



KERJA PRAKTIK - KI141330

## Rancang Bangun Aplikasi *Self Services* Sirkulasi Ruang Baca Berbasis *SLiMS*

Ruang Baca Teknik Informatika ITS  
Jalan Teknik Kimia, Surabaya - 60111

Oleh:

Adenuar Purnomo 5114100079

Alam Ar Raad Stone 5114100701

Pembimbing Departemen  
Arya Yudhi Wijaya, S.Kom., M.Kom.  
Pembimbing Lapangan  
Eva Mursidah AMd.

DEPARTEMEN TEKNIK INFORMATIKA  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017

*[Halaman ini sengaja dikosongkan]*



**KERJA PRAKTIK - KI141330**

## **Rancang Bangun Aplikasi Self Services Sirkulasi Ruang Baca Berbasis SLiMS**

**Ruang Baca Teknik Informatika ITS  
Jalan Teknik Kimia, Surabaya - 60111  
Periode:-2017**

Oleh:

Aenuar Purnomo	5114100079
Alam Ar Raad Stone	5114100701

Pembimbing Departemen  
Arya Yudhi Wijaya, S.Kom., M.Kom.  
Pembimbing Lapangan  
Eva Mursidah Amd.

DEPARTEMEN TEKNIK INFORMATIKA  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017

*[Halaman ini sengaja dikosongkan]*

# LEMBAR PENGESAHAN

## KERJA PRAKTIK

### Rancang Bangun Aplikasi *Self Services* Sirkulasi Ruang Baca Berbasis *SLiMS*

Oleh:

**Adenuar Purnomo**  
**Alam Ar Raad Stone**

5114100079  
5114100701

Disetujui oleh Pembimbing Kerja Praktik:

1. Arya Yudhi Wijaya, S.Kom., M.Kom. ....  
NIP. 19840904 201012 1 002 (Pembimbing Departemen)
  
2. Eva Mursidah AMd. ....  
NIP. 19740125 200112 2 001 (Pembimbing Lapangan)

**Surabaya**  
**November 2017**

*[Halaman ini sengaja dikosongkan]*

## **Rancang Bangun Aplikasi *Self Services* Sirkulasi Ruang Baca Berbasis *SLiMS***

**Nama Mahasiswa** : Adenuar Purnomo  
**NRP** : 5114100079  
**Nama Mahasiswa** : Alam Ar Raad Stone  
**NRP** : 5114100701  
**Departemen** : Teknik Informatika FTif-ITS  
**Pembimbing Departemen** : Arya Yudhi Wijaya, S.Kom.,  
M.Kom.  
**Pembimbing Lapangan** : Eva Mursidah AMD.

### **ABSTRAK**

Di dalam sebuah perpustakaan terdapat peminjaman maupun pengembalian buku. Biasanya perpustakaan menggunakan cara konvensional yaitu dengan melakukan pencatatan peminjaman dan pengembalian buku secara manual. Dari permasalahan tersebut, *SLiMS* dapat menjadi solusinya. Senayan Library Management System (*SLiMS*) adalah perangkat lunak sistem manajemen perpustakaan (library management system) sumber terbuka yang dilisensikan di bawah GPL v3.

Permasalahan baru timbul ketika sumber daya manusia yang mengelola perpustakaan kurang, sehingga diperlukan aplikasi *self services* peminjaman buku yang berbasis *SLiMS*. Metode yang digunakan dalam pengerjaan aplikasi ini adalah metode Waterfall. Bahasa pemrogramannya adalah HTML dan PHP. Basis datanya menggunakan MySQL. *Tools* dan *Editor* yang digunakan adalah XAMPP, serta Sublime Text 2.

Aplikasi ini nantinya digunakan untuk melakukan pencatatan peminjaman dan pengembalian buku dengan tidak menggunakan kertas atau *paperless*.

***Kata kunci: self services, SLiMS, perpustakaan, website.***

*[Halaman ini sengaja dikosongkan]*

## **KATA PENGANTAR**

Puji syukur penulis panjatkan kehadirat Tuhan Yang Maha Esa, karena dengan rahmat dan karunia-Nya lah penulis dapat menyelesaikan tugas kerja praktik dan laporan kerja praktik ini dengan baik.

Dalam proses penyusunan laporan kerja praktik ini penulis mendapatkan banyak bantuan dari berbagai pihak selama lebih kurang satu bulan. Besar harapan penulis agar laporan kerja praktik ini dapat memberi manfaat bagi penulis khususnya dan bagi pembaca pada umumnya untuk membantu dalam pembuatan dan perancangan sistem informasi.

Sebagai manusia yang tidak sempurna, tentu akan ditemui kekurangan-kekurangan, baik yang penulis sadari maupun tidak, dalam pelaksanaan kerja praktik dan dalam penyusunan laporan ini. Untuk itu penulis mohon maaf yang sebesar-besarnya.

Pada kesempatan ini penulis ingin mengucapkan terima kasih yang mendalam kepada:

1. Ibu Eva Mursidah Amd., selaku pembimbing lapangan pada pengerjaan tugas kerja praktik ini.
2. Bapak Arya Yudhi Wijaya, S.Kom., M.Kom., selaku dosen pembimbing kerja praktik.
3. Orang tua dan keluarga penulis yang senantiasa memberikan dukungan moral dan material selama pelaksanaan kerja praktik.

Akhir kata, semoga laporan kerja praktik ini dapat menjadi pedoman dan dorongan bagi kemajuan teknologi informasi. Semoga Tuhan Yang Maha Esa melimpahkan rahmat dan karunianya bagi semua pihak yang telah membantu penulis dalam pelaksanaan kerja praktik dan penyusunan laporan ini.

Surabaya, November 2017

Aduenar Purnomo  
Alam Ar Raad Stone

# DAFTAR ISI

LEMBAR PENGESAHAN .....	v
ABSTRAK .....	vii
KATA PENGANTAR.....	ix
DAFTAR ISI.....	xi
DAFTAR GAMBAR .....	xv
DAFTAR TABEL .....	xv
1 BAB I PENDAHULUAN .....	1
1.1. Latar Belakang .....	1
1.2. Tujuan .....	1
1.3. Manfaat .....	2
1.4. Rumusan Permasalahan .....	2
1.5. Lokasi dan Waktu Kerja Praktik.....	2
1.6. Metodologi Kerja Praktik .....	2
1.7. Sistematika Laporan .....	3
2 BAB II PROFIL INSTANSI .....	5
2.1. Tujuan dan Fungsi .....	5
2.2. Visi dan Misi .....	5
2.3. Tugas .....	6
2.4. Struktur Organisasi.....	7
3 BAB III TINJAUAN PUSTAKA .....	9
3.1. Management System .....	9
3.2. SLIMS .....	9
3.3. MYSQL.....	9
3.4. PHP .....	10
3.5. Bootstrap .....	11
4 BAB IV DESAIN DAN PERANCANGAN SISTEM.....	13
4.1. Deskripsi Sistem.....	13
4.2. Diagram Kasus Penggunaan .....	13
4.3. Spesifikasi Kasus Penggunaan.....	14
4.3.1. Meminjam buku.....	14
4.3.2. Mengembalikan buku .....	15
4.3.3. Mereservasi buku.....	16

4.3.4.	Melihat denda .....	17
4.4.	Diagram Aktivitas .....	18
4.4.1.	Meminjam Buku .....	18
4.4.2.	Mengembalikan Buku .....	19
4.4.3.	Mereservasi Buku .....	21
4.4.4.	Melihat Denda .....	22
4.5.	Karakteristik Pengguna.....	23
4.6.	Lingkungan Operasi .....	23
4.7.	Lingkungan Operasi .....	23
4.7.1.	Kebutuhan Fungsional .....	23
4.7.2.	Kebutuhan Non Fungsional .....	24
5	<b>BAB V IMPLEMENTASI SISTEM</b> .....	25
5.1.	Implementasi Lapisan Kontrol.....	25
5.2.	Implementasi Lapisan Data .....	26
5.2.1.	Basis Data .....	26
5.3.	Implementasi Program.....	29
5.3.1.	Kode Sumber Navigasi .....	29
5.3.2.	Kode Sumber Koneksi .....	30
5.3.3.	Kode Sumber Header Aplikasi .....	31
5.3.4.	Kode Sumber Login Aplikasi .....	32
5.3.5.	Kode Sumber Fitur Peminjaman .....	33
5.3.6.	Kode Sumber Fitur Pemesanan.....	40
5.3.7.	Kode Sumber Fitur Pengembalian .....	45
5.3.8.	Kode Sumber Fitur Denda .....	48
5.4.	Implementasi Antarmuka.....	50
6	<b>BAB VI PENGUJIAN DAN EVALUASI</b> .....	53
6.1.	Lingkungan Pengujian .....	53
6.1.1.	Server Aplikasi .....	53
6.1.2.	Perangkat Uji Coba.....	53
6.2.	Skenario Pengujian .....	53
6.3.	Evaluasi Pengujian .....	54
6.3.1.	Evaluasi Pengujian Kebutuhan Fungsional .....	54
6.3.2.	Evaluasi Pengujian Kebutuhan Non Fungsional .....	55
7	<b>BAB VII KESIMPULAN DAN SARAN</b> .....	57
7.1.	Kesimpulan.....	57

7.2. Saran.....	57
DAFTAR PUSTAKA .....	59
BIODATA PENULIS .....	61

*[Halaman ini sengaja dikosongkan]*

## DAFTAR GAMBAR

Gambar 2.4-1. Struktur Organisasi Ruang Baca Teknik Informatika.....	7
Gambar 4.2-1. Diagram Kasus Penggunaan .....	14
Gambar 4.4-1. Activity Diagram Meminjam Buku .....	19
Gambar 4.4-2. Activity Diagram Mengembalikan Buku .....	20
Gambar 4.4-3. Activity Diagram Memesan Buku .....	21
Gambar 4.4-4. Activity Diagram Melihat Denda .....	22
Gambar 5.4-1. Halaman Login Pengguna .....	50
Gambar 5.4-2. Halaman Peminjaman Buku .....	51
Gambar 5.4-3. Halaman Pengembalian Buku.....	51
Gambar 5.4-4. Halaman Reservasi Buku.....	52
Gambar 5.4-5. Halaman Lihat Denda.....	52

## DAFTAR TABEL

Tabel 4.5-1. Karakteristik Pengguna .....	23
Tabel 4.7-1. Kebutuhan Fungsional .....	24
Tabel 4.7-2. Kebutuhan Non Fungsional.....	24
Tabel 5.1-1. Implementasi Lapisan Kontrol .....	25
Tabel 5.2-1. Tabel Member .....	26
Tabel 5.2-2. Tabel Loan .....	27
Tabel 5.2-3. Tabel Item .....	27
Tabel 5.2-4. Tabel Reserve.....	28
Tabel 5.2-5. Tabel Fines .....	29
Tabel 6.3-1. Pengujian Kebutuhan Fungsional .....	54
Tabel 6.3-2. Pengujian Kebutuhan Non Fungsional .....	55



# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Pesatnya perkembangan teknologi saat ini membuat manusia melakukan aktifitas sehari-hari lebih cepat dan efisien. Efisiensi merupakan kebutuhan utama dalam meningkatkan kinerja sebuah sistem dan mempermudah pekerjaan semua entitas yang berada di dalam sistem tersebut. *Self services* peminjaman buku merupakan aplikasi berbasis SLiMS yang akan digunakan oleh Ruang Baca Teknik Informatika ITS.

