**SKPL-01**

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

SISTEM PENJADWALAN KARYA AKHIR

untuk:

STEI ITB

Dipersiapkan oleh:

Kelompok 03 :

23513018 – Rian Andrian

23513030 – Rickard Elsen

23513076 – Hastie Audytra

23513123 – M Akberita Djafar

23513190 – Lio Franklyn Kemit

Program Studi Teknik Informatika/Sistem dan Teknologi Informasi

STEI - ITB

Jl. Ganesha 10, Bandung 40132

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Program Studi Teknik Informatika / Sistem dan Teknologi Informasi**  **STEI – ITB** | Nomor Dokumen | | Halaman |
| *SKPL-00 <00:03>* | | *<1>/<jml 25* |
| Revisi | *00* | *Tgl: 28 April 2014* |

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 |
|  |  |  |  |

Table of Contents

[1. Pendahuluan 5](#_Toc388276015)

[1.1 Tujuan Penulisan Dokumen 5](#_Toc388276016)

[1.2 Lingkup Masalah 5](#_Toc388276017)

[1.3 Definisi, Istilah dan Singkatan 7](#_Toc388276018)

[1.4 Aturan Penomoran 8](#_Toc388276019)

[1.5 Referensi 8](#_Toc388276020)

[1.6 Deskripsi umum Dokumen (Ikhtisar) 8](#_Toc388276021)

[2 Deskripsi Umum Perangkat Lunak 8](#_Toc388276022)

[2.1 Deskripsi Umum Sistem 8](#_Toc388276023)

[2.2 Karakteristik Pengguna 9](#_Toc388276024)

[2.3 Batasan 10](#_Toc388276025)

[2.4 Lingkungan Operasi 10](#_Toc388276026)

[3 Deskripsi Kebutuhan 11](#_Toc388276027)

[3.1 Kebutuhan Antarmuka Eksternal 11](#_Toc388276028)

[3.1.1 Antarmuka pemakai 11](#_Toc388276029)

[3.1.2 Antarmuka Perangkat Keras 11](#_Toc388276030)

[3.1.3 Antarmuka Perangkat Lunak 11](#_Toc388276031)

[3.1.4 Antarmuka Komunikasi 11](#_Toc388276032)

[3.2 Kebutuhan Fungsional 12](#_Toc388276033)

[3.3 Model Use Case 14](#_Toc388276034)

[3.3.1 Diagram Use Case 14](#_Toc388276035)

[3.3.2 Definisi Actor 14](#_Toc388276037)

[3.3.3 Definisi Use Case 15](#_Toc388276038)

[3.3.4 Skenario Use Case 15](#_Toc388276039)

[3.3.4.6 Usecase kelola jadwal seminar 20](#_Toc388276040)

[3.3.4.7 Usecase kelola jadwal sidang 21](#_Toc388276041)

[3.3.4.8 Usecase Kelola Kesediaan Menguji 21](#_Toc388276042)

[3.3.4.9 Usecase Lock Jadwal 22](#_Toc388276043)

[3.4 Diagram Kelas 24](#_Toc388276044)

[3.5 Diagram Kelakuan 25](#_Toc388276045)

[3.6 Kebutuhan Non Fungsional 25](#_Toc388276046)

[3.7 Batasan Perancangan 26](#_Toc388276047)

[3.8 Kerunutan (traceability) 26](#_Toc388276048)

[3.8.1 Kebutuhan Fungsional vs Use Case 26](#_Toc388276049)

[3.8.2 Use Case vs Kelas Terkait 27](#_Toc388276050)

[3.9 Ringkasan Kebutuhan 27](#_Toc388276051)

[3.9.1 Kebutuhan Fungsional 27](#_Toc388276052)

[3.9.2 Kebutuhan Non Fungsional 29](#_Toc388276053)

# 1. Pendahuluan

Dokumen ini akan berisi Spesifikasi Kebutuhan Perangkat Lunak (SKPL) atau *Software Requirement Specification* (SRS) untuk Aplikasi Penjadwalan Karya Akhir. Untuk penamaan dokumen ini selanjutnya akan digunakan istilah SKPL.

## Tujuan Penulisan Dokumen

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) merupakan dokumen spesifikasi kebutuhan perangkat lunak yang akan dikembangkan. Dokumen ini digunakan oleh pengembang perangkat lunak sebagai acuan teknis pengembangan perangkat lunak pada tahap selanjutnya.

## Lingkup Masalah

Sistem penjadwalan sidang adalah subsistem dalam proses pengelolaan Karya Akhir (KA) yang menangani penjadwalan seminar prodi dan sidang akhir KA. Alur proses pelaksanaan seminar/sidang KA yang dilakukan saat ini adalah sebagai berikut :

