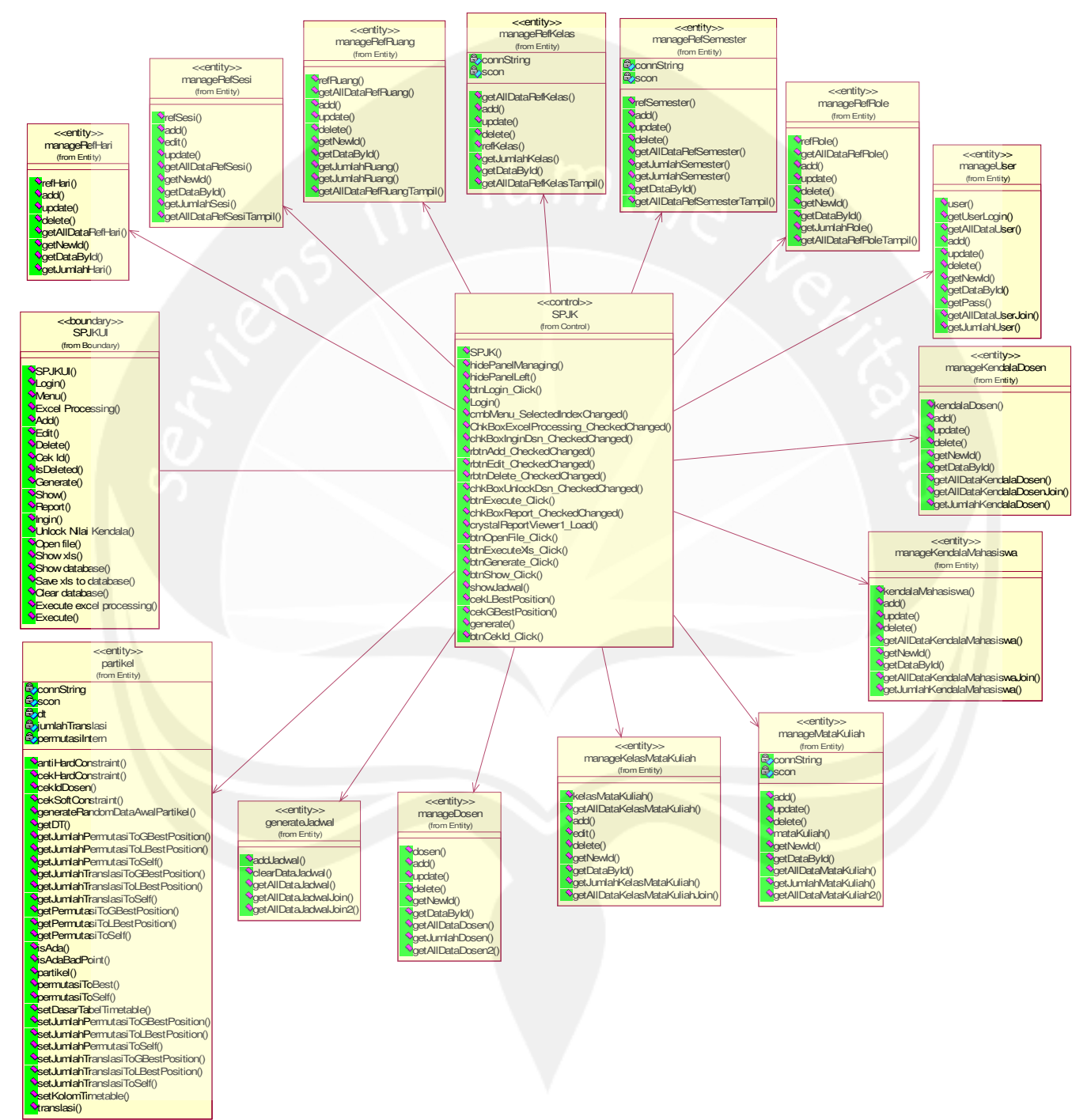
Gambar 52. Sequence Diagram: edit

14.3. delete



Gambar 53. Sequence Diagram: delete

15. Class Diagram



Gambar 54 Class Diagram

16. Class Diagram Spesific Description

a. Spesific Design Class SPJKUI

SPJKUI	<<boundary>>	
+ SPJKUI() Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas SPJKUI. + Login() Operasi ini digunakan untuk masuk ke dalam sistem. + Menu() Operasi ini digunakan untuk membuka pengaturan sesuai menu yang dipilih. + ExcelProcessing() Operasi ini digunakan untuk melakukan pemrosesan yang berhubungan dengan file excel. + Add() Operasi ini digunakan untuk memilih jika user akan menambah data sesuai dengan menu yang dipilih. + Edit() Operasi ini digunakan untuk memilih jika user akan mengubah data sesuai dengan menu yang dipilih. + Delete() Operasi ini digunakan untuk memilih jika user akan menghapus data sesuai dengan menu yang dipilih. + Cek Id() Operasi ini digunakan untuk mengecek apakah id yang diinputkan user ada pada database. + IsDeleted() Operasi ini digunakan untuk data ref sebagai penanda apakah data tidak digunakan atau masih digunakan. + Execute()		
Program Studi Teknik Informatika	DPPL- SPJK	63/ 102
Dokumen ini dan informasi yang dimilikinya adalah milik Program StudiTeknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Operasi ini digunakan untuk eksekusi operasi add, edit, maupun delete.

+ generate()

operasi ini digunakan untuk generate jadwal.

+ report()

Operasi ini digunakan untuk melihat report data jadwal yang sudah digenerate.

+ Ingin()

Operasi ini digunakan untuk menentukan keinginan atau ketidakinginan sesi tertentu.

+ Unlock Nilai Kendala ()

Operasi ini digunakan untuk menentukan nilai tingkat keinginan atau ketidakinginan dosen pada sesi tertentu.

+ Open File

Operasi ini digunakan untuk memilih file Xls yang akan diproses.

+ Show Xls

Operasi ini digunakan untuk memilih menampilkan data dari file xls.

+ Show Database

Operasi ini digunakan untuk memilih menampilkan data dari database.

+ Save Xls to Database

Operasi ini digunakan untuk memilih menyimpan data dari database.

+ Clear Database

Operasi ini digunakan untuk memilih menghapus data dari database.

Execute Excel Processing

Operasi ini digunakan untuk mengeksekusi pilihan antara Show Xls, Show Database, Save Xls to Database, atau

Clear Database.

b. Spesific Design Class SPJK

SPJK	<<control>>
+ actGenerateJadwal variable ini digunakan untuk memanggil fungsi pada kelas entity generateJadwal. + ActManageKelasMataKuliah variable ini digunakan untuk memanggil fungsi pada kelas entity manageKelasMataKuliah. + actManageDosen variable ini digunakan untuk memanggil fungsi pada kelas entity manageDosen. + ActManageKendaladosen variable ini digunakan untuk memanggil fungsi pada kelas entity manageKendalaDosen. + ActManageKendalaMahasiswa variable ini digunakan untuk memanggil fungsi pada kelas entity manageKendalaMahasiswa. + actManageMataKuliah variable ini digunakan untuk memanggil fungsi pada kelas entity manageMataKuliah. + actManageRefHari variable ini digunakan untuk memanggil fungsi pada kelas entity manageRefHari. + ActManageRefSesi variable ini digunakan untuk memanggil fungsi pada kelas entity manageRefSesi.	

+ ActManageRefRuang

variable ini digunakan untuk memanggil fungsi pada kelas entity manageRefRuang.

+ actManageRefKelas

variable ini digunakan untuk memanggil fungsi pada kelas entity manageRefKelas.

+ actManageRefSemester

variable ini digunakan untuk memanggil fungsi pada kelas entity manageRefSemester.

+ actManageUser

variable ini digunakan untuk memanggil fungsi pada kelas entity manageRefUser.

+ actManageRefRole

variable ini digunakan untuk memanggil fungsi pada kelas entity manageRefRole.

+ SPJK()

Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas entity SPJK.

+ hidePanelManaging()

Operasi ini digunakan untuk menyembunyikan panel pengelolaan yang ada.

+ hidePanelLeft()

Operasi ini digunakan untuk menonaktifkan atau menyembunyikan fungsi yang digunakan untuk pengelolaan.

+ btnLogin_Click()

Operasi ini digunakan untuk login sebagai user sesuai dengan input data.

+ Login()

Operasi ini digunakan untuk mengecek data inputan dengan database.

+ cmbMenu_SelectedIndexChanged()

Operasi ini digunakan untuk menentukan panel yang akan dibuka dan yang akan disembunyikan pada saat tombol menu selesai dipilih.

+ ChkBoxExcelProcessing_CheckedChanged()

Operasi ini digunakan untuk membuka panel pemrosesan excel dan menyembunyikan panel lain pada saat check box selesai dipilih.

+ chkBoxInginDsn_CheckedChanged()

Operasi ini digunakan untuk menentukan nilai constraint saat check box selesai dipilih.

+ rBtnAdd_CheckedChanged()

Operasi ini digunakan untuk menentukan apa saja yang tampil dan komputasi yang dilakukan saat user memilih radio button add.

+ rBtnEdit_CheckedChanged()

Operasi ini digunakan untuk menentukan apa saja yang tampil dan komputasi yang dilakukan saat user memilih radio button edit.

+ rBtnDelete_CheckedChanged()

Operasi ini digunakan untuk menentukan apa saja yang tampil dan komputasi yang dilakukan saat user memilih radio button delete.

+ btnCekId_Click()

Operasi ini digunakan untuk mengecek apakah id yang diinputkan user ada pada data yang diambil dari entity.

+ btnExecute_Click()

Operasi ini digunakan untuk melakukan komputasi sesuai dengan radio button yang dipilih.

+ chkBoxUnlockDsn_CheckedChanged()

Operasi ini digunakan untuk menentukan nilai tingkat keinginan atau ketidakinginan dosen pada sesi tertentu.

+ chkBoxReport_CheckedChanged()

Operasi ini digunakan untuk memanggil tampilan dari report data jadwal yang sudah digenerate.

+ crystalReportViewer1_Load()

Operasi ini digunakan untuk menampilkan data dari report data jadwal yang sudah digenerate.

+ btnOpenFile_Click()

Operasi ini digunakan untuk memilih file Xls yang akan diproses.

+ cekLBestPosition()

Operasi ini digunakan untuk mengecek posisi Lbest dari swarm.

+ cekGBestPosition()

Operasi ini digunakan untuk mengecek posisi Gbest dari swarm.

+ btnGenerate_Click()

Operasi ini digunakan untuk melakukan komputasi generate jadwal dengan memanggil fungsi fungsi di dalamnya.

+ showJadwal()

Operasi ini digunakan untuk menampilkan data generateJadwal yang dipanggil dari entity generateJadwal.

c. Spesific Design Class partikel

partikel	<<entity>>
+ connString variabel yang digunakan untuk string koneksi data	
+ scon variabel yang digunakan untuk menginisialisasi variable	

connString ke *SqlConnection*.

+ dt

variable yang digunakan untuk menampung data posisi partikel

+ jumlahTranslasi

variable yang digunakan untuk menampung data jumlah translasi pada pergerakan terhadap diri sendiri

+ permutasiIntern

variable yang digunakan untuk menampung data permutasi pada pergerakan terhadap diri sendiri

,+ partikel(jumlahHari, jumlahSesi, jumlahRuang, dsKelasMataKuliah, JumlahKelasMataKuliah, jumlahJadwal)
Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas partikel.

+ getDT()

Operasi ini digunakan untuk mengambil data partikel.

