

IMPLEMENTASI SISTEM TERINTEGRASI

(SISTEM GALERI BUKU & SISTEM PERPUSTAKAAN)

KATA PENGANTAR

Puji syukur penulis ucapkan atas kepada Allah SWT, atas berkat dan nikmat-Nya yang telah dilimpahkan kepada penulis selama ini. Syukur alhamdulillah dengan selesainya buku ini. Terima kasih atas semua dukungan dari pembimbing, penguji, dan rekan-rekan penulis yang selama ini telah berkontribusi serta mendukung dalam proses penulisan sampai dengan buku ini dapat diselesaikan dengan tepat waktu.

Buku ini merupakan hasil dari penggerjaan proyek III yang telah penulis kerjakan selama menempuh kuliah Semester 6 di Politeknik Pos Indonesia. Isi dari buku ini terdapat teori-teori serta panduan sederhana membuat sebuah aplikasi atau sistem yang dapat terintegrasi, didalamnya juga sudah terdapat beberapa *syntax* untuk membuat sebuah tabel atau *form* aplikasi.

Dengan adanya buku ini penulis berharap bisa membantu bagi para pembaca. Buku ini baik secara konteks maupun konten tentu kedepannya masih perlu perbaikan secara sempurna. Oleh karena itu, penulis membuka untuk saran dan kritik semoga buku ini bisa lakukan perbaikan dan dikembangkan dengan hal-hal baru.

Bandung, 2 Maret 2020

Penulis

DAFTAR ISI

KATA PENGANTAR.....	v
DAFTAR ISI	vi
BAGIAN 1 ANALISIS PERMASALAHAN.....	1
BAGIAN 2 TEORI PERPUSTAKAAN DIGITAL DAN INTEGRASI SISTEM.....	4
2.1 Definisi Perpustakaan.....	4
2.2 Definisi Integrasi Sistem	5
BAGIAN 3 KONSEP DASAR SISTEM INFORMASI DAN <i>WEB SERVICE</i>	6
3.1 Definisi Sistem.....	6
3.1.1 Karakteristik Sistem.....	7
3.1.2 Klasifikasi Sistem	9
3.2 Definisi Sistem Informasi.....	10
3.2.1 Kualitas Informasi.....	11
3.3 <i>Web Service</i>	11
BAGIAN 4 PENGENALAN DBMS DAN MYSQL	13
4.1 Tentang <i>Database</i> , DBMS, dan RDBMS	13
4.1.1 Definisi <i>Database</i>	13
4.1.2 Definisi DBMS (<i>Database Management System</i>)	15
4.2 Tentang MySQL	16
4.3 Instalasi MySQL di Windows	17
BAGIAN 5 PEMODELAN PADA SISTEM.....	22
5.1 <i>Unified Modeling Language</i> (UML).....	22
5.1.1 Representasi Arsitektur	22
5.1.2 Arsitektur Perangkat Lunak Dikembangkan Menggunakan UML Memberikan Manfaat Berikut	28
5.1.3 Kelemahan UML	30

BAGIAN 6 TATA CARA INSTALASI DAN PENGGUNAAN <i>TOOLS</i> YANG DIGUNAKAN	32
6.1 Xampp	32
6.1.1 Cara <i>Install</i> Xampp	32
6.1.2 Cara Menjalankan Aplikasi Xampp.....	37
6.2 <i>Sublime</i>	39
6.3 <i>Visual Code</i>	43
BAGIAN 7 TEORI <i>FRAMEWORK</i> DAN BAHASA YANG DAPAT DIGUNAKAN	48
7.1 <i>Framework</i> <i>CodeIgniter</i>	48
7.1.1 Pengertian <i>Framework</i>	48
7.1.2 Pengertian <i>CodeIgniter</i>	49
7.1.3 MVC (<i>Model View Controller</i>)	49
7.1.4 Cara Menggunakan <i>CodeIgniter</i>	50
7.2 Bahasa Pemrograman HTML Dan PHP.....	53
7.2.1 Definisi HTML (<i>Hyper Text Markup Language</i>).....	53
7.2.2 Definisi PHP (<i>Hypertext Preprocessor</i>).....	53
BAGIAN 8 PERANCANGAN <i>USER INTERFACE</i> DARI SISTEM YANG DIBANGUN	55
8.1 Pengertian Antar Muka Pengguna (<i>User Interface</i>)	55
8.2 Perancangan <i>User Interface</i>	57
8.2.1 <i>User Interface</i> Galeri Buku Registrasi.....	57
8.2.2 <i>User Interface</i> Galeri Buku <i>Login</i>	58
8.2.3 <i>User Interface</i> Galeri Buku Peminjaman.....	59
8.2.4 <i>User Interface</i> Galeri Buku Pengembalian	60
8.2.5 <i>User Interface</i> Galeri Buku <i>Upload</i> Bukti Kirim Kembali.....	61
8.2.6 <i>User Interface</i> Galeri Buku <i>Upload</i> Buku.....	62
8.2.7 <i>User Interface</i> Galeri Buku <i>View</i> Peminjaman	63
8.2.8 <i>User Interface</i> Galeri Buku <i>View</i> Pengembalian.....	64

8.2.9	<i>User Interface Upload</i> Bukti Kirim Peminjaman.....	65
8.2.10	<i>User Interface</i> Galeri Buku Kelola Data <i>User</i>	66
8.2.11	<i>User Interface</i> Galeri Buku Kelola Data Peminjaman.....	67
8.2.12	<i>User Interface</i> Galeri Buku Kelola Data Pengembalian	68
8.2.13	<i>User Interface</i> Galeri Buku Kelola Data Buku	69
8.2.14	<i>User Interface</i> Galeri Buku Konfirmasi Bukti Kirim	70
8.2.15	<i>User Interface</i> Perpustakaan Data Buku	71
8.2.16	<i>User Interface</i> Perpustakaan Data Anggota	72
8.2.17	<i>User Interface</i> Perpustakaan Data <i>User</i>	73
8.2.18	<i>User Interface</i> Perpustakaan Kategori Buku	74
8.2.19	<i>User Interface</i> Perpustakaan Peminjaman	75
BAGIAN 9	PENULISAN SYNTAX PADA SISTEM YANG TELAH	
DIRANCANG		77
9.1	Membuat Form <i>Login</i>	77
9.1.1	Membuat <i>Controller F_login.php</i>	77
9.1.2	Membuat <i>View signin.php</i>	80
9.1.3	Membuat Model <i>Auth_model.php</i>	84
9.1.4	Membuat Model <i>Crud_model.php</i>	85
9.2	Membuat Form Registrasi	86
9.2.1	Membuat <i>Controller F_register.php</i>	87
9.2.2	Membuat <i>View register.php</i>	101
9.2.3	Membuat Model <i>register_model.php</i>	124
9.3	Membuat Proses Pada Bagian Pemilik	125
9.3.1	Membuat <i>Controller F_owner.php</i>	125
9.3.2	Membuat Model <i>loan_model.php</i>	139
9.3.3	Membuat <i>View invest_buku.php</i>	140
9.3.4	Membuat Model <i>invest_book.php</i>	146
9.3.5	Membuat <i>View cek_peminjaman.php</i>	153

9.3.6	Membuat <i>View cek_pengembalian.php</i>	159
9.4	Membuat Proses Pada Bagian Peminjam.....	165
9.4.1	Membuat <i>Controller F_Borrower.php</i>	165
9.4.2	Membuat <i>View peminjaman.php</i>	178
9.4.3	Membuat <i>View pengembalian.php</i>	181
BAGIAN 10	PENGUJIAN DAN IMPLEMENTASI SISTEM YANG TELAH DIBANGUN	184
10.1	Perancangan Menggunakan Proses <i>Use Case Diagram</i>	184
10.2	Implementasi Sistem	185
DAFTAR PUSTAKA	214	

BAGIAN 1

ANALISIS PERMASALAHAN

Perpustakaan merupakan suatu satuan kerja organisasi badan atau Lembaga. Satuan unit kerja tersebut dapat berdiri sendiri, tetapi dapat juga merupakan bagian dari organisasi diatasnya yang lebih besar. Perpustakaan yang berdiri sendiri seperti perpustakaan umum, unit pelaksana teknis perpustakaan pada universitas, dan perpustakaan nasional. Sedangkan perpustakaan yang merupakan bagian dari suatu oragnisasi yang lebih besar seperti perpustakaan khusus atau kedianasan dan perpustakan sekolah.

Fungsi Perpustakaan sekolah sebagai tempat baca yang sekarang semakin kurang diminati oleh siswa untuk membaca buku dan mencari literatur ilmu pengetahuan, hal ini desebabkan karena terkalahkan oleh media informasi seperti internet yang lebih mudah digunakan dalam pencarian berbagai macam ilmu dan sumber bacaan.

Dalam dunia pendidikan, Perpustakaan sangat dibutuhkan oleh siswa untuk mencari bahan referensi namun perpustakaan saat ini memiliki banyak keterbatasan buku, maka dibutuhkan sebuah sistem yang dapat terintegrasi dengan galeri buku yang telah dirancang sebelumnya. Integrasi dapat diartikan adanya saling keterkaitan antara satu sistem dengan sistem yang lain.

Kehadiran teknologi informasi dan komunikasi (TIK) saat ini menjadi suatu yang penting dalam sebuah instansi. Kemajuan TIK dalam proses pertukaran informasi membuat kualitas pelayanan dari suatu instansi meningkat. Oleh karena itu, agar pertukaran informasi

dapat dilakukan dengan maksimal harus didukung dengan penggunaan teknologi dan komunikasi yang memadai.[1]

Instansi pendidikan membutuhkan sistem informasi dalam menjalankan kegiatan kerjanya, termasuk SMP 41 Bulukumba sebagai Sekolah Menengah Pertama. SMP 41 Bulukumba saat ini belum menggunakan Sistem Informasi Perpustakaan dalam menjalankan berbagai kegiatan kerjanya.

Perpustakaan dituntut untuk memberikan sebuah layanan informasi pada pengguna yang tepat dengan waktu yang tepat. Layanan dapat terlaksana dengan baik apabila perpustakaan dapat menghadirkan dan memanfaatkan perkembangan teknologi informasi dalam pengelolaan perpustakaan. [2]

Integrasi Sistem Informasi yang tepat dapat mendukung rencana dan pengembangan sistem informasi yang ada di SMPN 41 Bulukumba. Penerapan integrasi sistem jika penerapannya sesuai dengan tujuan yang dibangun dengan menerapkan strategi sistem yang selaras dengan strategi Sistem Galeri Buku. Ketersediaan SMPN 41 Bulukumba masih banyak buku yang tidak tersedia sehingga kurang membantu siswa yang ingin melakukan peminjaman. Sistem informasi belum juga tersedia sehingga dalam mengakses informasi ke sistem lainnya belum dapat dilakukan.

Oleh karena itu, akan dirancang sebuah Sistem informasi yang akan diintegrasikan dengan sistem yang telah dirancang sebelumnya, dimana sistem sebelumnya yaitu Sistem Galeri Buku akan memberikan akses kepada Sistem Informasi yang akan dirancang pada proyek ini.

Dengan integrasi sistem yang akan dibangun ini memiliki kelebihan yaitu dapat melakukan pertukaran data atau informasi dan

memiliki bahasa yang independent. Sehingga integrasi yang akan dilakukan ini yaitu sistem informasi perpustakaan dengan memanfaatkan teknologi *web service*.

Dalam perancangan sistem ini menggunakan sebuah kerangka kerja yaitu *codeigniter* yang sudah banyak dipakai dalam pengembangan *website*, yang ringan dan cepat dalam membangun pemrograman *website* dinamis. Berbeda dengan cms yang memang dapat di unduh secara gratis namun kurang dapat dikembangkan karena masih bersifat statis sehingga perancangan *website* Galeri Buku ini menggunakan kerangka kerja yang dapat dikembangkan dengan dinamis.

BAGIAN 2

TEORI PERPUSTAKAAN DIGITAL DAN INTEGRASI SISTEM

2.1 Definisi Perpustakaan

Perpustakaan adalah sebuah tempat yang digunakan oleh setiap orang mulai dari anak-anak hingga orang dewasa guna untuk menambah wawasan dengan cara membaca buku.

Seiring dengan berkembangnya dunia teknologi di zaman sekarang, perpustakaan tradisional di perbaharui agar sesuai dengan era digital. Perpustakaan elektronik dalam pengelolaan yang terdapat di dalamnya mirip dengan yang ada di perpustakaan tradisional, yaitu akuisisi, katalog, penyimpanan, peminjaman, dan pencarian. [3]

E-Library dapat dianggap sebagai suatu sistem yang dapat mengelola buku ataupun informasi tentang ilmu pengetahuan dengan menggunakan digital atau secara terkomputerisasi. *E-Library* semakin memungkinkan pengguna untuk jarak jauh untuk dapat mengakses berbagai sumber informasi tanpa mempertimbangkan kondisi penyimpanan fisik. *E-Library* sudah dianggap sebagai *effectifeway* untuk mengatur sumber daya informasi dan dapat memberikan layan yang terkait dengan pengguna. [4]

2.2 Definisi Integrasi Sistem

Sistem informasi yang terintegrasi merupakan sebuah sistem yang memungkinkan berbagi data untuk seluruh organisasi. Sistem terintegrasi juga disebut sistem ERP (*Enterprise Resource Planning*) karena semua data operasional terletak pada *database* pusat dan dapat diakses oleh pengguna disuatu organisasi. Dengan sebuah sistem terintegrasi, perusahaan dapat menyimpan informasi secara elektronik.

[5]

BAGIAN 3

KONSEP DASAR SISTEM INFORMASI DAN *WEB SERVICE*

3.1 Definisi Sistem

Sistem dapat didefinisikan dengan dua pendekatan yaitu: [3]

- 1) Dengan pendekatan prosedur, sistem dapat didefinisikan sebagai kumpulan dari prosedur-prosedur yang mempunyai tujuan tertentu.
- 2) Pendekatan komponen, sistem dapat didefinisikan sebagai kumpulan dari komponen yang saling berhubungan satu dengan yang lainnya membentuk satu kesatuan untuk mencapai tujuan tertentu.

Secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur komponen atau variabel yang terorganisasi saling berinteraksi saling bergantung satu sama lain dan terpadu.

Teori sistem secara umum pertama kali diuraikan oleh Kenneth Boulding, terutama menekankan pentingnya perhatian terhadap setiap bagian yang membentuk sebuah sistem titik teori Sistem mengatakan bahwa setiap unsur pembentuk organisasi itu penting dan harus mendapat perhatian yang utuh supaya manajer dapat bertindak lebih efektif.

Yang dimaksud unsur atau komponen pembentuk organisasi di sini bukan hanya bagian-bagian yang tampak secara fisik tetapi juga

hal-hal yang mungkin bersifat abstrak atau konseptual, seperti misi, pekerjaan, kegiatan kelompok informal, dan lain sebagainya. [4]

Pendekatan sistem memberikan banyak manfaat dalam memahami lingkungan titik pendekatan sistem Berusaha menjelaskan sesuatu yang dipandang dari sudut pandang sistem serta berusaha menemukan struktur unsur yang membentuk sistem tersebut titik Dengan memahami struktur sistem dan proses sistem seseorang akan menjelaskan mengapa tujuan suatu sistem tidak tercapai.

Dari definisi ini dapat disimpulkan pengertian sistem secara rinci, yaitu sebagai berikut: [4]

- a. Setiap sistem terdiri dari berbagai unsur. Unsur-unsur suatu sistem terdiri dari subsistem yang lebih kecil yang terdiri pula dari kelompok-kelompok unsur yang membentuk sistem tersebut.
- b. Unsur-unsur tersebut merupakan bagian yang tak terpisahkan dari sistem yang bersangkutan.
- c. Unsur-unsur di dalam sistem tersebut bekerja sama untuk mencapai tujuan sistem suatu.
- d. Sistem merupakan bagian dari sistem lain yang lebih besar.

3.1.1 Karakteristik Sistem

Suatu sistem memiliki karakteristik atau sifat-sifat tertentu, yaitu (Sutabri, 2004:12): [5]

1. Komponen-komponen (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang sering disebut dengan subsistem yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai

sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

2. Batas sistem (*Boundary*)

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (*scope*) sistem itu sendiri.

3. Lingkungan luar sistem (*Environments*)

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

4. Penghubung sistem (*Interface*)

Penghubung merupakan media penghubung antara subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya.

5. Masukan sistem (*Input*)

Masukan yaitu energi yang dimasukan ke dalam sistem, dimana dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). Masukkan perawatan adalah energi yang di-*input*-kan supaya sistem tersebut dapat beroperasi, sedang masukan sinyal adalah energi yang diproses untuk didapatkan keluaran.