1. Mahasiswa mencari pemimbing.
2. Mahasiswa mengajukan proposal ke pembimbing.
3. Jika pembimbing tidak setuju, maka kembali ke proses nomor 1.
4. Jika pembimbing setuju, maka pembimbing menandatangani proposal yang diajukan mahasiswa.
5. Mahasiswa membawa proposal yang sudah disetujui pembimbing ke TU.
6. TU menyerahkan proposal kepada ketua prodi.
7. Jika ketua prodi tidak setuju, maka kempali ke proses nomor 1.
8. jika ketua prodi setuju, maka ketua prodi akan membuat surat tugas bimbingan untuk pembimbing dan kartu bimbingan untuk mahasiswa.
9. Mahasiswa melakukan bimbingan dengan pembimbing.
10. Mahasiswa melakukan seminar mahasiswa dengan menghadirkan minimal 5 mahasiswa tamu.
11. Mahasiswa dan mahasiswa tamu mengisi berita acara seminar mahasiswa.
12. Pembimbing menentukan jadwal seminar prodi yang dihadiri pembimbing, mahasiswa, dan minimal 5 orang mahasiswa tamu.
13. Jika mahasiswa tidak melakukan seminar mahasiswa, maka mahasiswa harus menghadirkan minimal 10 mahasiswa tamu.
14. Mahasiswa menyerahkan jadwal seminar prodi ke TU.
15. TU menentukan ruangan dilaksanakannya seminar prodi.
16. Mahasiswa seminar pada ruang dan waktu yang telah ditentukan.
17. Jika pembimbing menyatakan lulus, maka pembimbing akan menjadwalkan sidang tugas akhir mahasiswa beserta pengujinya.
18. Jika pembimbing menyatakan tidak lulus, maka kembali ke proses 9.
19. Jika penguji tidak bisa memenuhi jadwal yang diberikan pembimbing atau pembimbing tidak menunjuk penguji saat menjadwalkan sidang akhir mahasiswa, maka mahasiswa akan mencari penguji.
20. Mahasiswa melanjutkan bimbingan sampai pembimbing menyetujui draft KA dan makalahnya.
21. Mahasiswa menyerakhan draft KA dan makalahnya ke TU.
22. TU mencetak formulir persetujuan menghadiri sidang akhir untuk Penguji.
23. Mahasiswa menyerahkan formulir ke penguji.
24. Jika penguji setuju, penguji menandatangani formulir.
25. Jika penguji tidak setuju, mahasiswa mencari penguji lain dan kembali ke proses 20.
26. Mahasiswa menyerakhan formulir yang sudah ditandatangani ke TU.
27. TU menyerahkan formulir yang sudah disetujui penguji ke ketua prodi.
28. Ketua prodi membuat surat tugas menguji untuk penguji.
29. Penguji menerima surat tugas menguji.
30. TU menentukan ruangan dilaksanakannya sidang.
31. Mahasiswa sidang pada ruang dan waktu yang telah ditentukan.
32. Jika lulus, maka mahasiswa diwajibkan mengumpulkan deliverable ke TU.
33. Jika lulus bersyarat, maka mahasiswa wajib memperbaiki draft dan ketika sudah selesai maka mahasiswa diwajibkan mengumpulkan deliverable ke TU.
34. Jika tidak lulus, maka kembali ke proses 20 atau proses 1.

Dari alur di atas, permasalahan yang ditemukan adalah :

1. Komunikasi antara pembimbing, TU, dan penguji masih melalui mahasiswa sehingga memakan waktu.
2. Penentuan penguji masih bersifat acak sehingga kemungkinan untuk ditolak penguji besar.

Untuk menyelesaikan masalah di atas, maka dibuat aplikasi penjadwalan yang mampu merekam dan menampilan ketersediaan jadwal penguji, memberikan notifikasi kepada stakeholder, dan membaca data dari subsistem lain seperti subsistem proses pengelolaan KA dan subsistem jadwal kuliah.

Aplikasi memfasilitasi komunikasi antara pembimbing dengan TU. Ketika pembimbing selesai mengisi jadwal seminar prodi pada aplikasi, TU akan mendapatkan notifikasi bahwa pembimbing tersebut telah melakukan penjadwalan. Aplikasi juga mampu membaca data ketersediaan ruangan dari subsistem penjadwalan kuliah sehingga sudah bisa ditentukan ruangan seminar akan dilaksanakan.

Aplikasi memfasilitasi komunikasi antara pembimbing dan penguji. Ketika pembimbing menentukan jadwal sidang akhir, aplikasi akan menampilkan penguji yang siap untuk menguji pada jadwal tersebut sesuai dengan isian penguji.

## Definisi, Istilah dan Singkatan

* SKPL adalah Spesifikasi Kebutuhan Perangkat Lunak, atau dalam bahasa Inggris-nya sering juga disebut sebagai *Software* *Requirements* *Spesification* (SRS), dan merupakan spesifikasi dari perangkat lunak yang akan dikembangkan.
* FR-xx adalah kode yang merepresentasikan kebutuhan (requirement) fungsional pada payroll application yang merupakan kode perangkat lunak, FR adalah kode fase dan xx adalah digit/nomor kebutuhan (requirement) fungsional.
* NFR-xx adalah kode yang merepresentasikan kebutuhan (requirement) non-fungsional pada payroll application yang merupakan kode perangkat lunak, NFR adalah kode fase dan xx adalah digit/nomor kebutuhan (requirement) non-fungsional.
* Use case diagram merupakan diagram yang menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah use case merepresentasikan sebuah interaksi antara aktor dengan sistem.

## Aturan Penomoran

* FR-xx aturan penomoran untuk functional requirement dan dimulai dari FR-01
* NFR-xx aturan penomoran untuk non functional requirement dan dimulai dari NFR-01
* UC-xx aturan penomoran untuk use case dan dimulai dari UC-01

## Referensi

1. IEEE Std 830-1998 : IEEE Recommended Practice for Software Requirements Specifications
2. GL01 ITB : Panduan Penggunaan Dan Pengisian Spesifikasi Kebutuhan Perangkat Lunak (SKPL)

## Deskripsi umum Dokumen (Ikhtisar)

Dokumen SKPL ini dibagi menjadi tiga bagian utama. Bagian utama berisi penjelasan tentang dokumen SKPL yang mencakup tujuan pembuatan dokumen ini, lingkup masalah yang diselesaikan oleh perangkat lunak yang dikembangkan, definisi, referensi dan deskripsi umum.