+ isAda(dtInput, x, jumlahData, column)

Operasi ini digunakan untuk melihat apakah variable yang ada sudah ada di dalam data random.

+ isAdaBadPoint(dsKelasMataKuliah, jumlahJadwal, jumlahKelasMataKuliah)

Operasi ini digunakan untuk melihat ada hard constraint yang dilanggar.

+ cekIdDosen(idKelasMataKuliah, dsKelasMataKuliah, jumlahKelasMataKuliah)

Operasi ini digunakan untuk mengecek id dosen sebelum melihat lbest gbest, dimana id dosen digunakan untuk mengecek kendala dosen

+ setKolomTimetable

Operasi ini digunakan untuk mengeset dimensi timetable.

+ setDasarTabelTimetable(jumlahHari, jumlahSesi,

jumlahRuang)

Operasi ini digunakan untuk mengeset nilai dimensi timetable.

```
+ generateRandomAwalPartikel(dsKelasMataKuliah,  
jumlahKelasMataKuliah, jumlahJadwal)
```

Operasi ini digunakan untuk mengeset nilai dimensi dimensi di dalam memory.

```
+ cekSoftConstraint(dsKelasMataKuliah, dsKendalaDosen,  
DataSet dsKendalaMahasiswa, jumlahJadwal,  
jumlahKelasMataKuliah, jumlahKendalaDosen,  
jumlahKendalaMahasiswa)
```

Operasi ini digunakan untuk mengecek nilai pelanggaran terhadap softConstraint.

```
+ cekHardConstraint(jumlahJadwal, dsKelasMataKuliah,  
jumlahKelasMataKuliah)
```

Operasi ini digunakan untuk mengecek nilai pelanggaran terhadap hardConstraint.

```
+ setJumlahTranslasiToSelf(int jumlahJadwal)
```

Operasi ini digunakan untuk mmengeset translasi terhadap diri sendiri.

```
+ setJumlahTranslasiToLBest(DataTable dtBestPosition,  
int jumlahJadwal)
```

Operasi ini digunakan untuk mmengeset translasi terhadap LBest.

```
+ setJumlahTranslasiToGBest(DataTable dtBestPosition,  
int jumlahJadwal)
```

Operasi ini digunakan untuk mmengeset translasi terhadap Gbest.

```
+ Translasi(int jumlahJadwal, int jumlahTranslasix)
```

Operasi ini merupakan operasi utama dimana komputasi pendekatan dengan translasi pada DPSO dalam memory

dilakukan.

```
+ setJumlahPermutasiToSelf(int jumlahJadwal)
```

Operasi ini digunakan untuk mengeset permutasi terhadap diri sendiri.

```
+ setJumlahPermutasiToLBest(DataTable dtLBestPosition,  
int jumlahJadwal)
```

Operasi ini digunakan untuk mengeset permutasi terhadap LBest.

```
+ setJumlahPermutasiToGBest(DataTable dtGBestPosition,  
int jumlahJadwal)
```

Operasi ini digunakan untuk mengeset permutasi terhadap GBest.

```
+ permutasiToSelf(int jumlahPermutasi, DataTable  
dtPendekatan)
```

Operasi ini merupakan operasi utama dimana komputasi pendekatan terhadap diri sendiri dengan permutasi pada DPSO dalam memory dilakukan.

```
+ permutasiToBest(int jumlahPermutasi, DataTable  
dtPendekatan)
```

Operasi ini merupakan operasi utama dimana komputasi pendekatan terhadap posisi terbaik dengan permutasi pada DPSO dalam memory dilakukan.

```
+ getJumlahTranslasiToSelf()
```

Operasi ini digunakan untuk mengambil jumlah translasi terhadap diri sendiri.

```
+ getJumlahTranslasiToLBest()
```

Operasi ini digunakan untuk mengambil jumlah translasi terhadap LBest.

```
+ getJumlahTranslasiToGBest()
```

Operasi ini digunakan untuk mengambil jumlah translasi terhadap Gbest.

+ getJumlahPermutasiToSelf()

Operasi ini digunakan untuk mengambil jumlah permutasi terhadap diri sendiri.

+ getJumlahPermutasiToLBest()

Operasi ini digunakan untuk mengambil jumlah permutasi terhadap LBest.

+ getJumlahPermutasiToGBest()

Operasi ini digunakan untuk mengambil jumlah permutasi terhadap GBest.

+ getPermutasiToSelf()

Operasi ini digunakan untuk mengambil data langhah permutasi terhadap diri sendiri.

+ getPermutasiToLBest()

Operasi ini digunakan untuk mengambil data langhah permutasi terhadap LBest.

+ getPermutasiToGBest()

Operasi ini digunakan untuk mengambil data langhah permutasi terhadap GBest.

+ antiHardConstraint(dsKelasMataKuliah, jumlahJadwal, jumlahKelasMataKuliah)

Operasi ini digunakan untuk melakukan transposisi khusus terhadap dimensi yang melanggar hardConstraint.

Operasi ini digunakan untuk melakukan transposisi khusus terhadap dimensi yang melanggar softConstraint kendala dosen.

d. Spesific Design Class generateJadwal

generateJadwal

<<entity>>

+ connString

variabel yang digunakan untuk string koneksi data

+ scon

variabel yang digunakan untuk menginisialisasi variable
connString ke *SqlConnection*.

+ generateJadwal()

Konstruktor, digunakan untuk inisialisasi semua
attribute dari kelas generateJadwal.

+ clearDataJadwal()

Operasi ini digunakan untuk mengosongkan data jadwal.

+ addJadwal(id, idKelasMataKuliah, idRefHari,
idRefSesi, idRefRuang)

Operasi ini digunakan untuk menambah data jadwal.

+ getAllDataJadwal()

Operasi ini digunakan untuk mengambil semua data
jadwal.

+ getAllDataJadwalJoin()

Operasi ini digunakan untuk mengambil semua data jadwal
yang sudah join dengan data lain.

+ getAllDataJadwalJoin2()

Operasi ini digunakan untuk mengambil semua data jadwal
yang sudah join dengan data lain.

e. Spesific Design Class manageDosen

manageDosen

<<entity>>

+ connString

variabel yang digunakan untuk string koneksi data

+ scon

variabel yang digunakan untuk menginisialisasi variable
connString ke *SqlConnection*.

+ manageDosen()

Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ManageDosen.

+ add(id, namaDosen, nip, aktif)

Operasi ini digunakan untuk menambah data dosen.

+ update(id, namaDosen, nip, aktif)

Operasi ini digunakan untuk mengubah data dosen.

+ delete(id)

Operasi ini digunakan untuk menghapus data dosen.

+ getNewId()

Operasi ini digunakan untuk melihat id tertinggi, dan memunculkan id baru yang memiliki angka lebih tinggi 1 point dari id tertinggi yang sudah ada.

+ getJumlahDosen()

Operasi ini digunakan untuk mengambil jumlah dosen.

+ getDataById(id)

Operasi ini digunakan untuk mengambil data dosen sesuai dengan id.

+ getAllDataDosen()

Operasi ini digunakan untuk mengambil semua data dosen.

+ getAllDataDosen2()

Operasi ini digunakan untuk mengambil semua data dosen.

f. Spesific

Design

Class

manageKelasMataKuliah

manageKelasMataKuliah	<<entity>>
+ connString variabel yang digunakan untuk string koneksi data	
+ scon variabel yang digunakan untuk menginisialisasi variable	

connString ke *SqlConnection*.

+ manageKelasMataKuliah()

Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ManageKelasMataKuliah.

+ add(id, int idMK, idDosen, kelas)

Operasi ini digunakan untuk menambah data kelasMataKuliah.

+ update (id, int idMK, idDosen, kelas)

Operasi ini digunakan untuk mengubah data kelasMataKuliah.

+ delete(id)

Operasi ini digunakan untuk menghapus data kelasMataKuliah.

+ getNewId()

Operasi ini digunakan untuk melihat id tertinggi, dan memunculkan id baru yang memiliki angka lebih tinggi 1 point dari id tertinggi yang sudah ada.

+ getJumlahKelasMataKuliah()

Operasi ini digunakan untuk mengambil jumlah kelas mata kuliah.

+ getDataById(id)

Operasi ini digunakan untuk mengambil data kelasMataKuliah sesuai dengan id.

+ getAllKelasMataKuliah()

Operasi ini digunakan untuk mengambil semua data kelasMataKuliah.

+ getAllKelasMataKuliahJoin()

Operasi ini digunakan untuk mengambil semua data kelasMataKuliah yang sudah join dengan data lain.

g. Spesific Design Class manageKendalaDosen

Program Studi Teknik Informatika	DPPL- SPJK	75/ 102
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

managekendalaDosen	<<entity>>
+ connString variabel yang digunakan untuk string koneksi data + scon variabel yang digunakan untuk menginisialisasi variable connString ke <i>SqlConnection</i> .	
+ manageKendalaDosen() Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ManageKendalaDosen. + add(id, idRefHari, idDosen, idRefSesi, nilaiKendala) Operasi ini digunakan untuk menambah data kendalaDosen. + update(id, idRefHari, idDosen, idRefSesi, nilaiKendala) Operasi ini digunakan untuk mengubah data kendalaDosen. + delete(id) Operasi ini digunakan untuk menghapus data kendalaDosen. + getNewId() Operasi ini digunakan untuk melihat id tertinggi, dan memunculkan id baru yang memiliki angka lebih tinggi 1 point dari id tertinggi yang sudah ada. + getJumlahKendalaDosen() Operasi ini digunakan untuk mengambil jumlah kendala dosen. + getDataById(id) Operasi ini digunakan untuk mengambil data kendalaDosen sesuai dengan id. + getAllDataKendalaDosen() Operasi ini digunakan untuk mengambil semua data kendalaDosen. + getAllDataKendalaDosenJoin()	

Operasi ini digunakan untuk mengambil semua data kendalaDosen yang sudah join dengan data lain.

h. Spesific

Design

Class

manageKendalaMahasiswa

managekendalaMahasiswa	<<entity>>
+ connString variabel yang digunakan untuk string koneksi data	
+ scon variabel yang digunakan untuk menginisialisasi variable connString ke <i>SqlConnection</i> .	
+ manageKendalaMahasiswa() Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ManageKendalaMahasiswa.	
+ add(id, idRefHari, idRefSesi, nilaiKendala) Operasi ini digunakan untuk menambah data kendalaMahasiswa.	
+ update(id, idRefHari, idRefSesi, nilaiKendala) Operasi ini digunakan untuk mengubah data kendalaMahasiswa.	
+ delete(id) Operasi ini digunakan untuk menghapus data kendalaMahasiswa.	
+ getNewId() Operasi ini digunakan untuk melihat id tertinggi, dan memunculkan id baru yang memiliki angka lebih tinggi 1 point dari id tertinggi yang sudah ada.	
+ getJumlahKendalaMahasiswa() Operasi ini digunakan untuk mengambil jumlah	

kendalaMahasiswa.

+ getDataById(id)

Operasi ini digunakan untuk mengambil data kendalaMahasiswa sesuai dengan id.

+ getAllDataKendalaMahasiswa()

Operasi ini digunakan untuk mengambil semua data kendalaMahasiswa.

+ getAllDataKendalaMahasiswaJoin()

Operasi ini digunakan untuk mengambil semua data kendalaMahasiswa yang sudah join dengan data lain.

i. Spesific Design Class manageMataKuliah

manageMataKuliah

<<entity>>

+ connString

variabel yang digunakan untuk string koneksi data

+ scon

variabel yang digunakan untuk menginisialisasi variable connString ke *SqlConnection*.

