**SKPL**-ELMA

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

(ELMA)  
ELECTRONIC LEARNING INFORMATIKA

untuk:

Mahasiswa Informatika, Universitas Telkom

Dipersiapkan oleh:

Raden Muhamad Irvan R (1301164339)

Ariani Fitria (1301164155)

Deryandaru Bhaskara (1301160239)

Fatkha Nuansa (1301164006)

Program Studi S1 Teknik Informatika – Fakultas Informatika

Universitas Telkom

Jalan Telekomunikasi Terusan Buah Batu, Bandung

Indonesia

|  |  |  |  |
| --- | --- | --- | --- |
| Program Studi S1 Teknik Informatika  -  Fakultas Informatika | Nomor Dokumen | | Halaman |
| *SKPL-xxx* | | 8 |
| Revisi | *<nomor revisi>* | *Tgl: <isi tanggal>* |

# Daftar Perubahan

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A | Bab 1  Bab 2  Flowmap |
| B | Context Diagram + DFD (3.2.1)  ERD (3.3)  Perbaikan.....  ..... |
| C | Deskripsi proses (3.2.2)  Deskripsi Data (3.3)  Kebutuhan antarmuka eksternal (3.1)  Kebutuhan Non Fungsional (3.4)  Perbaikan.....  ...... |
| D | Atribut Kualitas Perangkat Lunak (3.5)  Batasan Perancangan (3.6)  Matriks keterunutan (4)  Melengkapi Lampiran ()  Perbaikan.....  ........ |
| E |  |
| F |  |
| G |  |

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

# Daftar Halaman Perubahan

|  |  |  |  |
| --- | --- | --- | --- |
| Halaman | Revisi | Halaman | Revisi |
| 18 | B  B  B  C  D |  |  |

# Daftar Isi

[Daftar Perubahan 1](#_Toc512606773)

[Daftar Halaman Perubahan 2](#_Toc512606774)

[Daftar Isi 3](#_Toc512606775)

[1. Pendahuluan 3](#_Toc512606776)

[1.1 Tujuan Penulisan Dokumen 3](#_Toc512606777)

[1.2 Konvensi Dokumen 3](#_Toc512606778)

[1.3 Cakupan Produk 4](#_Toc512606779)

[1.4 Refernsi 4](#_Toc512606780)

[2. Overall Description 5](#_Toc512606781)

[2.1 Perspektif Produk 5](#_Toc512606782)

[2.2 Fungsi Produk 6](#_Toc512606783)

[2.3 Kelas dan Karakteristik Pengguna 7](#_Toc512606784)

[2.4 Lingkungan Operasi 7](#_Toc512606785)

[2.5 Batasan Perancangan dan Implementasi 8](#_Toc512606786)

[2.6 Dokumentasi Pengguna 8](#_Toc512606787)

[2.7 Asumsi dan Dependensi 8](#_Toc512606788)

[3. Requirements Antarmuka Eksternal 8](#_Toc512606789)

[3.1 Antarmuka Pengguna 8](#_Toc512606790)

[3.2 Antarmuka Perangkat Keras 12](#_Toc512606791)

[3.3 Antarmuka Perangkat Lunak 12](#_Toc512606792)

[3.4 Antarmuka Komunikasi 12](#_Toc512606793)

[4. Fitur Sistem (Use Cases) 12](#_Toc512606794)

[4.1 Use Case 1 12](#_Toc512606795)

[4.2 Use Case 2 13](#_Toc512606796)

[4.3 Use Case 3 14](#_Toc512606797)

[4.4 Use Case 4 15](#_Toc512606798)

[4.5 Use Case 5 16](#_Toc512606799)

[5. Requirements Nonfungsional Lainnya 16](#_Toc512606800)

[5.1 Requirements Performa 16](#_Toc512606801)

[5.2 Requirements Keselamatan 17](#_Toc512606802)

[5.3 Requirements Keamanan 17](#_Toc512606803)

[5.4 Atribut Kualitas Perangkat Lunak 17](#_Toc512606804)

[6. Requirements Lain 18](#_Toc512606805)

# Pendahuluan

## Tujuan Penulisan Dokumen

Tujuan dari penulisan dokumen ini yaitu sebagai dokumentasi dari segala aktivitas yang dilakukan kelompok kami selama pengerjaan aplikasi ELMA pembelajaran online berbasis *web*. Adapun tujuan dari aplikasi ini adalah untuk menjembatani antara dosen dengan mahasiswa sehingga memudahkan kedua pihak dalam proses pembelajaran sehari-hari. Proyek ini dikatakan berhasil apabila aplikasi *e-learning* ini dapat digunakan oleh dosen dan mahasiswa, serta dapat dirasakan manfaatnya oleh dosen dan mahasiswa.

Pengguna dari Dokumen ini adalah pengembang perangkat lunak untuk web aplikasi e-learning dan pengguna (user) dari perangkat lunak atau personil-personil yang terlibat dalam sistem . Dokumen ini akan digunakan sebagai bahan acuan dalam pengembangan dan sebagai bahan evaluasi pasca-pengembangan .