*Self services* Sirkulasi Ruang Baca merupakan aplikasi berbasis SLiMS yang membantu Ruang Baca Teknik Informatika ITS dalam melakukan pendataan peminjaman dan pengembalian buku di Ruang Baca.

Aplikasi ini mempercepat proses peminjaman maupun pengembalian buku di Ruang Baca Teknik Informatika ITS. Sebelumnya peminjaman dan pengembalian buku dilakukan dengan pencatatan manual, sehingga data yang ada di database Ruang Baca sering terjadi ketidakcocokan data. Hal ini dibuktikan dengan tidak beraturannya database Ruang Baca Teknik Informatika ITS. Aplikasi ini diharapkan dapat mempermudah pekerjaan pegawai Ruang Baca dengan tidak membingungkan para user yang memakainya.

Aplikasi ini berguna juga untuk meningkatkan kualitas setiap individu yang berada di Ruang Baca Teknik Informatika ITS.

### **1.2. Tujuan**

Tujuan pengerjaan kerja praktik ini adalah:

Membuat aplikasi *self services* Sirkulasi Ruang Baca berbasis SLiMS untuk memudahkan melakukan peminjaman dan pengembalian buku di Ruang Baca Teknik Informatika ITS .

### **1.3. Manfaat**

Manfaat yang diperoleh selama pengerjaan kerja praktik adalah:

1. Ruang Baca Teknik Informatika ITS memiliki sumber kode yang dapat dikembangkan di masa yang akan datang.
2. Mempermudah peminjaman buku di Ruang Baca Teknik Informatika ITS.
3. Mengubah kegiatan peminjaman buku di Ruang Baca Teknik Informatika ITS menjadi *paperless*.
4. Menambah ilmu baru yang belum pernah kami pelajari sebelumnya.

### **1.4. Rumusan Permasalahan**

Masalah yang akan dibahas dalam pengerjaan kerja praktik ini adalah:

1. Bagaimana cara untuk membuat aplikasi *self services* Sirkulasi Ruang Baca yang mudah digunakan oleh mahasiswa Teknik Informatika ITS dalam kegiatan peminjaman dan pengembalian buku serta membuatnya menjadi *paperless*.

### **1.5. Lokasi dan Waktu Kerja Praktik**

1. Lokasi Kerja Praktik:  
Ruang Baca Teknik Informatika ITS  
Jalan Teknik Kimia, Surabaya – 60111
2. Waktu Kerja Praktik  
Periode : 1 April 2017 – 14 Mei 2017.  
Hari Kerja : Sabtu – Minggu.  
Jam : 08.00 – 13.00 WIB.

### **1.6. Metodologi Kerja Praktik**

#### **1. Perumusan Masalah**

Pada tahap ini, pembimbing lapangan menjelaskan mengenai proses peminjaman buku di Ruang Baca Teknik Informatika ITS serta apa saja yang dibutuhkan dalam melaksanakan kegiatan ini. Diskusi lebih lanjut diterapkan oleh peserta magang untuk kebutuhan-kebutuhan tambahan yang mungkin dibutuhkan untuk perkembangan aplikasi kedepannya.

## **2. Studi Literatur**

Pada tahap ini dilakukan studi literatur tentang aplikasi yang akan dibangun, yaitu PHP dan HTML sebagai bahasa pemrograman dalam pembuatan aplikasi di sisi *back-end*, SLiMS sebagai *content management system* untuk perpustakaan, dan studi literatur mengenai pemasangan aplikasi pada server. Kemudian setiap referensi yang digunakan dalam pengerjaan aplikasi ini akan dituliskan pada daftar pustaka.

## **3. Analisis dan Perancangan Sistem**

Pada tahap ini dilakukan proses kebutuhan apa saja yang dibutuhkan oleh sistem, serta memberikan skenario yang mungkin terjadi ketika aplikasi ini berjalan. Kemudian dilanjutkan dengan diskusi mengenai kebutuhan-kebutuhan yang lebih mendalam.

## **4. Implementasi Sistem**

Pada tahap ini dilakukan pembelajaran mengenai cara guna SLiMS. Setelah dimengerti tentang cara menggunakan *content management system* tersebut, mulai dibuat kode pada sisi server menggunakan bahasa pemrograman PHP dan HTML.

## **5. Pengujian dan Evaluasi**

Pada tahap ini dilakukan pengujian terhadap kebutuhan fungsional sistem. Pengujian dilakukan dari prototype aplikasi. Semua kebutuhan fungsional diuji apakah sudah berjalan dengan baik atau belum. Jika dari prototype yang ditunjukkan sudah sesuai, maka selanjutnya dilakukan fiksasi terhadap fungsi. Evaluasi kinerja sistem didapat dari hasil pengujian yang dilakukan pengguna dengan mengetahui tanggapan terhadap bagaimana sistem itu bekerja.

## **1.7. Sistematika Laporan**

Laporan kerja praktik ini disusun dari 7 bab, yaitu:

### **1. Bab I Pendahuluan**

Bab ini dipaparkan mengenai garis besar kerja praktik yang meliputi latar belakang, tujuan kerja praktik, manfaat yang didapatkan dari

kerja praktik, rumusan permasalahan yang ditemukan ketika kerja praktik, lokasi dan waktu kerja praktik, metodologi kerja praktik dan sistematika laporan.

## **2. Bab II Profil Instansi**

Bab ini berisi penjelasan mengenai profil Ruang Baca Teknik Informatika ITS yang meliputi visi dan misi, struktur organisasi, dan informasi mengenai tugas dan fungsi Ruang Baca Teknik Informatika.

## **3. Bab III Tinjauan Pustaka**

Bab ini berisi penjelasan tentang istilah-istilah atau teori-teori yang digunakan dalam pembuatan kerja praktik dan pustaka yang dipakai.

## **4. Bab IV Analisis dan Perancangan Sistem**

Bab ini berisi penjelasan mengenai analisis terhadap sistem dan pemaparan mengenai kebutuhan untuk perancangan sistem yang akan dibangun dan dikembangkan.

## **5. Bab V Implementasi Sistem**

Bab ini berisi penjelasan tentang implementasi sistem dan hasil analisis yang telah dilakukan.

## **6. Bab VI Pengujian dan Evaluasi**

Bab ini berisi penjelasan tentang hasil pengujian sistem dan evaluasi yang dilakukan terhadap kinerja sistem secara menyeluruh.

## **7. Bab VII Kesimpulan dan Saran**

Bab ini berisi kesimpulan dan saran dari proses selama pengerjaan kerja praktik di Ruang Baca Teknik Informatika ITS.

## **BAB II**

### **PROFIL INSTANSI**

Ruang Baca merupakan salah satu perpustakaan kecil yang ada di lingkungan Institut Teknologi Sepuluh Nopember. Ruang Baca Jurusan Teknik Informatika merupakan sebuah perpustakaan yang digunakan untuk menunjang proses belajar mengajar yang dilakukan oleh Jurusan Teknik Informatika Institut Teknologi Sepuluh Nopember.

#### **2.1. Tujuan dan Fungsi**

Ruang Baca Jurusan Teknik Informatika memiliki tujuan dan fungsi sebagai berikut:

##### **1. Tujuan**

- Memperkaya pengetahuan yang relevan pada mahasiswa dan dosen.
- Meningkatkan motivasi belajar pada mahasiswa dan dosen.
- Menyampaikan informasi yang bermanfaat kepada seluruh civitas akademika dan masyarakat.

##### **2. Fungsi**

- Pusat layanan dan sumber informasi untuk program pendidikan dan pengajaran Jurusan Teknik Informatika ITS.
- Pusat layanan informasi dan sumber informasi untuk program penelitian.
- Pusat layanan informasi dan sumber informasi untuk program pengabdian pada masyarakat.

#### **2.2. Visi dan Misi**

Ruang Baca Jurusan Teknik Informatika memiliki visi dan misi sebagai berikut:

##### **1. Visi**

Menjadi mitra akademis bagi pelajar berdasarkan sikap peduli dan semangat belajar, memberikan layanan yang inovatif, berorientasi pada kepuasan pengguna, tidak terbatas oleh ruang dan waktu, dan pengembangan sains dan teknologi berdasarkan teknologi informasi.

## **2. Misi**

Menjadi mitra profesional bagi masyarakat belajar dengan bertindak sebagai pemandu dan peneliti pendidikan, menjadi pemandu proses belajar-mengajar dengan konsep belajar cara belajar, menjadi akses informasi utama bagi masyarakat global dimana sains yang menjadi fokus dari universitas.

### **2.3. Tugas**

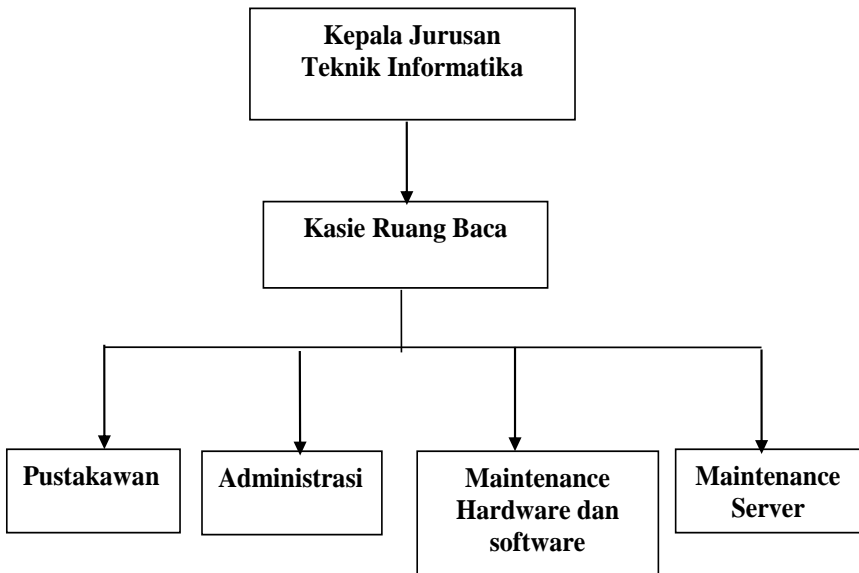
Ruang Baca Jurusan Teknik Informatika memiliki tugas sebagai berikut:

1. Menerima dan merekap usulan bahan pustaka baru dari pengguna ruang baca departemen Teknik Informatika, FTIF-ITS.
2. Pencarian deskripsi koleksi bahan pustaka yang akan di adakan lewat Internet atau book in print.
3. Penggandaan koleksi bahan pustaka sesuai kebutuhan.
4. Mendaftar bahan pustaka.
5. Melakukan klasifikasi bahan pustaka.
6. Melakukan katalogisasi bahan pustaka.
7. Menentukan subyek/keyword.
8. Membuat anotasi.
9. Entry Senayan Digital Library (Bibliography).
10. Melakukan pelabelan bahan pustaka.
11. Melakukan barcode koleksi.
12. Merawat bahan pustaka.
13. Membuat anotasi buku baru.
14. Menyebarkan informasi terkilat.
15. Mengengolah jajaran bahan pustaka.
16. Memberikan layanan kebutuhan informasi khusus.
17. Silang layanan antar ruang baca.
18. Statistik pengumpulan data.
19. Menyediakan koleksi bahan pustaka koleksi setempat.
20. Layanan sirkulasi.