6. Keluaran sistem (*Output*)

Keluaran yaitu hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

7. Pengolah sistem

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah *input* menjadi *output*.

8. Sasaran sistem (*Objective*)

Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Apabila suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak ada gunanya.

3.1.2 Klasifikasi Sistem

Sistem dapat diklasifikasikan dari berbagai sudut pandang, diantaranya adalah sebagai berikut (Sutabri, 2004:14): [5]

1. Sistem abstrak dan sistem fisik.

Sistem abstrak adalah suatu sistem yang berupa ide-ide pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya teologi yaitu sistem yang berupa pemikiran-pemikiran hubungan antara manusia dengan Tuhan.

Sedangkan sistem fisik adalah suatu sistem yang berupa pemikiran atau ide-ide yang nyata atau yang ada secara fisik. Misalnya sistem komputer, sistem akuntansi, sistem produksi dan lain sebagainya.

2. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia. Misalnya: sistem perputaran bumi. Sistem buatan manusia yang melibatkan interaksi antara manusia dengan mesin disebut dengan *human-machine system* atau ada yang menyebut dengan *man-machine system*.

3. Sistem tertentu dan sistem tak tentu

Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Contohnya: sistem komputer. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

4. Sistem tertutup dan sistem terbuka

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak berpengaruh dengan lingkungan luarnya. Sedangkan sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan.

3.2 Definisi Sistem Informasi

Informasi sangat erat hubungannya dengan data titik informasi berasal dari data. Keterikatan antara data dan informasi sebagai entitas penting pembentuk sistem informasi. Data merupakan nilai, keadaan, atau sifat yang berdiri sendiri lepas dari konteks apapun titik. Sementara informasi adalah data yang telah diolah menjadi suatu bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat itu atau mendatang. (Davis, 1995). Mc Leod (1995). [6]

Sistem informasi adalah kombinasi antara prosedur kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi. Dalam dunia teknologi sistem informasi yaitu kombinasi dari berbagai unit yang dikelola oleh orang dalam suatu organisasi yang meliputi perangkat keras, perangkat lunak, jaringan untuk menciptakan, mengubah dan menyebarkan informasi ke publik. [7]

3.2.1 Kualitas Informasi

Informasi yang baik adalah informasi yang berkualitas (Sutabri, 2004:25), informasi yang berkualitas ditentukan oleh beberapa hal, yaitu: [5]

- 1. Akurat (*accurate*)**

Informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan, informasi harus jelas mencerminkan maksudnya.

- 2. Tepat waktu (*time lines*)**

Informasi yang dihasilkan atau dibutuhkan tidak boleh terlambat, karena nantinya tidak mempunyai nilai yang baik, sehingga apabila dijadikan dasar dalam pengambilan keputusan akan berakibat fatal atau kesalahan pengambilan keputusan dan tindakan.

- 3. Relevan (*relevance*)**

Informasi harus memberikan manfaat yang baik untuk pemakai informasi tersebut.

3.3 Web Service

Web service merupakan kumpulan suatu layanan berbasis *web* dengan menggunakan jaringan protokol HTTP, layanan tersebut dapat diakses dan dimanfaatkan oleh pengguna dengan bahasa pemrograman, arsitektur dan sistem operasi yang berbeda (interoperability).

Web service harus bersifat *stateless*, sehingga dapat dibaca dan di akses multi-platform. Arsitektur pada *web service* dapat dibangun menggunakan *Simple Object Access Protocol* (SOAP) atau *Representational State Transfer* (REST), sementara *service* tersebut direpresentasikan dalam bentuk format teks, JSON atau XML. Pada

umumnya formatnya menggunakan JSON dan XML. Gambar 1 menunjukkan struktur logis dari tumpukan komunikasi untuk layanan *Web*. [8]

Jenis *web service* dapat dibagi menjadi dua menjadi dua, yaitu REST dan SOAP: [9]

1) *Representational State Transfer* (REST)

REST adalah salah satu jenis *web service* yang menerapkan konsep perpindahan antara *state*. *State* disini dapat digambarkan sebagai jika *browser* meminta suatu halaman *web* maka server akan mengirimkan halaman *web* yang sekarang ke *browser*. bernavigasi melalui *link-link*, sama halnya dengan mengganti step dari halaman *web* begitupula REST bekerja dengan bernavigasi melalui *link-link* HTTP untuk melakukan aktivitas tertentu seakan-akan terjadi perpindahan.

2) *Simple Object Access Control* (SOAP)

SOAP adalah protokol untuk saling bertukar pesan dalam format XML antar antar komputer di dalam jaringan, biasanya menggunakan HTTP/HTTPS. *Web service* ini menggunakan SOAP sebagai protokol pengiriman pesannya dalam menjalankan tugasnya soal menggunakan struktur XML tertentu dalam mengirimkan request kepada *web service* penerima respon dari *web service* memiliki struktur nya sendiri.

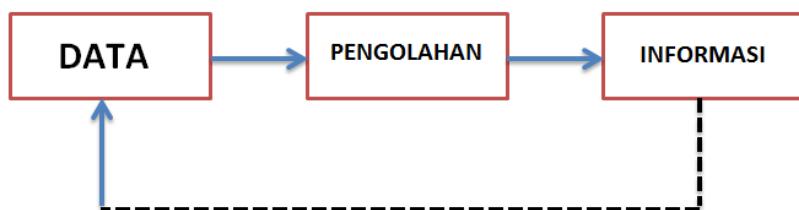
BAGIAN 4

PENGENALAN DBMS DAN MYSQL

4.1 Tentang *Database*, DBMS, dan RDBMS

4.1.1 Definisi *Database*

Secara sederhana *database* (basisdata) adalah suatu pengorganisasian data dengan bantuan computer yang memungkinkan suatu data tersebut dapat diakses dengan mudah dan cepat. [13]



Gambar 6.1 Data dan Informasi

Basis data dapat kita contohkan sebagai sebuah lemari arsip. Jika kita mempunyai sebuah lemari arsip dan bertugas untuk mengelolanya, maka kemungkinan kita akan memberikan hal-hal seperti: membuat sampul/map pada kumpulan arsip yang akan disimpan, menentukan jenis kelompok arsip dan juga membuat penomoran dengan kode tertentu. [14]

Dari contoh yang dijelaskan sebelumnya diatas dapat kita simpulkan definisi *database* secara rinci adalah: [14]

- Himpunan kelompok data (arsip) yang saling terhubung yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah,

- Kumpulan data yang saling terhubungan dan disimpan secara Bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu.
- Kumpulan *file*, tabel, arsip yang saling terhubung dan disimpan dalam media penyimpanan dalam bentuk elektronik.

Ditinjau berdasarkan komponen-komponennya, sistem basis data terdiri dari: [15]

- Perangkat Keras (*Hardware*)

Perangkat keras yang dimaksud adalah perangkat computer beserta perangkat periferalnya. Di dalam perangkat computer semua proses pengelolaan dan pengelolaan data dapat dilakukan, adanya unit pemrosesan utama CPU (*central processing unit*), *register* dan memori (*random acces memory* atau RAM). Pemakai dapat berinteraksi dengan Basis Data dengan menggunakan papan ketik (*keyboard*) ataupun *mouse*, dan pemakai dapat melihat hasil pengelolaannya dengan menggunakan layar computer (*monitor*).

- Perangkat Lunak (*software*)

Komponen kedua dalam sebuah sistem basis data adalah perangkat lunak atau *software*. Ada tiga jenis perangkat lunak yang terlibat dalam pengelolaan basis data, yaitu:

- Sistem operasi (*Operating System*)
- Sistem pengelola basis data (*Database Management System* atau DBMS)
- Aplikasi pengguna basis data.

- Pemakai (*Brainware*)

Ada beberapa tipe pemakai yang dibedakan berdasarkan jenis pekerjaan dan cara mereka berinteraksi dengan *database* yaitu:

Administrator Basis Data (*Database Administrator*)

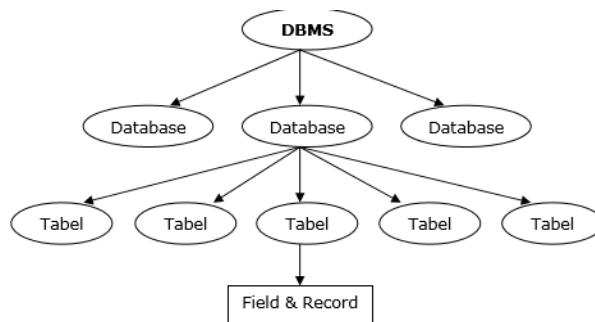
- Pengembang aplikasi (*Programmer/Developer*)
- Pemakai mahir (*Casual User*)
- Pemakai akhir/umum (*End-user-Naïve User*)
- Basis Data (*Database*)

Sebelumnya telah dibahas apa itu basis data secara singkat dan rinci. Kembali harus diketahui, basis data selalu terdiri atas 3 hal: kumpulan data yang terorganisir, relasi antar data dan objektifnya.

4.1.2 Definisi DBMS (*Database Management System*)

DBMS atau *Database Management System*. DBMS adalah sebuah perangkat lunak atau program computer yang dirancang secara khusus untuk memudahkan pengelolaan *database*. Salah satu macam DMBS yang sangat popular adalah RDBMS (*Relatioanl Database Management System*), yang menggunakan model basis data relasional atau dalam bentuk tabel yang saling terhubung antar satu sama lain.

[15]



Gambar 6.2 Struktur / Hirarki Database

DBMS ini yang nantinya akan menentukan bagaimana basis data dikelola dan diorganisir secara fisik dalam sebuah penyimpanan perangkat komputer. DBMS juga berperan dalam menerapkan

mekanisme pengamanan (*security*), dan menjaga integritas basis data jika dikelola dan dimanfaatkan dalam sebuah jaringan computer. [15]

Dapat di simpulkan bahwa dengan adanya DBMS ini akan sangat memudahkan *user* atau pengguna nantinya untuk mengakses *database* secara udah dan efisien, serta dapat menanipulasi dan juga mengotrol data yang telah ada dengan mudah.

4.2 Tentang MySQL

MySQL adalah salah satu contoh dari produk RDBMS yang sangat popular dilingkungan LINUX, MySQL juga tersedia di windows. RDBMS ini sudah banyak di gunakan oleh *programmer* dalam membuat sebuah *website* untuk dijadikan sebagai akses kedalam *database*.

MySQL adalah sebuah server *database open source* yang terkenal yang digunakan berbagai aplikasi tertentu terutama *server* atau membuat *WEB*. MySQL berfungsi sebagai SQL (*Structure Query Language*) yang memiliki sendiri dan sudah diperluas oleh mysql umumnya digunakan bersamaan dengan PHP untuk membuat sebuah *server* yang dinamis dan *powerfull*. [16]

Ada beberapa kelebihan MySQL antara lain: [17]

- *Free* (bebas didownload)
- Stabil dan Tangguh
- Fleksibel dengan berbagai pemrograman
- *Security* yang baik
- Dukungan dari banyak komunitas
- Kemudahan management *database*.
- Mendukung transaksi

- Perkembangan *software* yang cukup cepat.

4.3 Instalasi MySQL di Windows

- Pertama *download* MySQL pada *website*, untuk MySQL yang akan di *download* sebaiknya ada versi akhir atau *last version*, atau bisa di *download* di *link*:
<https://dev.mysql.com/downloads/mysql/>
- Setelah halaman terbuka, maka kita akan mendapatkan beberapa paket instalasi yang berbeda-beda. Pilih “*Go To Download Page*”

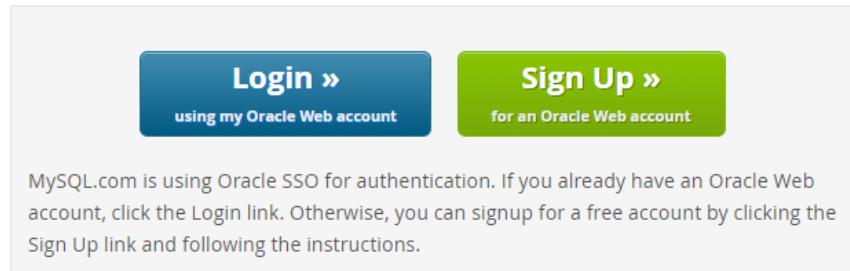
Recommended Download:



- Pada menu *Select Operation System* yang ada di bagian kiri atas kita dapat menentukan sistem operasi yang akan kita gunakan.
- Pilih Microsoft Windows pada *Select Operation System*
- Setelah itu pilih file yang akan kita download, di halaman ini menyiapkan dua file yang berbeda yaitu EXE dan ZIP, dan disini saya menilih file bagian .msi

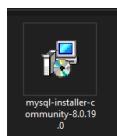
File	Version	Size	Action
Windows (x86, 32-bit), MSI Installer (mysql-installer-web-community-8.0.19.0.msi)	8.0.19	18.6M	Download
Windows (x86, 32-bit), MSI Installer (mysql-installer-community-8.0.19.0.msi)	8.0.19	398.9M	Download

- f. Setelah klik pada bagian *download* maka kita akan di arahkan ke bagian *login* dan *register*, jika tidak ingin melakukan proses registrasi akan klik saja link “*No Thanks, Just Start MyDownload*”



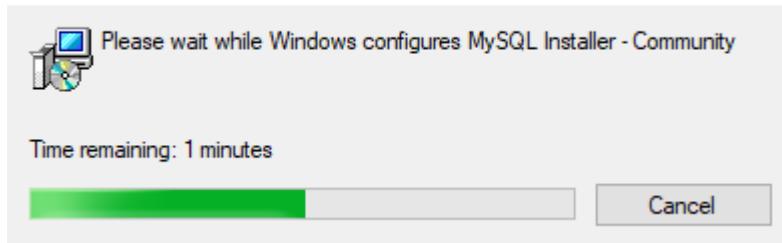
No thanks, just start my download

- g. Setelah file berhasil di *download*, lanjutkan dengan *Double click* file .msi tadi

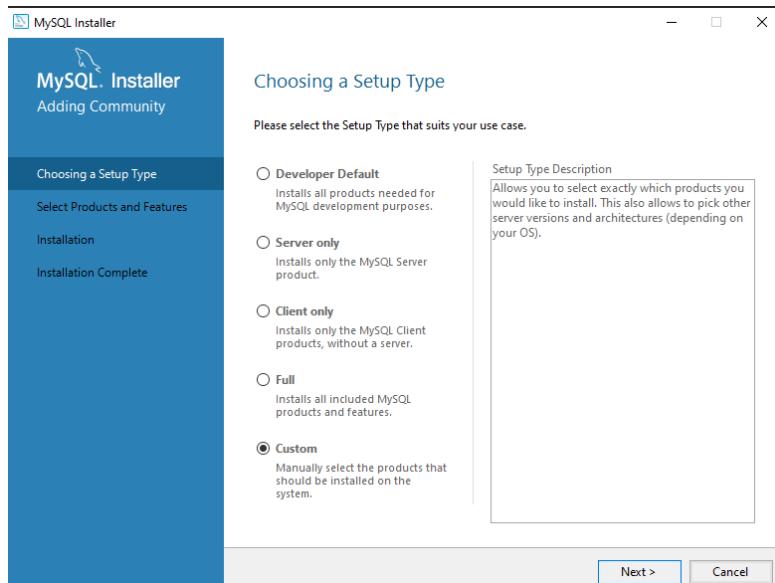


- h. Tunggu sampai proses intasllasi berhasil.

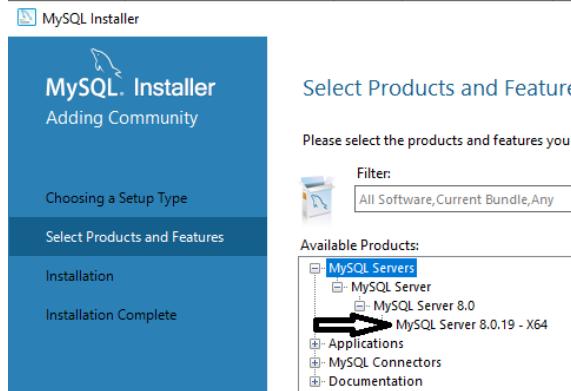
MySQL Installer - Community

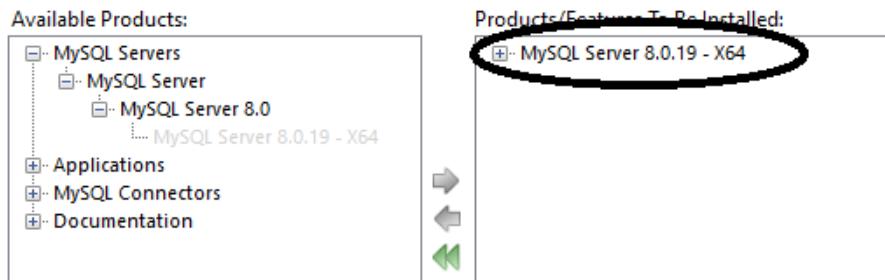


- i. Setelah proses instalasi selesai, maka akan muncul halaman *Choosing a setup type*. Ada beberapa pilihan untuk tipe instalasi seperti. Disini krna kita akan menggunakan MySQL untuk pengembangan produk, maka dipilih yang *custom* kemudian pilih next.

