**SISTEM INFORMASI PERPUSTAKAAN SEKOLAH**

**Diajukan untuk memenuhi tugas UTS Peraktikum Rekayasa Perangkat Lunak Lanjut**

*https://github.com/yopan/UTS-Praktikum-RPL.git*



Oleh:

Nama : Yopan Yuliana

Nim : 1137050222

Kelas : IF-D

**Teknik Informatika**

**UIN Sunan Gunung Djati Bandung**

**2015**

# Latar Belakang

Teknologi Informasi berperan penting dalam memperbaiki kualitas suatu Instansi. Penggunaannya tidak hanya sebagai proses otomatisasi terhadap akses informasi, tetapi juga menciptakan akurasi, kecepatan, dan kelengkapan sebuah sistem yang terintegrasi, sehingga proses organisasi yang terjadi akan efisien, terukur, fleksibel serta dapat memudahkan penggunanya dalam mengelola suatu sistem.

Bahkan dewasa ini perkembangan Teknologi Informasi mulai mendapat sambutan positif dari masyarakat. Perkembangannya tidak hanya disambut dan dinikmati oleh kalangan bisnis maupun pemerintah saja, tetapi juga mulai merambah dalam dunia pendidikan karena ketersediaan informasi yang terintegrasi makin penting dalam mendukung upaya menciptakan generasi penerus bangsa yang kompetitif.

Untuk mendukung kegiatan pembelejaran peran perpustakaan sangat dirasakan oleh akademika terutama para siswa yang sedang belajar akan lebih mudah dalam meminjam, mencari atau mengembalikan buku yang sudah selesesai masa peminjamanya.

Oleh karena itu pihak intansi terkait memiliki sumber daya yang memungkinkan untuk dibuatnya suatu Sistem Informasi manajemen Perpustakaan yang akan memudahkan dalam pengelolaan sistem yang berjalan di perpustakaan tersebut.

# Rumusan Masalah

1. Masih di gunakanya pendataan peminjaman buku secara manual dengan menggunakan buku peminjaman
2. Banyak data peminjaman yang hilang karena tercecer
3. Laporan peminjaman yang sering kurang lengkap

# Tujuan

1. Dapat mempermudah proses pencarian dan peminjaman buku.
2. Memudahkan dalam mengakses Informasi persediaan buku yang diterima pengunjung perpustakaan lebih cepat dan efektif.
3. Dapat melakukan registrasi secara komputerisasi.
4. Mengelola data anggota perpustakaan, dan informasi yang berhubungan dengan perpustakaan.

# Batasan

1. Perangkat lunak hanya dijalankan di Windows (XP,vista, 7,server 2008 dll).
2. Waktu pengembangan perangkat lunak yang singkat membuat adanya kemungkinan tidak semua fungsi yang ada dapat dilaksanakan.
3. Pengembangan perangkat lunak tidak akan merubah file-file ataupun database yang ada pada saat ini tanpa adanya izin dari pimpinan Perpustakaan.
4. Pengembangan perangkat lunak ini akan mengotomatisasi pengelolaan data-data yang ada di Perpustakaan, yang meliputi data anggota, data pengunjung, katalog buku, data sirkulasi buku dan laporan untuk pimpinan Perpustakaan.

# Deskripsi Perangkat Lunkak

## Deskripsi Umum

Deskripsi umum sistem ini menggunakan bahasa pemograman Microsoft Visual Basic 8.0 dengan Mysql dan koneksi mysql sebagai pendukung dalam pembuatan database perpustakan. Database perpustakaan ini merupakan tempat sekumpulan informasi tentang Pendaftaran anggota baru,Pengisian buku Pengunjung , peminjaman buku, pengembalian buku,denda yang harus dibayar saat pengembailan buku terlambat,yang dibuat dengan tujuan untuk mempermudah dan mempercepat proses kegiatan di perpustakaan.