+ manageMataKuliah()

Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas manageMataKuliah.

+ add(id, kodeMK, namaMK, semester, jumlahSKS)

Operasi ini digunakan untuk menambah data mataKuliah.

+ update(id, kodeMK, namaMK, semester, jumlahSKS)

Operasi ini digunakan untuk mengubah data mataKuliah.

+ delete(id)

Operasi ini digunakan untuk menghapus data mataKuliah.

+ getNewId()

Operasi ini digunakan untuk melihat id tertinggi, dan memunculkan id baru yang memiliki angka lebih tinggi 1

point dari id tertinggi yang sudah ada.

+ getJumlahMataKuliah()

Operasi ini digunakan untuk mengambil jumlah mataKuliah.

+ getDataById(id)

Operasi ini digunakan untuk mengambil data mataKuliah sesuai dengan id.

+ getAllDataMataKuliah()

Operasi ini digunakan untuk mengambil semua data mataKuliah.

+ getAllDataMataKuliah2()

Operasi ini digunakan untuk mengambil semua data mataKuliah.

j. Spesific Design Class manageRefHari

manageRefHari

<<entity>>

+ connString

variabel yang digunakan untuk string koneksi data

+ scon

variabel yang digunakan untuk menginisialisasi variable connString ke *SqlConnection*.

+ manageRefHari()

Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas manageRefHari.

+ add(id, hari)

Operasi ini digunakan untuk menambah data refHari.

+ update(id, hari)

Operasi ini digunakan untuk mengubah data refHari.

+ delete(id)

Operasi ini digunakan untuk menghapus data refHari.

+ getNewId()

Operasi ini digunakan untuk melihat id tertinggi, dan memunculkan id baru yang memiliki angka lebih tinggi 1 point dari id tertinggi yang sudah ada.

+ getJumlahHari()

Operasi ini digunakan untuk mengambil jumlah refHari.

+ getDataById(id)

Operasi ini digunakan untuk mengambil data refHari sesuai dengan id.

+ getAllDataRefHariTampil()

Operasi ini digunakan untuk mengambil semua data refHari.

+ getAllDataRefHari()

Operasi ini digunakan untuk mengambil semua data refHari yang isDeleted sama dengan 0.

k. Spesific Design Class manageRefSesi

manageRefSesi	<<entity>>
+ connString variabel yang digunakan untuk string koneksi data	
+ scon variabel yang digunakan untuk menginisialisasi variable connString ke <i>SqlConnection</i> .	
+ manageRefSesi() Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas manageRefSesi.	
+ add(id, jamMulai, jamSelesai) Operasi ini digunakan untuk menambah data refSesi.	
+ update(id, jamMulai, jamSelesai) Operasi ini digunakan untuk mengubah data refSesi.	
+ delete(id)	

Operasi ini digunakan untuk menghapus data refSesi.

+ getNewId()

Operasi ini digunakan untuk melihat id tertinggi, dan memunculkan id baru yang memiliki angka lebih tinggi 1 point dari id tertinggi yang sudah ada.

+ getJumlahSesi()

Operasi ini digunakan untuk mengambil jumlah refSesi.

+ getDataById(id)

Operasi ini digunakan untuk mengambil data refSesi sesuai dengan id.

+ getAllDataRefSesiTampil()

Operasi ini digunakan untuk mengambil semua data refSesi.

+ getAllDataRefSesi()

Operasi ini digunakan untuk mengambil semua data refSesi yang isDeleted sama dengan 0.

1. Spesific Design Class manageRefRuang

manageRefRuang

<<entity>>

+ connString

variabel yang digunakan untuk string koneksi data

+ scon

variabel yang digunakan untuk menginisialisasi variable connString ke *SqlConnection*.

+ manageRefRuang()

Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas manageRefRuang.

+ add(id, namaRuang)

Operasi ini digunakan untuk menambah data refRuang.

+ update(id, namaRuang)

Operasi ini digunakan untuk mengubah data refRuang.

+ delete(id)

Operasi ini digunakan untuk menghapus data refRuang.

+ getNewId()

Operasi ini digunakan untuk melihat id tertinggi, dan memunculkan id baru yang memiliki angka lebih tinggi 1 point dari id tertinggi yang sudah ada.

+ getJumlahRuang()

Operasi ini digunakan untuk mengambil jumlah refRuang.

+ getDataById(id)

Operasi ini digunakan untuk mengambil data refRuang sesuai dengan id.

+ getAllDataRefRuangTampil()

Operasi ini digunakan untuk mengambil semua data refRuang.

+ getAllDataRefRuang()

Operasi ini digunakan untuk mengambil semua data refRuang yang isDeleted sama dengan 0.

m. Spesific Design Class manageRefKelas

manageRefKelas	<<entity>>
+ connString variabel yang digunakan untuk string koneksi data	
+ scon variabel yang digunakan untuk menginisialisasi variable connString ke <i>SqlConnection</i> .	
+ manageRefKelas() Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas manageRefKelas.	
+ add(kelas)	

Operasi ini digunakan untuk menambah data refKelas.
+ update(oldKelas, newKelas)
Operasi ini digunakan untuk mengubah data refKelas.
+ delete(kelas)
Operasi ini digunakan untuk menghapus data refKelas.
+ getDataById
Operasi ini digunakan untuk mengambil data refRuang sesuai dengan id.
+ getJumlahKelas()
Operasi ini digunakan untuk mengambil jumlah refKelas.
+ getAllDataRefKelasTampil()
Operasi ini digunakan untuk mengambil semua data refKelas.
+ getAllDataRefKelas()
Operasi ini digunakan untuk mengambil semua data refKelas yang isDeleted sama dengan 0.

n. Spesific Design Class manageRefSemester

manageRefSemester	<<entity>>
+ connString variabel yang digunakan untuk string koneksi data + scon variabel yang digunakan untuk menginisialisasi variable connString ke <i>SqlConnection</i> .	
+ manageRefSemester() Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas manageRefSemester. + add(semester) Operasi ini digunakan untuk menambah data refSemester. + update(oldSemester, newSemester)	

Operasi ini digunakan untuk mengubah data refSemester.

```
+ delete(semester)
```

Operasi ini digunakan untuk menghapus data refSemester.

```
+ getDataById
```

Operasi ini digunakan untuk mengambil data refSemester sesuai dengan id.

```
+ getJumlahSemester()
```

Operasi ini digunakan untuk mengambil jumlah refSemester.

```
+ getAllDataRefSemesterTampil()
```

Operasi ini digunakan untuk mengambil semua data refSemester.

```
+ getAllDataRefSemester()
```

Operasi ini digunakan untuk mengambil semua data refSemester yang isDeleted sama dengan 0.

o. Spesific Design Class manageUser

manageUser	<<entity>>
<pre>+ connString</pre> variabel yang digunakan untuk string koneksi data	
<pre>+ scon</pre> variabel yang digunakan untuk menginisialisasi variable connString ke <i>SqlConnection</i>.	
<pre>+ manageUser()</pre> Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas manageUser.	
<pre>+ add(id, idRefRole, namaUser, password, alamat, telephone, email)</pre> Operasi ini digunakan untuk menambah data userLogin.	
<pre>+ update(id, idRefRole, namaUser, password, alamat,</pre>	

telephone, email)

Operasi ini digunakan untuk mengubah data userLogin.

+ delete(id)

Operasi ini digunakan untuk menghapus data userLogin.

+ getNewId()

Operasi ini digunakan untuk melihat id tertinggi, dan memunculkan id baru yang memiliki angka lebih tinggi 1 point dari id tertinggi yang sudah ada.

+ getJumlahUser()

Operasi ini digunakan untuk mengambil jumlah userLogin.

+ getDataById(id)

Operasi ini digunakan untuk mengambil data userLogin sesuai dengan id.

+ getAllDataUser()

Operasi ini digunakan untuk mengambil semua data userLogin.

+ getAllDataUserJoin()

Operasi ini digunakan untuk mengambil semua data userLogin.

+ getUserLogin(nama, password)

Operasi ini digunakan untuk mengambil data user login berdasar nama dan password

p. Spesific Design Class manageRefRole

manageRefRole

<<entity>>

+ connString

variabel yang digunakan untuk string koneksi data

+ scon

variabel yang digunakan untuk menginisialisasi variable connString ke *SqlConnection*.

+ manageRefRole()

Konstruktor, digunakan untuk inisialisasi semua attribute dari kelas manageRefRole.

+ add(id)

Operasi ini digunakan untuk menambah data refRole.

+ update(id, role)

Operasi ini digunakan untuk mengubah data refRole.

+ delete(id)

Operasi ini digunakan untuk menghapus data refRole.

+ getNewId()

Operasi ini digunakan untuk melihat id tertinggi, dan memunculkan id baru yang memiliki angka lebih tinggi 1 point dari id tertinggi yang sudah ada.

+ getDataById(id)

Operasi ini digunakan untuk mengambil data refRole sesuai dengan id.

+ getAllDataRefRoleTampil()

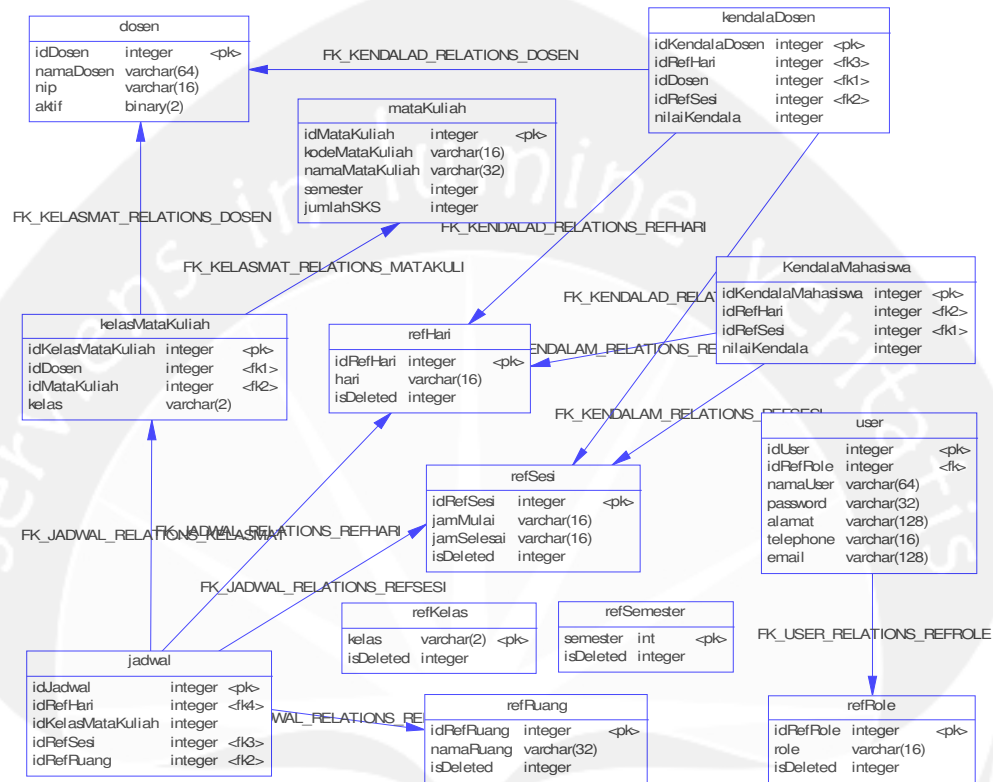
Operasi ini digunakan untuk mengambil semua data refRole.

+ getAllDataRefRole()

Operasi ini digunakan untuk mengambil semua data refRole yang isDeleted sama dengan 0.