Bagian kedua berisi penjelasan secara umum mengenai perangkat lunak yang akan dikembangkan meliputi fungsi dari perangkat lunak, karakteristik pengguna, batasan, dan asumsi yang diambil dalam pengembangan perangkat lunak.

Bagian ketiga berisi uraian kebutuhan perangkat lunak secara lebih rinci.

# Deskripsi Umum Perangkat Lunak

## Deskripsi Umum Sistem

Aplikasi Penjadwalan Karya Akhir adalah aplikasi yang membantu mengelola proses penjadwalan seminar/sidang karya akhir mahasiswa dijurusan Informatika STEI ITB. Aplikasi dapat menangani proses penjadwalan dimulai dari pendaftaran, kalender, waktu dan tempat, sampai status seminar/sidang. Dalam hal ini, semua history dapat ditelusuri dan terekam oleh sistem. Alur kerjanya adalah sebagai berikut :

* Koordinator membuat kalender yang merepresentasikan rentang waktu seminar/sidang. Kalender ini direkam oleh aplikasi dan dijadikan acuan penjadwalan seminar/sidang dan pengisian ketersediaan menguji oleh dosen penguji.
* Aplikasi mengirim notifikasi kepada semua dosen pembimbing dan penguji yang berisi permintaan untuk mengisi jadwal kesediaan menguji. Jadwal yang diisikan oleh pembimbing dan penguji akan direkam untuk dijadikan rujukan pembimbing memilih penguji.
* Aplikasi menampilkan daftar mahasiswa yang telah mendaftar. Pembimbing mengatur jadwal sidang/seminar mahasiswa bimbingannya lalu memilih penguji yang available pada hari itu.
* Aplikasi mengirimkan notification kepada calon penguji..
* Apabila calon penguji setuju, jadwal yang telah ditentukan di lock oleh sistem. Jika penguji tidak setuju, sistem mengirimkan notifikasi kepada pembimbing dilengkapi dengan daftar calon penguji pengganti yang available pada hari yang ditentukan agar pembimbing bisa memilih penguji pengganti
* Ketika semua fix, jadwal akan di create oleh aplikasi
* Jika pada H-3 yang dijadwalkan oleh pembimbing belum ada penguji yang bersedia, maka sidang/seminar tersebut tidak jadi dilaksanakan dan menunggu periode berikutnya
* Aplikasi menyimpan data pelaksanaan sidang/seminar yang dilaksanakan

## Karakteristik Pengguna

| **Kategori Pengguna** | **Tugas** | **Hak Akses ke aplikasi** |
| --- | --- | --- |
| Koordinator | Mengelola Kalender dan Jadwal seminar/siding | Create, Retrive, Update, Delete |
| Dosen (Penguji dan Pembimbing) | Menentukan jadwal ketersediaan untuk hadir dalam seminar/siding | Create, Retrive, Update, Delete |
| Mahasiswa | Melihat jadwal seminar/sidang yang telah di *publish* | *View* |

## Batasan

* Aplikasi akan memakai data dari sistem lain diantaranya :
  + Sistem Penjadwalan Kuliah
  + Master Data Management
* Aplikasi harus mencocokan data dengan sistem diatas
* Aplikasi harus berjalan pada browser Desktop, Tablet dan Mobile

## Lingkungan Operasi

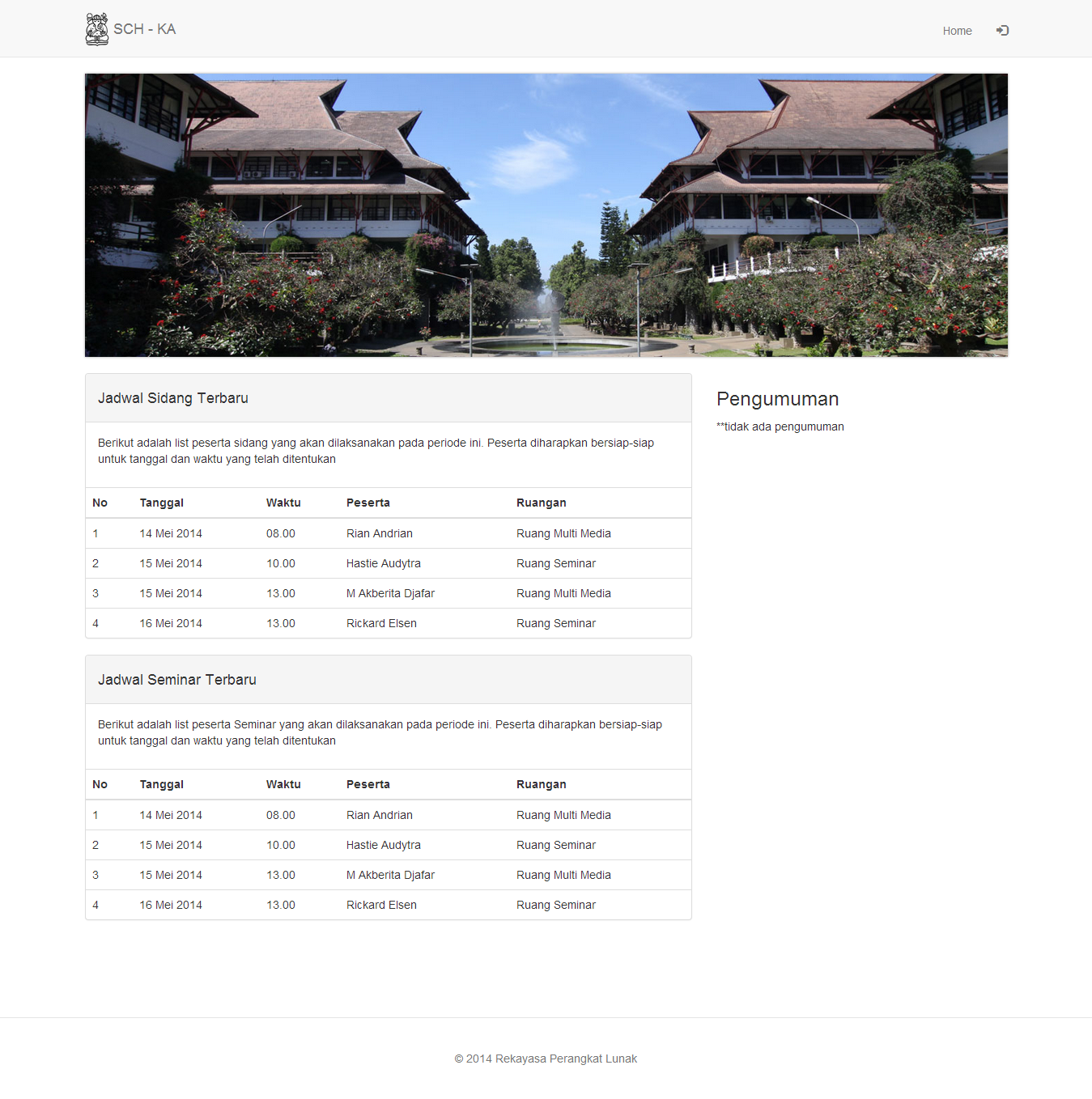
* Server : terpasang aplikasi web server Glassfish
* Client : terpasang aplikasi web browser
* Operating System : menggunakan OS yang mendukung manajemen jaringan komputer seperti MS Windows atau Linux
* DBMS : menggunakan DBMS MySql