## 2.4. Struktur Organisasi

Struktur Organisasi merupakan suatu kerangka yang mewujudkan pola tetap dari hubungan-hubungan diantara dibidang kerja maupun orang-orang yang menunjukkan kedudukan dan wewenang dan tanggung jawab dalam suatu sistem kerja. Suatu organisasi perlu diwujudkan dengan adanya kelompok orang yang berkumpul dan bekerja sama demi mencapai tujuan yang sudah disepakati bersama .

Struktur Organisasi Ruang Baca Jurusan Teknik Informatika:



Gambar 2.4-1. Struktur Organisasi Ruang Baca Teknik Informatika

*[Halaman ini sengaja dikosongkan]*

## **BAB III**

### **TINJAUAN PUSTAKA**

#### **3.1. *Management System***

Sistem manajemen adalah kerangka kebijakan, proses dan prosedur yang digunakan oleh sebuah organisasi untuk memastikan bahwa proyek tersebut dapat memenuhi semua tugas yang diperlukan untuk mencapai tujuannya. Tujuan ini akan menjadi campuran yang mencakup banyak aspek operasi organisasi (termasuk kesuksesan finansial, operasi yang aman, kualitas produk, hubungan klien, kesesuaian legislatif dan peraturan, manajemen pekerja, dll.).

#### **3.2. SLIMS**

Senayan, atau lengkapnya Senayan Library Management System (SLiMS), adalah perangkat lunak sistem manajemen perpustakaan (library management system) sumber terbuka yang dilisensikan di bawah GPL v3. Aplikasi web yang dikembangkan oleh tim dari Pusat Informasi dan Humas Departemen Pendidikan Nasional Republik Indonesia ini dibangun dengan menggunakan PHP, basis data MySQL, dan pengontrol versi Git.

#### **3.3. MYSQL**

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis. Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya; SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

### 3.4. PHP

PHP: Hypertext Preprocessor adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk memprogram situs web dinamis. PHP dapat digunakan untuk membangun sebuah CMS (Content Management System).

PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama FI (*Form Interpreted*), yang wujudnya berupa sekumpulan *script* yang digunakan untuk mengolah data form dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI, kependekan dari *Hypertext Preprocessing/Form Interpreter*. Dengan perilis kode sumber ini menjadi open source, maka banyak programmer yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini interpreter PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0.

Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek.

Kelebihan PHP dari bahasa pemrograman lain antara lain:

1. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. Web Server yang mendukung PHP dapat ditemukan dimana - mana dari mulai IIS sampai dengan apache, dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan developer yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena referensi yang banyak.

### 3.5. Bootstrap

Bootstrap adalah sebuah *library framework* CSS yang dibuat khusus untuk bagian pengembangan *front-end* website. Bootstrap juga merupakan salah satu framework HTML, CSS dan javascript yang paling populer di kalangan web developer yang digunakan untuk mengembangkan sebuah website yang responsive. Sehingga halaman website nantinya dapat menyesuaikan sesuai dengan ukuran monitor yang digunakan pengguna disaat mengakses website dari browser. Pada mulanya bootstrap bernama "*Twitter Blueprint*" yang dikembangkan oleh Mark Otto dan Jacob Thornton di Twitter sebagai kerangka kerja untuk mendorong konsistensi di alat internal.

Dengan menggunakan Bootstrap seorang developer dapat dengan mudah dan cepat dalam membuat *front-end* sebuah website. Developer hanya perlu memanggil *class-class* yang diperlukan, misalnya membuat tombol, grid, tabel, navigasi dan lainnya.

Bootstrap telah menyediakan kumpulan komponen *class interface* dasar yang telah dirancang sedemikian rupa untuk menciptakan sebuah tampilan yang menarik dan ringan. Selain komponen *class interface*, bootstrap juga memiliki grid yang berfungsi untuk mengatur layout pada halaman website. Selain itu developer juga dapat menambahkan class dan CSS sendiri, sehingga memungkinkan untuk membuat desain yang lebih variatif.

Bootstrap sendiri sudah kompatibel dengan versi terbaru dari beberapa browser seperti *google chrome*, *firefox*, *internet explorer*,

dan *safari browser*. Meskipun beberapa browser ini tidak didukung pada semua platform.

Beberapa alasan mengapa saat ini cukup banyak pengembang yang menggunakan Bootstrap dalam membuat *front-end* website, yaitu karena beberapa kelebihan yang dimiliki oleh Bootstrap itu sendiri yang antara lain:

1. Dapat mempercepat waktu proses pembuatan front-end website
2. Tampilan bootstrap yang sudah cukup terlihat modern.
3. Tampilan Bootstrap sudah responsive, sehingga mendukung segala jenis resolusi, baik itu PC, tablet, dan juga smartphone.

## **BAB IV**

### **DESAIN DAN PERANCANGAN SISTEM**

Bab ini membahas mengenai analisis dan perancangan sistem, meliputi deskripsi umum system, diagram kasus penggunaan, fitur, proses utama, diagram alir, arsitektur dan desain antar muka.

#### **4.1. Deskripsi Sistem**

Aplikasi *Self Services* Sirkulasi Ruang Baca ini merupakan aplikasi yang berguna untuk memberi kemudahan kepada pengguna Ruang Baca Teknik Informatika untuk melakukan peminjaman dan pengembalian buku. Aplikasi ini memiliki satu jenis pengguna yaitu user.

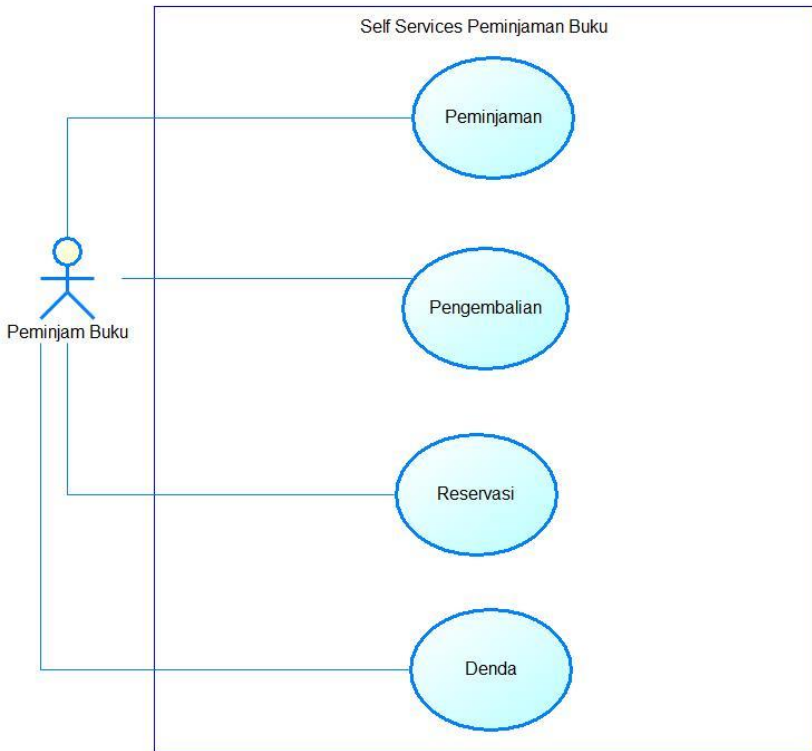
#### **4.2. Diagram Kasus Penggunaan**

Diagram kasus penggunaan adalah pemodelan untuk menggambarkan kelakuan sistem yang akan dibuat. Diagram kasus penggunaan menggambarkan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Secara sederhana, diagram kasus penggunaan digunakan untuk memahami fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang dapat menggunakan fungsi-fungsi tersebut.

Aplikasi ini memiliki beberapa fungsi dasar yang harus ada dan terintegrasi dengan baik pada sistem. Fungsi produk dari aplikasi ini adalah:

1. Meminjam buku di Ruang Baca Teknik Informatika ITS.
2. Mengembalikan buku yang telah dipinjam.
3. Melakukan booking peminjaman buku.
4. Melihat denda.

Diagram kasus penggunaan dapat dilihat pada gambar 4.2-1.



**Gambar 4.2-1. Diagram Kasus Penggunaan**

### 4.3. Spesifikasi Kasus Penggunaan

#### 4.3.1. Meminjam buku

Kode Use Case	: UC_001
Nama Use Case	: Meminjam Buku
Aktor	: Peminjam Buku
Deskripsi	: Kejadian dimana Peminjam buku meminjam buku.
Trigger	: Peminjam buku ingin meminjam buku

- Kondisi Awal : Buku ada dan tidak sedang dipinjam/direservasi
- Kondisi Akhir : Buku dipinjam
- Alur kejadian normal : **1.**Peminjam buku membuka aplikasi self services  
**2.**Sistem menampilkan Menu login  
**3.**Peminjam buku memasukkan NRP  
**4.**Sistem menampilkan Menu Utama  
**5.**Peminjam buku memilih menu meminjam buku  
**6.**Sistem menampilkan form peminjaman buku  
**7.**Peminjam buku memasukkan kode buku  
**8.**Sistem mengecek kode buku yang dimasukkan  
**9.**Sistem menampilkan pesan “peminjaman berhasil”  
**10.**Selesai
- Informasi Yang Terkait : **1.**Nama Peminjam buku  
**2.**Kode Buku  
**3.**Peminjaman

Berdasarkan spesifikasi kasus penggunaan UC\_001, maka dapat dibuat *activity diagram* seperti yang tertera pada gambar 4.4-1.

### **4.3.2. Mengembalikan buku**

- Kode Use Case : UC\_002
- Nama Use Case : Mengembalikan buku
- Aktor : Peminjam Buku
- Deskripsi : Kejadian dimana Peminjam buku mengembalikan buku yang dipinjam.
- Trigger : Peminjam Buku ingin mengembalikan buku yang dipinjam
- Kondisi Awal : Buku dipinjam oleh peminjam buku.

- Kondisi Akhir : Buku dikembalikan.
- Alur kejadian normal : **1.**Peminjam buku membuka aplikasi self services  
**2.**Sistem menampilkan Menu login  
**3.**Peminjam buku memasukkan NRP  
**4.**Sistem menampilkan Menu Utama  
**5.**Peminjam buku memilih menu mengembalikan buku  
**6.**Sistem menampilkan form pengembalian buku  
**7.**Peminjam buku memasukkan kode buku  
**8.**Sistem mengecek kode buku yang dimasukkan  
**9.**Sistem menampilkan pesan “buku berhasil dikembalikan”  
**10.**Selesai
- Informasi Yang Terkait : **1.**Nama Peminjam buku  
**2.**Kode Buku  
**3.**Peminjaman  
**4.**Tanggal Pengembalian

Berdasarkan spesifikasi kasus penggunaan UC\_002, maka dapat dibuat *activity diagram* seperti yang tertera pada gambar 4.4-2.

### **4.3.3. Mereservasi buku**

- Kode Use Case : UC\_003
- Nama Use Case : Mereservasi buku
- Aktor : Peminjam buku
- Deskripsi : Kejadian dimana peminjam buku mereservasi buku yang ingin dipinjam.
- Trigger : Peminjam buku ingin mereservasi buku yang ingin dipinjam.
- Kondisi Awal : Buku ada dan tidak sedang dipinjam/direservasi
- Kondisi Akhir : Buku direservasi

- Alur kejadian normal : **1.**Peminjam buku membuka aplikasi self services  
**2.**Sistem menampilkan Menu login  
**3.**Peminjam buku memasukkan NRP  
**4.**Sistem menampilkan Menu Utama  
**5.**Peminjam buku memilih menu reservasi buku  
**6.**Sistem menampilkan form reservasi buku  
**7.**Peminjam buku memasukkan kode buku  
**8.**Sistem mengecek kode buku yang dimasukkan  
**9.**Sistem menampilkan pesan “reservasi berhasil”  
**10.**Selesai
- Informasi Yang Terkait : **1.**Nama Peminjam buku  
**2.**Kode Buku

Berdasarkan spesifikasi kasus penggunaan UC\_003, maka dapat dibuat *activity diagram* seperti yang tertera pada gambar 4.4-3.