3. Perancangan Data (Data Design)

3.1 Pyhsical Data Model



Gambar 55 Physical Data Model

4. Perancangan Antarmuka (UI Design)

4.1 Antarmuka Login

Sistem Pengaturan Jadwal Kuliah			
Nama			
Password			
Login			
<table border="1"><tr><td></td><td></td></tr></table> ☐ Excel Processing ☐ Add ☐ Edit ☐ Delete Id Cek Id ☐ Is Deleted			

Gambar 56 Rancangan Antarmuka Login

Form ini digunakan oleh user untuk login sebagai admin, petugas TU untuk login. Menu di panel sebelah kiri belum dapat diakses, dan dapat diakses pada saat user sudah melakukan login. Untuk melakukan login, user mengisi kolom nama, kolom password, lalu menekan login. Setelah login dilakukan, tombol login akan berubah menjadi logout, dan muncul menu sesuai dengan tipe user.

4.2 Antarmuka Generate Jadwal

Gambar 57 Rancangan Antarmuka Generate Jadwal

Form ini digunakan oleh user setelah login sebagai Kaprodi dan setelah memilih menu Generate Jadwal. Menu excel processing, add, edit, delete, id, cek id, is deleted tidak bisa diakses (disable) karena tidak digunakan dalam generate jadwal. Untuk melakukan generate jadwal, user mengisi kolom jumlah partikel, epoch Setelah data selesai dimasukkan, user menekan generate. Setelah jadwal digenerate, data kendala dan Tabel Jadwal berisi data jadwal akan tampil. Untuk melihat report dari jadwal, user mencentang pilihan report.

4.3 Antarmuka Report

The screenshot shows the 'Main Report' window of the UAJS system. The window title is 'Main Report'. The main content area displays a table of course schedules for the date 28/02/2013. The table has the following columns: Hari, Sesi, Ruang, MataKuliah, Kelas, and Dosen. The data is organized by day and session, with each row representing a specific course and its instructor.

Hari	Sesi	Ruang	MataKuliah	Kelas	Dosen
senin	1	3216	Penrograman Visual	B	Yonathan Dri Hardarkho, ST, M.Eng
senin	1	3315	Kalkulus 2	C	Dr. Prsnowo, ST, MT
senin	1	3318	Rekayasa Perangkat Lunak	B	Thomas Adhi P.S., S.T., M.T
senin	1	3415	Penrograman Basis Data	B	Paulus Mudiartono, ST, MT
senin	1	3421			
senin	1	3422			
senin	2	3216	Projek Pembangunan PL Profesor	H	Kusworo Anindito, ST, MT
senin	2	3315	Interaksi Manusia dan Komputer	A	Yonathan Dri Hardarkho, ST, M.Eng
senin	2	3318	Kuliah Lapangan	C	Paulus Mudiartono, ST, MT
senin	2	3415	Komputasi Numerik 1	B	Ir. Alb. Joko Santoso, MT
senin	2	3421	Sistem Berbasis Pengetahuan	B	Dra. Emawati, MT
senin	2	3422	Basis Data	B	Findra Kartika Sari Dewi, S.T., M.T
senin	3	3216			
senin	3	3315	Kalkulus 2	A	Ir. Alb. Joko Santoso, MT
senin	3	3318	Aritektur Komputer	A	Eddy Julianto, ST, MT
senin	3	3415			
senin	3	3421	Algoritma dan Penrograman	C	Eduard Rusdianto, ST, MT
senin	3	3422	Metodologi Penelitian dan Publikasi	B	Prof. Ir. Suyoto, M.Sc, Ph.D
senin	4	3216	Projek Pembangunan PL Profesor	G	Thomas Adhi P.S., S.T., M.T
senin	4	3315	Projek Pembangunan PL Profesor	C	Findra Kartika Sari Dewi, S.T., M.T
senin	4	3318			
senin	4	3415	Jaringan Komputer	F	Thomas Susanto, ST, MT
senin	4	3421	Sistem Digital	A	Eddy Julianto, ST, MT
senin	4	3422	Sistem Digital	E	Th. Dewi Indrasari, ST, M.Sc
selasa	1	3216	Basis Data	C	Irya Winubuhadja, ST, MT
selasa	1	3315	Matematika Diskret	A	Dra. Emawati, MT
selasa	1	3318	Jaringan Komputer	A	Thomas Adhi P.S., S.T., M.T
selasa	1	3415	Pendidikan Pancasila dan Kewarg	B	Dosen MKU
selasa	1	3421			
selasa	1	3422	Projek Pembangunan PL Profesor	A	Yonathan Dri Hardarkho, ST, M.Eng
selasa	2	3216	Kuliah Lapangan	D	Ir. Alb. Joko Santoso, MT
selasa	2	3315	Sistem Digital	C	Thomas Adhi P.S., S.T., M.T
selasa	2	3318	Grafika Komputer Interaktif	B	Prof. Ir. Suyoto, M.Sc, Ph.D
selasa	3	3415			

The sidebar on the left contains the following elements:

- UAJS logo and tagline: 'Serviens in lumine veritatis'
- Generate Jadwal dropdown menu
- Excel Processing checkbox
- Add, Edit, Delete buttons
- Id input field
- Cek Id button
- IsDeleted checkbox
- Jumlah partikel input field (25)
- Epoch input field (25)
- Nilai pelanggaran input field (-990)
- Kendala dosen 1: 1 (2%)
- Kendala dosen 2: 16 (69%)
- Kendala mahasiswa: 4 (66%)
- Lama eksekusi: 00:11:29.8125000
- Generate button
- Report checkbox (checked)

At the bottom of the window, there is a status bar with the following information:

- Current Page No.: 1
- Total Page No.: 14
- Zoom Factor: 100%
- Creator: rochimandayarto@gmail.com

Gambar 58 Rancangan Antarmuka Report

Form ini digunakan oleh user setelah login sebagai Kaprodi dan setelah memilih menu Generate Jadwal dan mencentang pilihan report. Antar muka ini digunakan untuk mencetak jadwal.

4.4 Antarmuka Manage Dosen

Sistem Pengaturan Jadwal Kuliah		-	x
Nama		Password	Logout
Manage Dosen ☐ Excel Processing ☐ Add ☐ Edit ☐ Delete Id Cek Id ☐ Is Deleted Nama Nip Status Data Execute		Tabel Dosen	

Gambar 59 Rancangan Antarmuka Manage Dosen

Form ini digunakan oleh user setelah login sebagai Kaprodi dan setelah memilih menu Manage Dosen. Pada tampilan Manage Dosen, pilihan excel processing, add, edit, delete, id dan cek id dapat di akses. Untuk melakukan add, maka kolom id terisi otomatis. User mengisi kolom nama, nip, dan memilih status data setelah itu menekan execute. Untuk melakukan edit, user memasukkan id, lalu menekan tombol cek id. Setelah itu user mengganti data yang sesuai dan menekan execute. Untuk melakukan delete, user memasukkan id lalu menekan tombol Cek Id. Setelah memilih data user menekan execute. Untuk melakukan export-import data dari excel, user mencentang pilihan excel processing.

4.5 Antarmuka Manage Kelas Mata Kuliah

Sistem Pengaturan Jadwal Kuliah

Nama Password Logout

Manage Kelas Mata Kuliah

☐ Excel Processing
☐ Add ☐ Edit ☐ Delete

Id

Cek Id

☐ Is Deleted

MK

Dosen

Kelas

Execute

Tabel Kelas Mata Kuliah

Gambar 60 Rancangan Antarmuka Manage Kelas Mata Kuliah

Form ini digunakan oleh user setelah login sebagai Kaprodi dan setelah masuk ke tampilan Manage Kelas Mata Kuliah. Pada tampilan Manage Kelas Mata Kuliah, pilihan excel processing, add, edit, delete, id dan cek id dapat di akses. Untuk melakukan add, maka kolom id terisi otomatis. User memilih MK, dosen dan kelas. Setelah data selesai dimasukkan, user menekan execute. Untuk melakukan edit, user memasukkan id, lalu menekan tombol cek id. Setelah itu user mengganti data yang sesuai dan menekan execute. Untuk melakukan delete, user memasukkan id lalu menekan tombol Cek Id. Setelah memilih data, user menekan execute. Untuk melakukan export-import data dari excel, user mencentang pilihan excel processing.

4.6 Antarmuka Manage Kendala Mahasiswa

Sistem Pengaturan Jadwal Kuliah

Nama Password

☐ Excel Processing
☐ Add ☐ Edit ☐ Delete

Id

☐ Is Deleted

Nilai Kendala

Hari

Sesi

Tabel Kendala Mahasiswa

Gambar 61 Rancangan Antarmuka Manage Kendala Mahasiswa

Form ini digunakan oleh user setelah login sebagai Kaprodi dan setelah masuk ke tampilan Manage Kendala Mahasiswa. Pada tampilan Manage Kendala Mahasiswa, pilihan add, edit, delete, id dan cek id dapat diakses. Untuk melakukan add, maka kolom id terisi otomatis. User memilih hari dan sesi. Setelah data selesai dimasukkan, user menekan execute. Untuk melakukan edit, user memasukkan id, lalu menekan tombol cek id. Setelah memilih data, user mengganti data yang sesuai dan menekan execute. Untuk melakukan delete, user memasukkan id lalu menekan tombol Cek Id. Setelah memilih data, user menekan execute.

4.7 Antarmuka Manage Kendala Dosen

Gambar 62 Rancangan Antarmuka Manage Kendala Dosen

Form ini digunakan oleh user setelah login sebagai Kaprodi dan setelah masuk ke tampilan Manage Kendala Dosen. Pada tampilan Manage Kendala Dosen, pilihan add, edit, delete, id dan cek id dapat di akses. Untuk melakukan add, maka kolom id terisi otomatis. User memilih dosen, hari sesi, dan mencentang/tidak pilihan ingin. Untuk mengganti nilai kendala, user mencentang unlock nilai kendala. Setelah data selesai dimasukkan, user menekan execute. Untuk melakukan edit, user memasukkan id, lalu menekan tombol cek id. Setelah itu user mengganti data yang sesuai dan menekan execute. Untuk melakukan delete, user memasukkan id lalu menekan tombol Cek Id dan menekan execute.

4.8 Antarmuka Manage Mata Kuliah

Sistem Pengaturan Jadwal Kuliah

Nama Password Logout

Manage Mata Kuliah

☐ Excel Processing
☐ Add ☐ Edit ☐ Delete

Id

Cek Id

Kode MK

Nama MK

Semester

Jumlah SKS

Execute

Tabel Mata Kuliah

Gambar 63 Rancangan Antarmuka Manage Mata Kuliah

Form ini digunakan oleh user setelah login sebagai Petugas TU dan setelah masuk ke tampilan Manage Mata Kuliah. Pada tampilan Manage Mata Kuliah, pilihan excel processing, add, edit, delete, id dan cek id dapat diakses. Untuk melakukan add, maka kolom id terisi otomatis. User mengisi kode MK, nama MK, jumlah SKS dan memilih semester. Setelah data selesai dimasukkan, user menekan execute. Untuk melakukan edit, user memasukkan id, lalu menekan tombol cek id. Setelah itu user mengganti data yang sesuai dan menekan execute. Untuk melakukan delete, user memasukkan id lalu menekan tombol Cek Id. Setelah memilih data, user menekan execute. Untuk melakukan export-import data dari excel, user mencentang pilihan excel processing.

