**SKPL**-xxxx

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

Aplikasi Penjadwalalan (Neatly)

untuk:

Umum

Dipersiapkan oleh:

M. Rafi Galih S (1301170754)

Aditya Rahman (1301172728)

Zulfandi Ramdani (1301172730)

Ridho Bagus E (1301172738)

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* | | xx |
| Revisi | *<nomor revisi>* | *Tgl: <isi tanggal>* |

# Daftar Perubahan

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A |  |
| B |  |
| C |  |
| D |  |
| 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 |
|  |  |  |  |

# Daftar Isi

[Daftar Perubahan 1](#_Toc123127)

[Daftar Halaman Perubahan 2](#_Toc123128)

[Daftar Isi 3](#_Toc123129)

[1. Pendahuluan](#_Toc123130) 4

[1.1 Tujuan Penulisan Dokumen](#_Toc123131) 4

[1.2 Ruang Lingkup / Cakupan Dokumen](#_Toc123132) 4

[1.3 Definisi, Singkatan, dan Akronim 4](#_Toc123133)

[1.4 Referensi 4](#_Toc123134)

[2. Deskripsi Rinci Perangkat Lunak 4](#_Toc123135)

[2.1 Statement of Objective Perangkat Lunak](#_Toc123136) 5

[2.2 Perspektif dan Fungsi Perangkat Lunak](#_Toc123137) 5

[2.3 Profil dan Karakteristik Pengguna 5](#_Toc123138)

[2.4 Lingkungan Operasi 5](#_Toc123139)

[2.5 Batasan Perangkat Lunak / Sistem](#_Toc123140) 6

[2.6 Asumsi dan Dependensi](#_Toc123141) 6

[3. Deskrpsi Rinci Perangkat Lunak](#_Toc123142) 7

[3.1 Deskripsi Kebutuhan](#_Toc123143) 7

[3.1.1 Kebutuhan Fungsional](#_Toc123144) 7

[3.1.2 Kebutuhan Non-Fungsional](#_Toc123145) 8

[3.2 Pemodelan Analisis](#_Toc123146) 9

[3.2.1 Usecase Diagram](#_Toc123147) 9

[3.2.2 Class Diagram:](#_Toc123148) 14

[4. Requirements Antarmuka Eksternal](#_Toc123149) 15

[4.1 Antarmuka Pengguna](#_Toc123150) 15

[4.2 Antarmuka Perangkat Keras](#_Toc123151) 15

[4.3 Antarmuka Perangkat Lunak](#_Toc123152) 15

[4.4 Antarmuka Komunikasi](#_Toc123153) 15

[5. Requirements Lain](#_Toc123154) 16

# Pendahuluan

## Tujuan Penulisan Dokumen

Tujuan dari penulisan dokumen spesifikasi kebutuhan perangkat lunak adalah menjelaskan spesifikasi kebutuhan perangkat lunak dari aplikasi Penjadwalan berbasis timeboxing (Neatly) yang akan dikembangkan. Penulisan dokumen ini meliputi kebutuhan spesifik antarmuka, kebutuhan fungsional, kebutuhan struktural serta kebutuhan basis data.

## Ruang Lingkup / Cakupan Dokumen

Tujuan dikembangkannya aplikasi Neatly berbasis timeboxing agar mempermudah pengguna mengatur keseharian dengan penjadwalan yang teratur.

## Definisi, Singkatan, dan Akronim

Neatly merupakan aplikasi penjadwalan waktu berbasis *timeboxing*. Metode *timeboxing* merupakan teknik mengerjakan sesuatu dengan dibatasi waktu (mirip seperti waktu per ronde di *boxing*). Dengan metode *timeboxing*, pengguna dapat mengerjakan banyak aktifitas dalam waktu singkat. Aplikasi neatly bertujuan untuk membuat penggunanya menjadi pribadi yang lebih produktif.

Sebagai sumber pendapatan aplikasi neatly akan diberikan iklan. Aplikasi neatly menyasar kalangan umum. Dengan menyasar kalangan umum diharapkan akan banyak yang tertarik, sehingga *developer* memperoleh profit yang maksimal.

## Referensi

Fitur yang kami jadikan referensi adalah fitur manajemen waktu dari aplikasi DoNow dan menggabungkannya dengan metode *timeboxing*.

# Deskripsi Rinci Perangkat Lunak

## Statement of Objective Perangkat Lunak

## Netly merupakan jenis aplikasi baru namun hasil pembaharuan dari metode penjadwalan yang sudah ada. Untuk saat ini penerapan time boxing di penjadwalan hanya ada 1 aplikasi ya itu “do now”.