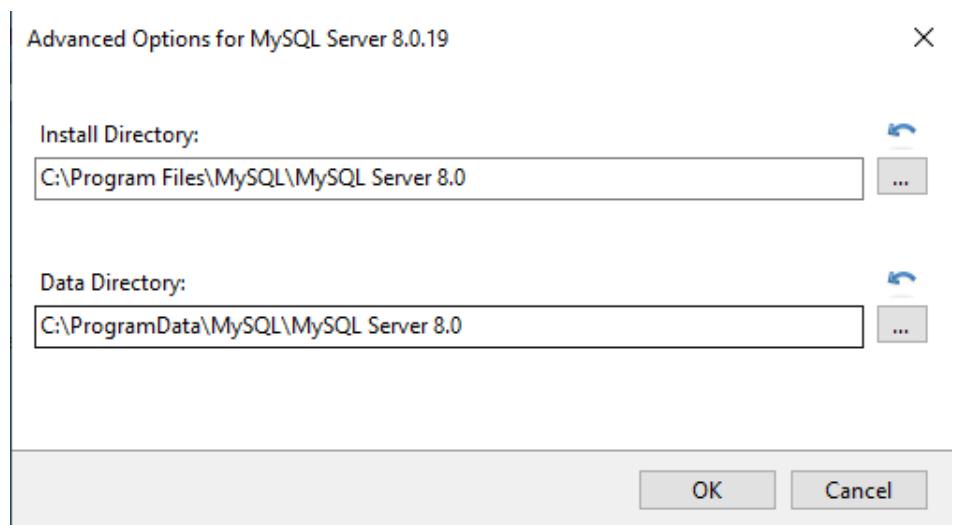


- j. Kemudian pada *Select Products and features* kita dapat memilih fitur apa saja yang diinginkan. Disini kita akan menggunakan MySQL Server 8.0.19 – X64 (*double clik*) maka akan muncul warna hijau pada panah di bagian tengah.
- k. *Double Clik* warna hijau untuk memindahkan ke bagian kiri. Seperti pada gambar di bawah ini.





- l. Selanjutnya klik MySQL Server 8.0.19 – X64 maka akan muncul *Advance Option* dibagian kiri bawah.



- m. Maka kan muncul dua *directory* yang akan menjadi tempat pemnyimpanan MySQL, dan kemudian OK.
- n. Kemudian pada *Check Requirements* pilih MySQL Server 8.0.19 dan *Execute* untuk melanjutkan.

Check Requirements

The following products have failing requirements. MySQL Installer will attempt to resolve them automatically. Requirements marked as manual cannot be resolved automatically. Click on each item to try and resolve it manually.

For Product	Requirement	Status
MySQL Server 8.0.19	Microsoft Visual C++ 2019 Redistrib...	

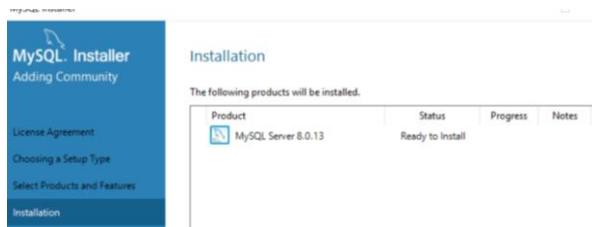
Requirement Details

MySQL Installer is trying to resolve this requirement automatically.
There is nothing you need to do.

Requirement: Microsoft Visual C++ 2019 Redistributable Package (x64) is not installed. Latest binary compatible version will be installed if agreed

Status:

- o. Setelah itu muncul halaman *Installation*. Pilih *Execute* dibagian bawah untuk melanjutkan.



Tunggu sampai proses instalasi selesai.

BAGIAN 5

PEMODELAN PADA SISTEM

Ada beberapa pemodelan yang dapat digunakan dalam merancang atau membangun sebuah *software* (perangkat lunak), seperti model *prototype*, model *Rapid Application Development (RAD)*, dan lain sebagainya. Namun pada buku ini di dalam perancangannya menggunakan *Unified Modeling Language (UML)* sebagai pemodelan *design* program berorientasi objek.

5.1 *Unified Modeling Language (UML)*

Langkah pertama dalam mengembangkan metodologi desain berorientasi objek untuk aplikasi komputer dan juga untuk sistem *database* adalah penggunaan UML (*Unified Modeling Language*) sebagai notasi standar untuk pemodelan objek dunia nyata. [18]

Unified Modeling Language (UML) memungkinkan analis, arsitek perangkat lunak, dan pengembang untuk menentukan, memvisualisasikan, dan mendokumentasikan seluruh sistem. Dengan menyediakan "bahasa" yang sama untuk berbicara tentang sistem yang kompleks, UML memungkinkan banyak kontributor yang berbeda, dengan beragam perspektif, untuk berkomunikasi berdasarkan kesamaan.[19]

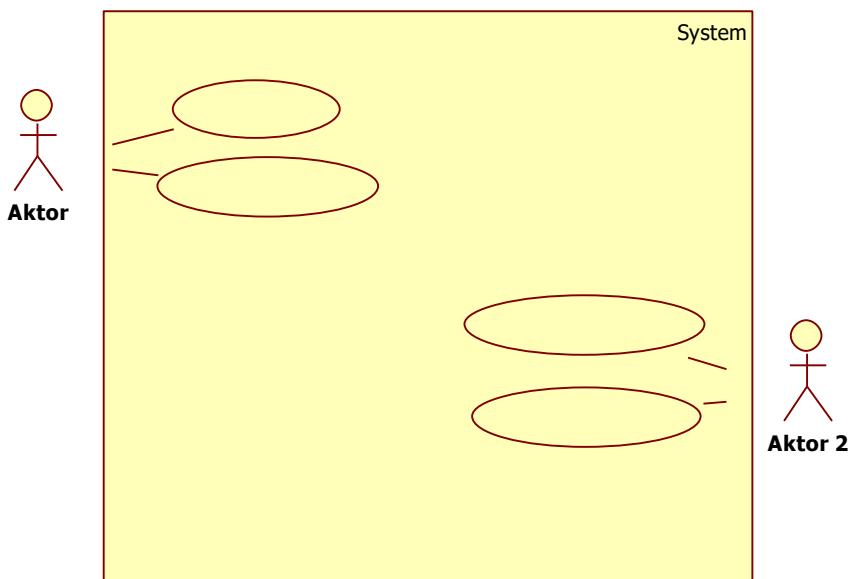
5.1.1 Representasi Arsitektur

Setiap diagram UML dirancang untuk memungkinkan pengembang dan pelanggan melihat sistem perangkat lunak dari

perspektif yang berbeda dan dalam berbagai tingkat abstraksi. Diagram UML yang biasa dibuat dalam alat pemodelan visual meliputi: [19]

a. *Use Case Diagram*

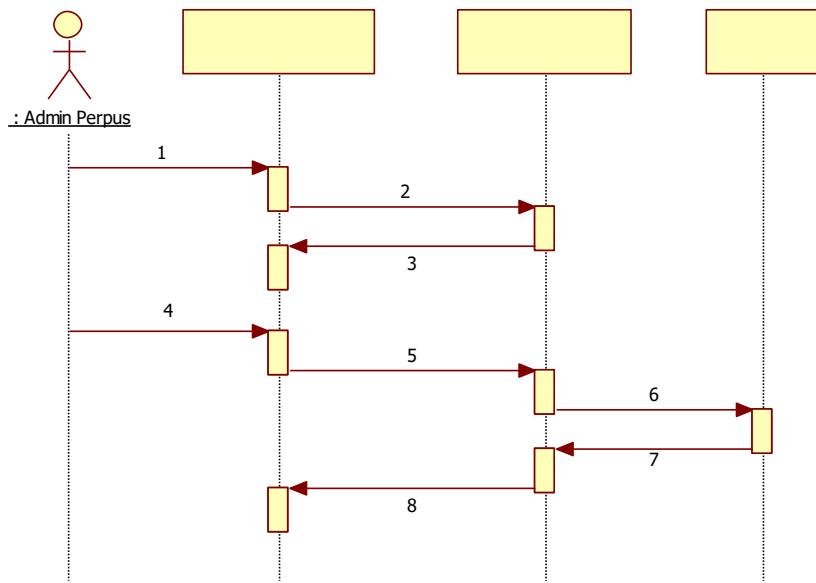
Menampilkan hubungan antara aktor dan kasus penggunaan. *Class Diagram* memodelkan struktur dan konten kelas menggunakan elemen desain seperti kelas, paket dan objek. Itu juga menampilkan hubungan seperti penahanan, warisan, asosiasi dan lain-lain.



Gambar 5.1 Contoh Use Case Diagram

b. *Sequence Diagram*

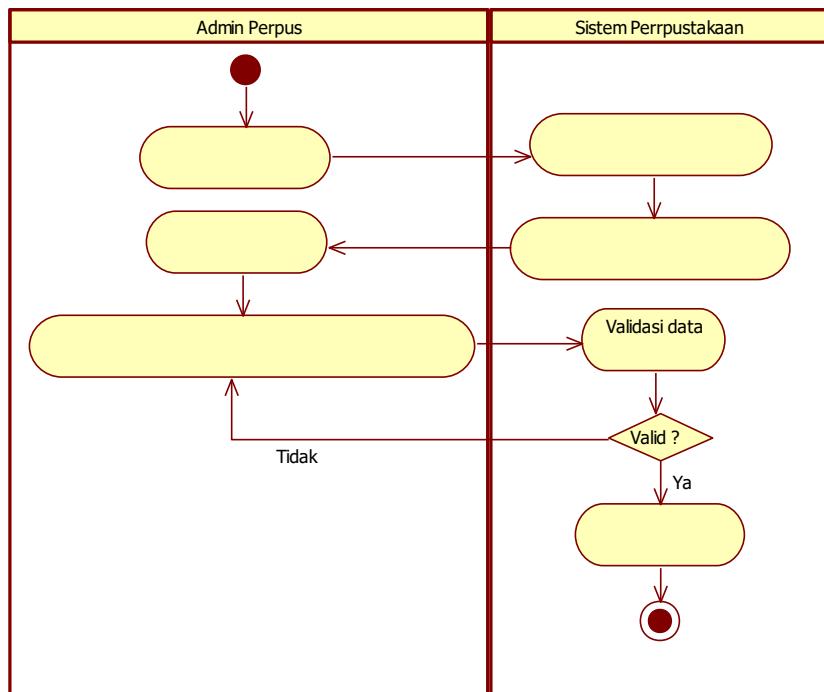
Menampilkan urutan waktu objek yang berpartisipasi dalam interaksi. Ini terdiri dari dimensi vertikal (waktu) dan dimensi horizontal (objek yang berbeda).



Gambar 5.2 Contoh Sequence Diagram

c. Activity Diagram

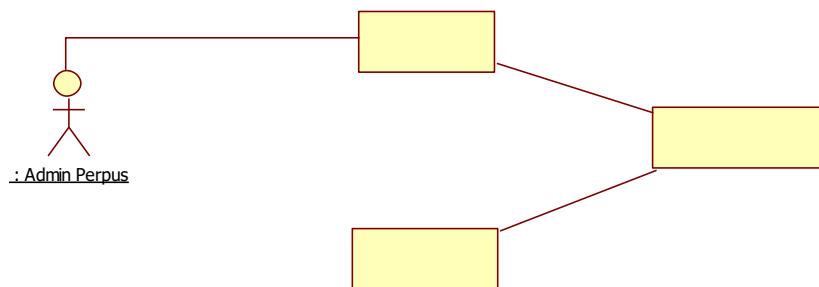
Menampilkan diagram keadaan khusus di mana sebagian besar keadaan adalah keadaan tindakan dan sebagian besar transisi dipicu oleh penyelesaian tindakan di negara sumber. Diagram ini berfokus pada aliran yang didorong oleh pemrosesan internal.



Gambar 5.3 Contoh Activity Diagram

d. Collaboration Diagram

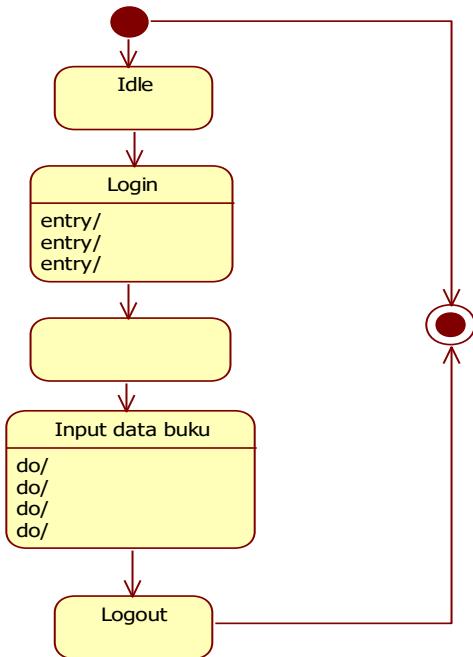
Menampilkan interaksi yang diatur di sekitar objek dan tautannya satu sama lain. Angka digunakan untuk menunjukkan urutan pesan.



Gamber 5.4 Contoh Collaboration Diagram

e. *Statechart Diagram*

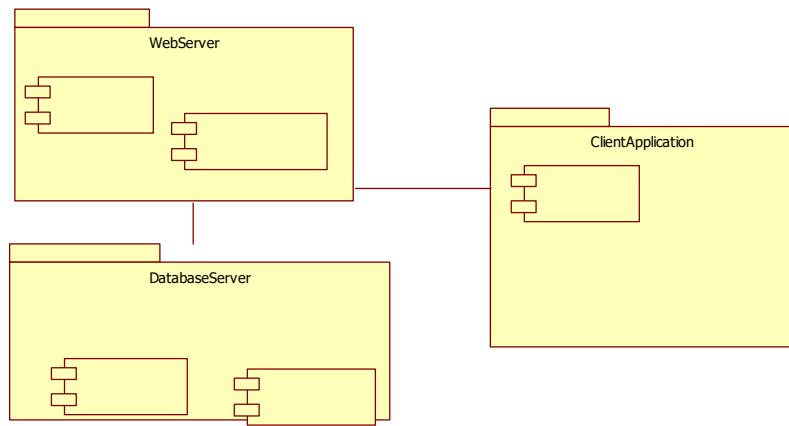
Menampilkan urutan keadaan yang dilalui objek interaksi selama hidupnya sebagai respons terhadap rangsangan yang diterima, bersama dengan respons dan tindakannya.



Gambar 5.5 Contoh Statechart Diagram

f. *Component Diagram*

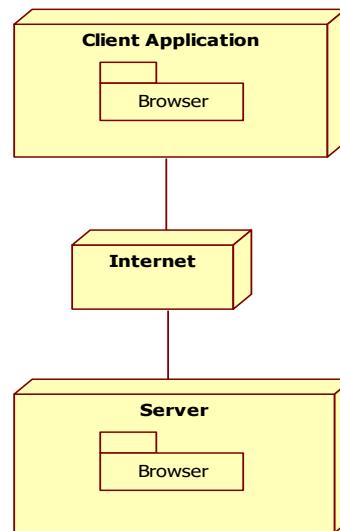
Menampilkan struktur paket tingkat tinggi dari kode itu sendiri. Ketergantungan antar komponen ditunjukkan, termasuk komponen kode sumber, komponen kode biner, dan komponen yang dapat dieksekusi. Beberapa komponen ada pada waktu kompilasi, pada waktu tautan, pada waktu menjalankan juga pada lebih dari satu waktu.



Gambar 5.6 Contoh Component Diagram

g. Deployment Diagram

Menampilkan konfigurasi elemen proses run-time dan komponen perangkat lunak, proses, dan objek yang hidup padanya. Contoh komponen perangkat lunak merupakan manifestasi run-time dari unit kode.