4.9 Antarmuka Manage Ref Hari

Sistem Pengaturan Jadwal Kuliah

Nama Password Logout

☐ Excel Processing
☐ Add ☐ Edit ☐ Delete

Id

☐ Is Deleted

Hari

Tabel Ref Hari

Gambar 64 Rancangan Antarmuka Manage Ref Hari

Form ini digunakan oleh user setelah login sebagai Petugas TU dan setelah masuk ke tampilan Manage Ref Hari. Pada tampilan Manage Ref Hari, pilihan add, edit, delete, is deleted, id dan cek id dapat di akses. Untuk melakukan add, maka kolom id terisi otomatis. User mengisi kolom hari. Setelah data selesai dimasukkan, user menekan execute. Untuk melakukan edit, user memasukkan id, lalu menekan tombol cek id. Setelah itu user mengganti data yang sesuai dan menekan execute. Untuk melakukan delete, user memasukkan id lalu menekan tombol Cek Id. Setelah memilih data user menekan execute.

4.10 Antarmuka Manage Ref Sesi

Sistem Pengaturan Jadwal Kuliah

Nama Password Logout

☐ Excel Processing
☐ Add ☐ Edit ☐ Delete

Id

☐ Is Deleted

Jam Mulai

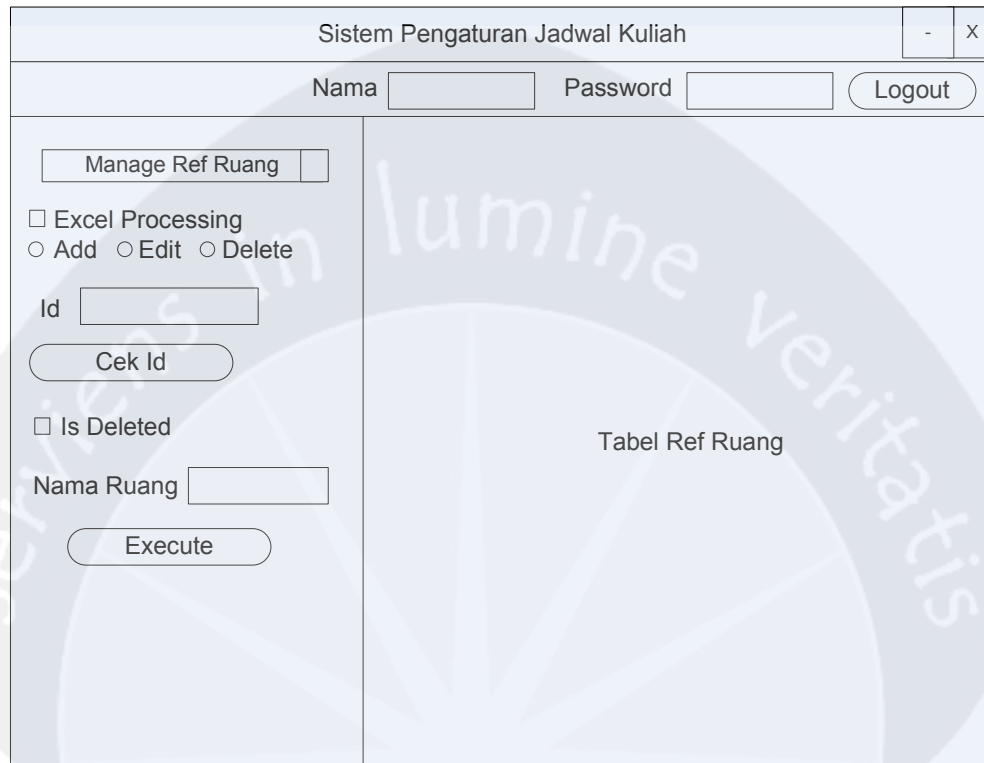
Jam Selesai

Tabel Ref Sesi

Gambar 65 Rancangan Antarmuka Manage Ref Sesi

Form ini digunakan oleh user setelah login sebagai Petugas TU dan setelah masuk ke tampilan Manage Ref Sesi. Pada tampilan Manage Ref Sesi, pilihan add, edit, delete, is deleted, id dan cek id dapat di akses. Untuk melakukan add, maka kolom id terisi otomatis. User mengisi kolom sesi. Setelah data selesai dimasukkan, user menekan execute. Untuk melakukan edit, user memasukkan id, lalu menekan tombol cek id. Setelah itu user mengganti data yang sesuai dan menekan execute. Untuk melakukan delete, user memasukkan id lalu menekan tombol Cek Id. Setelah memilih data user menekan execute.

4.11 Antarmuka Manage Ref Ruang



Sistem Pengaturan Jadwal Kuliah

Nama Password Logout

☐ Excel Processing
☐ Add ☐ Edit ☐ Delete

Id

☐ Is Deleted

Nama Ruang

Tabel Ref Ruang

Gambar 66 Rancangan Antarmuka Manage Ref Ruang

Form ini digunakan oleh user setelah login sebagai Petugas TU dan setelah masuk ke tampilan Manage Ref Ruang. Pada tampilan Manage Ref Ruang, pilihan add, edit, delete, is deleted, id dan cek id dapat di akses. Untuk melakukan add, maka kolom id terisi otomatis. User mengisi kolom ruang. Setelah data selesai dimasukkan, user menekan execute. Untuk melakukan edit, user memasukkan id, lalu menekan tombol cek id. Setelah itu user mengganti data yang sesuai dan menekan execute. Untuk melakukan delete, user memasukkan id lalu menekan tombol Cek Id. Setelah memilih data user menekan execute.

4.12 Antarmuka Manage Ref Kelas

Sistem Pengaturan Jadwal Kuliah

Nama Password Logout

☐ Excel Processing
☐ Add ☐ Edit ☐ Delete

Kelas

☐ Is Deleted

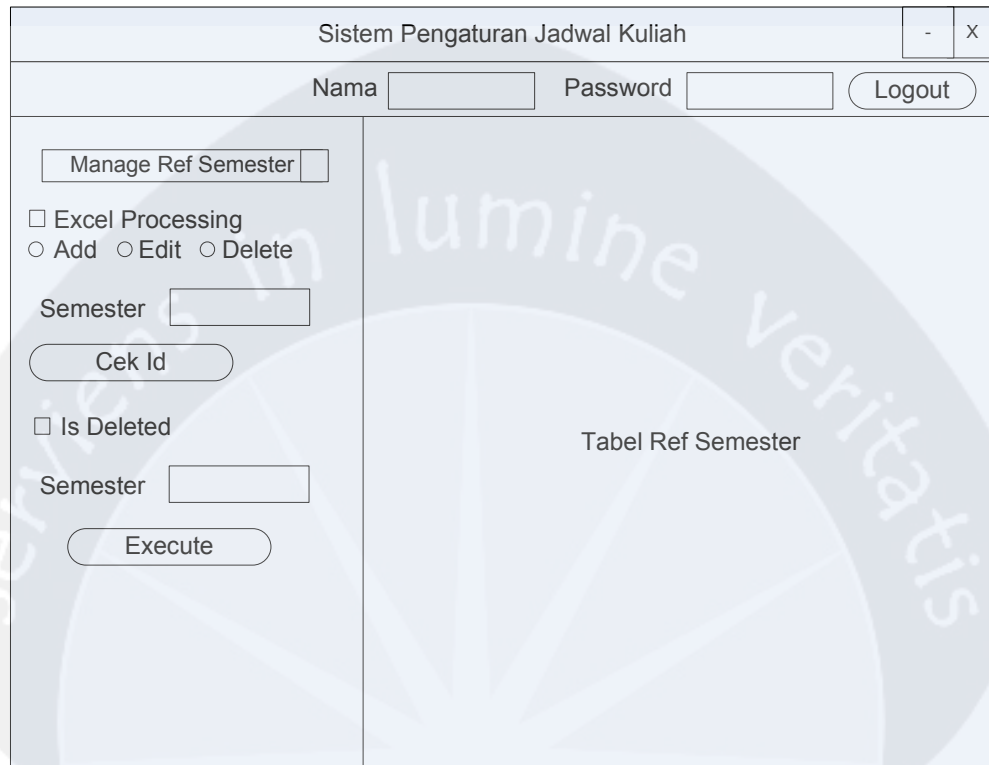
Kelas

Tabel Ref Kelas

Gambar 67 Rancangan Antarmuka Manage Ref Kelas

Form ini digunakan oleh user setelah login sebagai Petugas TU dan setelah masuk ke tampilan Manage Ref Kelas. Pada tampilan Manage Ref Kelas, pilihan add, edit, delete, is deleted, kelas dan cek id dapat diakses. Untuk melakukan add, maka user mengisi kolom kelas. Setelah data selesai dimasukkan, user menekan execute. Untuk melakukan edit, user memasukkan kelas, lalu menekan tombol cek id. Setelah itu user mengganti data yang sesuai dan menekan execute. Untuk melakukan delete, user memasukkan id lalu menekan tombol Cek Id. Setelah memilih data user menekan execute.

4.13 Antarmuka Manage Ref Semester



Sistem Pengaturan Jadwal Kuliah

Nama Password Logout

☐ Excel Processing
☐ Add ☐ Edit ☐ Delete

Semester

☐ Is Deleted

Semester

Tabel Ref Semester

Gambar 68 Rancangan Antarmuka Manage Ref Semester

Form ini digunakan oleh user setelah login sebagai Petugas TU dan setelah masuk ke tampilan Manage Ref Semester. Pada tampilan Manage Ref Semester, pilihan add, edit, delete, is deleted, id dan cek id dapat diakses. Untuk melakukan add, maka user mengisi kolom semester. Setelah data selesai dimasukkan, user menekan execute. Untuk melakukan edit, user memasukkan semester, lalu menekan tombol cek id. Setelah itu user mengganti data yang sesuai dan menekan execute. Untuk melakukan delete, user memasukkan semester lalu menekan tombol Cek Id. Setelah memilih data user menekan execute.

4.14 Antarmuka Manage User

Gambar 69 Rancangan Antarmuka Manage User

Form ini digunakan oleh user setelah login sebagai Admin dan setelah masuk ke tampilan Manage User. Pada tampilan Manage User, pilihan add, edit, delete, id dan cek id dapat di akses. Untuk melakukan add, maka kolom id terisi otomatis. User mengisi kolom nama, password, telephone, alamat dan email. Setelah data selesai dimasukkan, user menekan execute. Untuk melakukan edit, user memasukkan id, lalu menekan tombol cek id. Setelah itu user mengganti data yang sesuai dan menekan execute. Untuk melakukan delete, user memasukkan id lalu menekan tombol Cek Id. Setelah memilih data user menekan execute.

4.15 Antarmuka Manage Ref Role

Sistem Pengaturan Jadwal Kuliah

Nama Password Logout

Manage Ref Role

☐ Excel Processing
☐ Add ☐ Edit ☐ Delete

Id

Cek Id

☐ Is Deleted

Role

Execute

Tabel Ref Role

Gambar 70 Rancangan Antarmuka Manage Ref Role

Form ini digunakan oleh user setelah login sebagai Admin dan setelah masuk ke tampilan Manage User. Pada tampilan Manage Ref Role, pilihan add, edit, delete, is deleted, id dan cek id dapat di akses. Untuk melakukan add, maka kolom id terisi otomatis. User mengisi kolom role. Setelah data selesai dimasukkan, user menekan execute. Untuk melakukan edit, user memasukkan id, lalu menekan tombol cek id. Setelah itu user mengganti data yang sesuai dan menekan execute. Untuk melakukan delete, user memasukkan id lalu menekan tombol Cek Id. Setelah memilih data user menekan execute.