Dokumen ini berisi penjelasan pemakaian dan penulisan dokumen Spesifikasi Kebutuhan Perangkar Lunak (SKPL) atau *Software Requirement Spesification* (SRS) dengan pendekatan berorientasi objek. Uraian yang dituangkan di dalam dokumen ini digunakan sebagai acuan dalam menulis SKPL. Dokumen ini dibuat untuk membantu membuat spesifikasi perangkat lunak yang akan dikembangkan dengan ancangan berorientasi objek

## Konvensi Dokumen

Adapun konvensi pengetikan yang digunakan dalam SKPL ini adalah sebagai berikut:

1. Nama Bab per bagian ditulis menggunakan format *Heading* 1 dengan *font* Times New Roman dan *font* size 18.
2. Nama Sub-Bab ditulis menggunakan format *Heading 2* dengan *font* Times New Roman dan *font size* 14.
3. Isi tiap Sub-Bab ditulis menggunakan *font* Times New Roma dengan *font size* 12.
4. Istilah asing yang terkandung dalam dokumen ini ditulis miring.

## Cakupan Produk

Perangkat lunak kami yang diberi nama *ELMA* ini nantinya diharapkan mampu diterima di kalangan mahasiswa dan dosen Teknik Informatika, Telkom Univesitas. Aplikasi kami diharapkan mampu menjembatani proses belajar-mengajar antara dosen dengan mahasiswa. Dosen diharapkan merasa terbantu dengan adanya aplikasi ini, dimana aplikasi ini akan meringankan pekerjaan dosen dalam mengajar mahasiswa. Mahasiswa juga diharapkan bisa lebih mandiri dengan adanya aplikasi *ELMA* ini, serta mahasiswa diharapkan mampu memahami materi sebelum materi disampaikan oleh dosen di kelas. Mulai banyak kita jumpai bisnis start-up yang sedang berkembang. *ELMA* sendiri dibuat dengan tujuan agar mampu bersaing dengan bisnis *start-up* dalam bidang pendidikan yang lain.

## Refernsi

Penulisan dokumen ini merunjuk berdasarkan:

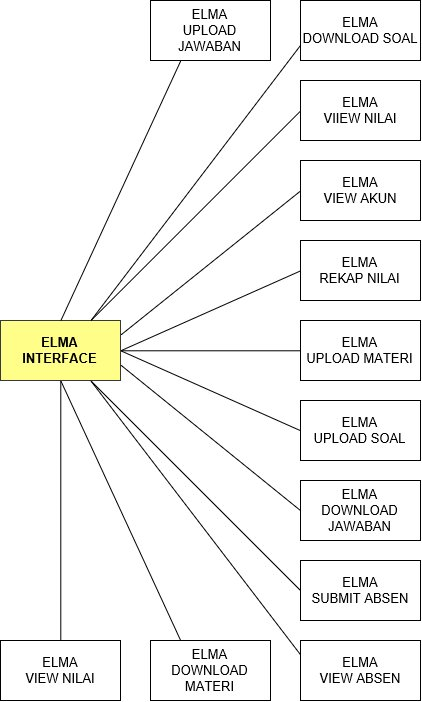
1. *Spesifikasi Kebutuhan Perangkat Lunak (SKPL)*. (2010, 05 26). Retrieved from Edugame Documentation: https://edugamemppl.wordpress.com/2010/05/26/spesifikasi-kebutuhan-perangkatlunak-skpl/

# Overall Description

## Perspektif Produk

*ELMA* adalah perangkat lunak untuk pembelajaran di tingkat perguruan tinggi yang mengedepankan kemandirian dari mahasiswa dalam proses belajar. *ELMA* sendiri adalah perangkat lunak penyempurna sistem pembelajaran online yang telah ada. Proses pembelajaran dilakukan dengan metode *self-learning.* *ELMA* akan menyimpan semua data *upload*-an materi dari dosen untuk mahasiswa berupa file dan soal-soal yang dijadikan tugas maupun *quiz* yang mana nilainya akan tersimpan ke *database* pusat dan diproses oleh *server* dan dijadikan nilai akhir hasil pembelajaran.

*ELMA* adalah perangkat lunak berbasis web yang dapat diakses pada *web browser* manapun. Perangkat lunak ini dapat diakses oleh pengguna, dengan menggunakan akun yang sudah ada. Gambar hubungan antar subsistem pada *ELMA* adalah sebagai berikut:



## Fungsi Produk

Fungsi utama dari *ELMA* ini adalah sebagai perantara terlaksananya pembelajaran yang fleksibel antara dosen dengan mahasiswa yang terkoordinasi dengan *systemr*. *ELMA* akan terintegrasi dengan sistem berupa data yang terupdate secara *real-time* dan diatur oleh Admin*.* Adapun fungsi-fungsi yang dimiliki oleh perangkat lunak ini adalah:

* Admin:

1. Login
2. Mengelola akun, CRUD(*Create, Read, Update and Delete*).
3. Merekap nilai dan absen mahasiswa.

* Dosen:

1. Login
2. Mengunggah materi(*Upload*) untuk mahasiswa
3. Mengunggah soal(*Upload*) untuk mahasiswa
4. Mengunduh jawaban (*Download*) mahasiswa untuk dinilai
5. Memasukan nilai kedalam system
6. *Submit* absen mahasiswa
7. Dapat melihat persentase Absen dan Nilai mahasiswa

* Mahasiswa:

1. Login
2. Melihat Absen dan Nilai
3. Mengunggah jawaban(*Upload*)
4. Mengunduh materi dan soal(*Download*)

## Kelas dan Karakteristik Pengguna

Pengguna web ini adalah seluruh mahasiswa Teknik Informatika di Universitas Telkom yang akan mengerjakan Tugas atau Kuis dan juga mereka yang menginginkan mengunduh (*Download*) materi yang disediakan oleh Dosen. Admin adalah *user* yang berperan sebagai pengelolaan data-data para *user* dan juga merekap data nilai dan absen mahasiswa yang sebelumnya sudah di-*submit* oleh Dosen. *User* Dosen adalah *user* yang dapat mengunggah (*Upload*) Tugas atau Kuis dan Materi agar dapat diikuti oleh mahasiswa. *User* Mahasiswa adalah user yang akan mengerjakan Kuis atau Tugas yang diberikan oleh dosen, dan mengunduh (*Download)* materi yang telah diunggah (*Upload*) oleh Dosen.