## Perspektif dan Fungsi Perangkat Lunak

Neatly merupakan sistem aplikasi yang sangat difokuskan kepada penjadwalan berbasis *timeboxing*, sistem aplikasi ini memberikan peringatan (*alert*) sebelum jadwal yang ditentukan. Neatly adalah aplikasi manajemen waktu berbasis *timeboxing*.

Aplikasi ini sangat cocok untuk pengguna yang memiliki jadwal yang padat, karna kita dapat mengatur jadwal secara teratur dengan peringatan jadwal yang seperti alarm, terlebih lagi menggunakan metode *timeboxing*.

Selain mengatur jadwal kita dapat menggunakan fitur seperti penggunaan *list* dimana di dalamnya pengguna dapat menginputkan aktifitas yang akan dikerjakan dan laporan pencapaian yaitu hasil yang di capai dari penjadwalan dalam aplikasi Neatly.

## Profil dan Karakteristik Pengguna

Pengguna aplikasi ini adalah masyarakat umum. Sedangkan admin adalah seseorang yang mengelola aplikasi.

A. Kategori pengguna: User

Tugas : Menginput kegiatan

Hak Akses:

1). Meng-*input* kegiatan.

2). Men-*delete* kegiatan.

3). Meng-*update* kegiatan.

4). *View* pencapaian.

5). Meng*-input* list.

6). Men*-delete* list.

7). Meng-*update* list.

B. Kategori pengguna: Admin

Tugas: Mengelola aplikasi

Hak akses:

1). Memperbaharui dan menambahkan fitur pada aplikasi.

2). Menyelesaikan *bug* pada aplikasi.

## Lingkungan Operasi

Software pada sisi server yang dibutuhkan:

• Sistem Operasi: Microsoft Windows 7/Vista/8/10, Linux

• DBMS: MySQL

• Software: Android Studio

Software pada sisi pengguna yang dibutuhkan:

• Sistem operasi: Android

• Internet Connection

• Ads

## Batasan Perangkat Lunak / Sistem

Batasan-bantasan program ini adalah:

1. Aplikasi ini hanya tersedia untuk android.

2. Sistem pencapayan yang terdapat dalam aplikasi tergantung kejujuran pengguna.

## Asumsi dan Dependensi

Asumsi

1. Penguna dapat mencatat kegiatan.

2. Pengguna memperoleh laporan pencapaian sesuai hasil kegiatan.

3. pengguna dapat mencatat list.

Depedensi

1. Data kegiatan ada ketika pengguna menginputkan kegiatan.

2. Data list ada ketika pengguna menginputkan list

# Deskripsi Rinci Perangkat Lunak

## Deskripsi Kebutuhan

### Kebutuhan Fungsional

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Kode Kebutuhan | Fungsi | Deskripsi |
| 1 | FR-AA | Login user(ofline) | Fungsi ini digunakan  oleh user untuk masuk dalam aplikasi |
| 2 | FR-BA | Input Jadwal | Fungsi ini digunakan oleh user untuk menginputkan jadwal ke sistem |
| 3 | FR-BB | Hapus Jadwal | Fungsi ini digunakan oleh user untuk menghapus jadwal ke sistem |
| 4 | FR-BC | Edit Jadwal | Fungsi ini digunakan untuk mengedit jadwal ke sistem |
| 5 | FR-CA | Input List | Fungsi ini digunakan untuk menginputkan list ke system |
| 6 | FR-CB | Hapus List | Fungsi ini digunakan untuk menghapus list ke system |
| 7 | FR-CC | Edit List | Fungsi ini digunakan untuk mengedit list ke sistem |
| 8 | FR-DA | View Pencapaian | Fungsi ini digunakan user untuk menampilkan hasil pencapaian pada |
| 9 | FR-BD | View Jadwal | Fungsi ini digunakan user untuk menampilkan menu jadwal |
| 10 | FR-CD | View list | Fungsi ini digunakan user untuk menampilkan menu list |
| 11 | FR-EA | View login | Fungsi ini digunakan user untuk menampilkan menu login |
| 13 | FR-FA | Maintenance aplikasi | Fungsi ini digunakan admin untuk menyelesaikan bug pada aplikasi. |

### Kebutuhan Non-Fungsional

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Quality | Kode Kebutuhan | Deskripsi |
| 1. | keamanan | NFR1 | Jika psword berturut turut salah dalam 5 kali, maka akan diberi peringatan. |
| 2. | Keamanan | NFR 2 | Data yang tersimpan tidak dapat dirubah oleh developr.(Developer hanya dapat menambah fitur juga menghapus bug pada aplikasi) |
| 3 | Keamanan | NFR 3 | Pada bagian view pencapaian, user tidak dapat menginputkan nilai secara langsung. Nila yang terdapat pada view pencapaian merupakan nilai yang disi oleh sistem secara otomatis dari data hasil melakukan suatu jadwal. |