LAMPIRAN 3



PDHUPL

PERENCANAAN DESKRIPSI dan HASIL UJI PERANGKAT LUNAK

SISTEM PENGATURAN JADWAL KULIAH

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Rochim Widaryanto/ 105301536

Program Studi Magister Teknik Informatika

Universitas Atma Jaya Yogyakarta

	Program Studi Magister Teknik Informatika	Nomor Dokumen		Halaman
		PDHUPL-SPJK		1/25
		Revisi		2013

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi
Program Studi Magister Teknik Informatika		PDHUPL – SPJK	3/ 25
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika			

Daftar Isi

1	Pendahuluan	3
1.1	Tujuan	3
1.2	Deskripsi Umum Sistem.....	3
1.3	Deskripsi Dokumen (ikhtisar)	3
1.5	Referensi	3
2	Lingkungan Pengujian Perangkat Lunak.....	3
2.1	Perangkat Lunak Pengujian.....	3
2.2	Perangkat Keras Pengujian.....	3
2.3	Material Pengujian.....	3
2.4	Sumber Daya Manusia	3
2.5	Prosedur Umum Pengujian.....	3
3	Identifikasi Rencana Pengujian.....	3
4	Deskripsi Hasil Uji	3
4.1	Identifikasi Kelas Pengujian Login.....	3
4.1.1	Identifikas Butir Pengujian Login.....	3
4.2	Identifikasi Kelas Pengujian manageDosen	3
4.2.1	Identifikas Butir Pengujian Add.....	3
4.2.2	Identifikasi Butir Pengujian Edit.....	3
4.2.3	Identifikasi Butir Pengujian Delete.....	3
4.3	Identifikasi Kelas Pengujian manageKelasMataKuliah.....	3
4.3.1	Identifikas Butir Pengujian Add.....	3
4.3.2	Identifikasi Butir Pengujian Edit.....	3
4.3.3	Identifikasi Butir Pengujian Delete.....	3
4.4	Identifikasi Kelas Pengujian manageKendalaDosen.....	3
4.4.1	Identifikas Butir Pengujian Add.....	3
4.4.2	Identifikasi Butir Pengujian Edit.....	3
4.4.3	Identifikasi Butir Pengujian Delete.....	3
4.5	Identifikasi Kelas Pengujian manageKendalaMahasiswa	3
4.5.1	Identifikas Butir Pengujian Add.....	3
4.5.2	Identifikasi Butir Pengujian Edit.....	3
4.5.3	Identifikasi Butir Pengujian Delete.....	3
4.6	Identifikasi Kelas Pengujian generateJadwal	3
4.6.1	Identifikas Butir Pengujian Generate Jadwal.....	3

Program Studi Magister Teknik Informatika	PDHUPL – SPJK	4/ 25
---	---------------	-------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

4.6.2	Identifikasi Butir Pengujian Show.....	3
4.6.3	Identifikasi Butir Pengujian Delete.....	3
4.7	Identifikasi Kelas Pengujian manageMataKuliah.....	3
4.7.1	Identifikas Butir Pengujian Add.....	3
4.7.2	Identifikasi Butir Pengujian Edit.....	3
4.7.3	Identifikasi Butir Pengujian Delete.....	3
4.8	Identifikasi Kelas Pengujian manageRefSesi.....	3
4.8.1	Identifikas Butir Pengujian Add.....	3
4.8.2	Identifikasi Butir Pengujian Edit.....	3
4.8.3	Identifikasi Butir Pengujian Delete.....	3
4.9	Identifikasi Kelas pengujian manageRefRuang.....	3
4.9.1	Identifikas Butir Pengujian Add.....	3
4.9.2	Identifikasi Butir Pengujian Edit.....	3
4.9.3	Identifikasi Butir Pengujian Delete.....	3
4.10	Identifikasi Kelas Pengujian manageRefHari.....	3
4.10.1	Identifikas Butir Pengujian Add.....	3
4.10.2	Identifikasi Butir Pengujian Edit.....	3
4.10.3	Identifikasi Butir Pengujian Delete.....	3
4.11	Identifikasi Kelas Pengujian manageRefKelas.....	3
4.11.1	Identifikas Butir Pengujian Add.....	3
4.11.2	Identifikasi Butir Pengujian Edit.....	3
4.11.3	Identifikasi Butir Pengujian Delete.....	3
4.12	Identifikasi Kelas Pengujian manageRefSemester.....	3
4.12.1	Identifikas Butir Pengujian Add.....	3
4.12.2	Identifikasi Butir Pengujian Edit.....	3
4.12.3	Identifikasi Butir Pengujian Delete.....	3
4.13	Identifikasi KelasPengujian manageUser.....	3
4.13.1	Identifikas Butir Pengujian Add.....	3
4.13.2	Identifikasi Butir Pengujian Edit.....	3
4.13.3	Identifikasi Butir Pengujian Delete.....	3
4.14	Identifikasi KelasPengujian manageRefRole.....	3
4.14.1	Identifikas Butir Pengujian Add.....	3
4.14.2	Identifikasi Butir Pengujian Edit.....	3
4.14.3	Identifikasi Butir Pengujian Delete.....	3

Daftar Tabel

Tabel 1. Definisi	8
Tabel 2. Identifikasi Pengujian	11
Tabel 3. Deskripsi Hasil Pengujian	21



1 Pendahuluan

1.1 Tujuan

Dokumen Perencanaan Deskripsi dan Hasil Uji Perangkat Lunak Time Table Perkuliahan dengan Discrete Particle Swarm Optimization (PDHUPL-SPJK) ini merupakan dokumen yang berisi perencanaan, deskripsi, dan hasil pengujian perangkat lunak SPJK. Dokumen ini dibuat untuk Universitas Atma Jaya Yogyakarta pada program S1. selanjutnya dokumen ini digunakan untuk menguji keseluruhan sistem SPJK.

1.2 Deskripsi Umum Sistem

SPJK adalah sistem yang bertugas untuk men-generate timetable perkuliahan. Komponen sistem ini digunakan untuk men-generate timetable, dan melakukan pengelolaan data yang digunakan untuk pembuatan timetable.

1.3 Deskripsi Dokumen (ikhtisar)

PDHUPL-SPJK mempunyai sistematika penulisan sebagai berikut

1. Pendahuluan
 - 1.1. Tujuan Pembuatan Dokumen
 - 1.2. Deskripsi Umum Sistem
 - 1.3. Deskripsi Dokumen atau Ikhtisar
 - 1.4. Definisi dan Singkatan
 - 1.5. Dokumen Referensi
2. Lingkungan pengujian perangkat lunak
 - 2.1. Perangkat Lunak Pengujian
 - 2.2. Perangkat Keras Pengujian
 - 2.3. Material Pengujian

Program Studi Magister Teknik Informatika	PDHUPL – SPJK	7/ 25
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2.4. Sumber Daya Manusia

2.5. Prosedur Umum

2.5.1. Pengenalan dan Latihan

2.5.2. Persiapan Awal

2.5.2.1. Persiapan Prosedural

2.5.2.2. Persiapan Perangkat Keras

2.5.2.3. Persiapan Perangkat Lunak

2.5.3. Pelaksanaan

2.5.4. Pelaporan Hasil

3. Identifikasi dan Rencana Pengujian

4. Deskripsi dan Hasil Uji

4.1. Identifikasi Kelas Pengujian

4.1.1. Identifikasi Butir Pengujian

1.4 Definisi dan Singkatan

Tabel 1. definisi

Keyword/Phrase	Definisi
PDHUPL	Merupakan dokumen pengujian dari perangkat lunak yang dikembangkan.
PDHUPL-SPJK-XXX	Kode yang merepresentasikan kebutuhan pada SPJK dimana XXX merupakan nomor fungsi produk.
SPJK	Program auto generate Timetable dengan menggunakan algoritma discrete particle swarm optimization.

1.5 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

Program Studi Magister Teknik Informatika	PDHUPL – SPJK	8/ 25
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Rochim Widaryanto, 2013, SKPL-SPJK, Yogyakarta, Tesis Universitas Atma Jaya Yogyakarta 2013.
2. Rochim Widaryanto, 2013, DPPL-SPJK, Yogyakarta, Tesis Universitas Atma Jaya Yogyakarta.
3. Template Laporan Pengujian Sistem Informasi Pelayanan Rencana Studi Mahasiswa (Sipersma) GL03-G08, Program Studi Teknik Informatika - Fakultas Teknologi Industri Universitas Atma Jaya Yogyakarta, <http://www.scribd.com/doc/48467978/Template-Laporan-Pengujian>.

2 Lingkungan Pengujian Perangkat Lunak

2.1 Perangkat Lunak Pengujian

Perangkat lunak pengujian berupa :

1. Windows dari Microsoft, sebagai sistem operasi.
2. Sql server 2003 dari Microsoft, sebagai DBMS penjalan aplikasi.
3. Basis data uji berupa file SPJK.mdf.

2.2 Perangkat Keras Pengujian

Perangkat keras pengujian berupa :

1. Komputer dengan spesifikasi processor intel® Core™ 2 4300 1.8 GHZ dengan memory 2gb.
2. Laptop dengan spesifikasi processor intel® dualcore 1.66 GHZ dengan memory 1gb.
3. Printer HP photosmart 7450.

2.3 Material Pengujian

Material tambahan untuk pengujian yaitu :

1. Hasil keluaran print out timetable.

2.4 Sumber Daya Manusia

Sumber daya pengujian ini yaitu :

Program Studi Magister Teknik Informatika	PDHUPL – SPJK	9/ 25
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Tester terdiri dari satu orang, tester dengan spesifikasi sarjana teknik informatika.
2. Sistem Analyst terdiri dari satu orang, sistem analyst dengan spesifikasi Magister Teknik Informatika.
3. Operator terdiri dari satu orang, operator dengan spesifikasi sarjana teknik informatika.

2.5 Prosedur Umum Pengujian

Sistem ini dapat dijalankan pada perangkat PC dan Notebook.

2.5.1. Persiapan Awal

2.5.1.1. Persiapan perangkat keras

1. Satu set komputer yang terdiri dari monitor, keyboard, mouse, cpu dengan spesifikasi processor intel® Core™ 2 4300 1.8 GHZ dengan memory 2gb.
2. Laptop dengan spesifikasi processor intel® dualcore 1.66 GHZ dengan memory 1gb.
3. Printer HP photosmart 7450.

2.5.1.2. Persiapan perangkat lunak

1. Perangkat Lunak SPJK terpasang di komputer.
2. Perangkat lunak SPJK terpasang di laptop.

2.5.2. Pelaksanaan

Pelaksanaan pengujian dilakukan dalam dua tahap, yaitu pengujian pada pengelolaan dan pengujian pada generate jadwal.