# Deskripsi Kebutuhan

## Kebutuhan Antarmuka Eksternal

Tidak ada.

### Antarmuka pemakai



### Antarmuka Perangkat Keras

N/A.

### Antarmuka Perangkat Lunak

Perangkat lunak memerlukan web service untuk mengambil data dari sistem lain.

### Antarmuka Komunikasi

N/A.

## Kebutuhan Fungsional

| **ID** | **Kebutuhan** | **Penjelasan** |
| --- | --- | --- |
| FR-01 | Aplikasi mampu menampilkan jadwal seminar dan sidang mahasiswa yang telah disetujui oleh pembimbing dan penguji. |  |
| FR-02 | Aplikasi mampu menampilkan event yang diinformasikan oleh koordinator pada tanggal atau rentang tanggal tertentu. |  |
| FR-03 | Aplikasi menyediakan fasilitas pencarian jadwal berdasarkan salah satu dari nim, inisial, tanggal, jam, dan ruang. |  |
| FR-04 | Aplikasi menyediakan form login untuk autentikasi pengguna aplikasi. |  |
| FR-05 | Aplikasi mampu membaca data mahasiswa dan karya akhir yang dibimbing oleh pembimbing untuk ditampilkan ketika pembimbing menjadwalkan seminar atau sidang. |  |
| FR-06 | Aplikasi menyediakan form untuk koordinator menyimpan informasi rentang tanggal (kalender) pelaksanaan seminar atau sidang. |  |
| FR-07 | Aplikasi menyediakan form untuk koordinator menyimpan informasi event pada tanggal atau rentang tanggal tertentu. |  |
| FR-08 | Aplikasi mengambil data dosen, ruangan, serta karya akhir dari subsystem aplikasi lain. |  |
| FR-09 | Aplikasi menyediakan form untuk dosen mengisi informasi kesediaan jadwalnya pada tanggal dan jam tertentu sesuai dengan kalender yang sudah ditentukan oleh koordinator. |  |
| FR-10 | Aplikasi menyediakan form untuk dosen menjadwalkan seminar atau sidang bagi mahasiswa yang dibimbingnya disesuaikan dengan jadwal kesediaan dosen pembimbing. |  |
| FR-11 | Aplikasi mampu membaca data kesediaan jadwal dosen lain saat pembimbing akan memilih penguji dengan menampilkan daftar dosen lain yang tersedia sesuai dengan tanggal dan jam sidang yang ditentukan pembimbing. |  |
| FR-12 | Aplikasi menyediakan fasilitas untuk pembimbing memilih penguji pada sidang yang dijadwalkan. |  |
| FR-13 | Aplikasi memberikan notifikasi kepada dosen untuk setiap permintaan kesediaan menjadi penguji. |  |
| FR-14 | Aplikasi menyediakan fasilitas untuk dosen menerima atau menolak permintaan menguji dari dosen lain. |  |
| FR-15 | Untuk setiap jadwal seminar yang sudah dijadwalkan pembimbing, atau jadwal sidang yang sudah dijadwalkan pembimbing dan disetujui oleh minimal 1 penguji, maka aplikasi akan mencari ruang yang tersedia mulai dari jam yang telah ditentukan selama rentang waktu pelaksanaan seminar (1 jam) atau sidang (2 jam) lalu mengunci jadwal pada ruangan tersebut. |  |

## Model Use Case

### Diagram Use Case

### 

### Definisi Actor

|  |  |  |
| --- | --- | --- |
| **No** | **Actor** | **Deskripsi** |
| 1 | Koordinator | Actor dengan role ini mempunyai wewenang untuk membuat rentan waktu pelaksanaan seminar / sidang dan mengumumkan event kepada guest |
| 2 | Dosen | Aktor dengan role ini mempunyai wewenang untuk menjadwalkan mahasiswa bimbingannya untuk seminar / sidang dan menguji mahasiswa dalam siang |
| 3 | Guest | Aktor dengan role ini memiliki wewenang untuk melihat jadwal seminar / sidang yang sudah final dan melihat event yang diumumkan oleh coordinator |