## Lingkungan Operasi

Perangkat lunak pada sisi *server* yang dibutuhkan oleh *ELMA* adalah:

SISTEM operasi: Microsoft Windows 8/10

Web server: Microsoft Personal Web Server(PWS)

*Scripting language:* Microsoft Active Pages (ASP)

DBMS: My Sql

Perangkat lunak pada sisi *client* yang dibutuhkanoleh *ELMA* adalah:

1. Sitem operasi: Microsoft Windows XP/Vista/7, Linux ubuntu/mint, macOS, Solaris.
2. Penjelajah situs (web browser) berbasis grafis atau teks: Microsoft® InternetExplorer, Netscape®4,Navigator, Opera, Modzilla, Konqueror, Nautilus, Lynx, Google Chrome

## Batasan Perancangan dan Implementasi

Batasan-batasan yang digunakan perancangan perangkat lunak ini adalah :

* Menggunakannnya harus tersambung dengan internet.
* Hanya untuk Mahasiswa Informatika, Universitas Telkom

## Dokumentasi Pengguna

* Mahasiswa

1. Login, Jika mahasiswa valid menginputkan username dan password akun mereka, maka Mahasiswa bisa masuk ke *ELMA.*
2. Mahasiswa dapat melihat Nilai dan Persentase Absen mereka.
3. Mahasiswa dapat mengunduh materi dan soal yang telah disiapkan oleh dosen yang terkait dengan matakuliah tertentu.
4. Mahasiswa dapat mengunggah jawaban dari soal yang telah diunduh oleh mahasiswa tersebut.

* Dosen

1. Login, Jika Dosen valid menginputkan username dan password akun mereka, maka Dosen bisa masuk ke *ELMA.*
2. Dosen dapat melihat Nilai dan Persentase Absen Mahasiswa
3. Dosen dapat melakukan Submit absensi dan Nilai
4. Dosen dapat mengunggah Materi dan Soal untuk diunduh oleh Mahasiswa
5. Dosen mengunduh jawaban yang telah diunggah oleh Mahasiswa untuk nantinya dinilai

* Admin

1. Login, Jika Admin valid menginputkan username dan password akun mereka, maka Admin bisa masuk ke *ELMA.*
2. CRUD untuk mengelola data akun Dosen dan Mahasiswa