#### **4.3.4. Melihat denda**

- Kode Use Case : UC\_004  
Nama Use Case : Melihat denda  
Aktor : Peminjam buku  
Deskripsi : Kejadian dimana peminjam buku melihat denda peminjam.  
Trigger : Peminjam buku ingin melihat denda peminjaman  
Kondisi Awal : Denda belum dilihat  
Kondisi Akhir : Denda sudah dilihat.  
Alur kejadian normal : **1.**Peminjam buku membuka aplikasi self services  
**2.**Sistem menampilkan Menu login  
**3.**Peminjam buku memasukkan NRP

- 4.Sistem menampilkan Menu Utama
- 5.Peminjam buku memilih menu denda
- 6.Sistem menampilkan jumlah denda
- 7.Selesai

Informasi Yang Terkait : 1>Nama peminjam buku

Berdasarkan spesifikasi kasus penggunaan UC\_004, maka dapat dibuat *activity diagram* seperti yang tertera pada gambar 4.4-4.

#### 4.4. Diagram Aktivitas

Diagram Aktivitas menunjukkan bagaimana sistem melakukan respon terhadap input dari user. Diagram Aktivitas dibuat berdasarkan Diagram Kasus Penggunaan pada sistem.

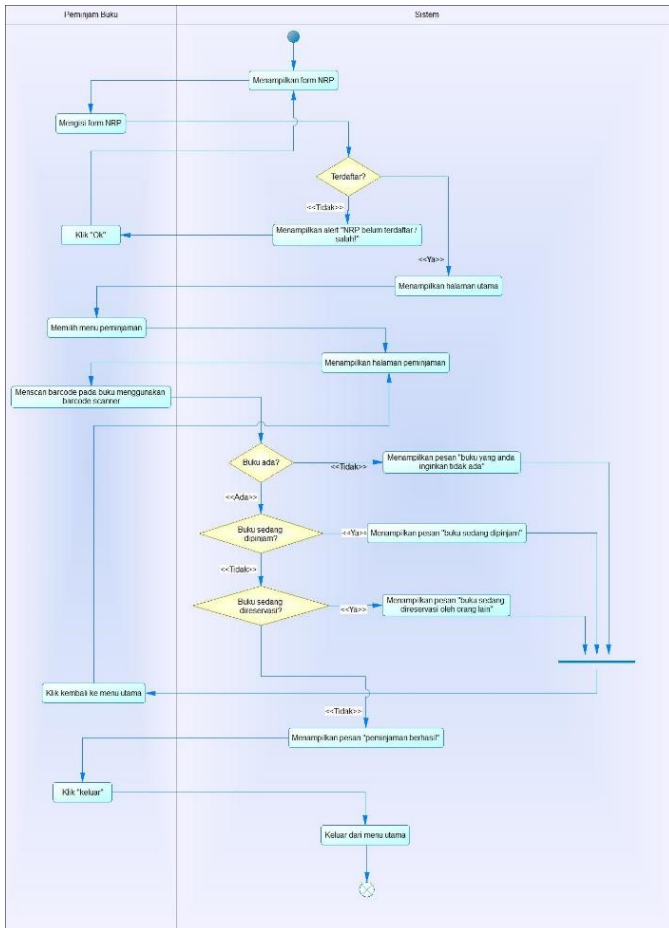
Diagram Aktivitas juga digunakan sebagai representasi grafis dari *workflow* dari kegiatan dan tindakan bertahap dengan dukungan untuk pilihan, *iterasi*, dan *concurrency*. Dalam *Unified Modeling Language*, Diagram Aktivitas dimaksudkan untuk model kedua proses komputasi dan organisasi (yaitu *workflow*). Diagram Aktivitas menunjukkan aliran keseluruhan kontrol.

Diagram Aktivitas dibangun dari sejumlah bentuk, dihubungkan dengan panah. Bentuk-bentuk yang dimaksud antara lain:

1. Persegi panjang bulat merupakan tindakan.
2. Berlian merupakan keputusan.
3. Bar mewakili awal (split) atau akhir (bergabung) kegiatan bersamaan.
4. Lingkaran merupakan awal (initial state) dari alur kerja.
5. Lingkaran hitam dengan silang mewakili akhir (keadaan akhir).

##### 4.4.1. Meminjam Buku

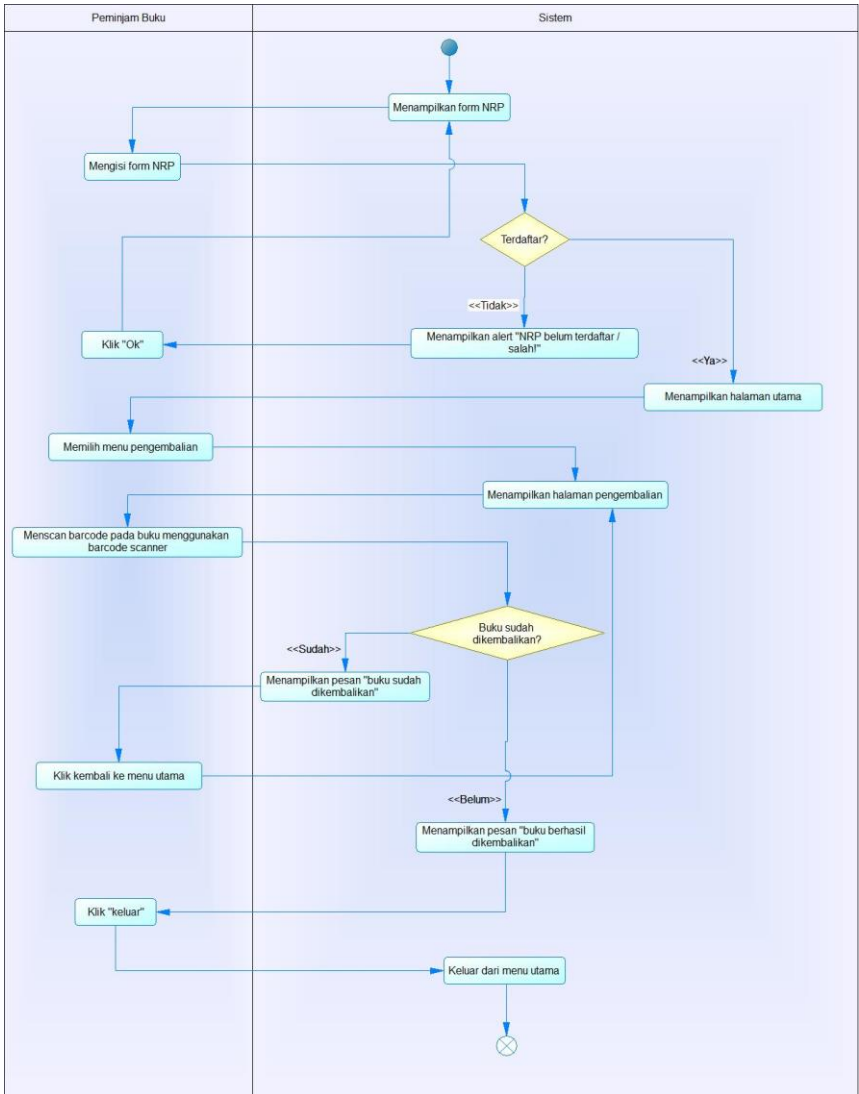
Pada gambar 4.4-1., disajikan diagram aktivitas ketika pengguna melakukan peminjaman buku pada Ruang Baca Teknik Informatika. Diagram aktivitas pada gambar 4.4.-1., dibuat berdasarkan spesifikasi kasus penggunaan UC\_001.



**Gambar 4.4-1. Activity Diagram Meminjam Buku**

### 4.4.2. Mengembalikan Buku

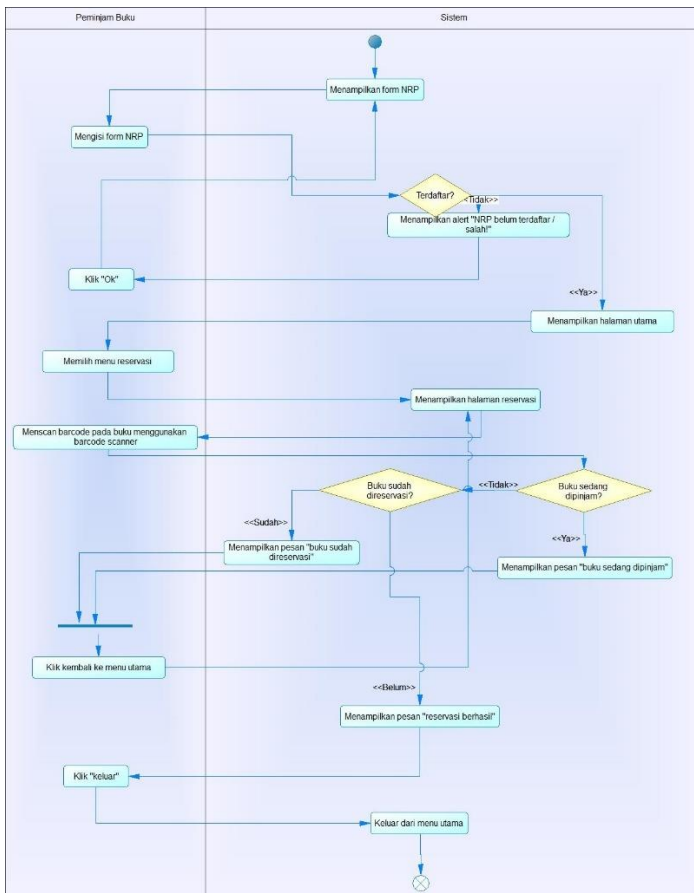
Pada gambar 4.4-2., disajikan diagram aktivitas ketika pengguna melakukan pengembalian buku pada Ruang Baca Teknik Informatika. Diagram aktivitas pada gambar 4.4.-2., dibuat berdasarkan spesifikasi kasus penggunaan UC\_002.



**Gambar 4.4-2. Activity Diagram Mengembalikan Buku**

### 4.4.3. Mereservasi Buku

Pada gambar 4.4-3., disajikan diagram aktivitas ketika pengguna melakukan reservasi buku pada Ruang Baca Teknik Informatika. Diagram aktivitas pada gambar 4.4.-3., dibuat berdasarkan spesifikasi kasus penggunaan UC\_003.



**Gambar 4.4-3. Activity Diagram Mereservasi Buku**



## 4.5. Karakteristik Pengguna

Pada aplikasi ini akan ditentukan mengenai karakteristik pengguna, mulai dari kategori pengguna, apa saja tugas dari setiap kategori pengguna, dan bagaimana hak akses setiap kategori pengguna terhadap aplikasi ini. Penjelasan dari hal tersebut akan dijelaskan pada tabel 4.5-1.