Dengan adanya aplikasi perpustakaan ini petugas perpustakaan tidak lagi menggunakan cara manual dalam melakukan proses kegiatan perpustakaan, mulai dari pengisian formulir anggota, hingga proses peminjaman buku, dan pengembalian buku kini dapat dilakukan dengan mudah, dengan menginputkan dari database yang telah dibuat. Aplikasi ini juga dilengkapi dengan form pencarian buku sehingga anggota perpustakaan yang ingin meminjam buku dapat dengan mudah menemukan buku tersebut. Cukup dengan mengetik keyword buku yang ingin dipinjamnya.

Denda bagi anggota yang melanggar peraturan perpustakaan juga tersedia di aplikasi perpustakaan ini, denda ini akan terjadi secara otomatis disaat anggota melakukan pelanggaran atau keterlambatan pengembalian, sistem denda ini mencetak id anggota dan nama anggota secara otomatis disaat proses pengembaliannya mengalami masalah. Aplikasi ini juga mempunyai tampilan yang menarik dan mudah dimengerti sehingga petugas dapat dengan mudah untuk mengoperasikan aplikasi perpustakaan ini

## Pengguna

| Kategori Pengguna | Tugas | Hak Akses ke aplikasi |
| --- | --- | --- |
| Operator | Mencatat Peminjaman Buku, Pengembalian, Jumlah Buku | Mengakses From Peminjaman Dan Mencetak Laporan |
| Peminjam | Menminjam Dan Mengembalikan Buku Yang Di Pinjam | Mengakses Daftar Buku Yang Tersedia |
| Admin | Menerima Loparan Peminjaman | Mengakses Jumlah Pengunjung Dan Data Buku |

# Requirement

## Functional Requirement

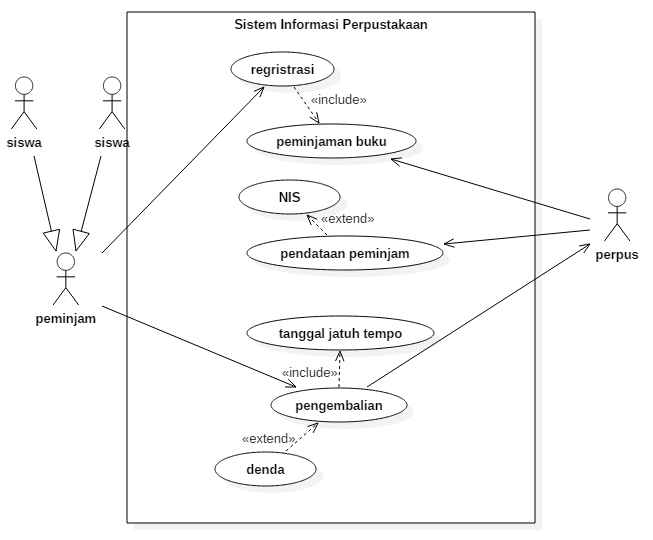
| ID | Kebutuhan | Penjelasan |
| --- | --- | --- |
| FR01 | Sistem Harus Dapat Melakukan Pencarian Sederhana Dan Menampilkan Daftar Buku Yang Tersedia | Fungsionalitas Ini Digunakan Oleh Pengunjung atau Peminjam Untuk Mencari Dan Menampilkan Daftar Buku |
| FR02 | Sistem Harus Mampu Melakukan Pencarian Berdasarkan Kata Kunci Yang Di Masukan Oleh Pengunjung AtauPun Peminjam | Fungsianalitas Ini Digunakan Oleh Anggota (Guru Dam Peminjam) Untuk Mencari Buku berdasarkan Kata Kunci Yang Di Inputkan |
| FR03 | Sistem Mampu Melakukan Cek Status Bulu | Fungsionalitas Ini Digunakan Oleh Peminjam Dengan Memasukan NIS (No Induk Siswa) Terlebih Dahulu Untuk Mengecek Status Peminjaman |
| FR04 | Sistem Dapat Menghitung Dendan Berdasarkan Jumlah Keterlabatan Hari Pengembalian Dan Melakukan Konfirmasi Pengembalian | Fungsonalitas Ini Dapat Menampilkan Peminjaman Pengembalian Buku Dengan Informasi NIS, Jenis Transaksi (Peminjaman atau Pengembalian), Id Buku Dan Nama Buku |
| FR05 | Sistem Harus Mampu Menampilkan Halaman Login | Fungsionalitas Ini Di Gunakan Oleh Admin Dan Operator Sebelum Memasuki Halaman Dan Terlebih Dahulu Harus Memasukan ID Dan Password |