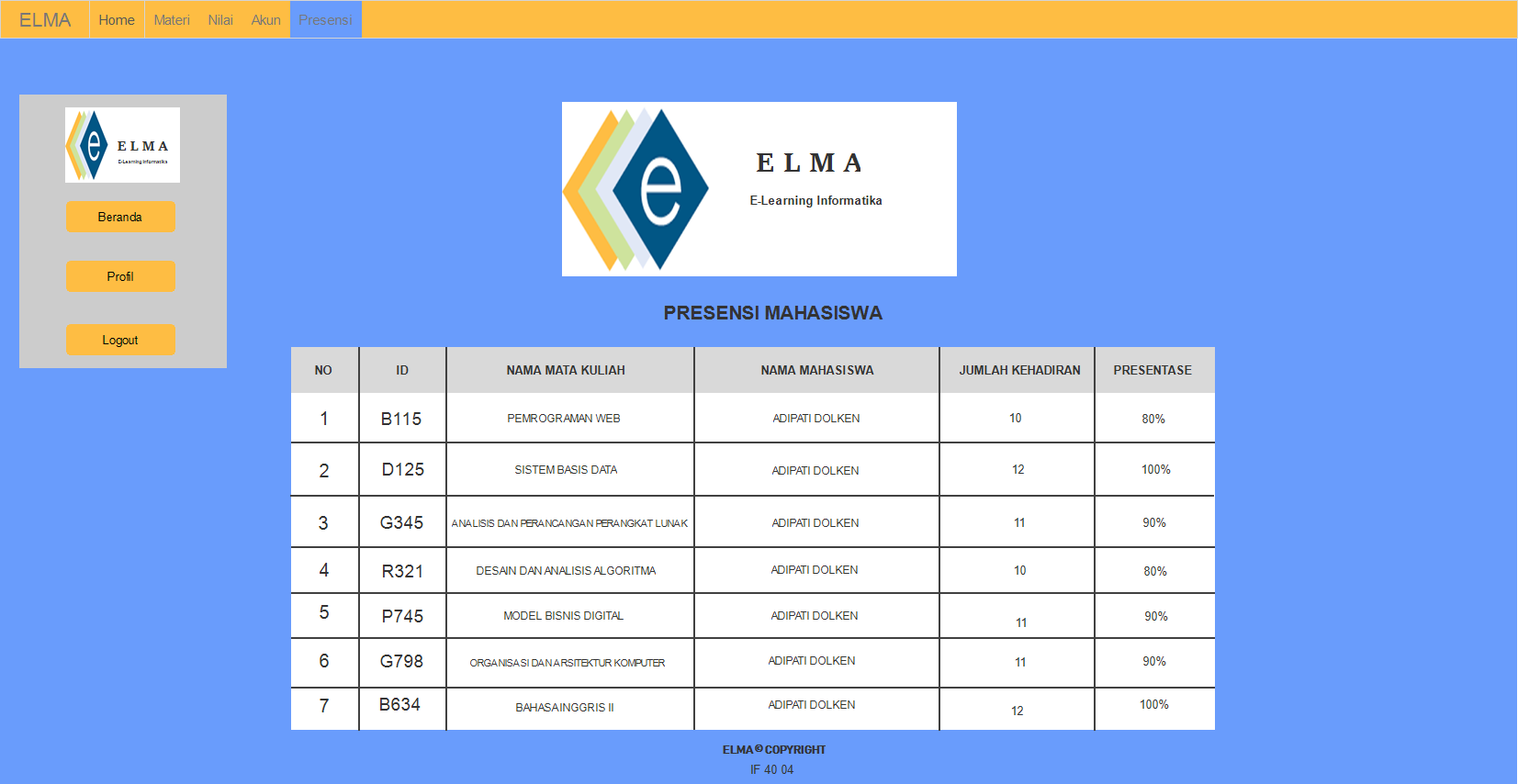
## Asumsi dan Dependensi

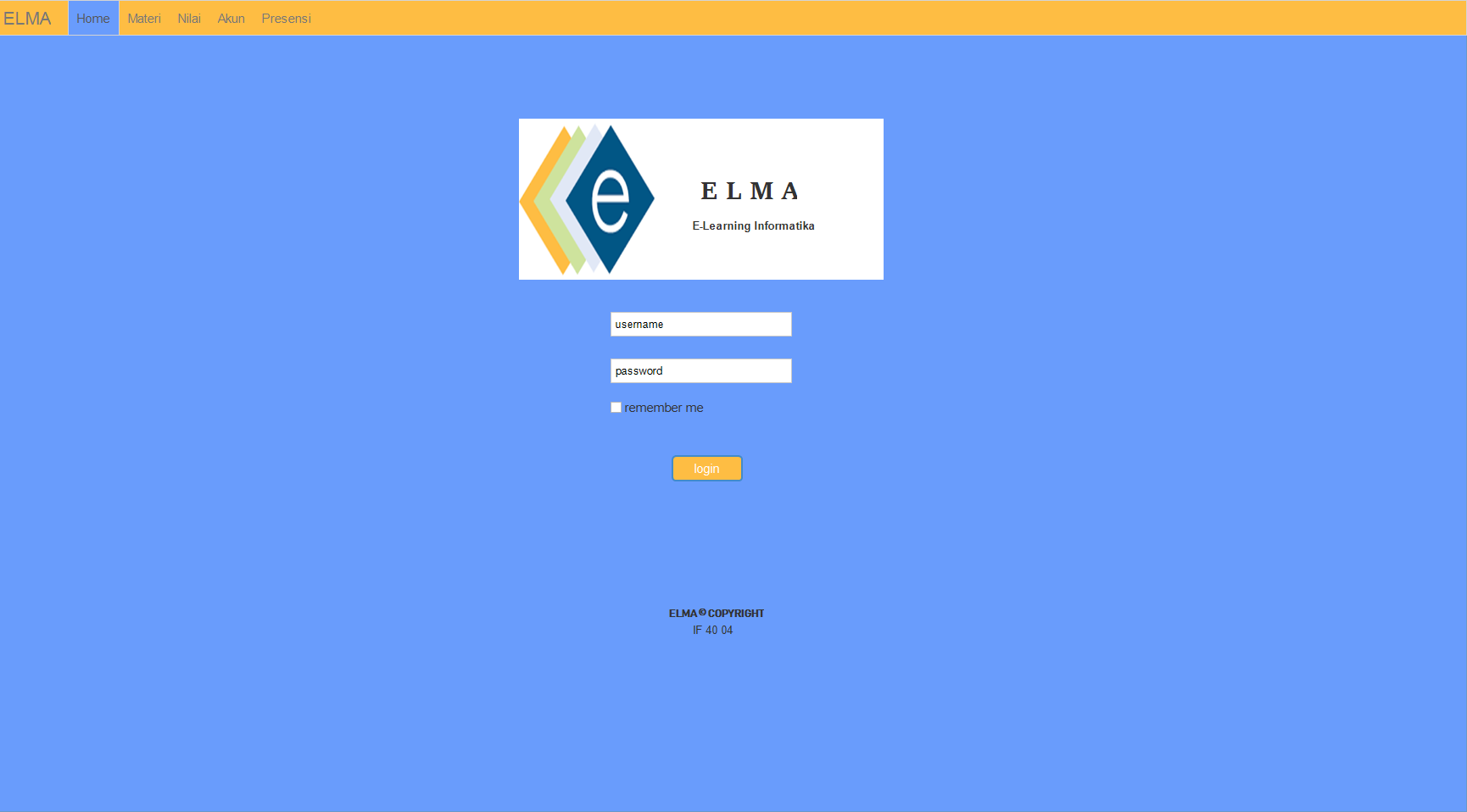
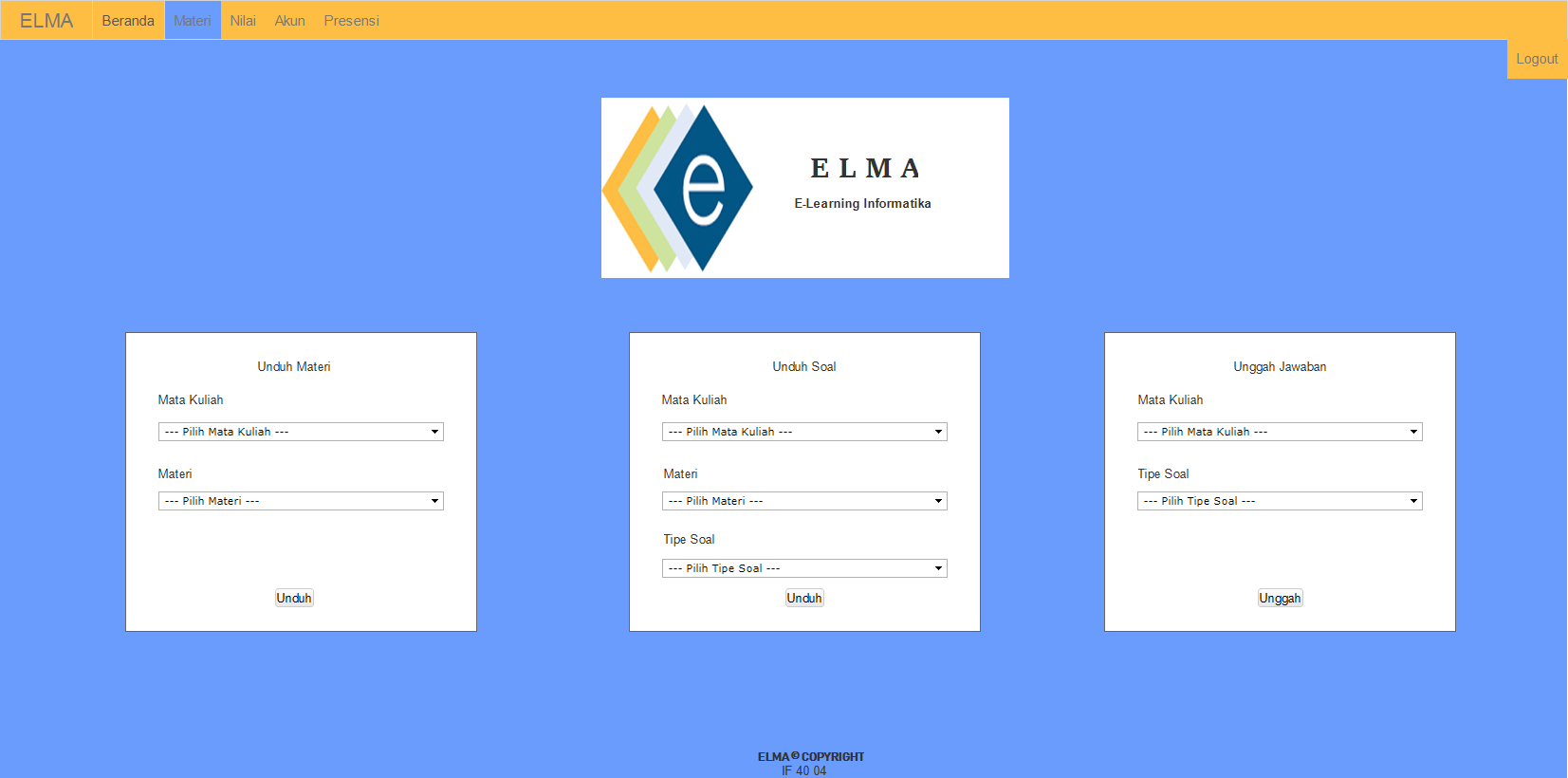
<Buatkan daftar faktor-faktor yang diasumsikan (sebagai lawan dari fakta yang telah diketahui) yang dapat mempengaruhi requirements dalam SKPL ini. Hal ini dapat mencakup pihak ketiga atau komponen komersial yang Anda berencana untuk gunakan, isu seputar development atau lingkungan operasi, atau kendala yang akan dihadapi. Proyek ini bisa terpengaruh jika asumsi ini tidak benar, tidak disebarluaskan, atau berubah. Juga identifikasi dependensi yang dimiliki proyek pada faktor-faktor eksternal, seperti komponen perangkat lunak yang Anda berniat untuk gunakan kembali dari proyek lain, kecuali jika komponen tersebut sudah didokumentasikan di tempat lain (misalnya, dalam dokumen visi dan ruang lingkup atau rencana proyek).>