### Definisi Use Case

|  |  |  |
| --- | --- | --- |
| *No* | *Use Case* | *Deskripsi* |
| *UC-01* | Cronjob | Use case ini menggambarkan proses pengambilan data dari sistem lain untuk selanjutnya digunakan oleh aplikasi penjadwalan. |
| *UC-02* | Login | Use case ini menggambarkan proses otentikasi yang dilakukan oleh aplikasi penjadwalan terhadap user beserta hak aksesnya |
| *UC-03* | Pengelolaan Kalender | Use case ini menggambarkan proses pengelolaan kalender periode pelaksanaan seminar/sistem yang akan menjadi acuan bagi Dosen untuk menentukan jadwal Seminar/sidang yang akan dilakukan |
| *UC-04* | Kelola Event | Use case ini menggambarkan proses pengelolaan event pengumuman yang berkaitan dengan penjadwalan yang bisa diakses dihalaman utama oleh siapapun |
| *UC-05* | Kelola jadwal kesediaan | Use case ini menggambarkan proses pengelolaan jadwal availability dosen untuk melaksanakan seminar/siding |
| *UC-06* | Kelola jadwal seminar | Use case ini menggambarkan proses pengelolaan penjadwalan seminar karya akhir dari mahasiswa bimbingan. |
| *UC-07* | Kelola jadwal siding | Use case ini menggambarkan proses pengelolaan penjadwalan sidang karya akhir dari mahasiswa bimbingan. |
| *UC-08* | Kelola kesediaan menguji | Use case ini menggambarkan proses penerimaan atau penolakan tawaran menguji yang ditawarkan kepada dosen yang bersangkutan yang kemudian akan di lock oleh aplikasi |
| *UC-09* | Kelola Lock Jadwal | Use case ini menggambarkan proses penguncian jadwal yang telah disetujui oleh penguji dan pembimbing, serta pengaturan ruangan secara otomatis oleh aplikasi |

### Skenario Use Case

#### Usecase Cronsjob

|  |  |  |
| --- | --- | --- |
| **Nomor** | 001 | |
| **Nama** | Cronjob | |
| **Tujuan** | Proses ini digunakan untuk mengambil data dari sistem lain yang berhubungan dengan aplikasi penjadwalan karya akhir | |
| **Deskripsi** | Use case ini menggambarkan proses pengambilan data dari sistem lain untuk selanjutnya digunakan oleh aplikasi penjadwalan. | |
| **Aktor** | Web service SubSys Dosen, SubSys Ruangan, SubSys Karya Akhir | |
| **Pre Kondisi** | Aplikasi telah berjalan pada web server | |
| **Skenario Utama** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
|  | | 1. Aplikasi menjalankan cronsjob secara otomatis |
|  | | 1. Aplikasi mengirimkan permintaan data ke web service |
| 1. Web Service memvalidasi permintaan data dari Aplikasi | |  |
| 1. Web service mengirimkan data yang telah diminta | |  |
|  | | 1. Aplikasi menyimpan data yang dikirimkan oleh web service kedalam database |
| **Skenario Alternatif : Validasi Permintaan Gagal** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
|  | | 1. Aplikasi menjalankan cronsjob secara otomatis |
|  | | 1. Aplikasi mengirimkan permintaan data ke web service |
| 1. Web Service memvalidasi permintaan data dari Aplikasi | |  |
| 1. Validasi gagal dilakukan | |  |
|  | | 1. Cronsjob akan dijalankan kembali sesuai dengan waktu yang ditentukan |
| **Post Kondisi** | Aplikasi menyimpan data yang dikirimkan oleh web service kedalam database | |

#### Usecase login

|  |  |  |
| --- | --- | --- |
| **Nomor** | 002 | |
| **Nama** | Login | |
| **Tujuan** | Proses ini digunakan untuk mengotentikasi user yang mengakses aplikasi | |
| **Deskripsi** | Use case ini menggambarkan proses otentikasi yang dilakukan oleh aplikasi penjadwalan terhadap user beserta hak aksesnya | |
| **Aktor** | Koordinator, Dosen | |
| **Pre Kondisi** | Aktor telah mengakses halaman login | |
| **Skenario Utama** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. Aktor mengisi form login | |  |
|  | | 1. Aplikasi memeriksa kesesuaian informasi akun dengan yang ada di database |
|  | | 1. Aplikasi mengotentikasi aktor berdasarkan role |
| 1. Aktor masuk kehalaman dashboard sesuai dengan role aktor yang bersangkutan | |  |
|  | |  |
| **Skenario Alternatif : Otentikasi Aktor Gagal** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. Aktor mengisi form login | |  |
|  | | 1. Aplikasi memeriksa kesesuaian informasi akun dengan yang ada di database |
|  | | 1. Aplikasi gagal mengotentikasi aktor berdasarkan role |
| 1. Aktor menerima pesan gagal | |  |
| **Post Kondisi** | Aktor masuk kehalaman dashboard sesuai dengan role aktor yang bersangkutan | |