## Non Functional Requirement

| ID | Parameter | Kebutuhan |
| --- | --- | --- |
| NFR01 | Availability | Sistem Harus Dapat Berjalan Selama 8 Jam Sehari |
| NFR02 | Reliability | Sistem Harus Dapat Menampilkan Semua Daftar Buku yang Tersedia |
| NFR02 | Ergonomy | Sistem Harus Dapa Mencari Buku Sesuai Inputan Pengguna |
|  | Portability | N/A |
|  | Memory | Sistem Harus Dapat Menyimpan 500 Giga |
|  | Response time | N/A |
|  | Safety | N/A |
|  | Security | Sistem Harus Menyediakan Form Login Yang Di Isi Sesuai User Name Dan Password Dan Tersimpan Di Memory |
|  |  |  |
|  | Others 1: Bahasa komunikasi | Misalnya : semua tanya jawab harus dalam bahasa Indonesia |
|  |  | Setiap layar harus mengandung logo PT Pos Indonesia |
|  |  |  |

# Usecase

## Usecase Diagram*.*



## Definisi Actor

|  |  |  |
| --- | --- | --- |
| No | Actor | Deskripsi |
| 1 | Admin | Orang Yang Mengelola Sistem Pepustakaan |
| 2 | Operator | Orang Yang Bertanggung Jawab Mengelola Transaksi Peminjaman Dan Pengembalian |
| 3 | Peminjam | Orang Yang Meminjam Dan Mengembalikan Buku |

## Definisi Usecase

|  |  |  |
| --- | --- | --- |
| No | Use Case | Deskripsi |
| 1 | Regristrasi | Proses Pemasukan Anggota Baru Kedalam Sistem |
| 2 | Peminjaman Buku | Proses Transaksi Peminjaman Buku Antara Operator Dengan Peminjam |
| 3 | Pendataan Peminjaman Buku | Menampillkan Proses Untuk Mengelola Data Peminjam Buku |
| 4 | Pengembalian | Proses Pengembalian Buku Sebelum Jatuh Tempo |

## Skenario Usecase

**Nama Use Case : Registrasi**

**Skenario :**

|  |  |
| --- | --- |
| ***Aksi Actor*** | ***Reaksi Sistem*** |
| Skenario Normal |  |
| 1. Memasukkan data anggota   sesuai field yang ada |  |
| 1. Mengklik tombol tambah data |  |
|  | 1. Mengecek valid tidaknya data masukkan |
|  | 1. Menyimpan data anggota ke basis data |
|  | 1. Menampilkan pesan sukses disimpan. |
| Skenario Alternatif |  |
| 1. Memasukkan data anggota   sesuai field yang ada |  |
| 1. Mengklik tombol tambah data |  |
|  | 1. Mengecek valid tidaknya data masukkan |
|  | 1. Menampilkan pesan *error* |
| 1. Memasukkan data anggota yang   valid sesuai field yang ada |  |
| 1. Mengklik tombol tambah data |  |
|  | 1. Mengecek valid tidaknya data masukkan |
|  | 1. Menyimpan data anggota ke basis data |
|  | 1. Menampilkan pesan sukses disimpan |