Gamber 5.7 Contoh Component Diagarm

5.1.2 Arsitektur Perangkat Lunak Dikembangkan Menggunakan UML Memberikan Manfaat Berikut

Manfaat dari UML dapat di lihat dari uraian di bawah: [19]

- *Preliminary Analysis Model / Model Analisis Pendahuluan*
Di luar diagram *use case*, pemodelan UML dapat sangat membantu dalam formalisasi model analisis pendahuluan. Pemodelan proses bisnis dapat didasarkan pada diagram aktivitas dan model pertama aplikasi Anda dapat direalisasikan menggunakan diagram kelas. Persyaratan *Objecteering / UML* menyediakan asisten untuk membantu dalam pembuatan model dari analisis persyaratan dan kamus. Dengan menggunakan diagram alir, Persyaratan *Objecteering / UML* memberi Anda representasi unik dari definisi persyaratan Anda melalui arus informasi yang dipertukarkan antara elemen sistem

- *Component-Based Development / Pengembangan Berbasis Komponen*

Untuk pengembangan sistem bisnis yang kompleks, pengembangan berbasis komponen adalah cara yang sudah terbukti. Dengan *Rational Rose*, pengguna sekarang dapat memodelkan komponen dan antarmuka mereka dengan lebih efektif. Komponen desain yang diperlukan dapat dengan mudah direkayasa ulang untuk menjelajahi antarmuka dan keterkaitan antara komponen lain dalam model, nafsu tarik dan lepas komponen dari sistem file ke diagram komponen. Kemudian Anda dapat merekayasa balik, menggunakan kembali, memvisualisasikan, beradaptasi, memperoleh, dan membuat komponen untuk sistem Anda dengan cepat dan efisien.

- *Multi-Language and Multi-Platform Development / Pengembangan Multi-Bahasa dan Multi-i*

Proyek perangkat lunak berskala besar sering melibatkan berbagai bahasa. *Rational Rose* menyediakan dukungan multi bahasa sehingga Anda dapat membangun komponen dalam bahasa campuran. Rose mendukung C++, Visual C++, Java, Visual Basic, Ada, dan dapat menghasilkan *Interface Definition Language* (IDL) untuk aplikasi CORBA, serta *Data Description Language* (DDL) untuk aplikasi *database*. Rose yang rasional juga *platform* independen, yang berarti tim Anda dapat bekerja di lingkungan Windows dan UNIX.

- *Complete Team Support / Dukungan Tim Lengkap*

Rational Rose menyediakan cara mudah untuk mengoordinasikan tim pengembangan yang terdiri dari sepuluh hingga seratus atau lebih. Analis, arsitek, dan *engineers* dapat beroperasi di ruang kerja pribadi yang berisi tampilan individual dari keseluruhan model, dan memodifikasi bagian modelnya sendiri hingga siap untuk digabung dengan model lain menggunakan Integrator Model. Perubahan dibuat tersedia untuk orang lain dengan memeriksanya ke dalam *configuration management and version control* (CMVC). *Rational Rose* terintegrasi dengan alat CMVC utama termasuk *Rational ClearCase* dan *Microsoft SourceSafe*. Rose juga sangat terintegrasi dengan *Microsoft Repository* dan Unisys UREP dan terbuka untuk yang lain repositori.

- *Ease of Use / Kemudahan Penggunaan*

Rational Rose adalah alat pemodelan visual pilihan untuk berbagai jenis proyek perangkat lunak. Selain menggunakan UML, Rose mendukung elemen GUI yang sudah dikenal seperti *drag and drop* dengan otomatisasi OLE. Ini memungkinkan Anda berintegrasi

dengan aplikasi seperti Microsoft Word, dan mendorong Rose dari antarmuka server otomatisasi OLE. Kompatibilitas Rose yang terbukti dengan *platform Windows* telah diuji di lab kegunaan Microsoft.

5.1.3 Kelemahan UML

Ada dua kekurangan lain dari UML, tetapi ini dapat diatasi, baik melalui penggunaan stereotip atau dengan memaksakan disiplin pada cara UML digunakan. [19]

- Dalam kasus pertama, UML dapat ditingkatkan secara signifikan dengan meningkatkan disiplin dalam penggunaan nama hubungan. Paling umum nama hubungan di UML adalah kata kerja tunggal yang menggambarkannya dalam satu arah. Apakah ini satu-satunya pilihan, itu tidak dapat diterima. Namun, dimungkinkan untuk menambahkan "peran" di setiap akhir hubungan. Ini memberikan kemampuan untuk menggambarkan bagaimana suatu entitas dilihat dari perspektif entitas lain. Sementara tidak ada seorang pun di luar dunia *Barker* melakukan ini, akan sangat berharga jika nama-nama peran ini dibatasi untuk mengikuti konvensi penamaan *Barker*.
- Kedua, UML hanya menangani sebagian pengidentifikasi unik. Filosofi di balik orientasi objek adalah bahwa tidak perlu secara eksplisit untuk menunjukkan pengidentifikasi unik. Tetapi kemudian ternyata dari sudut pandang entitas induk, seringkali perlu untuk mengidentifikasi kejadian entitas anak. Jadi "*qualified associations*" memungkinkan ini untuk diungkapkan. Tetapi Anda hanya diizinkan mengidentifikasi kejadian ke entitas induk. Anda tidak diizinkan mengidentifikasinya ke dunia pada umumnya. Ini berarti bahwa alih-alih simbol sederhana yang dilampirkan pada hubungan atau atribut untuk menunjukkan pengidentifikasi unik

secara universal, Anda harus menambahkan kotak baru yang artinya dibatasi dan membingungkan.

- Membandingkan file diagram satu sama lain: Solusi saat ini menyediakan penampil penggabungan konten tidak berfungsi untuk kasus penggunaan "Bandingkan Dengan Satu Sama Lain" ketika file yang dipilih di bagian tampilan berisi file ASCII non-diagram bersama dengan file diagram (*.dnx). Dalam hal ini, infrastruktur bandingkan / gabungkan dalam Eclipse menentukan penampil penggabungan konten dengan memeriksa konten file. Inspeksi konten file diagram mengungkapkan bahwa tipe file adalah ASCII dan karenanya *Eclipse* membuat penampil penggabungan teks default. Meskipun penampil default memungkinkan Anda untuk mengedit file dan menggabungkan perbedaan individu, melakukannya berpotensi merusak file diagram. Juga, jika Anda menyimpan sesi "Bandingkan Dengan Satu Sama Lain" dengan diagram terbuka, Anda mendapatkan kesalahan dan pengecualian dalam *file log*.

BAGIAN 6

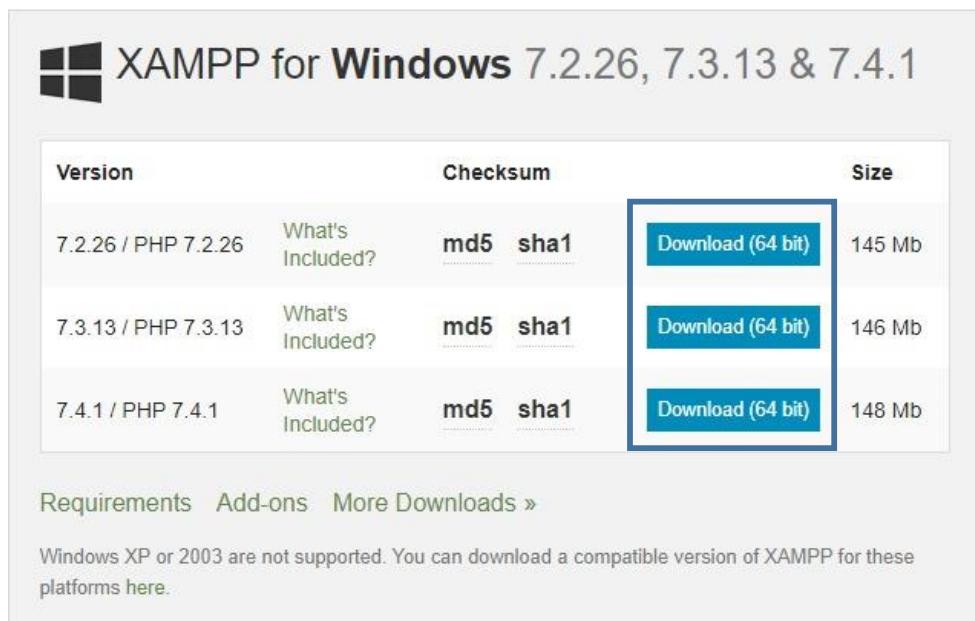
TATA CARA INSTALASI DAN PENGGUNAAN

TOOLS YANG DIGUNAKAN

6.1 Xampp

6.1.1 Cara *Install* Xampp

1. *Download* aplikasi Xampp terbaru, *link*:
<https://www.apachefriends.org/download.html>. Pilih salah satu sesuai dengan yang anda butuhkan:

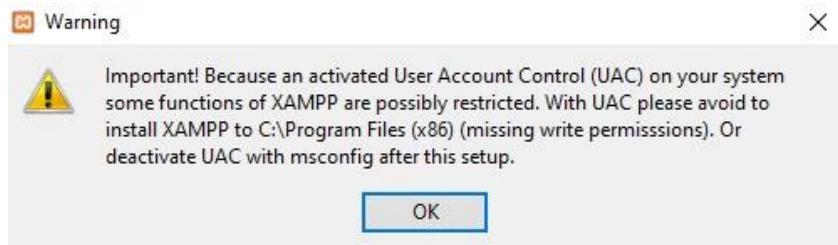


Version	Checksum	Size
7.2.26 / PHP 7.2.26	What's Included? md5 sha1	145 Mb
7.3.13 / PHP 7.3.13	What's Included? md5 sha1	146 Mb
7.4.1 / PHP 7.4.1	What's Included? md5 sha1	148 Mb

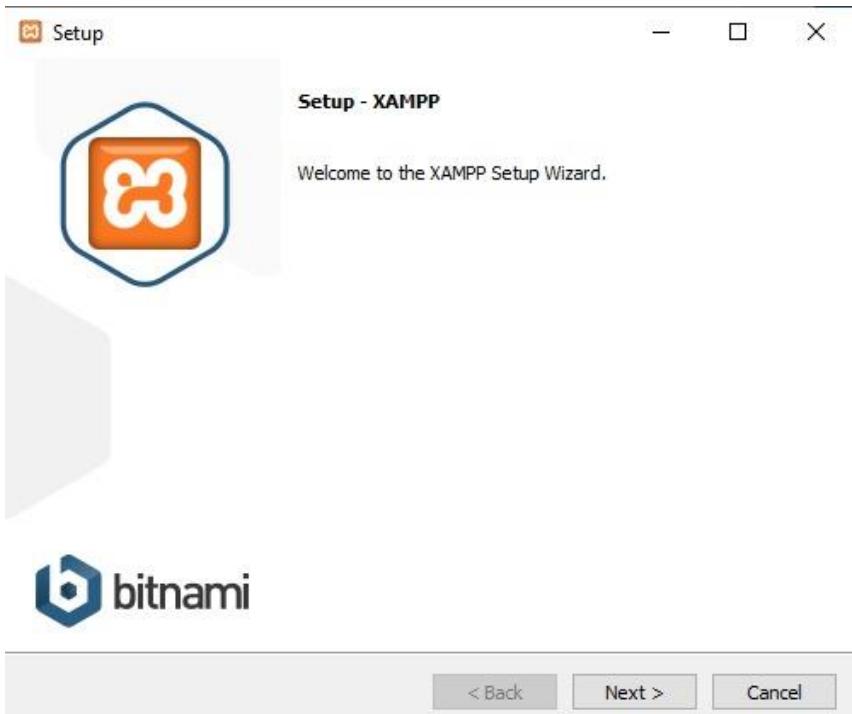
Requirements Add-ons More Downloads »

Windows XP or 2003 are not supported. You can download a compatible version of XAMPP for these platforms [here](#).

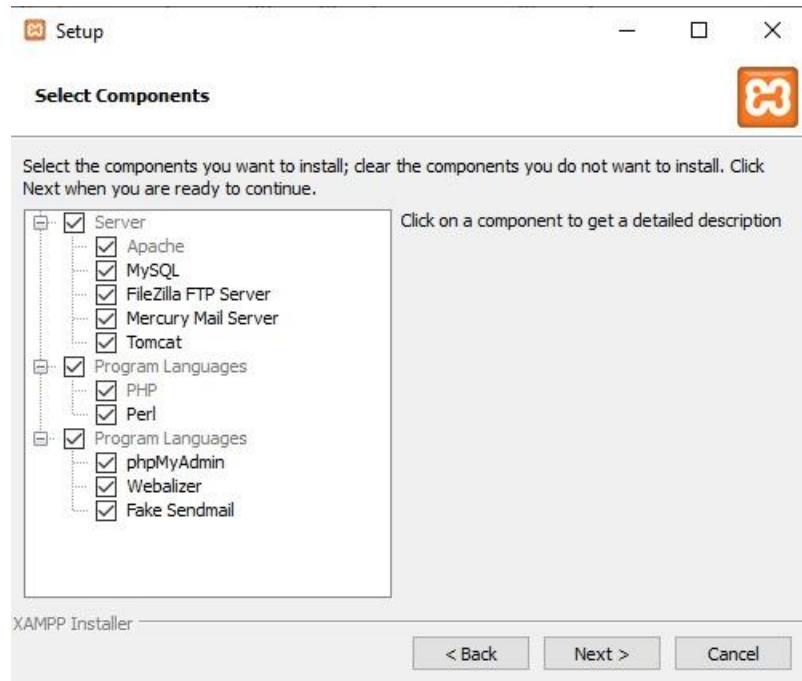
2. Klik dua kali pada file xampp yang baru saja anda *download*, jika muncul pesan *error* seperti gambar berikut abaikan saja dan pilih oke untuk melanjutkan instalasi



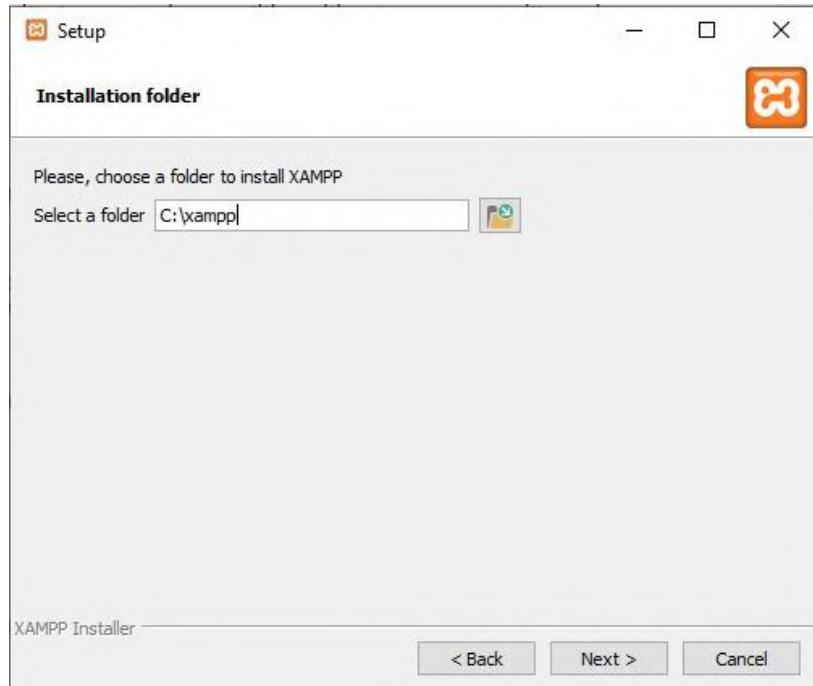
3. Selanjutnya anda akan dialihkan ke jendela yang isinya meminta anda untuk menginstall aplikasi Xampp.