#### Usecase Kelola Kalender

|  |  |  |
| --- | --- | --- |
| **Nomor** | 003 | |
| **Nama** | Pengelolaan Kalender | |
| **Tujuan** | Proses ini digunakan untuk mengelola kalender periode pelaksanaan seminar/sidang | |
| **Deskripsi** | Use case ini menggambarkan proses pengelolaan kalender periode pelaksanaan seminar/sistem yang akan menjadi acuan bagi Dosen untuk menentukan jadwal Seminar/sidang yang akan dilakukan | |
| **Aktor** | Koordinator | |
| **Pre Kondisi** | Aktor telah masuk kehalaman dashboard | |
| **Skenario Utama** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. Aktor masuk kehalaman pengelolaan kalender | |  |
| 1. Aktor mengisi form pembuatan kalender periode seminar/siding | |  |
|  | | 1. Aplikasi menyimpan kalender periode seminar/siding |
|  | | 1. Kalender periode seminar/sidang telah berhasil dibuat |
| **Skenario Alternatif** | | |
| **Post Kondisi** | Kalender periode seminar/sidang telah berhasil dibuat | |

#### Usecase kelola event

|  |  |  |
| --- | --- | --- |
| **Nomor** | 004 | |
| **Nama** | Kelola Event | |
| **Tujuan** | Proses ini digunakan untuk mengelola pengumuman yang berkaitan dengan penjadwalan seminar/sidang | |
| **Deskripsi** | Use case ini menggambarkan proses pengelolaan event pengumuman yang berkaitan dengan penjadwalan yang bisa diakses dihalaman utama oleh siapapun | |
| **Aktor** | Koordinator | |
| **Pre Kondisi** | Aktor telah masuk kehalaman dashboard | |
| **Skenario Utama** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. Aktor masuk kehalaman pengelolaan Event | |  |
| 1. Aktor mengisi form pembuatan event | |  |
|  | | 1. Aplikasi menyimpan event kedalam database |
|  | | 1. Aplikasi mempublish event/pengumuman pada halaman utama |
| **Skenario Alternatif :** | | |
| **Post Kondisi** | Aplikasi mempublish event/pengumuman pada halaman utama | |

#### Usecase kelola jadwal kesediaan

|  |  |  |
| --- | --- | --- |
| **Nomor** | 005 | |
| **Nama** | Kelola jadwal kesediaan | |
| **Tujuan** | Proses ini digunakan untuk merekan availability dosen | |
| **Deskripsi** | Use case ini menggambarkan proses pengelolaan jadwal availability dosen untuk melaksanakan seminar/siding | |
| **Aktor** | Dosen Pembimbing, Dosen Penguji | |
| **Pre Kondisi** | Aktor telah masuk kehalaman dashboard | |
| **Skenario Utama** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. Aktor masuk kehalaman pengelolaan kesediaan jadwal | |  |
| 1. Aktor mengisi form jadwal kesediaan melaksanakan seminar/sidang | |  |
|  | | 1. Aplikasi menyimpan jadwal kesediaan/availability dosen |
| **Skenario Alternatif :** | | |
| **Post Kondisi** | Aplikasi menyimpan jadwal kesediaan/availability dosen | |

## Usecase kelola jadwal seminar

|  |  |  |
| --- | --- | --- |
| **Nomor** | 006 | |
| **Nama** | Kelola jadwal seminar | |
| **Tujuan** | Proses ini digunakan untuk mengelola penjadwalan seminar | |
| **Deskripsi** | Use case ini menggambarkan proses pengelolaan penjadwalan seminar karya akhir dari mahasiswa bimbingan. | |
| **Aktor** | Dosen Pembimbing | |
| **Pre Kondisi** | Aktor telah masuk kehalaman dashboard | |
| **Skenario Utama** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. Aktor masuk kehalaman pengelolaan penjadwalan seminar | |  |
| 1. Aktor mengisi form penjadwalan seminar karya akhir mahasiswa bimbingan | |  |
|  | | 1. Aplikasi menyimpan jadwal seminar karya akhir mahasiswa bimbingan |
| **Skenario Alternatif :** | | |
| **Post Kondisi** | Aplikasi menyimpan jadwal seminar karya akhir mahasiswa bimbingan | |

## Usecase kelola jadwal sidang

|  |  |  |
| --- | --- | --- |
| **Nomor** | 007 | |
| **Nama** | Kelola jadwal siding | |
| **Tujuan** | Proses ini digunakan untuk mengelola penjadwalan sidang | |
| **Deskripsi** | Use case ini menggambarkan proses pengelolaan penjadwalan sidang karya akhir dari mahasiswa bimbingan. | |
| **Aktor** | Dosen Pembimbing | |
| **Pre Kondisi** | Aktor telah masuk kehalaman dashboard | |
| **Skenario Utama** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. Aktor masuk kehalaman pengelolaan penjadwalan siding | |  |
| 1. Aktor mengisi form penjadwalan sidang karya akhir mahasiswa bimbingan | |  |
|  | | 1. Aplikasi menyimpan jadwal sidang karya akhir mahasiswa bimbingan |
| **Skenario Alternatif :** | | |
| **Post Kondisi** | Aplikasi menyimpan jadwal sidang karya akhir mahasiswa bimbingan | |