**Nama Use Case : Peminjaman**

**Skenario :**

|  |  |
| --- | --- |
| **Aksi Actor** | **Reaksi Sistem** |
| Skenario Normal |  |
| 1. Memasukkan data peminjaman sesuai field yang ada |  |
| 1. Mengklik tombol tambah data |  |
|  | 1. Menyimpan data peminjaman ke basis data |
|  | 1. Menampilkan pesan sukses disimpan. |
| Skenario Alternatif |  |
| 1. Memasukkan data peminjaman   sesuai field yang ada |  |
| 1. Mengklik tombol tambah data |  |
|  | 1. Mengecek valid tidaknya data masukkan |
|  | 1. Menampilkan pesan error |
| 1. Memasukkan data peminjaman   yang valid sesuai field yang ada |  |
| 1. Mengklik tombol tambah data |  |
|  | 1. Mengecek valid tidaknya data masukkan |
|  | 1. Menyimpan data peminjaman ke basis data |
|  | 1. Menampilkan pesan sukses disimpan. |

**Nama Use Case : Pendataan Peminjaman**

**Skenario :**

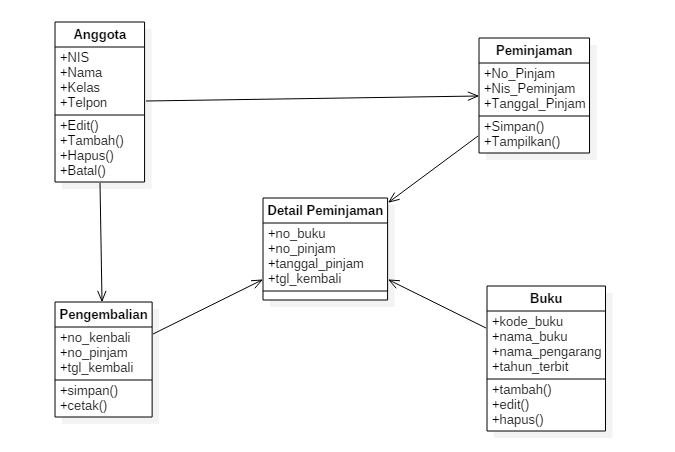
|  |  |
| --- | --- |
| **Aksi Actor** | **Reaksi Sistem** |
| Skenario Normal |  |
| 1. Memasukkan data peminjaman   sesuai field yang ada |  |
| 1. Mengklik tombol tambah data |  |
|  | 1. Menyimpan data peminjaman ke basisdata |
|  | 1. Menampilkan pesan sukses disimpan |
| Skenario Alternatif |  |
| 1. Memasukkan data peminjaman sesuai field yang ada |  |
| 1. Mengklik tombol tambah data |  |
|  | 1. Mengecek valid tidaknya data masukkan |
|  | 1. Menampilkan pesan *error* |
| 1. Memasukkan data peminjaman yang valid sesuai field yang ada |  |
| 1. Mengklik tombol tambah data |  |
|  | 1. Mengecek valid tidaknya data masukkan |
|  | 1. Menyimpan data peminjaman ke basis data |
|  | 1. Menampilkan pesan sukses disimpan |

**Nama Use Case : Pengembalian**

**Skenario :**

|  |  |
| --- | --- |
| **Aksi Actor** | **Reaksi Sistem** |
| Skenario Normal |  |
| 1. Memilih data peminjaman yang akan di kembalikan, pada tabel data peminjaman |  |
| 1. Mengklik tombol kembalikan |  |
|  | 1. Menampilkan pesan konfirmasi apakah buku telah di kembalikan |
| 1. Mengklik pilihan setuju data pengembalian telah di simpan |  |
|  | 1. Menyimpan data pengembalian dari   basis data |
| Skenario Alternatif |  |
| 1. Memilih data peminjaman yang akan yang akan mengembalikan pada tabel data peminjaman |  |
| 1. Mengklik tombol pengembalian |  |
|  | 1. Menampilkan pesan konfirmasi apakah buku sudah di kembalikan |
| 1. Mengklik pilihan tidak |  |
|  | 1. Kembali ke form data peminjaman |

# Class Diagram

**

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas | Jenis |
| 1 | Anggota | Interfaces Class |
| 2 | Pengembalian | Control Class |
| 3 | Buku | EntityClass |
| 4 | Peminjaman | Control Class |