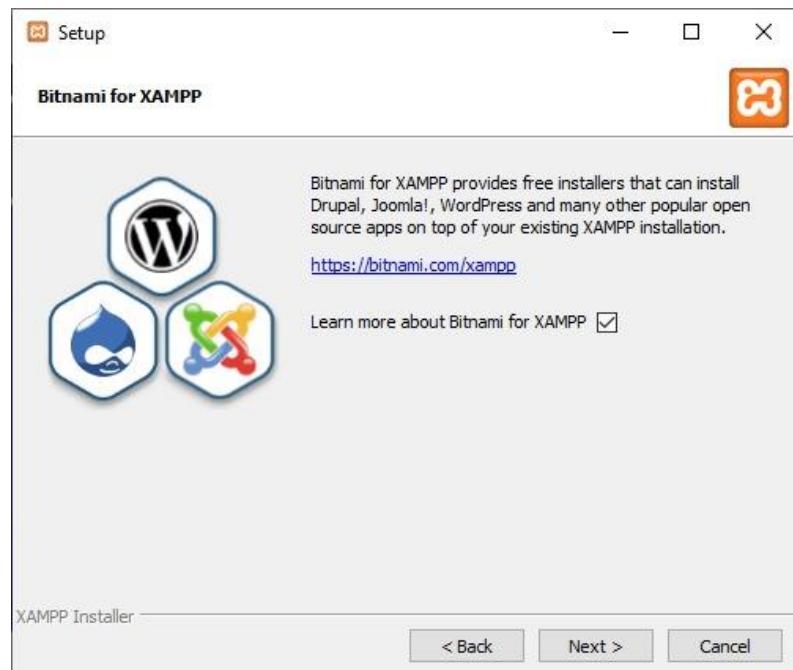
4. Klik *Next* untuk melanjutkan.
5. Selanjutkan anda akan diminta untuk memilih aplikasi yang mau di *install*. Centang saja semua pilihan dan klik *Button Next*.



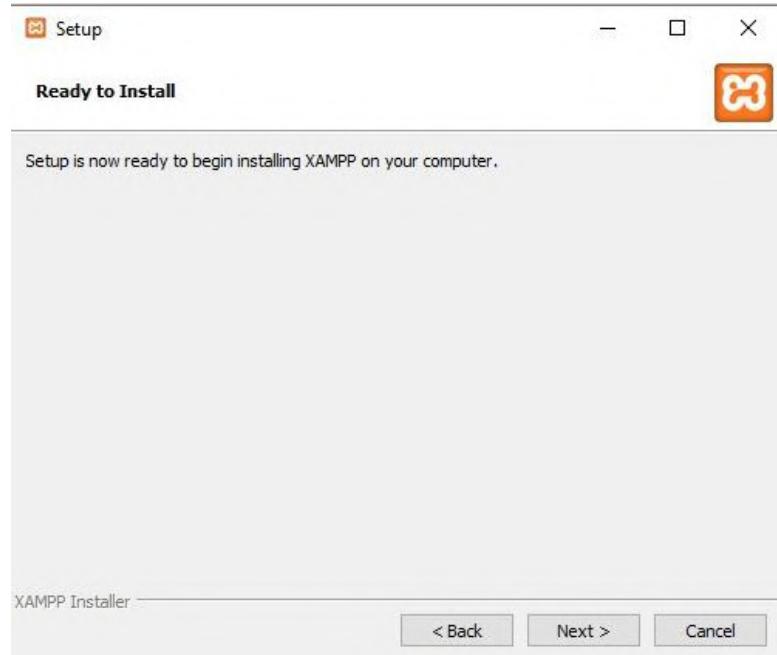
6. Kemudian anda akan diminta untuk menentukan lokasi folder penyimpanan file-file dan folder xampp. Secara default akan diarahkan ke lokasi C:\Xampp.
7. Tetapi jika anda ingin menyimpannya di folder lain anda bisa mengklik *browse* kemudian menentukan secara manual folder yang anda ingin gunakan.
8. Jika sudah selesai lanjutkan dan silahkan klik *Button Install*.



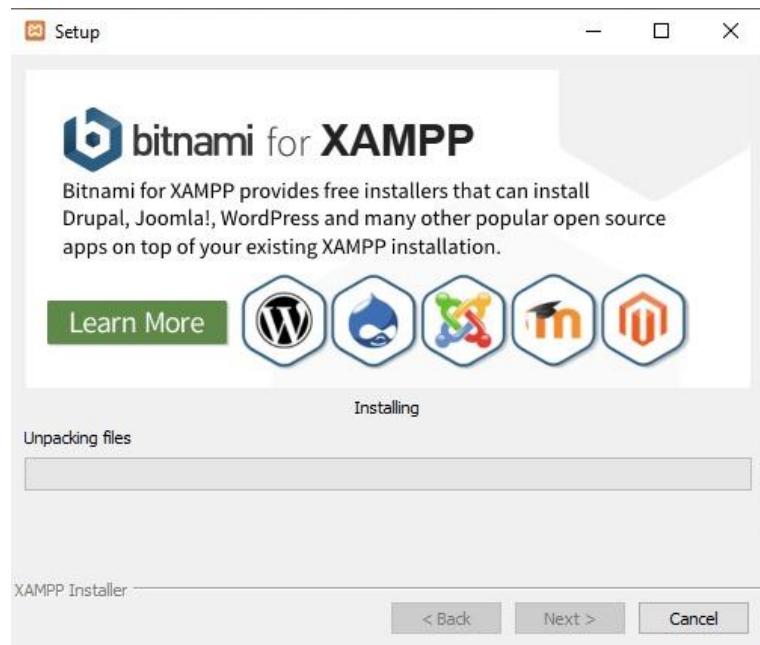
9. Selanjutnya akan muncul tampilan seperti gambar berikut, anda tinggal klik *Next* saja.



10. Selanjutnya akan muncul tampilan bahwa xampp siap untuk di *install*. Anda tinggal klik *Next* saja.

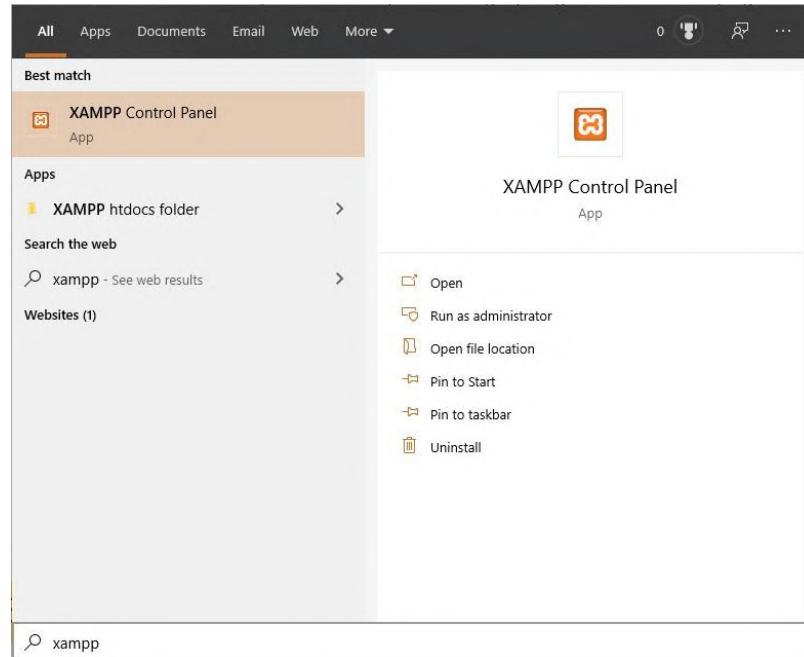


11. Tunggu sampai proses instalasi selesai.

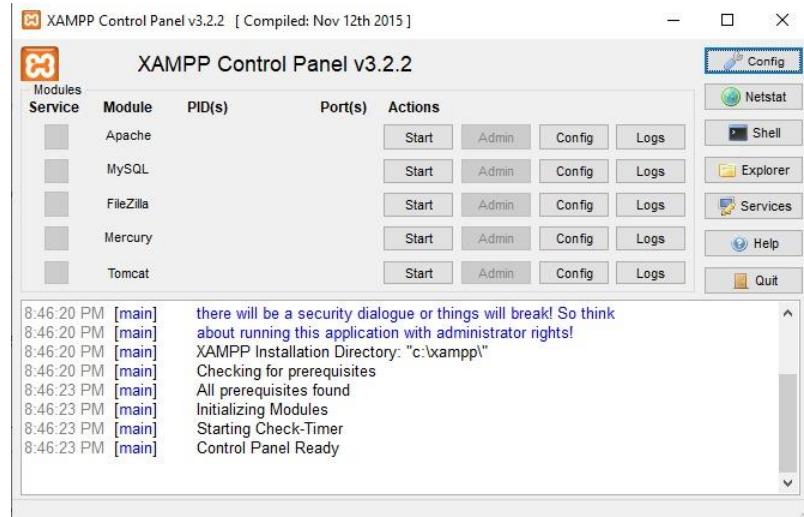


6.1.2 Cara Menjalankan Aplikasi Xampp

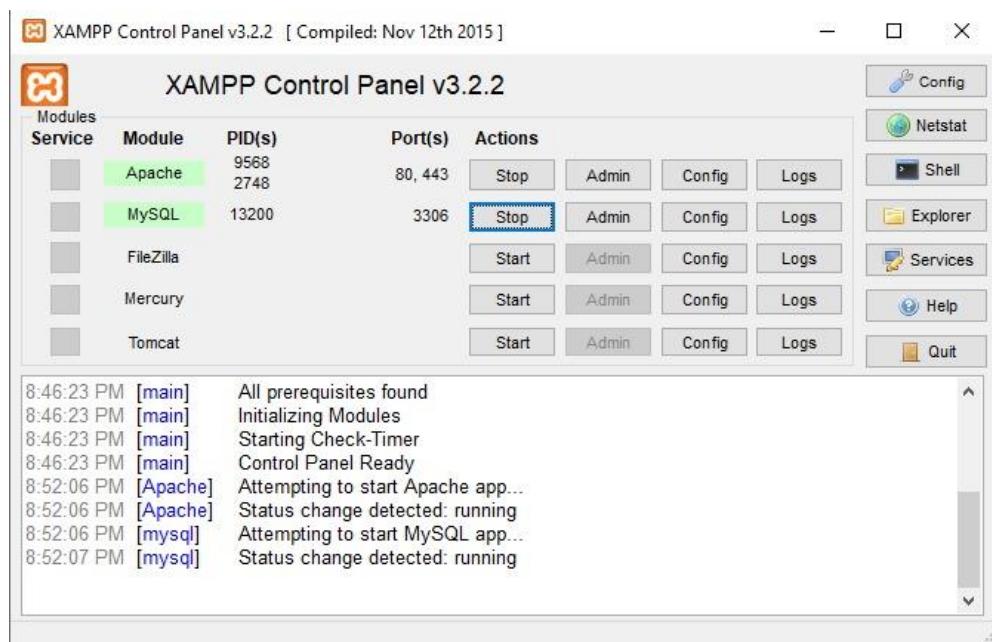
1. Buka aplikasi Xampp bisa melalui pencarian di task bar atau di desktop



2. Klik dua kali pada xampp tersebut, maka akan muncul tampilan seperti berikut:



3. Silahkan klik *Button Start* pada kolom *action* sehingga *Button* tersebut berubah menjadi *stop*.
4. Dengan mengklik *Button* tersebut artinya bahwa aplikasi tersebut telah dijalankan. Biasanya saya menggunakan xampp yang *Start* hanyalah aplikasi Apache dan MySQL karena saya tidak memerlukan aplikasi lainnya.



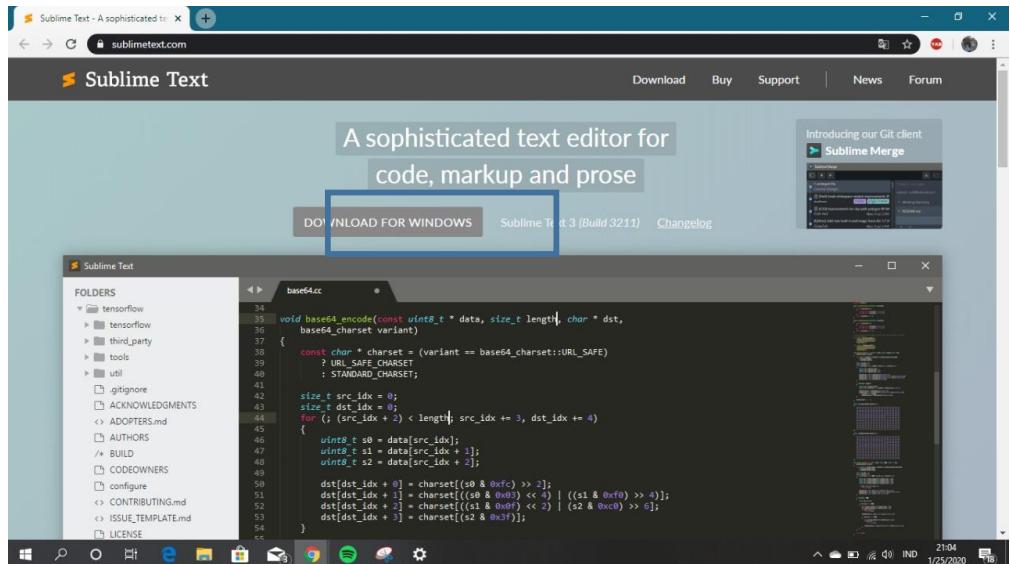
5. Selanjutnya buka *browser* anda dan coba ketikkan <http://localhost/> di *address bar*, jika muncul tampilan seperti gambar dibawah ini berarti proses instalasi telah berhasil.



6.2 Sublime

1. Download terlebih dahulu file sublime di situs resmi sublime,

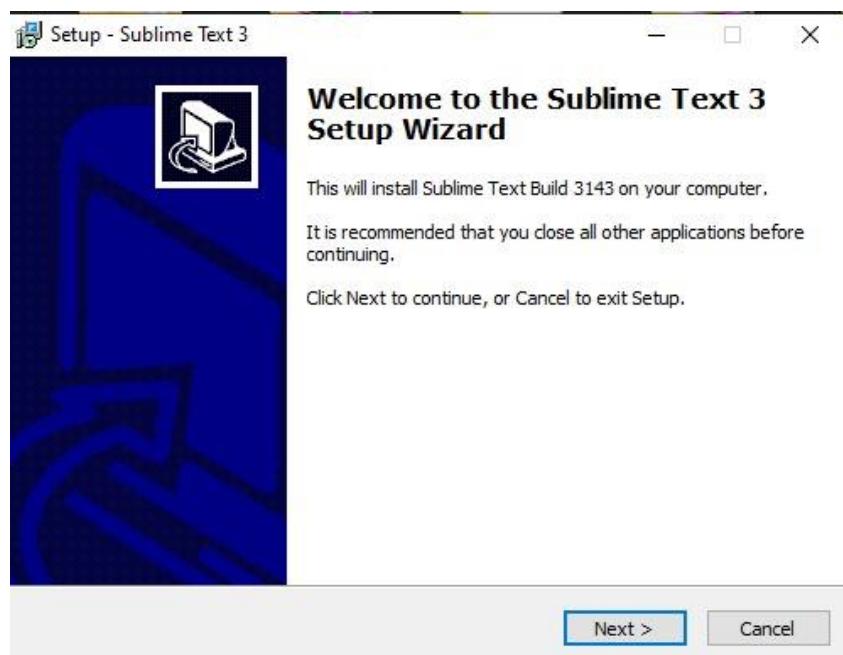
link : <https://www.sublimetext.com/>



- Setelah selesai di *download* klik dua kali pada file .exe

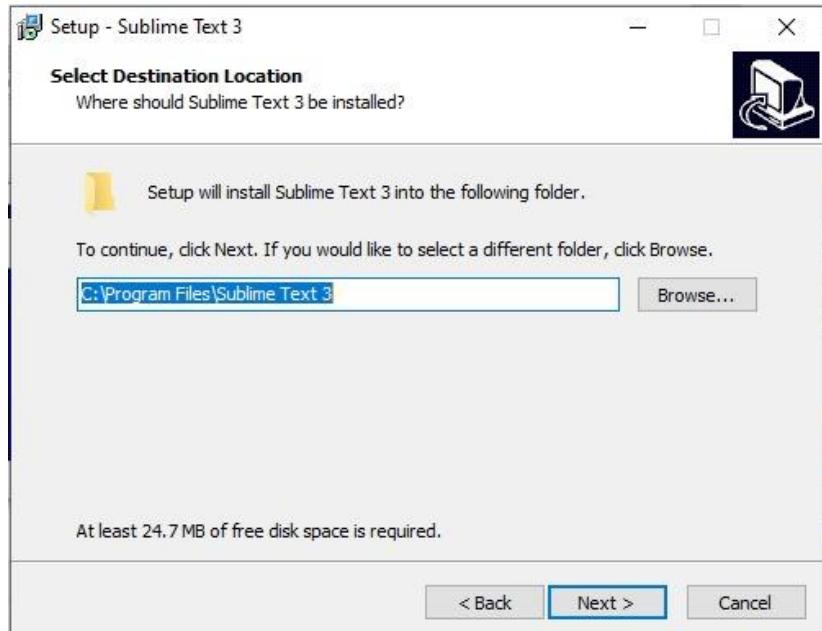


- Selanjutnya akan muncul tampilan awal untuk proses instalasi.