**Tabel 4.5-1. Karakteristik Pengguna**

Kategori Pengguna	Tugas
Peminjam Buku	<ol style="list-style-type: none"><li>1. Meminjam buku</li><li>2. Mengembalikan buku</li><li>3. Memesan buku</li><li>4. Melihat denda</li></ol>

## 4.6. Lingkungan Operasi

1. Sistem Operasi Server: Windows 8.1
2. Web Server: Apache
3. Database: MySQL
4. Bahasa Pemrograman: PHP, HTML
5. Library yang digunakan: *Bootstrap*

## 4.7. Lingkungan Operasi

Deployment diagram adalah Aplikasi *Self Services* Sirkulasi Ruang Baca yang dibangun disini memiliki beberapa kebutuhan yang dibagi menjadi kebutuhan fungsional dan kebutuhan non fungsional.

### 4.7.1. Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang berisi layanan apa saja yang nantinya harus disediakan oleh sistem, mencakup bagaimana sistem harus bereaksi pada input tertentu dan bagaimana perilaku sistem pada situasi tertentu. Kebutuhan fungsional dapat dilihat pada tabel 4.7-1.

**Tabel 4.7-1. Kebutuhan Fungsional**

No.	Kebutuhan	Deskripsi Kebutuhan
F001	Meminjam Buku	Peminjam Buku dapat meminjam buku yang sedang tidak dipinjam/direservasi, atau sudah direservasi sebelumnya oleh Peminjam Buku
F002	Mengembalikan Buku	Peminjam Buku dapat mengembalikan buku yang telah dipinjam
F003	Reservasi Buku	Peminjam Buku dapat mereservasi buku yang sedang tidak direservasi/dipinjam
F004	Melihat Denda	Peminjam Buku dapat melihat denda yang didapatkan

#### **4.7.2. Kebutuhan Non Fungsional**

Kebutuhan non fungsional adalah kebutuhan yang menitikberatkan pada properti perilaku yang dimiliki oleh sistem. kebutuhan fungsional juga sering disebut sebagai batasan layanan atau fungsi yang ditawarkan sistem. Kebutuhan non fungsional dapat dilihat pada tabel 4.7-2.

**Tabel 4.7-2. Kebutuhan Non Fungsional**

No.	Kebutuhan	Deskripsi Kebutuhan
NF01	<i>Availability</i>	Bisa diakses selama waktu aktif Ruang Baca Teknik Informatika (7.30 – 16.00) pada hari aktif Ruang Baca Teknik Informatika (senin – jumat).
NF02	<i>Security</i>	Autentikasi NRP pada setiap pengguna.

## BAB V IMPLEMENTASI SISTEM

### 5.1. Implementasi Lapisan Kontrol

Lapisan kontrol merupakan lapisan yang berfungsi untuk menjembatani model data, sehingga data-data yang diambil dari basis data bisa tertata dan ditampilkan pada antarmuka pengguna. Tabel 5.1-1 memaparkan bagian lapisan kontrol beserta fungsinya.

**Tabel 5.1-1. Implementasi Lapisan Kontrol**

<b>Nama Kontrol</b>	<b>Nama Fungsi</b>	<b>Deskripsi</b>
Halaman	Login	Menangani proses autentikasi pengguna dan menampilkan halaman login.
	Logout	Menangani proses logout.
	Peminjaman	Menangani proses peminjaman buku dan menampilkan halaman peminjaman.
	Pengembalian	Menangani proses pengembalian buku dan menampilkan halaman pengembalian.
	Reservasi	Menangani proses reservasi buku dan menampilkan halaman reservasi.
	Denda	Menghitung total denda pengguna dan menampilkan halaman pengguna.

## 5.2. Implementasi Lapisan Data

### 5.2.1. Basis Data

Lapisan data untuk mengoperasikan perangkat lunak didukung oleh MySQL. Basis data ini menyimpan informasi-informasi yang dibutuhkan oleh perangkat lunak agar bisa menjalankan fungsinya. Penjelasan mengenai informasi-informasi yang dibutuhkan aplikasi beserta contoh atribut dalam setiap tabel dalam basis data aplikasi tertera pada deskripsi berikut:

1. Informasi : Pengguna Aplikasi  
Tabel : *member*

**Tabel 5.2-1. Tabel Member**

Attribut	Keterangan
member_id	ID pengguna
member_name	Nama pengguna
gender	Jenis kelamin pengguna
birth_date	Tanggal lahir pengguna
member_address	Alamat pengguna
member_email	E-mail pengguna
member_phone	No. Telepon pengguna
register_date	Tanggal register
mpasswd	Password pengguna
last_login	Terakhir kali login
last_login_ip	IP address terakhir kali login
last_update	Terakhir kali update data

Tabel *member* berguna untuk menyimpan data-data pengguna yang terdaftar pada Ruang Baca Teknik Informatika ITS. Daftar atribut beserta keterangan pada tabel *member* dapat dilihat pada tabel 5.2-1.

2. Informasi : Peminjaman  
Tabel : *loan*

Tabel *loan* berguna untuk menyimpan data-data peminjaman pada Ruang Baca Teknik Informatika ITS. Daftar atribut beserta keterangan pada tabel *loan* dapat dilihat pada tabel 5.2-2.

**Tabel 5.2-2. Tabel Loan**

Attribut	Keterangan
loan_id	ID peminjaman
item_code	Kode barang
member_id	ID pengguna
loan_date	Tanggal peminjaman
due_date	Tanggal terakhir pengembalian
is_lent	Sedang dipinjamkan (Ya/Tidak)
is_return	Sudah dikembalikan (Sudah/Belum)
return_date	Tanggal pengembalian
input_date	Tanggal memasukkan data
last_update	Terakhir kali update data

3. Informasi : Barang  
Tabel : *item*

Tabel *item* berguna untuk menyimpan data-data barang pada Ruang Baca Teknik Informatika ITS. Daftar atribut beserta keterangan pada tabel *item* dapat dilihat pada tabel 5.2-3.

**Tabel 5.2-3. Tabel Item**

Attribut	Keterangan
loan_id	ID peminjaman
item_code	Kode barang
member_id	ID pengguna

loan_date	Tanggal peminjaman
due_date	Tanggal terakhir pengembalian
is_lent	Sedang dipinjamkan (Ya/Tidak)
is_return	Sudah dikembalikan (Sudah/Belum)
return_date	Tanggal pengembalian
last_update	Terakhir kali update data

4. Informasi : Pemesanan  
Tabel : *reserve*

Tabel *reserve* berguna untuk menyimpan data-data pemesanan peminjaman pada Ruang Baca Teknik Informatika ITS. Daftar atribut beserta keterangan pada tabel *reserve* dapat dilihat pada tabel 5.2-4.

**Tabel 5.2-4. Tabel Reserve**

Attribut	Keterangan
reserve_id	ID pemesanan
member_id	ID pengguna
item_code	Kode barang
reserve_date	Tanggal pemesanan

5. Informasi : Denda  
Tabel : *fines*

Tabel *fines* berguna untuk menyimpan data-data denda pengguna di Ruang Baca Teknik Informatika ITS. Daftar atribut beserta keterangan pada tabel *fines* dapat dilihat pada tabel 5.2-5.

Tabel 5.2-5. Tabel Fines

Attribut	Keterangan
fines_id	ID denda
fines_date	Tanggal denda
member_id	ID pengguna
debit	Pemasukan
credit	Pengeluaran
description	Keterangan pembayaran denda

### 5.3. Implementasi Program

Pada bagian ini akan dijelaskan kode program yang digunakan untuk membuat aplikasi self services.

#### 5.3.1. Kode Sumber Navigasi

Berikut ini kode sumber yang berfungsi untuk menambahkan menu navigasi yang mengarahkan pengguna untuk dapat masuk ke aplikasi *self services*. Tautan untuk menuju aplikasi *self services* dapat dilihat pada baris 10.

```

1. <nav class="s-menu-content animated-fast" role="navigation">
2. <a href="#" id="hide-menu" class="s-menu-
   toggle"><span></span></a>
3. <h1>Menu</h1>
4. <ul>
5.     <li><a href="index.php"><?php echo __('Home'); ?></a></li>
6.     <li><a href="index.php?p=news"><?php echo __('Library
   News'); ?></a></li>
7.     <li><a href="index.php?p=libinfo"><?php echo __('Library
   Information'); ?></a></li>
8.     <li><a href="index.php?p=member"><?php echo __('Member
   Area'); ?></a></li>
9.     <li><a href="index.php?p=login"><?php echo __('Librarian
   LOGIN'); ?></a></li>
10.    <li><a href="index.php?p=login2"><?php echo __('Self
   Services Peminjaman Buku'); ?></a></li>

```

```

11.     <li><a href="http://yudisium.if.its.ac.id"><?php echo
      __('Yudisium Teknik Informatika ITS'); ?></a></li>
12. </ul>
13.
14. <div class="s-menu-info">
15. <form class="language" name="langSelect" action="index.php"
      method="get">
16.     <label class="language-info" for="select_lang"><?php echo
      __('Select Language'); ?></label>
17.     <span class="custom-dropdown custom-dropdown--emerald
      custom-dropdown--small">
18.         <select name="select_lang" id="select_lang"
      title="Change language of this site"
      onchange="document.langSelect.submit();" class="custom-
      dropdown__select custom-dropdown__select--emerald">
19.             <?php echo $language_select; ?>
20.         </select>
21.     </span>
22. </form>
23. </div>
24. </nav>

```

### 5.3.2. Kode Sumber Koneksi

Berikut ini kode sumber yang berfungsi untuk melakukan koneksi database *mysql* pada aplikasi *self services*.

```

1. <?php
2. $servername = "localhost";
3. $username = "root";
4. $password = "";
5. $dbname = "sql_rbt";
6. // Create connection
7. global $conn;
8. $conn = new mysqli($servername, $username, $password,
  $dbname);
9. // Check connection
10. if ($conn->connect_error) {
11. die("Connection failed: " . $conn->connect_error);
12. }
13. ?>

```



### 5.3.4. Kode Sumber Login Aplikasi

Berikut ini kode sumber yang berfungsi untuk menampilkan formulir login bagi pengguna yang akan menggunakan aplikasi.

```
1. <?php
2. require 'template/connect.php';
3. if($_SERVER['REQUEST_METHOD'] == 'POST')
4. {
5.     session_start();
6.     $NRP=$_POST['nrp'];
7.     $result=$conn->query("SELECT member_id FROM member");
8.     if(!$result) die(mysql_error());
9.     if($result->num_rows > 0)
10.    {
11.        while($row = $result->fetch_assoc())
12.        {
13.            if($NRP==$row['member_id'])
14.            {
15.                $_SESSION['nrp'] = $NRP;
16.                header("location: pinjam.php");
17.                exit();
18.            }
19.        }
20.        echo "<script>alert('NRP belum terdaftar /
    salah!')</script>";
21.    }
22.    session_destroy();
23. }
24. mysqli_close($conn);
25. ?>
26.
27. <!doctype html>
28. <html>
29. <head>
30. <?php include 'template/include.php';?>
31. </head>
32. <body>
33. <?php include 'template/header.php';?>
34. <div class="container">
```

```

35. <div class="row">
36.     <div class="col-xs-4 col-xs-offset-4 text-center">
37.     <div class="form-top">
38.         <h3 class="text-center">Masukkan
         <b>NRP</b></h3>
39.     </div>
40.     <div class="form-bottom">
41.         <form role="form" action="index.php"
         method="post">
42.             <div class="form-group">
43.                 <input type="number" name="nrp"
         class="form-control" autofocus required>
44.             </div>
45.             <button type="submit" name="send"
         class="btn btn-primary">Next</button>
46.         </form>
47.     </div>
48. </div>
49. </div>
50. </div>
51. </div>
52. </body>
53. </html>