Program Studi Magister Teknik Informatika	PDHUPL – SPJK	10/ 25
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

3 Identifikasi Rencana Pengujian

Tabel 2. Identifikasi Pengujian

Kelas Uji	Butir Uji	Identifikasi		Tingkat Pengujian	Jenis Pengujian	Jadwal
		SKPL	PDHUPL			
Pengujian antarmuka login	Pengujian login	SKPL - 001	AU 01	Pengujian Unit	Black Box	06-01-2013
Pengujian antarmuka manage Dosen	Pembangkitan data Dosen	SKPL - 002	AU 02	Pengujian Unit	Black Box	06-01-2013
	Penambahan data dosen	SKPL - 002-01	AU 02_01	Pengujian Unit	Black Box	06-01-2013
	Pengubahan data dosen	SKPL - 002-02	AU 02_02	Pengujian Unit	Black Box	06-01-2013
	Penghapusan data dosen	SKPL - 002-03	AU 02_03	Pengujian Unit	Black Box	06-01-2013
Pengujian antarmuka manage Kelas Mata Kuliah	Pembangkitan data kelas Mata Kuliah	SKPL - 003	AU 03	Pengujian Unit	Black Box	06-01-2013
	Penambahan data kelas Mata Kuliah	SKPL - 003-01	AU 03_01	Pengujian Unit	Black Box	06-01-2013
	Pengubahan data kelas Mata Kuliah	SKPL - 003-02	AU 03_02	Pengujian Unit	Black Box	06-01-2013
	Penghapusan data kelas Mata Kuliah	SKPL - 003-03	AU 03_03	Pengujian Unit	Black Box	06-01-2013
Pengujian antarmuka manage Kendala Dosen	Pembangkitan data kendala Dosen	SKPL - 004	AU 04	Pengujian Unit	Black Box	06-01-2013
	Penambahan data kendala Dosen	SKPL - 004-01	AU 04_01	Pengujian Unit	Black Box	06-01-2013
	Pengubahan data kendala Dosen	SKPL - 004-02	AU 04_02	Pengujian Unit	Black Box	06-01-2013
	Penghapusan data kendala Dosen	SKPL - 004-03	AU 04_03	Pengujian Unit	Black Box	06-01-2013
Pengujian antarmuka manage Kendala Mahasiswa	Pembangkitan data kendala Mahasiswa	SKPL - 005	AU 05	Pengujian Unit	Black Box	06-01-2013
	Penambahan data kendala	SKPL - 005-01	AU 05_01	Pengujian Unit	Black Box	06-01-2013

Program Studi Magister Teknik Informatika

PDHUPL – SPJK

11/ 25

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

	Mahasiswa					
	Pengubahan data kendala Mahasiswa	SKPL - 005-02	AU 05_02	Pengujian Unit	Black Box	06-01-2013
	Penghapusan data kendala Mahasiswa	SKPL - 005-03	AU 05_03	Pengujian Unit	Black Box	06-01-2013
Pengujian antarmuka generate jadwal	Pembangkitan user interface	SKPL - 006	AU 06	Pengujian Unit	Black Box	06-01-2013
	Pembentukan timetable mata kuliah	SKPL - 006-01	AU 06_01	Pengujian Unit	Black Box	06-01-2013
	pembangkitan data timetable mata kuliah	SKPL - 006-02	AU 06_02	Pengujian Unit	Black Box	06-01-2013
	Report data timetable mata kuliah	SKPL - 006-03	AU 06_03	Pengujian Unit	Black Box	06-01-2013
Pengujian antarmuka manage Mata Kuliah	Pembangkitan data mata Kuliah	SKPL - 007	AU 07	Pengujian Unit	Black Box	06-01-2013
	Penambahan data mata Kuliah	SKPL - 007-01	AU 07_01	Pengujian Unit	Black Box	06-01-2013
	Pengubahan data mata Kuliah	SKPL - 007-02	AU 07_02	Pengujian Unit	Black Box	06-01-2013
	Penghapusan data mata Kuliah	SKPL - 007-03	AU 07_03	Pengujian Unit	Black Box	06-01-2013
Pengujian antarmuka manage RefSesi	Pembangkitan data refSesi	SKPL - 008	AU 08	Pengujian Unit	Black Box	06-01-2013
	Penambahan data refSesi	SKPL - 008-01	AU 08_01	Pengujian Unit	Black Box	06-01-2013
	Pengubahan data refSesi	SKPL - 008-02	AU 08_02	Pengujian Unit	Black Box	06-01-2013
	Penghapusan data refSesi	SKPL - 008-03	AU 08_03	Pengujian Unit	Black Box	06-01-2013
Pengujian antarmuka manage RefRuang	Pembangkitan data refRuang	SKPL - 009	AU 09	Pengujian Unit	Black Box	06-01-2013
	Penambahan data refRuang	SKPL - 009-01	AU 09_01	Pengujian Unit	Black Box	06-01-2013
	Pengubahan data refRuang	SKPL - 009-02	AU 09_02	Pengujian Unit	Black Box	06-01-2013
	Penghapusan data refRuang	SKPL - 009-03	AU 09_03	Pengujian Unit	Black Box	06-01-2013

Pengujian antarmuka manage refHari	Pembangkitan data refHari	SKPL - 010	AU 10	Pengujian Unit	Black Box	06-01-2013
	Penambahan data refHari	SKPL - 010-01	AU 10_01	Pengujian Unit	Black Box	06-01-2013
	Pengubahan data refHari	SKPL - 010-02	AU 10_02	Pengujian Unit	Black Box	06-01-2013
	Penghapusan data refHari	SKPL - 010-03	AU 10_03	Pengujian Unit	Black Box	06-01-2013
Pengujian antarmuka manage refKelas	Pembangkitan data refKelas	SKPL - 011	AU 11	Pengujian Unit	Black Box	06-01-2013
	Penambahan data refKelas	SKPL - 011-01	AU 11_01	Pengujian Unit	Black Box	06-01-2013
	Pengubahan data refKelas	SKPL - 011-02	AU 11_02	Pengujian Unit	Black Box	06-01-2013
	Penghapusan data refKelas	SKPL - 011-03	AU 11_03	Pengujian Unit	Black Box	06-01-2013
Pengujian antarmuka manage refSemester	Pembangkitan data ref Semester	SKPL - 012	AU 12	Pengujian Unit	Black Box	06-01-2013
	Penambahan data ref Semester	SKPL - 012-01	AU 12_01	Pengujian Unit	Black Box	06-01-2013
	Pengubahan data ref Semester	SKPL - 012-02	AU 12_02	Pengujian Unit	Black Box	06-01-2013
	Penghapusan data ref Semester	SKPL - 012-03	AU 12_03	Pengujian Unit	Black Box	06-01-2013
Pengujian antarmuka manage User	Pembangkitan data User	SKPL - 013	AU 13	Pengujian Unit	Black Box	06-01-2013
	Penambahan data User	SKPL - 013-01	AU 13_01	Pengujian Unit	Black Box	06-01-2013
	Pengubahan data User	SKPL - 013-02	AU 13_02	Pengujian Unit	Black Box	06-01-2013
	Penghapusan data User	SKPL - 013-03	AU 13_03	Pengujian Unit	Black Box	06-01-2013
Pengujian antarmuka manage User	Pembangkitan data refRole	SKPL - 014	AU 14	Pengujian Unit	Black Box	06-01-2013
	Penambahan data refRole	SKPL - 014-01	AU 14_01	Pengujian Unit	Black Box	06-01-2013
	Pengubahan data refRole	SKPL - 014-02	AU 14_02	Pengujian Unit	Black Box	06-01-2013
	Penghapusan data refRole	SKPL - 014-03	AU 14_03	Pengujian Unit	Black Box	06-01-2013

4 Deskripsi Hasil Uji

4.1 Identifikasi Kelas Pengujian Login

Kelas Pengujian antarmuka pengguna user adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan user sebagai penggunanya

4.1.1 Identifikas Butir Pengujian Login

Butir pengujian ini menguji login oleh user dengan pemasukan input nama user dan password.

4.2 Identifikasi Kelas Pengujian manageDosen

Kelas Pengujian antarmuka pengguna kaprodi adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan kaprodi sebagai penggunanya

4.2.1 Identifikas Butir Pengujian Add

Butir pengujian ini menguji penambahan data oleh kaprodi dengan mengklik kiri radio button add. Pengujian berikutnya akan berupa klik kiri button execute.

4.2.2 Identifikasi Butir Pengujian Edit

Butir pengujian ini menguji pengubahan data oleh kaprodi dengan mengklik kiri radio button edit. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari buttir ini berupa klik kiri button execute.

4.2.3 Identifikasi Butir Pengujian Delete

Butir pengujian ini menguji penghapusan data oleh kaprodi dengan mengklik kiri radio button delete. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari buttir ini berupa klik kiri button execute.

4.3 Identifikasi Kelas Pengujian manageKelasMataKuliah

Kelas Pengujian antarmuka pengguna kaprodi adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan kaprodi sebagai penggunanya

Program Studi Magister Teknik Informatika	PDHUPL – SPJK	14/ 25
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4.3.1 Identifikas Butir Pengujian Add

Butir pengujian ini menguji penambahan data oleh kaprodi dengan mengklik kiri radio button add. Pengujian berikutnya akan berupa klik kiri button execute.

4.3.2 Identifikasi Butir Pengujian Edit

Butir pengujian ini menguji pengubahan data oleh kaprodi dengan mengklik kiri radio button edit. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.3.3 Identifikasi Butir Pengujian Delete

Butir pengujian ini menguji penghapusan data oleh kaprodi dengan mengklik kiri radio button delete. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.4 Identifikasi Kelas Pengujian manageKendalaDosen

Kelas Pengujian antarmuka pengguna kaprodi adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan kaprodi sebagai penggunanya

4.4.1 Identifikas Butir Pengujian Add

Butir pengujian ini menguji penambahan data oleh kaprodi dengan mengklik kiri radio button add. Pengujian berikutnya akan berupa klik kiri button execute.

4.4.2 Identifikasi Butir Pengujian Edit

Butir pengujian ini menguji pengubahan data oleh kaprodi dengan mengklik kiri radio button edit. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.4.3 Identifikasi Butir Pengujian Delete

Butir pengujian ini menguji penghapusan data oleh kaprodi dengan mengklik kiri radio button delete. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

Program Studi Magister Teknik Informatika	PDHUPL – SPJK	15/ 25
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4.5 Identifikasi Kelas Pengujian manageKendalaMahasiswa

Kelas Pengujian antarmuka pengguna kaprodi adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan kaprodi sebagai penggunaanya

4.5.1 Identifikas Butir Pengujian Add

Butir pengujian ini menguji penambahan data oleh kaprodi dengan mengklik kiri radio button add. Pengujian berikutnya akan berupa klik kiri button execute.

4.5.2 Identifikasi Butir Pengujian Edit

Butir pengujian ini menguji pengubahan data oleh kaprodi dengan mengklik kiri radio button edit. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.5.3 Identifikasi Butir Pengujian Delete

Butir pengujian ini menguji penghapusan data oleh kaprodi dengan mengklik kiri radio button delete. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.6 Identifikasi Kelas Pengujian generateJadwal

Kelas Pengujian antarmuka pengguna kaprodi adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan kaprodi sebagai penggunaanya

4.6.1 Identifikas Butir Pengujian Generate Jadwal

Butir pengujian ini menguji generate jadwal oleh kaprodi dengan mengklik kiri button generate.

4.6.2 Identifikasi Butir Pengujian Show

Butir pengujian ini menguji tampil data oleh kaprodi dengan mengklik kiri button show.

4.6.3 Identifikasi Butir Pengujian Delete

Butir pengujian ini menguji report data oleh kaprodi dengan mengklik kiri button report.

4.7 Identifikasi Kelas Pengujian manageMataKuliah

Kelas Pengujian antarmuka pengguna petugas TU adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan petugas TU sebagai pengguna

4.7.1 Identifikas Butir Pengujian Add

Butir pengujian ini menguji penambahan data oleh petugas TU dengan mengklik kiri radio button add. Pengujian berikutnya akan berupa klik kiri button execute.