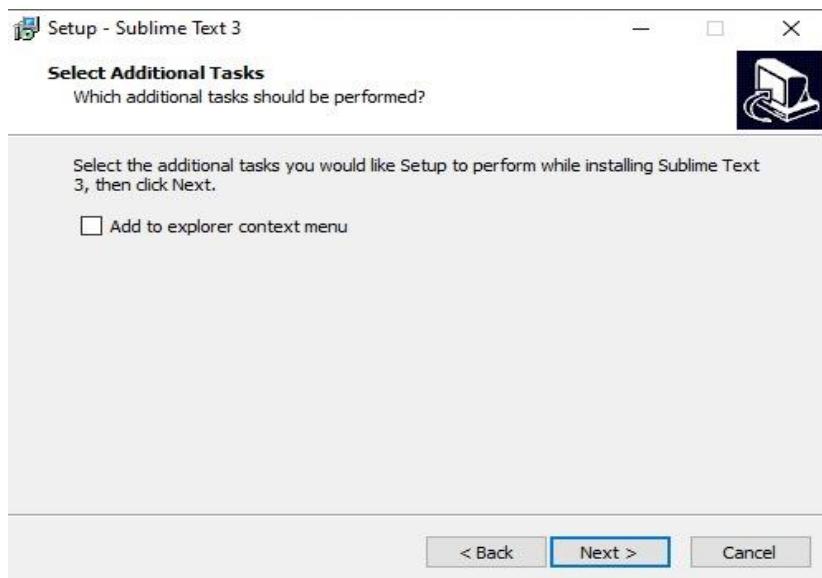


- Klik *Next* untuk melanjutkan.
- Setelah itu akan muncul tampilan library penyimpanan sublime. Secara default akan diarahkan ke lokasi C:\Program Files\Sublime Text 3
- Tetapi jika anda ingin menyimpannya di folder lain anda bisa mengklik *browse* kemudian menentukan secara manual folder yang anda ingin gunakan.

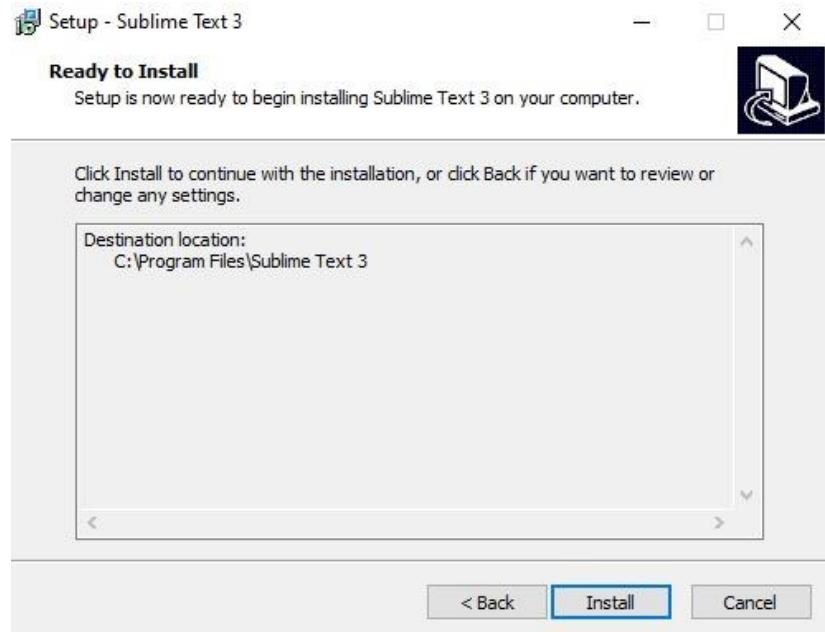
7. Jika sudah selesai lanjutkan dan silahkan klik *Button Next*.



8. Selanjutkan akan muncul tampilan, abaikan dan kli *Next*.



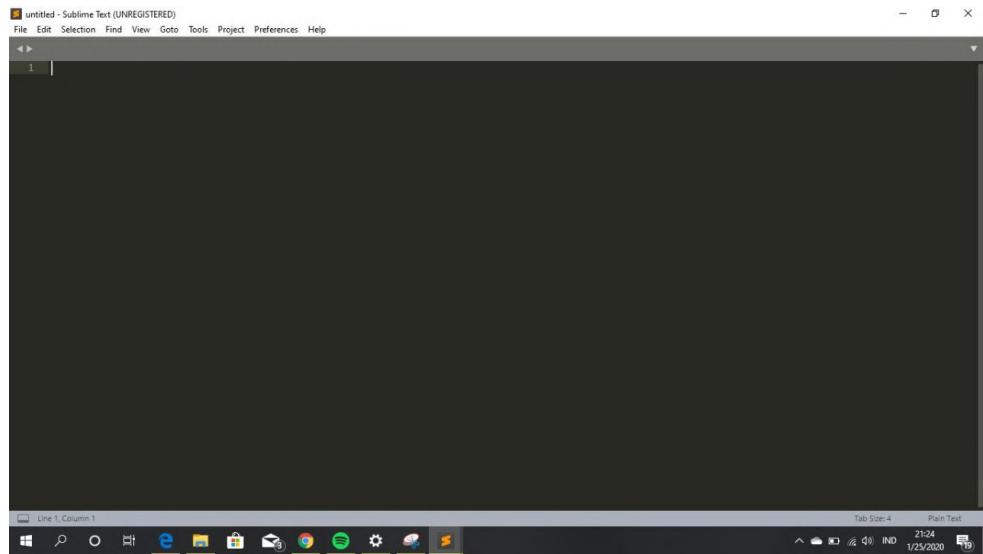
9. Setelah itu klik *install* untuk melanjutkan proses instalasi, tunggu sampai selesai.



10. Setelah proses instalasi berhasil klik *finish*.

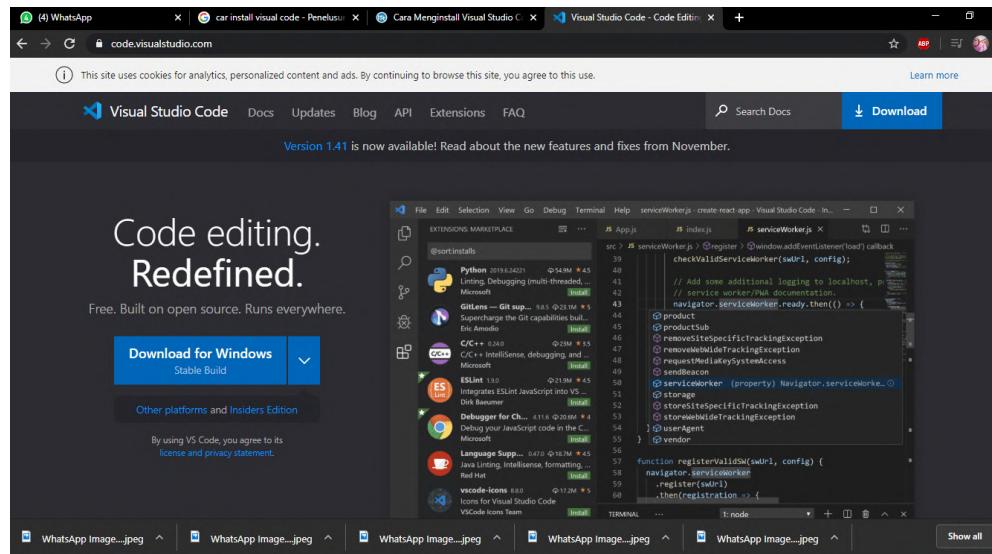


11. Tampilan awal sublime

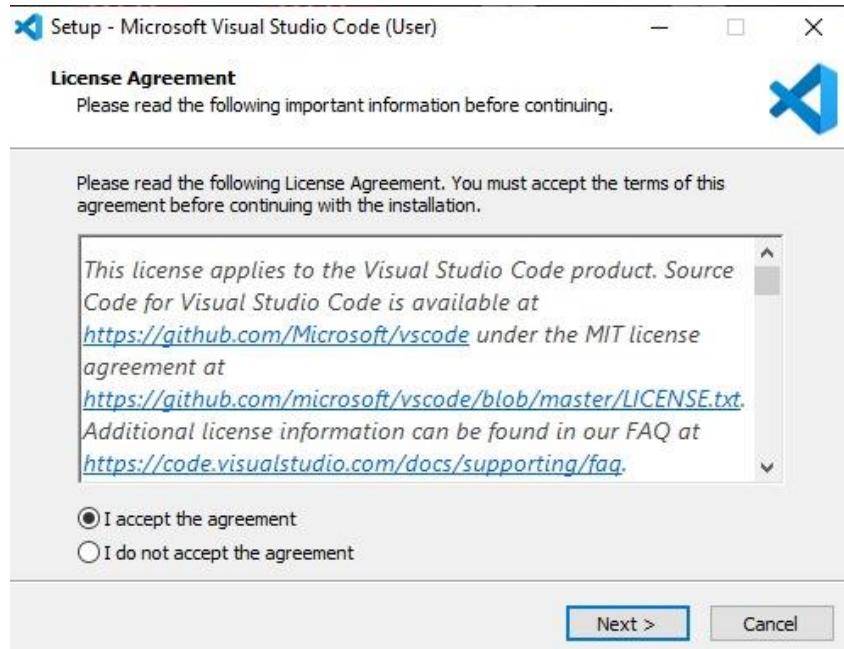


6.3 Visual Code

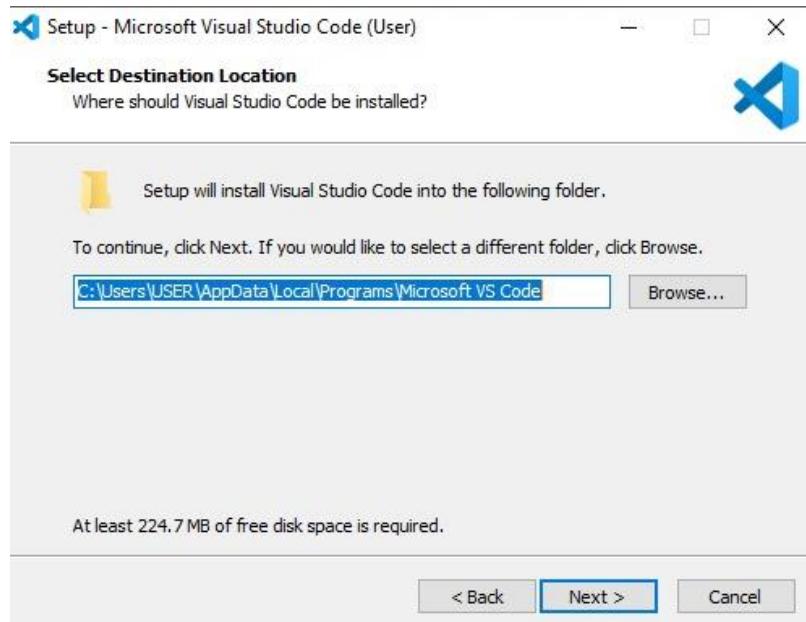
1. Pertama download aplikasi visual code, *link* <https://code.visualstudio.com/>.
2. Jika *link* sudah di buka klik download di pojok kanan . seperti pada gambar dibawah ini.



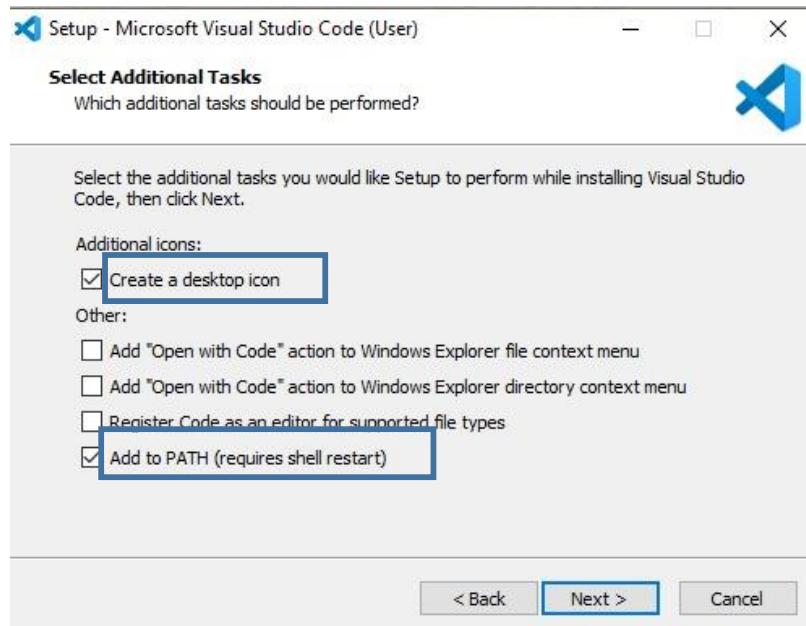
3. Setelah download selesai, buka aplikasi visual code yang telah didownload tadi, maka akan muncul *license agreement*.



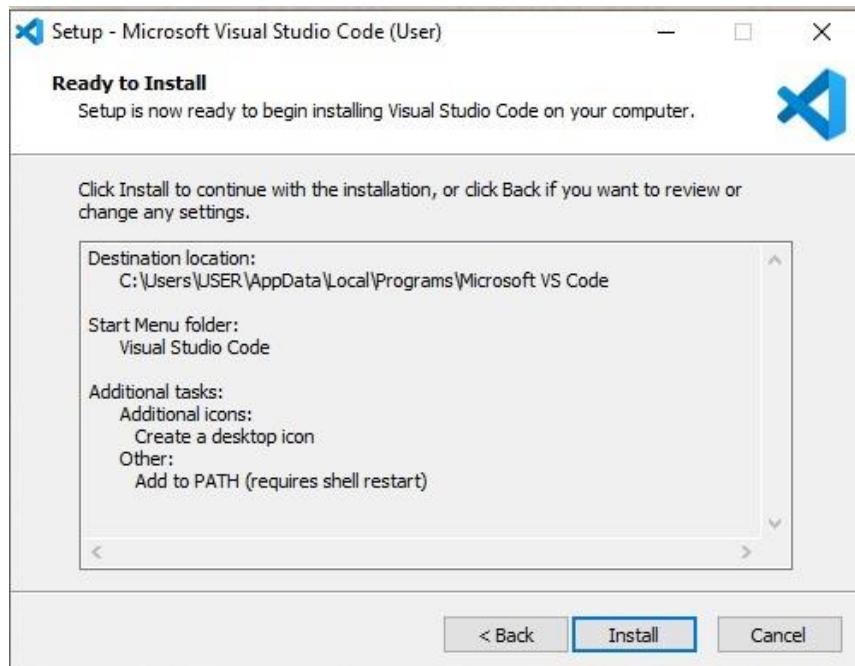
4. Pilih "*I accept the agreement*" untuk menyetujui kebijakan visual code, lalu untuk melanjutkan klik *Next*.
5. Setelah itu akan muncul tampilan library penyimpanan sublime. Secara default akan diarahkan ke lokasi C:\User\USER\AppData\Local\Programs\Microsoft VS Code
6. Tetapi jika anda ingin menyimpannya di folder lain anda bisa mengklik *browse* kemudian menentukan secara manual folder yang anda ingin gunakan.
7. Jika sudah selesai lanjutkan dan silahkan klik *Button Next*.



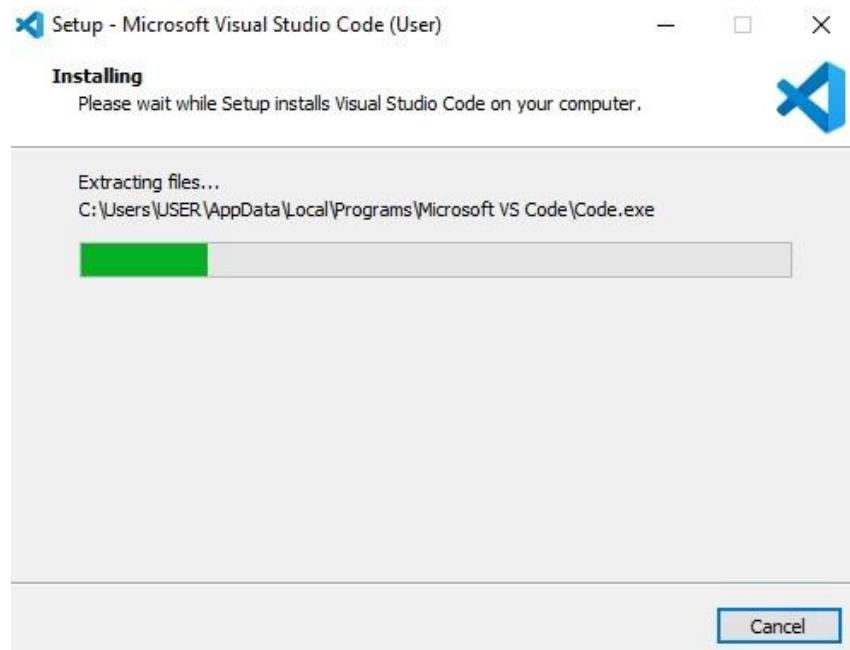
8. Setelah itu akan tampil seperti pada gambar di bawah ini, ceklis “*Create Desktop Icon*”. Jika ingin membuat *shortcut* VSCode-nya, ceklis “*Add to PATH (require shell restart)*”. Jika sudah klik *Next*.