## Pemodelan Analisis

### C:\Users\ASUS\Desktop\Use Case.jpgUsecase Diagram

#### Usecase Scenario #1

|  |  |
| --- | --- |
| Use Case | Login |
| Input | Password |
| Output | Login berhasil |
| Actor | User |
| Precondition | User ingin login ke dalam aplikasi |
| Post Condition | User telah berhasil login ke aplikasi |
| Description | Untuk mengunakan aplikasi |
| Typical Course of Event | |  |  | | --- | --- | | user | system | | 1. Membuka aplikasi Netly |  | |  | 1. Menampilkan halaman login aplikasi | | 1. memasukan password |  | | 1. menekan tombol login |  | |  | 1. Menampilkan halaman utama aplikasi | |

#### Usecase Scenario #2

|  |  |
| --- | --- |
| Use Case | Input Jadwal |
| Input | Data Jadwal |
| Output | Data Jadwal |
| Actor | User |
| Precondition | User ingin menginputkan jadwal dan melakukan login |
| Post Condition | User telah melakukan input data jdwal |
| Description | Untuk menginputkan jadwal oleh user |
| Typical Course of Event | |  |  | | --- | --- | | user | system | | 1. Membuka menu input jadwal |  | |  | 1. Menampilkan menu input jadwal | | 1. memasukan data jadwal |  | | 1. menekan tombol simpan |  | |  | 1. menyimpan data user | |

#### Usecase Scenario #3

|  |  |
| --- | --- |
| Use Case | Hapus jadwal |
| Input | Data Jadwal |
| Output | Data Jadwal |
| Actor | User |
| Precondition | User ingin menghapus jadwal dan melakukan login |
| Post Condition | User telah melakukan hapus data jdwal |
| Description | Untuk menghapus jadwal oleh user |
| Typical Course of Event | |  |  | | --- | --- | | user | system | | 1. Membuka menu hapus jadwal |  | |  | 1. Menampilkan menu hapus jadwal | | 1. menghapus data jadwal |  | |  |  | |  | 1. menghapus data user | |

#### Usecase Scenario #4

|  |  |
| --- | --- |
| Nama Use Case | Edit Jadwal |
| Actor | User |
| Input | Data Jadwal |
| Output | Data Jadwal |
| Pre-Condition | User ingin mengupdate jadwal |
| Post-Conditon | User telah mengupdate jadwal dan telah disimpan oleh program |
| Deskripsi | User mengupdate jadwal untuk merubah data sebelumnya |
| Typical Course of Event | |  |  | | --- | --- | | Actor | System | | 1. Mengklik ikon Update Jadwal |  | |  | 2. Menampilkan menu Update Jadwal | | 3. Mengupdate data Jadwal |  | | 4. Menekan tombol simpan |  | |  | 5. Data disimpan | |

#### Usecase Scenario #5

|  |  |
| --- | --- |
| Use Case | Input list |
| Input | Data list |
| Output | Data list |
| Actor | User |
| Precondition | User ingin menghapus List |
| Post Condition | User telah melakukan input data list |
| Description | Untuk menginputkan list oleh user |
| Typical Course of Event | |  |  | | --- | --- | | user | system | | 1. Membuka menu input list |  | |  | 1. Menampilkan menu input list | | 1. memasukan data list |  | |  |  | |  | 1. menyimpan data user | |

#### Usecase Scenario #6

|  |  |
| --- | --- |
| Use Case | Hapus List |
| Input | Data List |
| Output | Data List |
| Actor | User |
| Precondition | User ingin menghapus data list yang ada |
| Post Condition | User telah menghapus list yang ada |
| Description | Untuk menghapus list oleh user |
| Typical Course of Event | |  |  | | --- | --- | | user | system | | 1. Membuka menu hapus list |  | |  | 1. Menampilkan menu hapus list | | 1. Menghapus list yang diinginkan | 1. List berhasil dihapus | | 1. menekan tombol simpan |  | |  | 1. menyimpan data user | |