# Requirements Antarmuka Eksternal

## Antarmuka Pengguna

Antarmuka pengguna akan dikembangkan dengan menggunakan modus grafik dan berbasis web. Pemakai berinteraksi dengan perangkat lunak *ELMA* melalui antarmuka penjelajah situs (web browser). *ELMA* merupakan sistem pembelajaran online yang dimana operasi dari setiap fungsionalitas akan memerlukan data tertentu. Maka itu *ELMA* akan menerima masukan dari pengguna melalui fungsionalitas yang telah tersedia. Fungsionalitas dari *ELMA* ini cukup mudah dipahami karena tidak begitu banyak fungsionalitas dalam aplikasi ini, dan memudahkan pengguna untuk mengoprasikannya. Berikut beberapa antarmuka penggunanya:

1. Antarmuka Presensi

1. Antarmuka Login
2. Antarmuka Materi
3. Antarmuka Nilai
4. Antarmuka Akun

## Antarmuka Perangkat Keras

Kebutuhan minimum perangkat keras yang dapat digunakan oleh ELMA adalah:

* PC IBM *Compatible* (PC *Standart*)
* Papan Kunci (*Keyboard*)
* Tetikus (*Mouse*)
* Modem

## Antarmuka Perangkat Lunak

Kebutuhan perangkat lunak yang dapat digunakan untuk bisa mengakses ELMA adalah Web Browser seperti Mozila Firefox, Google Chorme, Internet Exploler dll.