9. Selanjutnya klik *install* untuk melakukan proses instalasi.



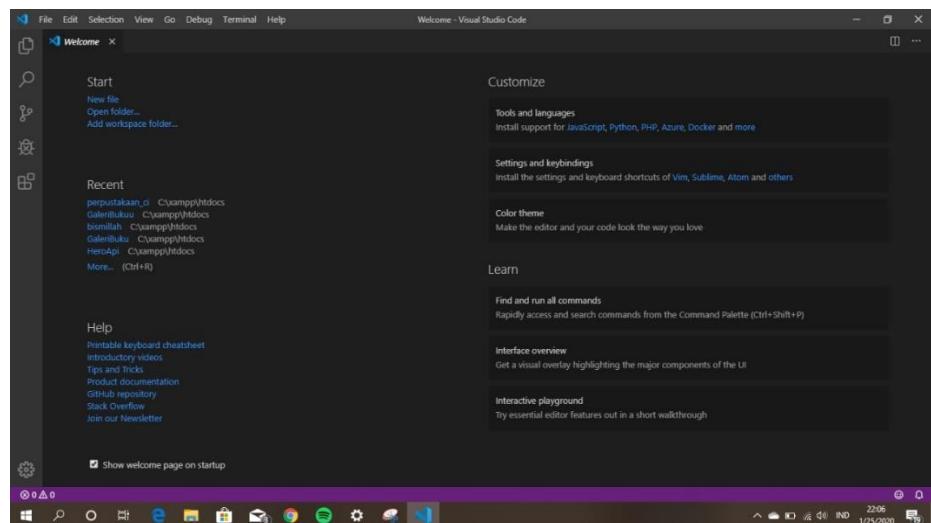
10. Proses instalasi dimulai, tunggu sampai proses selesai.



11. Setelah muncul tampilan seperti gambar dibawah ini, artinya instalasi telah berhasil.



12. Berikut adalah tampilan awal dari visual code.



BAGIAN 7

TEORI FRAMEWORK DAN BAHASA YANG DAPAT DIGUNAKAN

7.1 Framework CodeIgniter

7.1.1 Pengertian Framework

Framework adalah kumpulan perintah atau fungsi dasar yang membentuk aturan-aturan tertentu dan saling berinteraksi satu sama lain sehingga dalam pembuatan aplikasi *website* kita harus mengikuti aturan dari penemuan tersebut titik dengan *Framework* dalam hal ini PHP kita tidak perlu memikirkan kode perintah atau fungsi dasar dari aplikasi *website* kita. Dalam *Framework* ini kita hanya memikirkan apa kode SQL-nya dan ditampilkan ke mana? di dalam *framework* ini juga menunjang koneksi *database* validasi form, GUI, dan keamanan. [20]

Jadi, keuntungan yang dapat diperoleh dari penggunaan *Framework* adalah: [20]

- Waktu pembuatan aplikasi *website* kita jauh lebih singkat kode
- Aplikasi *website* menjadi lebih mudah dibaca karena sedikit dan sifatnya pokok. detailnya adalah kode dari dan ini mungkin tidak perlu dipikirkan.
- *Website* kita menjadi lebih mudah diperbaiki karena kita tidak perlu fokus ke semua komponen kode *website* terutama kode system *framework*.
- Kita tidak perlu lagi membuat kode penunjang aplikasi *website* seperti koneksi *database* validasi form GUI dan keamanan apa

yang ditampilkan dan lain apa saja yang diberikan ke *website* tersebut kita hanya perlu fokus ke permasalahan tersebut.

7.1.2 Pengertian *CodeIgniter*

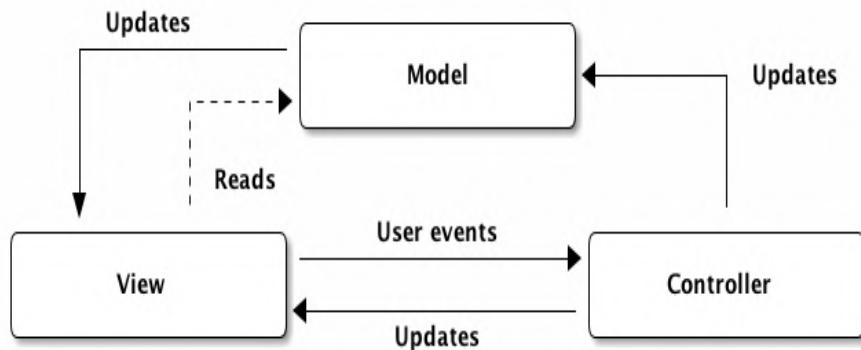
Codeigniter adalah aplikasi *open source* berupa *framework* dengan model MVC (*Model, View, Controller*) untuk membangun *web* dinamis dengan menggunakan PHP. [6]

7.1.3 MVC (*Model View Controller*)

Model view Controller merupakan suatu konsepr yang cukup popular dalam pembangunann aplikasi *web*, berawal pada bahasa pemrograan *Small Talk*, MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, *user interface*, dan bagian yang menjadi control aplikasi. [16]

Tiga jenis Komponen yang membangun MVC dalam satu aplikasi yaitu: [16]

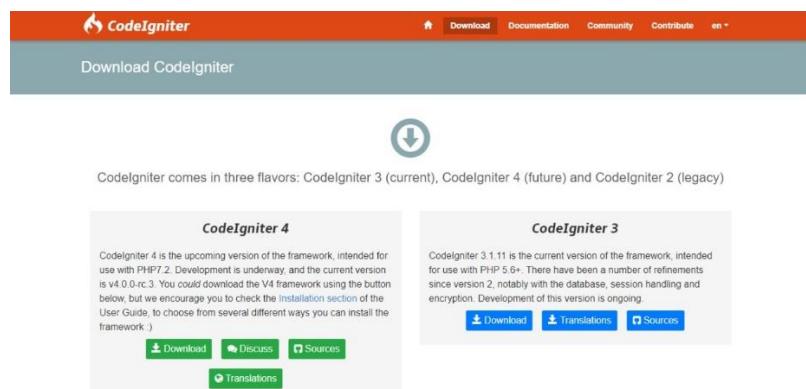
1. *View*, adalah bagian yang menangani *presentation logic*. *View* berfungsi untuk menerima dan mempresentasikan data kepada *user*. Bagian ini tidak memiliki akses langsung terhadap bagian *model*.
2. *Model*, biasanya berhubungan langsung dengan *database* untuk memanipulasi data (*insert, update, delete, search*), menangani validasi dari bagian *controller*, namun tidak dapat berhubungan langsung dengan bagian *view*.
3. *Controller*, adalah bagian yang mengatur hubungan antara bagian *model* dan bagian *view*, *controller* ini berfungsi untuk menerima *request* dan data dari *user* kemudian menentukan apa yang akan diproses oleh aplikasi.



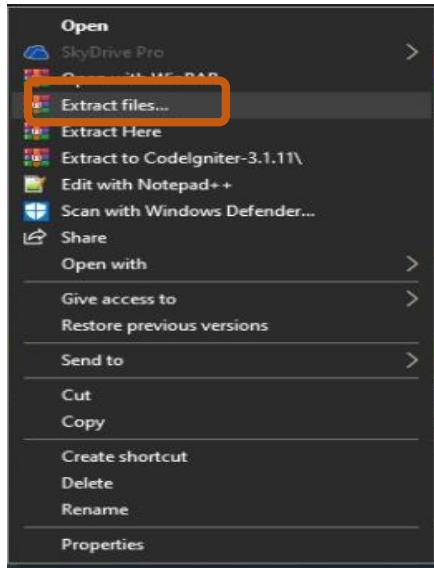
Gambar 6.1 MVC Patterns

7.1.4 Cara Menggunakan *CodeIgniter*

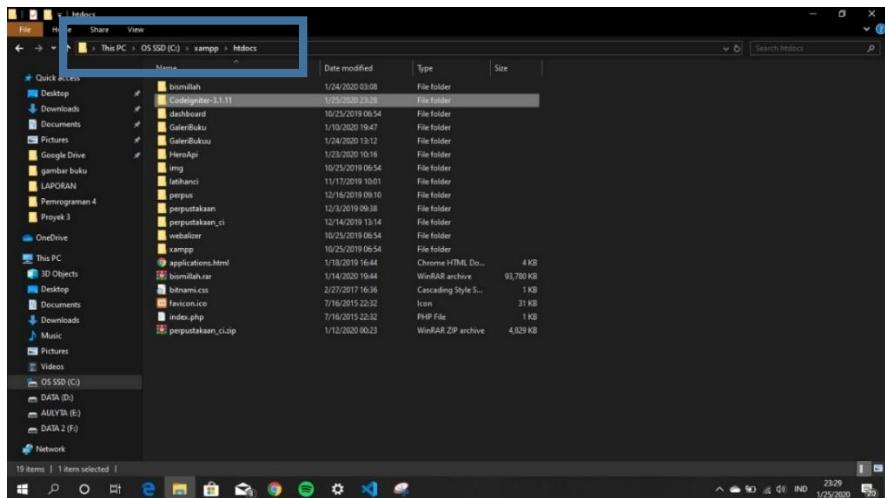
1. Download terlebih dahulu *codeigniter* di *link* <https://codeigniter.com/download>
2. setelah *link* tersebut di buka, pilih sesuai dengan kebutuhan masing masing



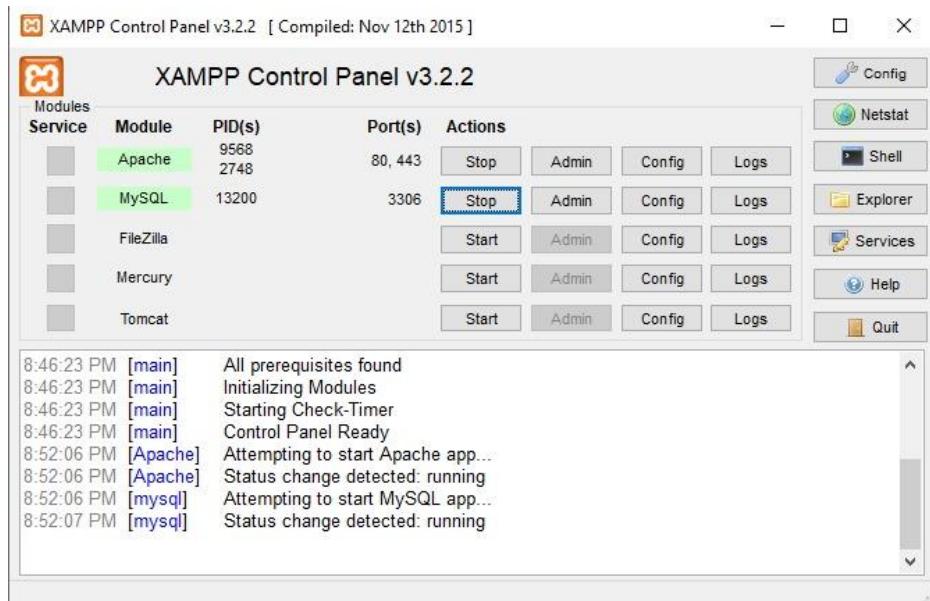
3. kemudian *extract* file yang telah berhasil di *download*.



- Setelah itu pindahkan folder yang telah di *extract* tadi kedalam folder htdocs (C:\xampp\htdocs)

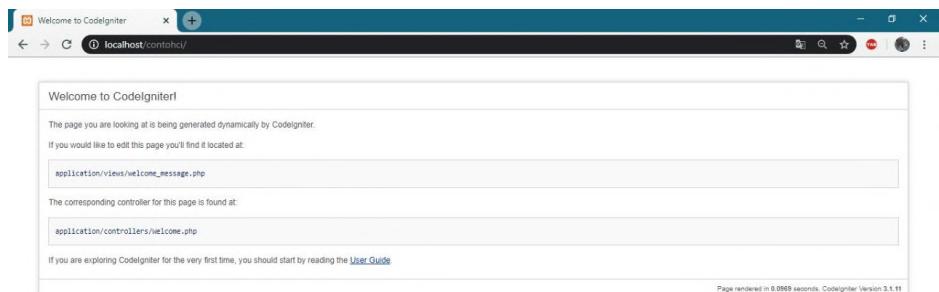


- Kemudian ubah nama file tersebut sesuai dengan yang anda inginkan, disini saya menggunakan nama file contohCI.
- Setelah itu buka aplikasi Xampp dan jalankan xampp dengan mengklik pada bagian *actions* sampai berubah menjadi *stop*.



7. Selanjutnya buka browser dan ketikkan

<http://localhost/contohci/>



8. Selamat *codeigniter* berhasil di *install*.

9. Setelah itu buka VisualCode yang telah di *install* tadi, hal ini dilakukan untuk mengenali struktur direktori pada *codeigniter*.

7.2 Bahasa Pemrograman HTML Dan PHP

7.2.1 Definisi HTML (*Hyper Text Markup Language*)

HTML (*Hyper Text Markup Language*) adalah sekumpulan simbol-simbol atau tag-tag yang dituliskan dalam sebuah file yang dimaksudkan untuk menampilkan halaman pada *web browser*. Tag-tag tadi memberitahu *browser* bagaimana menampilkan halaman *web* dengan lengkap kepada pengguna. Tag-tag HTML selalu diawali dengan dan diakhiri dengan dimana x tag HTML seperti b, i, u dan sebagainya. [21]

7.2.2 Definisi PHP (*Hypertext Preprocessor*)

PHP (*PHP: Hypertext Preprocessor*) adalah sebuah bahasa pemrograman yang perintahnya dilaksanakan server dan kemudian hasilnya ditampilkan pada komputer *client*. PHP juga merupakan HTML embedded, yaitu sintaks PHP yang dituliskan bersamaan dengan sintaks HTML. Jadi PHP dan HTML adalah sinergi dua bahasa pemrograman yang saling menguatkan. Walaupun sebagian orang berpendapat HTML bukan sebuah bahasa pemrograman. [21]

Beberapa kelebihan bahasa pemrograman PHP sebagai berikut: [21]

a. Keamanan

Keamanan sebuah program selain sistem operasi menjadi sangat penting. PHP menyediakan 3 jenis authentikasi *user*, yaitu http authentikasi, penggunaan cookies dan penggunaan session.

b. Integrasi Dengan *Database*

PHP mendukung integritas, kecepatan dan efisiensi akses ke *database* yang kebanyakan menggunakan *database* berjenis

relational seperti MySQL, PostgreSQL, Oracle, SQLite dan lain-lain.

c. *Cross-Platform*

PHP mendukung berbagai jenis sistem operasi seperti semua varian Linux, Microsoft Windows, Mac OS dan lain-lain.

d. Reliability

PHP merupakan salah satu bahasa pemrograman yang bebas *web*. Alasan utama adalah dukungan dokumentasi yang lengkap, aman dan banyak komunitas helpdesk untuk membantu para pengembang *web* sistem yang menggunakan PHP.

e. Harga

PHP berada dalam lisensi GPL (*GNU Public License*). Hal ini berarti bahwa PHP bebas digunakan dan didistribusikan serta gratis.

f. Kemudahan Berimigrasi

Tujuannya adalah memperbaiki kinerja dan menambah fitur-fitur baru. Kelebihan ini karena banyaknya dukungan terhadap PHP sehingga berdampak PHP terus menerus dikembangkan

BAGIAN 8

PERANCANGAN *USER INTERFACE* DARI SISTEM YANG DIBANGUN

Untuk menghasilkan sebuah aplikasi yang mudah dikenali dan digunakan, diperlukan perancangan (desain) yang baik, meliputi desain *user interface*, pembuatan kode (*coding*) dan pengujian. Desain merupakan inti teknis dalam fase pengembangan bagi setiap produk atau sistem yang direkayasa. Tujuan dari perancang adalah untuk menghasilkan suatu model atau representasi dari entitas yang kemudian akan dibangun. Istilah ramah dengan pengguna (*user friendly*) digunakan untuk merujuk pada karakteristik yang dimiliki oleh perangkat lunak atau program aplikasi yang mudah dioperasikan.