```

### 5.3.5. Kode Sumber Fitur Peminjaman

Berikut ini kode sumber yang berfungsi untuk menampilkan fitur peminjaman buku yang dilihat oleh pengguna.

```

1. <?php
2. require 'template/connect.php';
3. session_start();
4. if (isset($_session['nrp']) == false)
5. {
6.     header("location: index.php");
7.     exit();
8. }
9. else
10. {
11.     $nrp=$_session['nrp'];

```

```

12.     $result=$conn->query("select * from member");
13.     if (!$result) die(mysql_error());
14.     mysqli_close($conn);
15. }
16. ?>
17.
18. <!doctype html>
19. <html>
20. <head>
21. <?php include 'template/include.php';?>
22. </head>
23. <body>
24. <?php
25. $active_page = 'pinjam';
26. include 'template/header.php';
27. ?>
28. <div class="container">
29. <div class="row">
30. <div class="col-xs-4 col-xs-offset-4 text-center">
31.     <div class="form-top">
32.         <h3 class="text-center">masukkan <b>book
code/barcode</b></h3>
33.     </div>
34.     <div class="form-bottom">
35.         <form role="form" action="pinjam2.php" method="post">
36.             <div class="form-group">
37.                 <input type="text" name="code" class="form-control"
aoutofocus required>
38.             </div>
39.             <button type="submit" name="send" class="btn btn-
primary">next</button>
40.         </form>
41.     </div>
42. </div>
43. </div>
44. </div>
45. </div>
46. </body>
47. </html>

```

Berikut ini kode sumber yang berfungsi untuk mengecek apakah buku yang ingin dipinjam oleh pengguna sedang dipinjam pengguna lain ataukah ada pengguna yang memesan buku tersebut atau buku yang ingin dipinjam tidak ada di ruang baca, kode sumber ini memberikan keluaran apakah peminjaman pengguna berhasil atau tidak.

```

48. <?php
49. $message = "";
50. require 'template/connect.php';
51. session_start();
52. $NRP=$_SESSION['nrp'];
53. $kode=$_POST['code'];
54.
55. $flag=0; $flag2=0;$flag3=0;$x=0;
56. $dt = date("Y-m-d");
57. $dt2= date( "Y-m-d", strtotime("+1 week"));
58. $dt3=date( "Y-m-d", strtotime("-1 week"));
59. $flag4= array();
60. while ( $x<= 90000) {
61.     $flag4[$x]=0;
62.     $x++;
63. }
64.
65. $result3=$conn->query("SELECT * FROM loan");
66. if ($result3->num_rows > 0)
67. {
68.     while($row3 = $result3->fetch_assoc())
69.     {
70.         if($kode==$row3['item_code'])
71.         {
72.             $flag4[$x-1]=$row3['loan_id'];
73.         }
74.         $x--;
75.         if($x<=0){
76.             break;
77.         }
78.     }
79.     while($x<=90000)

```

```

80.     {
81.         if($flag4[$x]>$flag3)
82.             {
83.                 $flag3=$flag4[$x];
84.             }
85.         $x++;
86.     }
87. }
88.
89. $result=$conn->query("SELECT item_code FROM item");
90. if ($result->num_rows > 0)
91.     {
92.         while($row = $result->fetch_assoc())
93.             {
94.                 if($kode==$row['item_code'])
95.                     {
96.
97.                         $result2=$conn->query("SELECT * FROM reserve");
98.                         if ($result2->num_rows > 0)
99.                             {
100.                                while($row2 = $result2->fetch_assoc())
101.                                    {
102.                                        if($kode==$row2['item_code'] &&
$NRP==$row2['member_id'])
103.                                            {
104.                                                $sql = "delete from reserve where
member_id='$NRP' and item_code='$kode'";
105.                                                if ($conn->query($sql) === TRUE)
106.                                                    {
107.                                                        $message = "reserve deleted";
108.                                                        $sql2 = "INSERT INTO loan
(member_id,item_code, loan_date, due_date,is_lent) VALUES
('$NRP', '$kode', '$dt', '$dt2',1)";
109.                                                        if ($conn->query($sql2) === TRUE)
110.                                                            {
111.                                                                $flag=0;
112.                                                                $message = "peminjaman berhasil";
113.                                                            }
114.                                                        break;

```

```

115.         }
116.     }
117.     else if($kode==$row2['item_code'])
118.     {
119.         $tgl_pesan = strtotime($row2['reserve_date']);
120.         $tgl_batas = strtotime($dt3);
121.         if($tgl_batas>$tgl_pesan)
122.         {
123.             $sql = "delete from reserve where
reserve_date='$tgl_pesan' and item_code='$kode'";
124.             if ($conn->query($sql) === TRUE)
125.             {
126.                 $sql2 = "INSERT INTO loan
(member_id,item_code, loan_date, due_date,is_lent) VALUES
('$NRP', '$kode', '$dt', '$dt2',1)";
127.                 if ($conn->query($sql2) === TRUE)
128.                 {
129.                     $flag=0;
130.                     $message = "peminjaman berhasil";
131.                 }
132.                 break;
133.             }
134.         }
135.     else
136.     {
137.         $flag=3;
138.     }
139. }
140. else $flag2=1;
141. }//eow
142. if($flag2==1)
143. {
144.     $result3=$conn->query("SELECT * FROM loan");
145.     if ($result3->num_rows > 0)
146.     {
147.         while($row3 = $result3->fetch_assoc())
148.         {
149.             if($kode==$row3['item_code'] &&
$row3['is_return']==1 && $row3['loan_id']==$flag3)

```

```

150.         {
151.             $sql3 = "INSERT INTO loan (item_code,
member_id, loan_date, due_date, is_lent) VALUES ( '$kode', '$NRP',
'$dt', '$dt2', 1)";
152.             if ($conn->query($sql3) === TRUE)
153.                 {
154.                     $flag=0;
155.                     $message = "peminjaman berhasil";
156.                     break;
157.                 }
158.             }
159.             else
160.                 {
161.                     $flag=2;
162.                 }
163.             }
164.         }
165.     }
166.     break;
167. }
168. else
169. {
170.     $result3=$conn->query("SELECT * FROM loan");
171.     if ($result3->num_rows > 0)
172.     {
173.         while($row3 = $result3->fetch_assoc())
174.         {
175.
176.             $ret=$row3['is_return'];
177.             if($kode==$row3['item_code'] &&
$row3['is_return']==1 && $row3['loan_id']==$flag3)
178.                 {
179.                     $sql3 = "INSERT INTO loan (item_code,
member_id, loan_date, due_date, is_lent) VALUES ( '$kode', '$NRP',
'$dt', '$dt2', 1)";
180.                     if ($conn->query($sql3) === TRUE)
181.                         {
182.                             $flag=0;
183.                             $message = "peminjaman berhasil";

```

```

184.                 break;
185.                 }
186.             }
187.         else
188.         {
189.             $flag=2;
190.         }
191.     }
192. }
193. }
194.     break;
195. }
196. else
197. {
198.     $flag=1;
199. }
200. }//eow
201. }
202. else
203. {
204.     $flag=1;
205. }
206.
207. if($flag==1)
208. {
209.     $message = "buku yang anda inginkan tidak ada."<br>;
210. }
211. else if($flag==2)
212. {
213.     $message = "buku sedang dipinjam<br>";
214. }
215. else if($flag==3)
216. {
217.     $message = "buku sedang dipeservasi oleh orang lain<br>";
218. }
219. ?>
220.
221. <!doctype html>
222. <html>

```

```

223. <head>
224. <?php include 'template/include.php';?>
225. </head>
226. <body>
227. <?php
228. $active_page = 'pinjam';
229. include 'template/header.php';
230. ?>
231. <div class="container">
232. <h1><?php echo $message; ?></h1>
233. <h3>Kembali ke <a href='pinjam.php'>menu utama</a></h3>
234. </div>
235. </div>
236. </body>
237. </html>

```

### 5.3.6. Kode Sumber Fitur Pemesanan

Berikut ini kode sumber yang berfungsi untuk menampilkan fitur pemesanan buku yang dilihat oleh pengguna.

```

1. <?php
2. require 'template/connect.php';
3. session_start();
4. if (isset($_SESSION['nrp']) == false)
5. {
6.     header("location: index.php");
7.     exit();
8. }
9. else
10. {
11.     $NRP=$_SESSION['nrp'];
12.     $result=$conn->query("SELECT * FROM member");
13.     if (!$result) die(mysql_error());
14.     mysqli_close($conn);
15. }
16. ?>
17. <!doctype html>
18. <html>
19. <head>
20. <?php include 'template/include.php';?>

```

```

21. </head>
22. <body>
23. <?php
24. $active_page = 'reservasi';
25. include 'template/header.php';
26. ?>
27. <div class="container">
28. <div class="row">
29. <div class="col-xs-4 col-xs-offset-4 text-center">
30. <div class="form-top">
31. <h3 class="text-center">Masukkan <b>Book
    Code/Barcode</b></h3>
32. </div>
33. <div class="form-bottom">
34. <form role="form" action="reservasi2.php" method="post">
35. <div class="form-group">
36. <input type="text" name="code" class="form-control"
    autofocus required>
37. </div>
38. <button type="submit" name="send" class="btn btn-
    primary">Next</button>
39. </form>
40. </div>
41. </div>
42. </div>
43. </div>
44. </div>
45. </body>
46. </html>

```

Berikut ini kode sumber yang berfungsi untuk mengecek apakah buku yang ingin dipesan oleh pengguna sedang dipesan pengguna lain atau buku tersebut sedang dipinjam oleh pengguna lain ataukah buku yang ingin dipinjam tidak ada di ruang baca, kode sumber ini memberikan keluaran apakah pemesanan pengguna berhasil atau tidak.

```

47. <?php
48. $message = "";
49. require 'template/connect.php';

```

```

50. session_start();
51. $NRP=$_SESSION['nrp'];
52. $kode=$_POST['code'];
53. $flag=0; $flag2=0;$flag3=0;$x=0;
54. $dt=date('Y-m-d');
55. $flag4= array();
56. while ( $x<= 90000) {
57.     $flag4[$x]=0;
58.     $x++;
59. }
60. $result4=$conn->query("SELECT * FROM item");
61. if ($result4->num_rows > 0)
62. {
63.     while($row4 = $result4->fetch_assoc())
64.     {
65.         if($kode==$row4['item_code'])
66.         {
67.             $asd=$row4['biblio_id'];
68.         }
69.     }
70. }
71.
72. $result3=$conn->query("SELECT * FROM loan");
73. if ($result3->num_rows > 0)
74. {
75.     while($row3 = $result3->fetch_assoc())
76.     {
77.         if($kode==$row3['item_code'])
78.         {
79.             $flag4[$x-1]=$row3['loan_id'];
80.         }
81.         $x--;
82.         if($x<=0){
83.             break;
84.         }
85.     }
86.     while($x<=90000)
87.     {
88.         if($flag4[$x]>$flag3)