## Antarmuka Komunikasi

Yang dibutuhkan hanya sebuah komputer *server* dan satu atau beberapa komputer *client* yang terhubung secara *client-server* dalam lingkup jaringan Internet atau intranet berbasis protokol *Transmission Control Protocol/Internet Protocol* (TCP/IP).

# Fitur Sistem (Use Cases)

## Use Case 1

* Nama Use Case: Login
* Tujuan: Untuk dapat mengakses aplikasi ELMA, sebelum bisa mengoprasikan fungsionalitas yang ada pada aplikasi ini
* Input:Username & Password User
* Output: Dapat mengakses ELMA
* Skenario Utama: Admin, Mahasiswa, dan Dosen menginputkan username dan password untuk melakukan login dan data tersimpan di database
* Prakondisi: Actor belum melakukan login dan memasuki sistem
* Langkah-langkah:

|  |  |
| --- | --- |
| Actor | System |
| 1. Actor menginputkan username dan password. |  |
| 1. Actor melakukan login. |  |
|  | 1. System memberikan notifikasi kepada actor, apakah login berhasil atau gagal. |
| 1. Actor menerima notifikasi dari system |  |
| 1. Actor melakukan login kembali jika gagal. |  |
|  | 1. Data login tersimpan di database |

* Pascakondisi: Actor berhasil melakukan login dan telah memasuki sistem
* Skenario eksepsional 1
* Contoh :

## Use Case 2

* Nama Use Case: Kelola Akun
* Tujuan: Mengelola data akun yang tersimpan pada database
* Input: Memasukan Data yang ingin diperbaharui atau dihapus
* Output: Data akun yang baru tersimpan
* Skenario Utama: Admin akan mengubah data atau menambahkan data yang nanti nya akan disimpah kedalam database.
* Prakondisi: Actor ingin mengubah data akun
* Langkah-langkah:

|  |  |
| --- | --- |
| Actor | System |
| 1. Actor memasuki menu kelola akun |  |
|  | 1. System memberikan pilihan pengelolaan akun, ada Update, Delete, New dan Edit akun. |
| 1. Actor memilih salah satu option pengelolaan akun |  |
| 1. Actor akan Mengubah atau menghapus data akun |  |
| 1. Actor akan menyimpan data akaun |  |
|  | 1. System akan menerima data baru yang telah diperbaharui oleh Actor dan menyimpannya ke Database |
| Actor | System |
| 1. Actor menginputkan username dan password. |  |
| 1. Actor melakukan login. |  |
|  | 1. System memberikan notifikasi kepada actor, apakah login berhasil atau gagal. |
| 1. Actor menerima notifikasi dari system |  |
| 1. Actor melakukan login kembali jika gagal. |  |
|  | 1. Data login tersimpan di database |

* Pascakondisi: Actor berhasil melakukan perubahan data akun dan menyimpan datanya kembali
* Skenario eksepsional 1
* Contoh :

## Use Case 3

* Nama Use Case: Kelola Absen
* Tujuan: Mengelola data absen, yang nantinya akan ditampilkan dan disimpan
* Input: Actor memasukan data absen yang masuk
* Output: Data absen tersimpan
* Skenario Utama: mahasiswa melakukan absen di sistem yang divalidasi oleh dosen dan tersimpan di database.
* Prakondisi: mahasiswa belum melakukan absen di sistem dan dosen belum memvalidasi data absen
* Langkah-langkah:

|  |  |
| --- | --- |
| Actor | System |
| 1. Mahasiswa menginputkan absen di sistem |  |
|  | 1. System memberikan notifikasi bahwa absen berhasil atau gagal. |
| 1. Mahasiswa menerima notifikasi dari system |  |
| 1. Mahasiswa melakukan absen kembali jika gagal |  |
|  | 1. Data absen tersimpan di database |
|  | 1. System memberikan notifikasi kepada dosen bahwa mahasiswa telah melakukan absen |
| 1. Dosen menerima notifikasi dari system |  |
| 1. Dosen melakukan validasi data absen |  |
|  | 1. System memberikan notifikasi kepada dosen bahwa validasi berhasil atau gagal |
| 1. Dosen menerima notifikasi |  |
| 1. Dosen melakukan validasi kembali jika gagal |  |
|  | 1. System menyimpan data absen yang telah tervalidasi |
| 1. Mahasiswa melihat daftar absen |  |

* Pascakondisi: mahasiswa dapat melihat daftar absen dan dosen telah memvalidasi data absen.
* Skenario eksepsional 1
* Contoh