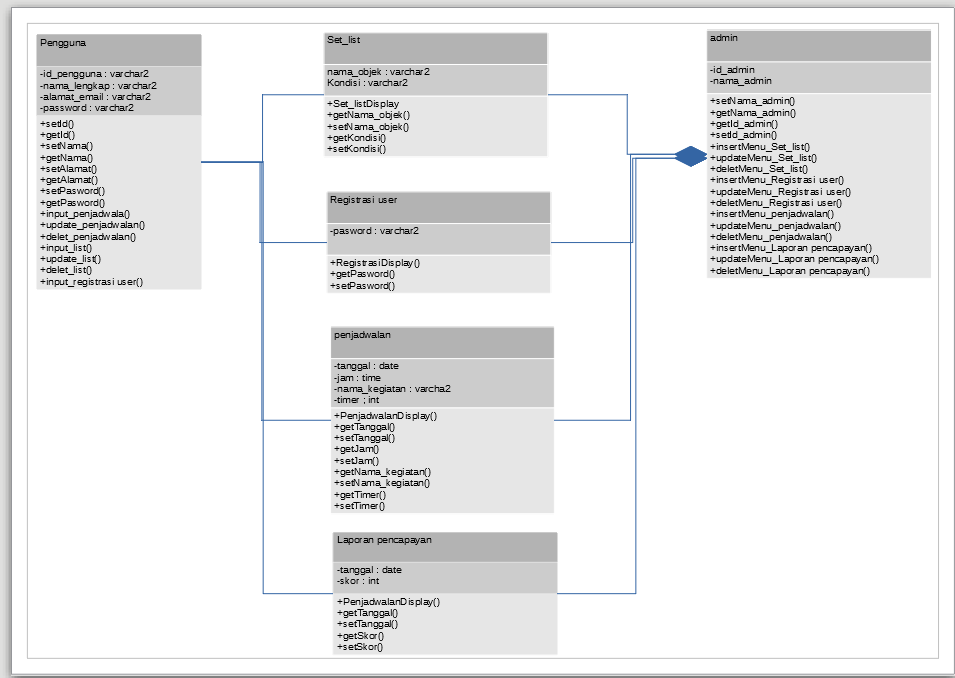
#### Usecase Scenario #7

|  |  |
| --- | --- |
| Nama Use Case | Edit List |
| Actor | User |
| Input | Data Jadwal |
| Output | Data Jadwal |
| Pre-Condition | User ingin mengupdate list |
| Post-Conditon | User telah mengupdate list dan telah disimpan oleh program |
| Deskripsi | User mengupdate list untuk merubah data sebelumnya |
| Typical Course of Event | |  |  | | --- | --- | | Actor | System | | 20. Mengklik ikon Update List |  | |  | 21. Menampilkan menu update List | | 22. Mengupdate data list |  | | 23. Menekan tombol simpan |  | |  | 24. Data disimpan | |

#### Usecase Scenario #8

|  |  |
| --- | --- |
| Use Case | Update aplikasi |
| Input | Data Jadwal |
| Output | Data pencapaian |
| Actor | Admin |
| Precondition | Admin ingin memperbarui aplikasi |
| Post Condition | User telah melakukan input data jdwal |
| Description | Untuk menginputkan jadwal oleh user |
| Typical Course of Event | |  |  | | --- | --- | | user | system | | 1. Membuka menu input jadwal |  | |  | 1. Menampilkan menu input jadwal | | 1. memasukan data jadwal |  | | 1. menekan tombol simpan |  | |  | 1. menyimpan data user | |

### Class Diagram



jadwal

list

# Requirements Antarmuka Eksternal

## Antarmuka Pengguna

Antarmuka yang digunakan pengguna (*user*) untuk aplikasi Neatly ini adalah antarmuka berbasis aplikasi Android/Smartphone. Yang memudahkan pengguna (*user*) berinteraksi secara *mobile* *interface*.

## Antarmuka Perangkat Keras

Kebutuhan perangkat keras:

1. Ponsel pintar (Smartphone)

2. Komputer (Monitor, CPU, keyboard, mouse)

## Antarmuka Perangkat Lunak

Aplikasi ini hanya hanya dapat digunakan oleh Android. Aplikasi Neatly dapat diakses menggunakan berbagai jenis Smartphone berbasis Android. Dalam membangun aplikasi ini menggunakan Bahasa Javascript.

## Antarmuka Komunikasi

Aplikasi netly merupakan aplikasi ofline sehingga tidak memerlukan akses ke serfer

# Requirements Lain

Lampiran A: Daftar Kata-Kata Sukar

- SKPL: Spesifikasi Kebutuhan Perangkat Lunak merupakan spesifikasi dari perangkat

lunak yang akan dikembangkan.

- Usecase Scenario: Narasi untuk menjelaskan usecase yang telah dibuat sehingga custemer lebih memahami fungsi dari usecase yang ada

- Usecase diagram: Diagram yang mendeskripsikan fungsionalitas sistem berdasarkan sudut pandang user

- Class Diagram: model statis yang menggambarkan struktur dan deskripsi class serta hubungannya antara class.

- DBMS: kependekan dari data base managjment system

- MySQL: suatu DBMS berbasis SQL

- Android Studio: suatu IDE untuk mengolah bahasa pemograman javascript yang ditujukan untuk pembuatan aplikasi di os android

- Javascript: suatu nama dari bahasa pemograman

Lampiran B: Analysis Models

ER Diagram