```

```

89.     {
90.         $flag3=$flag4[$x];
91.     }
92.     $x++;
93. }
94. }
95.
96. $result=$conn->query("SELECT * FROM loan");
97. if ($result->num_rows > 0) {
98.     // output data of each row
99.     while($row = $result->fetch_assoc())
100.    {
101.        if($kode==$row['item_code']&&$row['is_return']==1&&
102.        $row['loan_id']==$flag3)
103.        {
104.            $result2=$conn->query("SELECT * FROM reserve");
105.            if ($result2->num_rows > 0)
106.            {
107.                // output data of each row
108.                while($row2 = $result2->fetch_assoc())
109.                {
110.                    if($kode=$row2['item_code'])
111.                    {
112.                        $flag=0;
113.                        $message = "buku sudah direservasi<br>";
114.                    }
115.                    else
116.                    {
117.                        $sql = "INSERT INTO reserve (member_id, biblio_id,
118.                        item_code, reserve_date) VALUES ('$NRP', '$asd', '$kode', '$dt)";
119.                        if ($conn->query($sql) === TRUE)
120.                        {
121.                            $flag=0;
122.                            $message = "Reservasi Berhasil<br>";
123.                        }
124.                    }
125.                }
126.            }
127.            break;
128.        }
129.    }
130.    break;

```

```

126.     }
127.     else{
128.         $sql = "INSERT INTO reserve (member_id, biblio_id,
item_code, reserve_date) VALUES ('$NRP', '$asd', '$kode', '$dt)";
129.         if ($conn->query($sql) === TRUE)
130.             {
131.                 $flag=0;
132.                 $message = "Reservasi Berhasil<br>";
133.                 break;
134.             }
135.         }
136.     }
137.     else $flag=1;
138. }
139. }
140. if($flag==1)
141. {
142.     $message = "buku sedang dipinjam<br>";
143. }
144. ?>
145.
146. <!doctype html>
147. <html>
148. <head>
149. <?php include 'template/include.php';?>
150. </head>
151. <body>
152. <?php
153. $active_page = 'reservasi';
154. include 'template/header.php';
155. ?>
156. <div class="container">
157. <h1><?php echo $message; ?></h1>
158. <h3>Kembali ke <a href='pinjam.php'>menu utama</a></h3>
159. </div>
160. </div>
161. </body>
162. </html>

```

### 5.3.7. Kode Sumber Fitur Pengembalian

Berikut ini kode sumber yang berfungsi untuk menampilkan fitur pengembalian buku yang dilihat oleh pengguna.

```
1. <?php
2. require 'template/connect.php';
3. session_start();
4. if (isset($_SESSION['nrp']) == false)
5. {
6.     header("location: index.php");
7.     exit();
8. }
9. else
10. {
11.     $NRP=$_SESSION['nrp'];
12.     $result=$conn->query("SELECT * FROM member");
13.     if (!$result) die(mysql_error());
14.     //echo "welcome ". $NRP;
15.     mysqli_close($conn);
16. }
17. ?>
18. <!doctype html>
19. <html>
20. <head>
21. <?php include 'template/include.php';?>
22. </head>
23. <body>
24. <?php
25.     $active_page = 'kembali';
26.     include 'template/header.php';
27.     ?>
28. <div class="container">
29.     <div class="row">
30.         <div class="col-xs-4 col-xs-offset-4 text-center">
31.             <div class="form-top">
32.                 <h3 class="text-center">Masukkan <b>Book
Code/Barcode</b></h3>
33.             </div>
34.         <div class="form-bottom">
```

```

35.     <form role="form" action="kembali2.php" method="post">
36.     <div class="form-group">
37.         <input type="text" name="code" class="form-control"
aoutofocus required>
38.         </div>
39.         <button type="submit" name="send" class="btn btn-
primary">Next</button>
40.     </form>
41. </div>
42. </div>
43. </div>
44. </div>
45. </div>
46. </body>
47. </html>

```

Berikut ini kode sumber yang berfungsi untuk mengecek apakah buku yang ingin dikembalikan oleh pengguna sudah dikembalikan oleh pengguna tersebut atau belum, kode sumber ini memberikan keluaran apakah pengembalian pengguna berhasil atau tidak.

```

48. <?php
49. $message = "";
50. require 'template/connect.php';
51. session_start();
52. $NRP=$_SESSION['nrp'];
53. $kode=$_POST['code'];
54. $dt=date('Y-m-d');
55.
56. $flag=0;$flag3=0;$x=0;
57. $flag4= array();
58. while ( $x<= 90000)
59. {
60.     $flag4[$x]=0;
61.     $x++;
62. }
63.
64. $result3=$conn->query("SELECT * FROM loan");
65. if ($result3->num_rows > 0)

```

```

66.     {
67.     while($row3 = $result3->fetch_assoc())
68.     {
69.         if($kode==$row3['item_code'])
70.         {
71.             $flag4[$x-1]=$row3['loan_id'];
72.         }
73.         $x--;
74.         if($x<=0)
75.         {
76.             break;
77.         }
78.     }
79.     while($x<=90000)
80.     {
81.         if($flag4[$x]>$flag3)
82.         {
83.             $flag3=$flag4[$x];
84.         }
85.         $x++;
86.     }
87. }
88. $result=$conn->query("SELECT * FROM loan");
89. if ($result->num_rows > 0)
90. {
91.     // output data of each row
92.     while($row = $result->fetch_assoc())
93.     {
94.         if($kode==$row['item_code']&& $row['is_return']=0&&
95. $row['loan_id']==$flag3)
96.         {
97.             $sql = "UPDATE loan SET is_return=1,return_date='$dt'
98. WHERE member_id='$NRP' and item_code='$kode' and
99. is_return='0'";
100.            if ($conn->query($sql) === TRUE) {
101.                $flag=0;
102.                $message = "buku berhasil dikembalikan";
103.                break;
104.            }

```

```

102.     }
103.     else {
104.         $flag=1;
105.     }
106.     }
107. }
108.
109. if($flag==1){
110.     $message = "buku sudah dikembalikan";}
111. ?>
112.
113. <!doctype html>
114. <html>
115. <head>
116. <?php include 'template/include.php';?>
117. </head>
118. <body>
119. <?php
120.     $active_page = 'kembali';
121.     include 'template/header.php';
122.     ?>
123.     <div class="container">
124.         <h1><?php echo $message; ?></h1>
125.         <h3>Kembali ke <a href='pinjam.php'>menu utama</a></h3>
126.     </div>
127. </div>
128. </body>
129. </html>

```

### 5.3.8. Kode Sumber Fitur Denda

Berikut ini kode sumber yang berfungsi untuk menampilkan denda yang dimiliki oleh pengguna.

```

1. <?php
2. $message = "";
3. require 'template/connect.php';
4. session_start();
5. $NRP=$_SESSION['nrp'];
6. $result=$conn->query("SELECT * FROM fines");
7. if (!$result) {

```

```

8.     die(mysql_error());
9. }
10. $dbt=0;
11. $krd=0;
12. if ($result->num_rows > 0) {
13.     // output data of each row
14.     while($row = $result->fetch_assoc())
15.     {
16.         if($NRP==$row['member_id'])
17.         {
18.             $dbt+=$row['debet'];
19.             $krd+=$row['credit'];
20.         }
21.     }
22. }
23. $tot=$dbt-$krd;
24. if($tot>0) $message = "Total denda anda sebesar: " . $tot . "
    rupiah<br>Silahkan bayar denda anda di petugas perpustakaan
    <br>";
25. else $message = "Anda tidak mempunyai denda <br>";
26. ?>
27.
28. <!doctype html>
29. <html>
30. <head>
31. <?php include 'template/include.php';?>
32. </head>
33. <body>
34. <?php
35.     $active_page = 'denda';
36.     include 'template/header.php';
37.     ?>
38.     <div class="container">
39.         <h1><?php echo $message; ?></h1>
40.         <h3>Kembali ke <a href='pinjam.php'>menu utama</a></h3>
41.     </div>
42. </div>
43. </body>
44. </html>

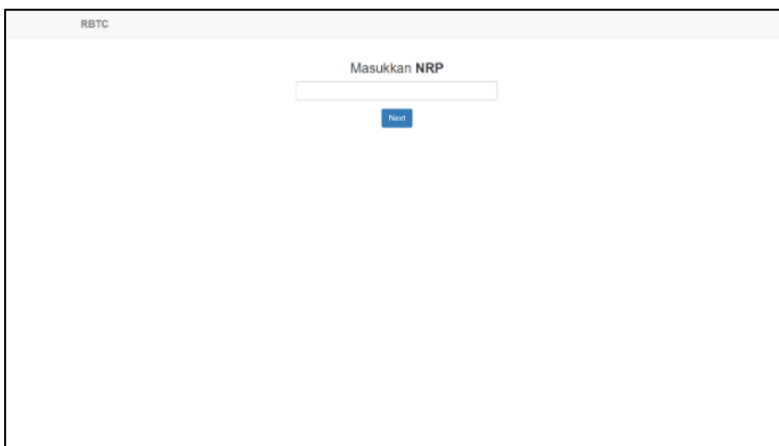
```

## 5.4. Implementasi Antarmuka

Terdapat 5 kategori implementasi tampilan antarmuka pengguna, diantaranya :

1. Halaman Login
2. Halaman Peminjaman Buku
3. Halaman Pengembalian Buku
4. Halaman Reservasi Buku
5. Halaman Lihat Denda

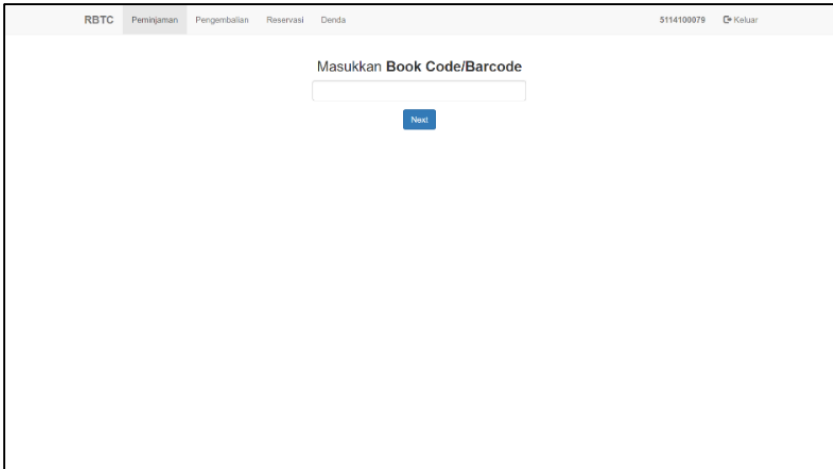
Untuk mengakses tampilan tersebut, pengguna harus memasukkan NRP yang valid sebagai autentikasi awal masuk ke tampilan peminjaman buku. Halaman login dapat dilihat pada gambar 5.3-1.



**Gambar 5.4-1. Halaman Login Pengguna**

### 1. Halaman Peminjaman Buku

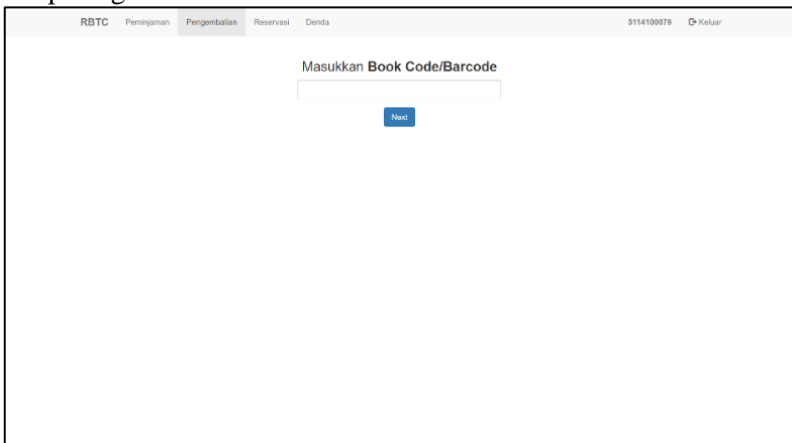
Halaman peminjaman buku digunakan oleh pengguna untuk meminjam buku. Halaman peminjaman buku dapat dilihat pada gambar 5.3-2.