## Use Case 4

* Nama Use Case: Tahap Pembelajaran
* Tujuan: Menjelaskan tentang sistem pembelajaran dalam ELMA
* Input: Masing-masing actor akan memasukan data atau dokumen
* Output: Data atau dokumen tersebut akan tersimpan
* Skenario Utama: Dosen akan memberika Soal, tugas dan materi kepada mahasiswa, melalui system E-Learning ini, yang nanti mahasiswa akan menjawab dan mengupload jawaban kedalam system, lalu jawaban tersebut akan dinilai oleh dosen.
* Prakondisi: Actor belum melakukan proses pembelajaran pada sistem
* Langkah-langkah:

|  |  |
| --- | --- |
| Actor | System |
| 1. Dosen mengupload Materi dan soal |  |
|  | 1. System memproses upload dari Dosen |
|  | 1. Sytem menyimpan Materi dan Soal kedalam data base |
| 1. Mahasiswa mengunduh Materi ataupun Soal pada system |  |
| 1. Mahasiwa mengerjakan/menjawab Soal |  |
| 1. Mahasiswa mengupload jawaban |  |
|  | 1. System memproses upload dari mahasiswa |
|  | 1. System menyimpan Jawaban kedalam database |
| 1. Dosen mengunduh jawaban yang telah diupload oleh mahasiswa |  |
| 1. Dosen menilai jawaban dari mahasiswa |  |

* Pascakondisi: Actor sudah melakukan berbagai rangkaian proses pembelajaran pada sistem.
* Skenario eksepsional 1
* Contoh

## Use Case 5

* Nama Use Case: Kelola Absen
* Tujuan: Untuk mengelola data absen
* Input: Mahasiswa akan melakukan tapping dan dosen akan melakukan submit
* Output: Data absen disimpan didalam database dan sistem akan menampilkan persentase absen mahasiswa tersebut.
* Skenario Utama: Dosen akan melakukan peng-input-an absen mahasiswa yang nantinya akan disimpan didalam database
* Prakondisi: mahasiswa belum melakukan absensi dan dosen belum melakukan submit absensi
* Langkah-langkah:

|  |  |
| --- | --- |
| Actor | System |
| 1. Mahasiswa melakukan *tapping* |  |
|  | 1. System menerima detail kapan mahasiswa tersebut melakukan absen |
| 1. Dosen masuk ke menu Absen |  |
| 1. Dosen melakukan submit |  |
|  | 1. System akan memasukannya kedalam database dan menampilkan persentase absen mahasiswa |

* Pascakondisi: mahasiswa sudah melakukan absen dan dosen sudah melakukan submit absen
* Skenario eksepsional 1
* Contoh

# Requirements Nonfungsional Lainnya

## Requirements Performa

|  |  |
| --- | --- |
| **Kriteria Kebutuhan** | **Tuntutan Kebutuhan** |
| Ketersedian Pelayanan | Pelayanan ELMA harus tersedia selama 1x24 jam |
| Jumlah akses simultan yang dapat ditangani | Minimal 30 *user* yang dapat mengakses secara bersamaan |
| Kecepatan Pelayanan | Proses pelayanan maksimal 10 detik untuk setiap pelayanan |
| Kebutuhan pengaksesan *ELMA* | Dapat diakses disetiap web browser |
| Bahan pembangun program | JAVA |
| Durasi Perawatan | Proses backup data dan maintenance maksimal 1 jam |

## Requirements Keselamatan

<Tentukan persyaratan yang berkaitan dengan kehilangan yang mungkin terjadi, kerusakan, atau kerugian yang mungkin berasal dari penggunaan produk. Definisikan perlindungan atau tindakan yang harus diambil, serta tindakan yang harus dicegah. Dengan mengacu pada setiap kebijakan atau peraturan negara tentang keamanan yang mempengaruhi desain produk atau penggunaan eksternal. Definisikan sertifikasi keselamatan yang harus dipenuhi.>

## Requirements Keamanan

|  |  |
| --- | --- |
| **Kriteria Kebutuhan** | **Tuntutan Kebutuhan** |
| Integritas Data | Program dilengkapi dengan sertifikasi keamanan |
| Keselamatan Data | Dilakukan backup data seminggu sekali secara otomatis bukan di hari jam kerja |
| Pengamanan Jaringan | Database dilindungi oleh dua *firewall* |
| Pengamanan Fisik | Server disimpan pada suatu ruangan khusus, yang dilengkapi dengan CCTV dan Smartlock yang hanya bisa diakses oleh Administrator dan Pemilik |
| Tenaga Cadangan | Disediakannya beberapa generator untuk pencegahan hilang sumber daya listrik. |

## Atribut Kualitas Perangkat Lunak

|  |  |
| --- | --- |
| **Kriteria Kualitas Kehandalan** | **Tuntutan Kualitas** |
| Kehandalan(*Reliability*) | Sistem dapat diakses 1x24 jam |
| Ketersedian(*Availability*) | Data materi dan quiz ter-*update* setiap dosen *update* materi dan quiz. Proses nilai quiz dilakukan secara manual oleh dosen, gimana dosen akan menginputkan nilainya kedalam sistem. Perhitungan index nilai dilakukan oleh sistem. Sistem dapat diakses oleh 5000 user secara bersamaan. |
| Keamanan(*System Security*) | Dari sisi power, disediakan UPS untuk menjaga server tetap hidup. Dari sisi keamanan jaringan, digunakan *double filtering* yang dilakukan oleh dua perangkat *firewall*. Dari sisi fisikal *server* diletakan di ruangan yang khusus dan hanya beberapa orang saja yang boleh masuk. |
| Perawatan (*Maintainability*) | Perawatan tidak perlu dilakukan oleh user, semua perawatan system dilakukan oleh Administrator. |
| Kepemindahan (*Portability*) | Sistem menggunakan teknik mirroring atau RAID 1, sehingga, apabila *system down* akan ada *backup* dengan kondisi serupa seperti yang sebelumnya. |