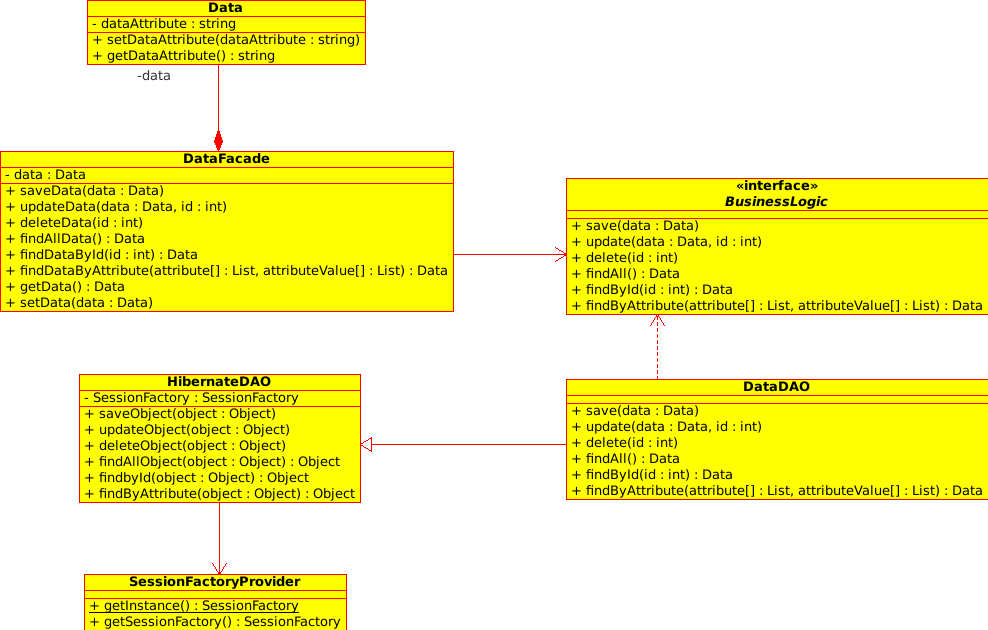
## Usecase Kelola Kesediaan Menguji

|  |  |  |
| --- | --- | --- |
| **Nomor** | 008 | |
| **Nama** | Kelola kesediaan menguji | |
| **Tujuan** | Proses ini digunakan untuk mengolah penerimaan atau penolakan tawaran menguji | |
| **Deskripsi** | Use case ini menggambarkan proses penerimaan atau penolakan tawaran menguji yang ditawarkan kepada dosen yang bersangkutan yang kemudian akan di lock oleh aplikasi | |
| **Aktor** | Dosen | |
| **Pre Kondisi** | Aktor telah masuk kehalaman dashboard | |
| **Skenario Utama** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. Aktor masuk kehalaman pengelolaan kesediaan menguji | |  |
| 1. Aktor menerima tawaran menguji | |  |
|  | | 1. Aplikasi menyimpan penerimaan tawaran kesediaan menguji |
|  | | 1. Aplikasi mengirimkan notifikasi ke dosen penguji terkait jadwal pengujian yang telah |
| **Skenario Alternatif : Tawaran menguji ditolak** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. Aktor masuk kehalaman pengelolaan kesediaan menguji | |  |
| 1. Aktor menolak tawaran menguji | |  |
|  | | 1. Aplikasi menyimpan penolakan tawaran kesediaan menguji |
|  | | 1. Aplikasi mengirimkan notifikasi ke dosen pembimbing terkait penolakan kesediaan penguji dan memberikan daftar calon penguji lain untuk dipilih ulang |
| **Post Kondisi** | Aplikasi mengirimkan notifikasi ke dosen penguji terkait jadwal pengujian yang telah | |

## Usecase Lock Jadwal

|  |  |  |
| --- | --- | --- |
| **Nomor** | 009 | |
| **Nama** | Kelola Lock Jadwal | |
| **Tujuan** | Proses ini digunakan untuk menghasilkan jadwal yang siap dipublish | |
| **Deskripsi** | Use case ini menggambarkan proses penguncian jadwal yang telah disetujui oleh penguji dan pembimbing, serta pengaturan ruangan secara otomatis oleh aplikasi | |
| **Aktor** | Aplikasi | |
| **Pre Kondisi** | Penguji telah menyetujui tawaran menguji | |
| **Skenario Utama** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
|  | | 1. Aplikasi menyimpan penerimaan tawaran kesediaan menguji |
|  | | 1. Aplikasi secara otomatis mencari ruangan yang tersedia untuk jadwal tersebut |
|  | | 1. Aplikasi mengubah status jadwal dari kandidat menjadi final |
| **Skenario Alternatif : Ruangan Penuh** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
|  | | 1. Aplikasi menyimpan penerimaan tawaran kesediaan menguji |
|  | | 1. Aplikasi secara otomatis mencari ruangan yang tersedia untuk jadwal tersebut |
|  | | 1. Aplikasi tidak menemukan ruangan yang tersedia/kosong untuk pelaksanaan seminar/siding |
|  | | 1. Aplikasi mengirimkan notifikasi kepada dosen pembimbing, dosen penguji dan koordinator bahwa seminar/sidang belum bisa dilaksanakan. |
| **Post Kondisi** | Aplikasi mengubah status jadwal dari kandidat menjadi final | |

## Diagram Kelas



|  |  |  |
| --- | --- | --- |
| No | Nama Kelas | Jenis |
| 1 | Data |  |
| 2 | DataFacade |  |
| 3 | BusinessLogic | Interface |
| 4 | HibernateDAO |  |
| 5 | DataDAO |  |
| 6 | SessionFactoryProvider |  |

|  |  |  |
| --- | --- | --- |
| Nama Kelas | Daftar Tanggung-Jawab | Daftar Atribut |
| Kelas Data | 1. setDataAttribute()  2. getDataAttribute() | 1. dataAttribute |
| Kelas DataFacade | 1. saveData ()  2. deleteData ()  3. updateData ()  4. findAllData()  5. findDataById()  6. findDataByAttribute()  7. setData ()  8. getData () | 1. data |
| Kelas BusinessLogic | 1. save ()  2. delete ()  3. update ()  4. findAll ()  5. findById()  6. findByAttribute() |  |
| Kelas HibernateDAO | 1. saveObject()  2. deleteObject ()  3. updateObject ()  4. findAllObject ()  5. findById()  6. findByAttribute() | 1. SessionFactory |
| Kelas DataDAO | 1. save ()  2. delete ()  3. update ()  4. findAll ()  5. findById()  6. findByAttribute() |  |
| Kelas SessionFactoryProvider | 1. getInstance()  2. getSessionFactory() |  |