**Gambar 5.4-2. Halaman Peminjaman Buku**

## 2. Halaman Pengembalian Buku

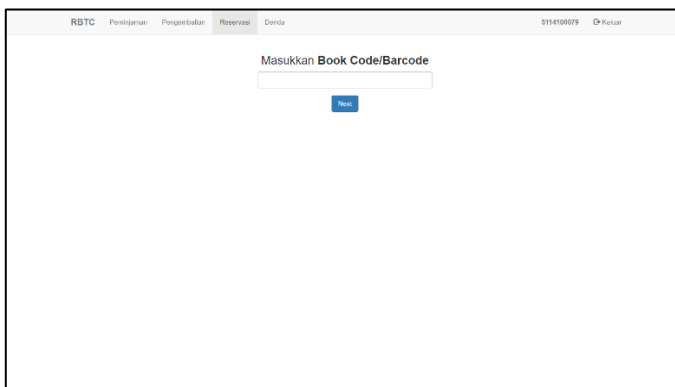
Halaman pengembalian buku digunakan oleh pengguna untuk mengembalikan buku. Halaman pengembalian buku dapat dilihat pada gambar 5.3-3.



**Gambar 5.4-3. Halaman Pengembalian Buku**

### 3. Halaman Reservasi Buku

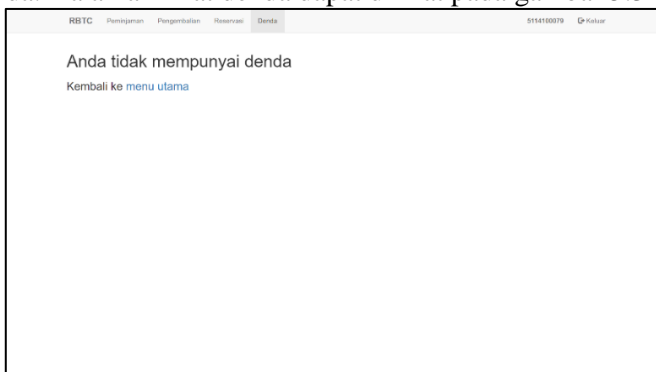
Halaman reservasi buku digunakan oleh pengguna untuk memesan buku yang ingin dipinjam. Halaman reservasi buku dapat dilihat pada gambar 5.3-4.



**Gambar 5.4-4. Halaman Reservasi Buku**

### 4. Halaman Lihat Denda

Halaman lihat denda digunakan oleh pengguna untuk melihat denda. Halaman lihat denda dapat dilihat pada gambar 5.3-5.



**Gambar 5.4-5. Halaman Lihat Denda**

## **BAB VI**

### **PENGUJIAN DAN EVALUASI**

#### **6.1. Lingkungan Pengujian**

Pengujian sistem pada pengerjaan kerja praktik ini dilakukan pada lingkungan dan alat kaku sebagai berikut:

##### **6.1.1. Server Aplikasi**

*Processor* : Intel® Core™ i5-7200U *Processor*  
(2.5 GHz, Cache 3 MB)  
*Memory* : 4.00 GB  
*Jenis Device* : Laptop  
*Sistem Operasi* : Microsoft Windows 8.1 64 bit  
*Browser* : Google Chrome Versi 62.0.3202.75  
*Web Server* : Apache

##### **6.1.2. Perangkat Uji Coba**

###### **Perangkat 1:**

*Processor* : Intel® Core™ i5-7200U *Processor*  
(2.5 GHz, Cache 3 MB)  
*Memory* : 4.00 GB  
*Jenis Device* : Laptop  
*Sistem Operasi* : Microsoft Windows 8.1 64 bit  
*Browser* : Google Chrome Versi 62.0.3202.75

#### **6.2. Skenario Pengujian**

Skenario pengujian aplikasi adalah sebagai berikut:

- Sebuah *server* penyedia *web service* dan aplikasi basis data ditanamkan pada alat kaku pengujian yang tertera pada poin 6.1.1. *Server* penyedia *web service* yang digunakan adalah Apache dengan database MySQL.

- Pengujian dilakukan dengan cara mengakses dan mencoba semua fitur aplikasi menggunakan *browser* Google Chrome pada alat perangkat uji coba.
- Aktivitas yang dilakukan pada saat pengujian, antara lain: meminjam buku, mengembalikan buku, mereservasi buku, dan melihat denda.

### 6.3. Evaluasi Pengujian

Pada subbab ini akan diberikan hasil evaluasi dari pengujian-pengujian yang telah dilakukan. Evaluasi yang diberikan meliputi evaluasi pengujian kebutuhan fungsional, evaluasi hasil pengelompokan, dan evaluasi kegunaan sistem.

#### 6.3.1. Evaluasi Pengujian Kebutuhan Fungsional

**Tabel 6.3-1. Pengujian Kebutuhan Fungsional**

No. Kebutuhan	Kebutuhan	Uji Coba	Status
F001	Meminjam Buku	Pengguna dapat meminjam buku yang belum direservasi atau tidak dalam keadaan sedang dipinjam .	Berhasil
F002	Mengembalikan Buku	Pengguna dapat mengembalikan buku yang telah dipinjam.	Berhasil
F003	Reservasi Buku	Pengguna dapat mereservasi buku yang belum dipinjam atau tidak dalam keadaan direservasi pengguna lainnya.	Berhasil

No. Kebutuhan	Kebutuhan	Uji Coba	Status
F004	Melihat Denda	Pengguna dapat melihat denda yang dimilikinya	Berhasil

### 6.3.2. Evaluasi Pengujian Kebutuhan Non Fungsional

Tabel 6.3-2. Pengujian Kebutuhan Non Fungsional

No.	Kebutuhan	Uji Coba	Status
NF01	<i>Availability</i>	Aplikasi dipasang pada server dan dibiarkan bekerja selama beberapa hari.	Belum bisa dipastikan karena pengujian hanya berlangsung pada perangkat pengembang, bukan pada server yang sesungguhnya.
NF02	<i>Security</i>	Pengguna harus menjalani proses autentikasi terlebih dahulu sebelum mengakses fitur-fitur yang ada. Jika pengguna langsung mengakses fitur melalui URL tanpa melakukan autentikasi, maka akan diarahkan ke halaman autentikasi.	Berhasil.

*[Halaman ini sengaja dikosongkan]*

## **BAB VII**

### **KESIMPULAN DAN SARAN**

#### **7.1. Kesimpulan**

Kesimpulan yang kami peroleh dari hasil kerja praktik ini, antara lain:

- Terdapat beberapa kelemahan pada rancangan sistem dari aplikasi yang sudah ada sebelumnya, antara lain: pengguna Ruang Baca Teknik Informatika ITS tidak dapat melakukan peminjaman, pengembalian, dan pemesanan buku secara mandiri atau harus diinputkan oleh pegawai/karyawan Ruang Baca Teknik Informatika ITS, serta pengguna Ruang Baca Teknik Informatika ITS belum dapat melihat berapa jumlah denda yang harus dibayarkan kepada ruang baca teknik informatika ITS secara mandiri.
- Tujuan dari kerja praktik ini tercapai karena telah berhasil menambahkan fitur-fitur baru yang sesuai dengan kebutuhan Ruang Baca Teknik Informatika ITS.
- Penambahan fitur baru yang telah dibuat pada aplikasi yang lama ini mempermudah pengguna dalam melakukan peminjaman, pengembalian, dan pemesanan buku di Ruang Baca Teknik Informatika ITS. Serta mempermudah pengguna dalam melakukan pengecekan denda.
- Fitur yang dibuat lebih mudah digunakan daripada hanya pegawai/karyawan saja yang dapat memasukkan data peminjaman, pengembalian, dan pemesanan buku oleh pengguna.

#### **7.2. Saran**

Penulis menyadari pentingnya keberadaan fitur baru yang telah dibuat ini, namun penulis menemukan beberapa hal yang kami rasa perlu untuk diperbaiki dan ditingkatkan, antara lain:

- Perlunya *maintenance* dan improvisasi berkala terhadap aplikasi yang ada saat ini, mengingat sistem kerja pada

perpustakaan seperti Ruang Baca Teknik Informatika ITS dapat dengan mudah berubah.

- *Barcode scanner* yang digunakan dalam pengenalan kode buku, terkadang tidak dapat mengenali kode seperti semestinya.
- Kedepannya disarankan dikembangkannya sistem ini dengan menggunakan *QR code* daripada menggunakan *barcode*. Dengan penggunaan *QR code*, pengenalan kode buku diharapkan akan lebih mudah dilakukan.
- Kekurangan-kekurangan yang penulis temui pada aplikasi sebelumnya sudah ditangani pada fitur yang telah ditambahkan. Fitur tersebut ingin digunakan, akan lebih baik jika dilakukan uji coba terlebih dahulu selama minimal satu minggu untuk memastikan aplikasi baru tersebut benar-benar memenuhi semua kebutuhan.

## DAFTAR PUSTAKA

- [1] RBTC, "RBTC | Ruang Baca Jurusan Teknik Informatika - ITS :: OPAC," [Online]. Available: <http://rbtc.if.its.ac.id/v4/>. [Accessed 29 Oktober 2017].
- [2] SLiMS, "SLiMS - Senayan Library Management System - About SLiMS," [Online]. Available: <https://slims.web.id/web/pages/about/>. [Accessed 29 Oktober 2017].
- [3] Twitter, "Bootstrap - The most popular HTML, CSS, and JS library in the world.," [Online]. Available: <http://getbootstrap.com/>. [Accessed 29 Oktober 2017].
- [4] AKHMAD, "Pengertian Bootstrap, fungsi dan juga kelebihannya - Galeri Programmer," 22 August 2016. [Online]. Available: <http://galeriprogramer.blogspot.co.id/2016/08/pengertian-bootstrap-fungsi-dan-juga.html>. [Accessed 12 December 2017].
- [5] Arjayay, "Management System - Wikipedia," Wikipedia Foundation, 25 November 2017. [Online]. Available: [https://en.wikipedia.org/wiki/Management\\_system](https://en.wikipedia.org/wiki/Management_system). [Accessed 5 December 2017].

*[Halaman ini sengaja dikosongkan]*

## BIODATA PENULIS



Aenuar Purnomo, lahir pada tanggal 14 Januari 1997 di Pasuruan. Penulis merupakan seorang mahasiswa yang sedang menempuh studi di Departemen Teknik Informatika Institut Teknologi Sepuluh Nopember. Memiliki beberapa hobi antara lain membaca komik dan tidur. Selama menempuh pendidikan di kampus, penulis juga aktif dalam organisasi kemahasiswaan, antara lain Koordinator Departemen Kerohanian Tim Pembina Kerohanian Buddha ITS pada tahun ke-2, serta Ketua Harian Tim Pembina Kerohanian Buddha ITS pada tahun ke-3.



Alam Ar Raad Stone, lahir pada tanggal 9 Maret 1997 di Pacitan. Penulis merupakan seorang mahasiswa yang sedang menempuh studi di Departemen Teknik Informatika Institut Teknologi Sepuluh Nopember. Memiliki ketertarikan pada *Computer Vision* dan *Machine Learning*. Selama menempuh pendidikan di kampus, penulis aktif sebagai programmer tim Robot Sepak Bola Beroda ITS.