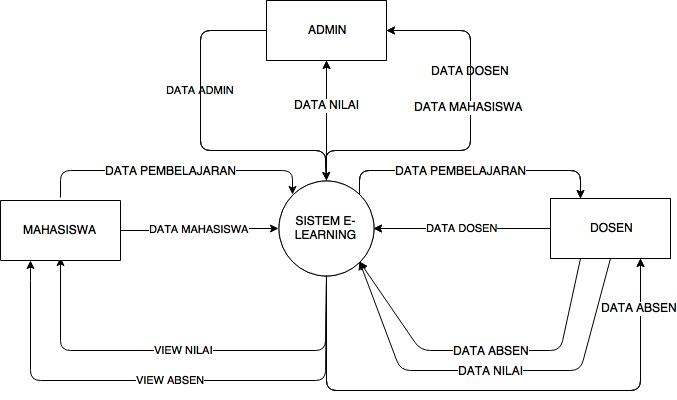
# Requirements Lain

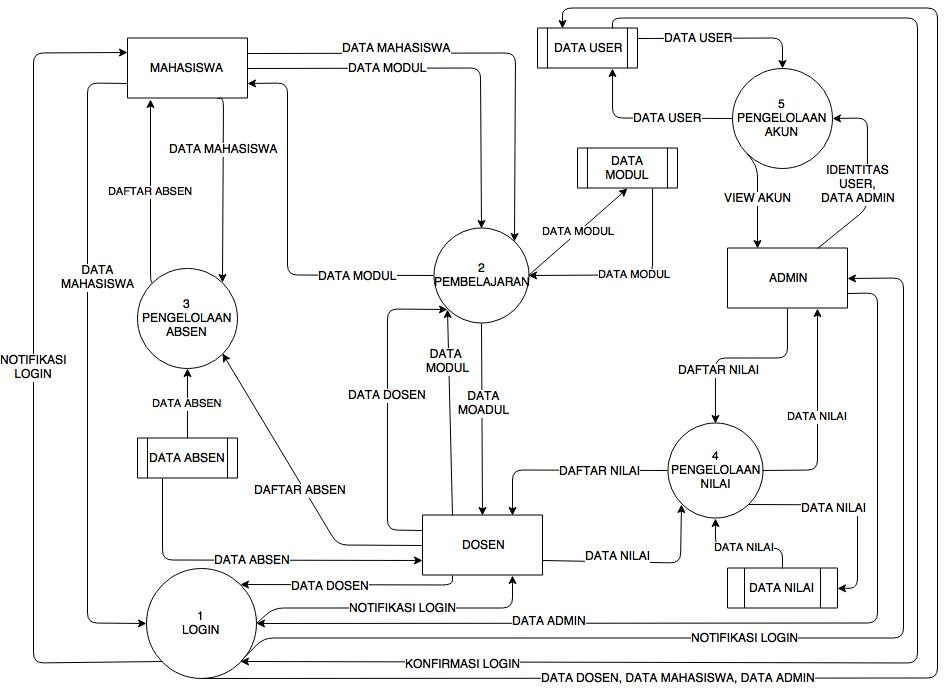
|  |  |
| --- | --- |
| **Kriteria Kebutuhan** | **Tuntutan Kebutuhan** |
| Bahasa | Bahasa Indonesia |
| Database | MySQL |
| Server | Apache2 |
| Sistem Operasi Server | Linux (Ubuntu atau CentOS 6.x) atau Windows 8/10 |
| Lingkup Jaringan | WAN |

Lampiran A: Daftar Kata-Kata Sukar

|  |  |
| --- | --- |
| Kata Kunci atau Frase | Definisi atau akronim |
| SRS | *Software Requirement Spesification*  *adalah dokumen yang menjelaskan tentang berbagai kebutuhan yang harus dipenuhi oleh suatu software.* |
| SKPL | Spesifikasi Kebutuhan Perangkat Lunak adalah spesifikasi dari perangkat lunak yang akan dikembangkan. |
| DFD | *iData Flow Diagram* adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, terstruktur dan jelas. |
| ERD | *Entity Relationship Diagram* adalah suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objekobjek dasar data yang mempunyai hubungan antar relasi. |
| DPPL | Deskripsi Perancangan Perangkat Lunak adalah dokumen yang berisi tentang deskripsi perangkat lunak yang akan dikembangkan. |
| DBMS | *Database Management System* sistem pengorganisasian dan sistem pengolahan database pada komputer. |
| PHP | *Hypertext Preprocessor* adalah Bahasa transkrip yang dapat ditanamkan atau disisipkan ke dalam HTML. |
| *User* | Setiap orang yang memakai atau menjalankan aplikasi |
| *Administrator* | Setiap orang yang dapat mengubah system aplikasi dan memutuskan apa saja yang bisa dilakukan oleh user. |
| *Server* | Sesuatu yang dapat menjalankan dan mengatur system administrasi jaringan melalui perangkat lunak administrative untuk mengontrol akses terhdapap jaringan dan sumber daya di dalamnya. |
| *Framework* | Sebuah software yang berguna untuk memudahkan programmer membuat aplikasi atau web yang isinya adalah berbagai fungsi plugin dan konsep sehingga membentuk suatu system tertentu. |

Lampiran B: Analysis Models

* **DFD LEVEL 0**
* **DFD LEVEL 1**



* **DFD LEVEL 2**