## Diagram Kelakuan

N/A

## Kebutuhan Non Fungsional

| **ID** | **Parameter** | **Kebutuhan** |
| --- | --- | --- |
| NFR01 | Availability | Aplikasi harus tersedia dari jam 06:00 hingga 23:00 |
|  | Reliability | N/A |
|  | Ergonomy | N/A |
| NFR02 | Portability | Aplikasi harus bisa dijalankan di berbagai sistem operasi |
|  | Memory | N/A |
|  | Response time | N/A |
|  | Safety | N/A |
| NFR03 | Security | Aplikasi harus menyediakan form login untuk membedakan wewenang setiap pengguna |

## Batasan Perancangan

N/A

## Kerunutan (traceability)

### Kebutuhan Fungsional vs Use Case

|  |  |
| --- | --- |
| **ID Kebutuhan Fungsional** | **ID Use Case Terkait** |
| FR-01 | UC-09 |
| FR-02 | UC-04 |
| FR-03 | UC-06  UC-07 |
| FR-04 | UC-02 |
| FR-05 | UC-06  UC-07 |
| FR-06 | UC-03 |
| FR-07 | UC-04 |
| FR-08 | UC-01 |
| FR-09 | UC-05 |
| FR-10 | UC-06  UC-07 |
| FR-11 | UC-05  UC-06  UC-07 |
| FR-12 | UC-06  UC-07 |
| FR-13 | UC-06  UC-07 |
| FR-14 | UC-08 |
| FR-15 | UC-09 |

### Use Case vs Kelas Terkait

|  |  |
| --- | --- |
| **ID Use Case** | **Kelas Terkait** |
| UC-01 | Data |
| UC-02 | SessionFactoryProvider |
| UC-03 | DataDAO |
| UC-04 | DataFacade |
| UC-05 | DataFacade |
| UC-06 | DataFacade |
| UC-07 | DataFacade |
| UC-08 | DataFacade |
| UC-09 | DataFacade |

## Ringkasan Kebutuhan

### Kebutuhan Fungsional

| **ID** | **Kebutuhan** |
| --- | --- |
| FR-01 | Aplikasi mampu menampilkan jadwal seminar dan sidang mahasiswa yang telah disetujui oleh pembimbing dan penguji. |
| FR-02 | Aplikasi mampu menampilkan event yang diinformasikan oleh koordinator pada tanggal atau rentang tanggal tertentu. |
| FR-03 | Aplikasi menyediakan fasilitas pencarian jadwal berdasarkan salah satu dari nim, inisial, tanggal, jam, dan ruang. |
| FR-04 | Aplikasi menyediakan form login untuk autentikasi pengguna aplikasi. |
| FR-05 | Aplikasi mampu membaca data mahasiswa dan karya akhir yang dibimbing oleh pembimbing untuk ditampilkan ketika pembimbing menjadwalkan seminar atau sidang. |
| FR-06 | Aplikasi menyediakan form untuk koordinator menyimpan informasi rentang tanggal (kalender) pelaksanaan seminar atau sidang. |
| FR-07 | Aplikasi menyediakan form untuk koordinator menyimpan informasi event pada tanggal atau rentang tanggal tertentu. |
| FR-08 | Aplikasi menyediakan form untuk koordinator mendaftarkan akun dosen berdasarkan informasi email, password, dan inisial dosen. |
| FR-09 | Aplikasi mengambil data dosen, ruangan, serta karya akhir dari subsystem aplikasi lain. |
| FR-10 | Aplikasi menyediakan form untuk dosen menjadwalkan seminar atau sidang bagi mahasiswa yang dibimbingnya disesuaikan dengan jadwal kesediaan dosen pembimbing. |
| FR-11 | Aplikasi mampu membaca data kesediaan jadwal dosen lain saat pembimbing akan memilih penguji dengan menampilkan daftar dosen lain yang tersedia sesuai dengan tanggal dan jam sidang yang ditentukan pembimbing. |
| FR-12 | Aplikasi menyediakan fasilitas untuk pembimbing memilih penguji pada sidang yang dijadwalkan. |
| FR-13 | Aplikasi memberikan notifikasi kepada dosen untuk setiap permintaan kesediaan menjadi penguji. |
| FR-14 | Aplikasi menyediakan fasilitas untuk dosen menerima atau menolak permintaan menguji dari dosen lain. |
| FR-15 | Untuk setiap jadwal seminar yang sudah dijadwalkan pembimbing, atau jadwal sidang yang sudah dijadwalkan pembimbing dan disetujui oleh minimal 1 penguji, maka aplikasi akan mencari ruang yang tersedia mulai dari jam yang telah ditentukan selama rentang waktu pelaksanaan seminar (1 jam) atau sidang (2 jam) lalu mengunci jadwal pada ruangan tersebut. |

### Kebutuhan Non Fungsional

| **ID** | **Parameter** | **Kebutuhan** |
| --- | --- | --- |
| NFR01 | Availability | Aplikasi harus tersedia dari jam 06:00 hingga 23:00 |
| NFR02 | Portability | Aplikasi harus bisa dijalankan di berbagai sistem operasi |
| NFR03 | Security | Aplikasi harus menyediakan form login untuk membedakan wewenang setiap pengguna |