4.7.2 Identifikasi Butir Pengujian Edit

Butir pengujian ini menguji pengubahan data oleh petugas TU dengan mengklik kiri radio button edit. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.7.3 Identifikasi Butir Pengujian Delete

Butir pengujian ini menguji penghapusan data oleh petugas TU dengan mengklik kiri radio button delete. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.8 Identifikasi Kelas Pengujian manageRefSesi

Kelas Pengujian antarmuka pengguna petugas TU adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan petugas TU sebagai pengguna

4.8.1 Identifikas Butir Pengujian Add

Butir pengujian ini menguji penambahan data oleh petugas TU dengan mengklik kiri radio button add. Pengujian berikutnya akan berupa klik kiri button execute.

4.8.2 Identifikasi Butir Pengujian Edit

Butir pengujian ini menguji pengubahan data oleh petugas TU dengan mengklik kiri radio button edit. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.8.3 Identifikasi Butir Pengujian Delete

Butir pengujian ini menguji penghapusan data oleh petugas TU dengan mengklik kiri radio button delete. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.9 Identifikasi Kelas pengujian manageRefRuang

Kelas Pengujian antarmuka pengguna petugas TU adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan petugas TU sebagai penggunaanya

4.9.1 Identifikas Butir Pengujian Add

Butir pengujian ini menguji penambahan data oleh petugas TU dengan mengklik kiri radio button add. Pengujian berikutnya akan berupa klik kiri button execute.

4.9.2 Identifikasi Butir Pengujian Edit

Butir pengujian ini menguji pengubahan data oleh petugas TU dengan mengklik kiri radio button edit. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.9.3 Identifikasi Butir Pengujian Delete

Butir pengujian ini menguji penghapusan data oleh petugas TU dengan mengklik kiri radio button delete. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.10 Identifikasi Kelas Pengujian manageRefHari

Kelas Pengujian antarmuka pengguna petugas TU adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan petugas TU sebagai penggunaanya

4.10.1 Identifikas Butir Pengujian Add

Butir pengujian ini menguji penambahan data oleh petugas TU dengan mengklik kiri radio button add. Pengujian berikutnya akan berupa klik kiri button execute.

4.10.2 Identifikasi Butir Pengujian Edit

Butir pengujian ini menguji pengubahan data oleh petugas TU dengan mengklik kiri radio button edit. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.10.3 Identifikasi Butir Pengujian Delete

Butir pengujian ini menguji penghapusan data oleh petugas TU dengan mengklik kiri radio button delete. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.11 Identifikasi Kelas Pengujian manageRefKelas

Kelas Pengujian antarmuka pengguna petugas TU adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan petugas TU sebagai pengguna

4.11.1 Identifikas Butir Pengujian Add

Butir pengujian ini menguji penambahan data oleh petugas TU dengan mengklik kiri radio button add. Pengujian berikutnya akan berupa klik kiri button execute.

4.11.2 Identifikasi Butir Pengujian Edit

Butir pengujian ini menguji pengubahan data oleh petugas TU dengan mengklik kiri radio button edit. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.11.3 Identifikasi Butir Pengujian Delete

Butir pengujian ini menguji penghapusan data oleh petugas TU dengan mengklik kiri radio button delete. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.12 Identifikasi Kelas Pengujian manageRefSemester

Kelas Pengujian antarmuka pengguna petugas TU adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan petugas TU sebagai pengguna

4.12.1 Identifikas Butir Pengujian Add

Butir pengujian ini menguji penambahan data oleh petugas TU dengan mengklik kiri radio button add. Pengujian berikutnya akan berupa klik kiri button execute.

4.12.2 Identifikasi Butir Pengujian Edit

Butir pengujian ini menguji pengubahan data oleh petugas TU dengan mengklik kiri radio button edit. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.12.3 Identifikasi Butir Pengujian Delete

Butir pengujian ini menguji penghapusan data oleh petugas TU dengan mengklik kiri radio button delete. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.13 Identifikasi KelasPengujian manageUser

Kelas Pengujian antarmuka pengguna admin adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan admin sebagai penggunanya

4.13.1 Identifikas Butir Pengujian Add

Butir pengujian ini menguji penambahan data oleh admin dengan mengklik kiri radio button add. Pengujian berikutnya akan berupa klik kiri button execute.

4.13.2 Identifikasi Butir Pengujian Edit

Butir pengujian ini menguji pengubahan data oleh admin dengan mengklik kiri radio button edit. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.13.3 Identifikasi Butir Pengujian Delete

Butir pengujian ini menguji penghapusan data oleh admin dengan mengklik kiri radio button delete. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari butir ini berupa klik kiri button execute.

4.14 Identifikasi KelasPengujian manageRefRole

Kelas Pengujian antarmuka pengguna admin adalah Kelas pengujian yang meliputi pengujian pengujian yang melibatkan fungsi antarmuka dengan admin sebagai penggunanya

4.14.1 Identifikas Butir Pengujian Add

Butir pengujian ini menguji penambahan data oleh admin dengan mengklik kiri radio button add. Pengujian berikutnya akan berupa klik kiri button execute.

4.14.2 Identifikasi Butir Pengujian Edit

Butir pengujian ini menguji pengubahan data oleh admin dengan mengklik kiri radio button edit. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari buttir ini berupa klik kiri button execute.

4.14.3 Identifikasi Butir Pengujian Delete

Butir pengujian ini menguji penghapusan data oleh admin dengan mengklik kiri radio button delete. Pengujian berikutnya akan berupa klik kiri button cek id. Pengujian terakhir dari buttir ini berupa klik kiri button execute.

Tabel 3. Deskripsi hasil pengujian

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang didapat	Kesimpulan
AU 01	Pengujian login oleh user	mengisi nama atau password yang tidak sesuai	klik kiri pada button login	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 02	Pengujian tampil data oleh kaprodi	membuka menu manage Dosen					Handal
AU 02_01	Pengujian input data oleh kaprodi	input dengan data kosong	klik kiri pada button execute	message "data belum dimasukkan"	keluar message "data belum dimasukkan"	keluar message "data belum dimasukkan"	Handal
AU 02_02	Pengujian edit data oleh kaprodi	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 02_03	Pengujian delete data oleh kaprodi	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal

AU 03	Pengujian tampil data oleh kaprodi	membuka menu manage Kelas Mata Kuliah					Handal
AU 03_01	Pengujian input data oleh kaprodi	input dengan data kosong	klik kiri pada button execute	message "data belum dimasukkan"	keluar message "data belum dimasukkan"	keluar message "data belum dimasukkan"	Handal
AU 03_02	Pengujian edit data oleh kaprodi	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 03_03	Pengujian delete data oleh kaprodi	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 04	Pengujian tampil data oleh kaprodi	membuka menu manage Kendala Dosen					Handal
AU 04_01	Pengujian input data oleh kaprodi	input dengan data kosong	klik kiri pada button execute	message "data belum dimasukkan"	keluar message "data belum dimasukkan"	keluar message "data belum dimasukkan"	Handal
AU 04_02	Pengujian edit data oleh kaprodi	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 04_03	Pengujian delete data oleh kaprodi	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 05	Pengujian tampil data oleh kaprodi	membuka menu manage Kendala Mahasiswa					Handal
AU 05_01	Pengujian input data oleh kaprodi	input dengan data kosong	klik kiri pada button execute	message "data belum dimasukkan"	keluar message "data belum dimasukkan"	keluar message "data belum dimasukkan"	Handal
AU 05_02	Pengujian edit data oleh kaprodi	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 05_03	Pengujian delete data oleh kaprodi	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 06_01	Pengujian tampil data oleh kaprodi	membuka menu generate Jadwal	klik kiri pada button generate				Handal
AU 06_02	Pengujian tampil data oleh kaprodi	membuka menu generate	klik kiri pada				Handal

	kaprodi	Jadwal	button show				
AU 06_03	Pengujian tampil data oleh kaprodi	membuka menu generate Jadwal	klik kiri pada button report				Handal
AU 07	Pengujian tampil data oleh petugas TU	membuka menu manage Mata Kuliah					Handal
AU 07_01	Pengujian input data oleh petugas TU	input dengan data kosong	klik kiri pada button execute	message "data belum dimasukkan"	keluar message "data belum dimasukkan"	keluar message "data belum dimasukkan"	Handal
AU 07_02	Pengujian edit data oleh petugas TU	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 07_03	Pengujian delete data oleh petugas TU	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 08	Pengujian tampil data oleh petugas TU	membuka menu manage Ref Sesi					Handal
AU 08_01	Pengujian input data oleh petugas TU	input dengan data kosong	klik kiri pada button execute	message "data belum dimasukkan"	keluar message "data belum dimasukkan"	keluar message "data belum dimasukkan"	Handal
AU 08_02	Pengujian edit data oleh petugas TU	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 08_03	Pengujian delete data oleh petugas TU	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 09	Pengujian tampil data oleh petugas TU	membuka menu manage Ref Ruang					Handal
AU 09_01	Pengujian input data oleh petugas TU	input dengan data kosong	klik kiri pada button execute	message "data belum dimasukkan"	keluar message "data belum dimasukkan"	keluar message "data belum dimasukkan"	Handal
AU 09_02	Pengujian edit data oleh petugas TU	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 09_03	Pengujian delete data oleh petugas TU	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 10	Pengujian tampil	membuka menu					Handal

	data oleh petugas TU	manage Ref Hari					
AU 10_01	Pengujian input data oleh petugas TU	input dengan data kosong	klik kiri pada button execute	message "data belum dimasukkan"	keluar message "data belum dimasukkan"	keluar message "data belum dimasukkan"	Handal
AU 10_02	Pengujian edit data oleh petugas TU	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 10_03	Pengujian delete data oleh petugas TU	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 11	Pengujian tampil data oleh petugas TU	membuka menu manage Ref Kelas					Handal
AU 11_01	Pengujian input data oleh petugas TU	input dengan data kosong	klik kiri pada button execute	message "data belum dimasukkan"	keluar message "data belum dimasukkan"	keluar message "data belum dimasukkan"	Handal
AU 11_02	Pengujian edit data oleh petugas TU	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 11_03	Pengujian delete data oleh petugas TU	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 12	Pengujian tampil data oleh petugas TU	membuka menu manage Ref Semester					Handal
AU 12_01	Pengujian input data oleh petugas TU	input dengan data kosong	klik kiri pada button execute	message "data belum dimasukkan"	keluar message "data belum dimasukkan"	keluar message "data belum dimasukkan"	Handal
AU 12_02	Pengujian edit data oleh petugas TU	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 12_03	Pengujian delete data oleh petugas TU	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 13	Pengujian tampil data oleh Admin	membuka menu manage User					Handal
AU 13_01	Pengujian input data oleh admin	input dengan data kosong	klik kiri pada button execute	message "data belum dimasukkan"	keluar message "data belum dimasukkan"	keluar message "data belum dimasukkan"	Handal
AU 13_02	Pengujian edit data	mengisi id yang	klik kiri	message "data tidak	keluar message	keluar message "	Handal

	oleh admin	tidak ada	pada button Cek Id	ditemukan"	"data tidak ditemukan"	data tidak ditemukan"	
AU 13_03	Pengujian delete data oleh admin	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 14	Pengujian tampil data oleh Admin	membuka menu manage Ref Role					Handal
AU 14_01	Pengujian input data oleh admin	input dengan data kosong	klik kiri pada button execute	message "data belum dimasukkan"	keluar message "data belum dimasukkan"	keluar message "data belum dimasukkan"	Handal
AU 14_02	Pengujian edit data oleh admin	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal
AU 14_03	Pengujian delete data oleh admin	mengisi id yang tidak ada	klik kiri pada button Cek Id	message "data tidak ditemukan"	keluar message "data tidak ditemukan"	keluar message "data tidak ditemukan"	Handal