Sebagai *user interface* nya dirancang sebuah sistem “Integrase Sistem Informasi: Akses Sistem Informasi Perpustakaan SMPN 41 Bulukumba Dengan Sistem Galeri Buku” yang menampilkan data secara real time. Adapun perangkat lunak yang dibutuhkan adalah sebagai berikut:

- Sistem operasi windows 10
- XAMPP
- Balsamiq Mockups 3.5.16

8.1 Pengertian Antar Muka Pengguna (*User Interface*)

Perancangan antarmuka pengguna merupakan suatu proses yang kompleks, hal ini didasari karena antarmuka pengguna merupakan bagian dari sistem yang akan dikendalikan oleh

pengguna dan merupakan tahap persiapan untuk rancang bangun implementasi. Oleh karena itu untuk mendapatkan sistem yang dapat berjalan sesuai dengan fungsi yang diharapkan perlu pengalaman dalam merancang antarmuka pengguna, kreativitas yang tinggi, analisis tugas dan dapat menyesuaikan dengan kebutuhan serta kemampuan pengguna.[22]

Prinsip-prinsip dalam merancang *user interface* adalah sebagai berikut: [23]

- *User familiarity* / Mudah dikenali: gunakan istilah, konsep dan kebiasaan *user* bukan computer (misal: sistem perkantoran gunakan istilah letters, documents, folders bukan directories, file, identifiers. -- jenis document open office.
- *Consistency* / “selalu begitu”: Konsisten dalam operasi dan istilah di seluruh sistem sehingga tidak membingungkan. -- layout menu di open office mirip dgn layout menu di MS office.
- *Minimal surprise* / Tidak buat kaget *user*: Operasi bisa diduga prosesnya berdasarkan perintah yang disediakan.
- *Recoverability* / pemulihan: *Recoverability* ada dua macam: *Confirmation of destructive action* (konfirmasi terhadap aksi yang merusak) dan ketersediaan fasilitas pembatalan (undo)
- *User guidance* / bantuan: Sistem manual online, *menu help*, *caption* pada *icon* khusus tersedia.
- *User diversity* / keberagaman: Fasilitas interaksi untuk tipe *user* yang berbeda disediakan. Misalnya ukuran huruf bisa diperbesar.

User interface yang baik harus menyatuakan interaksi pengguna (*user interaction*) dan penyajian informasi (*information presentation*). Ada 5 tipe utama interaksi untuk *user interaction*:

8.2 Perancangan *User Interface*

Dari konsep yang telah disusun, maka dapat dilakukan proses perancangan *user interface* aplikasi dengan menyusun kerangka atau *frame* dari halaman *interface*. Adapun rancangan *interface* yang telah dibuat adalah sebagai berikut:

8.2.1 *User Interface* Galeri Buku Registrasi

The screenshot shows a web browser window with the URL http://localhost/Perpustakaan/Frontend/Website/F_register/register. The page title is "Form Registrasi". The form is divided into two sections: "Peminjam" (radio button selected) and "Pemilik" (radio button available). The "Peminjam" section contains fields for "Username" (text input), "Nama (Sesuai KTP)" (text input), "Tempat Lahir" (text input), "Tanggal lahir" (date input with calendar icon), "No Telepon" (text input), and "Alamat Lengkap" (text input). The "Pemilik" section contains fields for "Username" (text input with dropdown arrow), "Password" (text input), "Masukkan Kembali Password" (text input), "Upload KTP" (file input), and "Pilih File" (button). Below the "Pemilik" section is an "Image Not Available" placeholder with a diagonal cross. At the bottom right is a "Register" button.

Gambar 8.1 *User Interface* Galeri Buku Registrasi

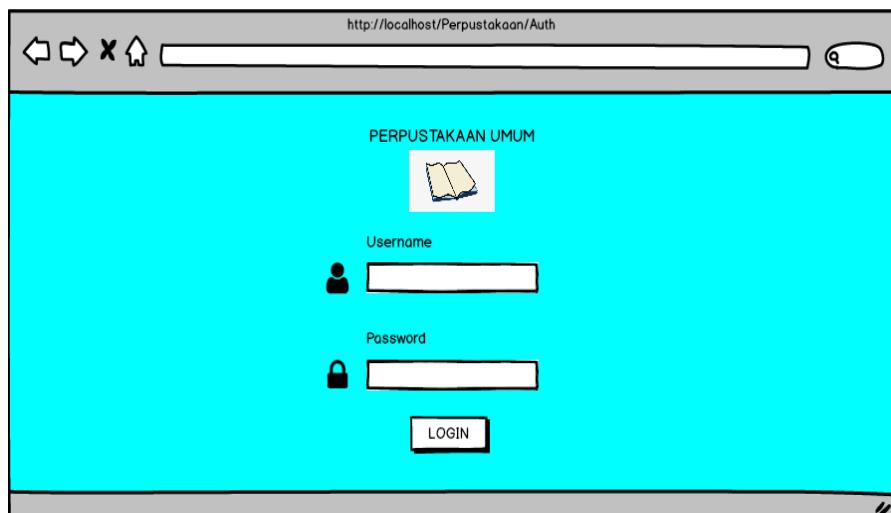
Penjabaran dari bagian pada *frame interface* diatas adalah sebagai berikut:

- *Form* menu registrasi di bagian atas terdiri dari *Icon Circle Outline*, *Text Input*, *Date Chooser*, *Image*, *Cared Square Outline Down* dan *Button*.
- Ada dua *circle outlined* pada *interface* diatas yaitu untuk peminjam dan juga untuk pemilik. *Icon Circle outlined*

merupakan tempat untuk nantinya *user* akan memilih dia akan menjadi pemilik atau peminjam pada saat registrasi.

- Kemudian pada bagian *Text Input* adalah untuk melakukan *inputan* data sesuai dengan permintaan sistem.
- Pada *button* registrasi diklik saat semua data yang di akan di *inputkan* sudah benar.

8.2.2 User Interface Galeri Buku Login



Gambar 8.2 User Interface Galeri Buku Login

Penjabaran dari bagian pada *frame interface* diatas adalah sebagai berikut:

- *Form* menu login dibagian atas terdiri dari *Text Input*, *Icon User*, *Icon Lock*, dan *Button*.
- Ada dua *Text Input* yaitu untuk melakukan *inputan username* dan *password*.
- *Button* adalah tempat untuk mengirimankan data setelah dilakukan *inputan* yang benar sesuai dengan yang ada di *form*.

8.2.3 User Interface Galeri Buku Peminjaman



Gambar 8.3 User Interface Galeri Buku Peminjaman

Penjabaran dari bagian *frame interface* diatas adalah sebagai berikut:

- *Form* menu buku peminjaman di bagian atas terdiri dari *Icon Tachometer*, *Icon Book*, *Icon User*, *Image* dan *Button*.
- Pada *form* bagian list buku merupakan tempat di mana peminjam dapat melihat buku apa saja yang ada didalam galeri buku tersebut.

8.2.4 User Interface Galeri Buku Pengembalian

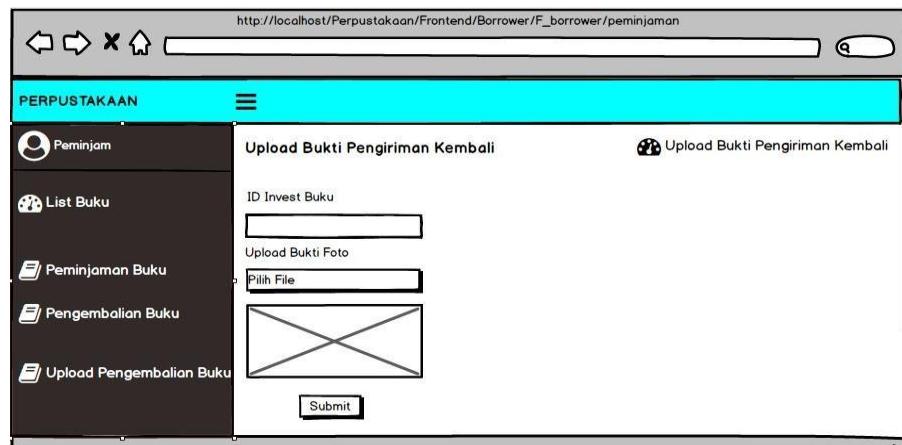
The screenshot shows a web-based application interface for a library system. The URL in the browser bar is http://localhost/Perpustakaan/Frontend/Borrower/F_borrower/peminjaman. The page title is "PERPUSTAKAAN". On the left, there is a sidebar with icons and labels: "Peminjam", "List Buku", "Peminjaman Buku", "Pengembalian Buku", and "Upload Pengembalian Buku". The main content area has a title "Pengembalian Buku" and several input fields: "Nama Lengkap" (Name), "Tempat Lahir" (Place of Birth), "Tanggal Lahir" (Date of Birth), "Gender", "No HP" (Phone Number), "Kode pos" (Postal Code), and "Alamat" (Address). At the bottom right of the form area are two buttons: "Ajukan Pengembalian" (Apply Return) and "Tutup" (Close).

Gambar 8.4 User Interface Galeri Buku Pengembalian

Penjabaran dari bagian diatas pada *frame interface* diatas adalah sebagai berikut:

- *Form* menu buku pengembalian di bagian atas terdiri dari *Icon Tachometer*, *Input Text*, *Text Area*, dan *Button*.
- *Form* diatas adalah bagian pengembalian buku bagi peminjam yang akan melakukan mengembalikan buku.
- Pada *menu* diatas nantinya data yang ada akan secara langsung muncul dengan otomatis, yang mana data tersebut akan sesuai dengan data yang di *inputkan* peminjam pada saat melakukan peminjaman di awal.

8.2.5 User Interface Galeri Buku Upload Bukti Kirim Kembali



Gambar 8.5 User Interface Galeri Buku Bukti Kirim Kembali

Penjabaran dari bagian pada *frame interface* diatas adalah sebagai berikut:

- *Form* menu bukti kirim kembali di bagian atas terdiri dari *Icon Tachometer*, *Icon Book*, *Icon User*, *Text Input*, *Image* dan *Button*.
- Pada *Form* diatas merupakan tempat dimana nantinya peminjam akan melakukan pengiriman bukti kirim kembali kepada admin dan akan diteruskan kepada pemilik buku.
- Peminjam akan melakukan *inputan* data sesuai dengan permintaan yang ada pada *Form* seperti diatas.

8.2.6 User Interface Galeri Upload Buku

The screenshot shows a web-based user interface for managing books. On the left, there is a sidebar with icons and labels: 'Pemilik' (Owner), 'Data Buku' (Book Data), 'Invest Buku' (Invest Book), 'Cek Peminjaman' (Check Borrowing), 'Cek Pengembalian' (Check Return), and 'Upload Bukti Pengiriman' (Upload Shipment Proof). The main area has a header 'PERPUSTAKAAN' and a sub-header 'Proses Invest buku'. It contains several input fields: 'Nama Buku' (Book Name), 'Nama Pengarang' (Author Name), 'Penerbit' (Publisher), 'Tahun Terbit' (Year Published) with a date picker icon, 'Kategori' (Category), 'Resume' (Summary), and 'Upload Foto' (Upload Photo) with a 'Choose Image' button. A 'Submit' button is located at the bottom right.

Gambar 8.6 User Interface Galeri Buku Upload Buku

Penjabaran dari bagian pada *frame interface* diatas adalah sebagai berikut:

- *Form* menu *upload* buku di bagian atas terdiri dari *Icon Tachometer*, *Icon Book*, *Icon User*, *Text Input*, *Date Chooser*, dan *Button*.
- *Form* di atas adalah tempat yang nantinya ketika seorang pemilik akan melakukan investasi buku ke dalam galeri buku.
- Untuk *date chooser* sebagai tempat untuk memudahkan *user* untuk memilih tanggal.
- Pemilik melakukan *inputan* data sesuai dengan yang permintaan yang ada pada *form* diatas dengan menyesuaikan data yang ada di buku yang akan di *invest*.

8.2.7 User Interface Galeri Buku View Peminjaman

The screenshot shows a web-based application interface for a library system. The title bar at the top displays the URL: http://localhost/Perpustakaan/Frontend/Owner/F_owner/cek_peminjaman. The main header is 'PERPUSTAKAAN'. On the left sidebar, there is a menu with icons and labels: 'Pemilik' (User), 'Data Buku' (Book Data), 'Invest Buku' (Invest Book), 'Cek Peminjaman' (Check Loan), 'Cek Pengembalian' (Check Return), and 'Upload Bukti Pengiriman' (Upload Delivery Proof). The main content area is titled 'Cek Peminjaman' and contains two input fields for 'Start Date' and 'End Date', each with a date picker icon. Below these fields is a data grid table with columns: No, Id Peminjam, Tanggal Peminjam, Nama Peminjam, Nama Buku, Harga, Status, and Action. At the bottom of the grid, it says 'Total' and shows the value '0'. There are also navigation icons for back, forward, and search.

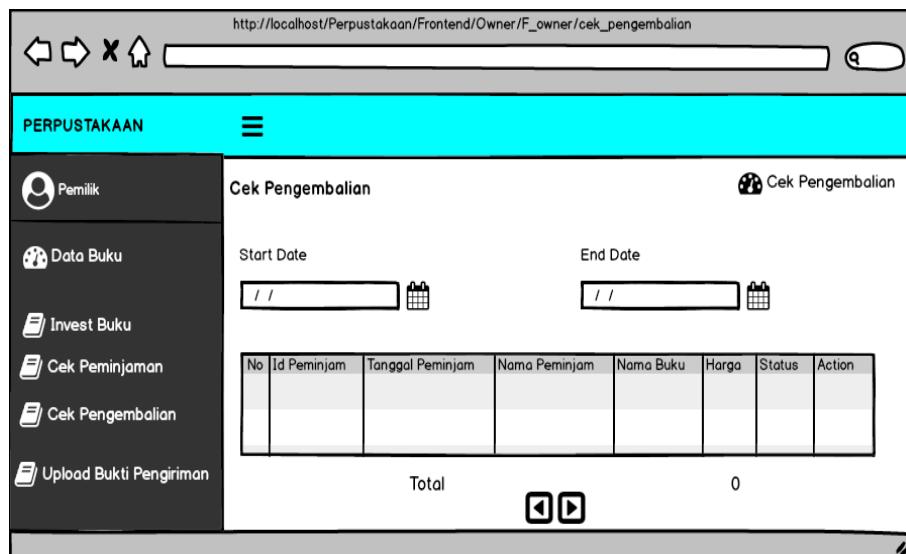
Gambar 8.7 User Interface Galeri Buku View Peminjaman

Penjabaran dari bagian frame view peminjaman diatas adalah sebagai berikut:

- *Form* buku view peminjaman di bagian atas terdiri dari *Icon Tachometer*, *Icon Book*, *Icon User*, *Date Chooser*, *Data Grid*, *Cared Square Outlined Right*, *Cared Square Outlined Left*.
- *Form* diatas terdapat *data grid* sebagai tempat untuk menampilkan hasil proses *query database* kita dalam bentuk tabel.
- *Icon cared squared outlined right* disini berfungsi untuk menampilkan isi slide berikutnya yang ada pada tabel *data grid*. Sedangkan untuk *icon cared squared outlined left* berfungsi untuk kembali ke isi slide sebelumnya yang terdapat table.

- *Form* di atas akan tampil secara otomatis, dimana semua data-data *user* yang telah melakukan peminjaman akan muncul pada tabel.

8.2.8 User Interface Galeri Buku View Pengembalian



Gambar 8.8 User Interface Galeri Buku View Pengembalian

Penjabaran dari bagian pada *frame view pengembalian* diatas adalah sebagai berikut:

- *Form* buku *view pengembalian* di bagian atas terdiri dari *Icon Tachometer*, *Icon Book*, *Icon User*, *Date Chooser*, *Data Grid*, *Cared Square Outlined Right*, *Cared Square Outlined Left*, dan *Button*.
- *Form* diatas terdapat *data grid* sebagai tempat untuk menampilkan hasil proses *query database* kita dalam bentuk tabel.
- Untuk *icon cared squared outlined right* disini berfungsi untuk menampilkan isi slide berikutnya yang ada pada tabel *data grid*. Sedangkan untuk *icon cared squared*

outlined left berfungsi untuk kembali ke isi slide sebelumnya yang terdapat table.

- Selanjutnya pada *form* di atas akan tampil secara otomatis, dimana semua data-data *user* yang telah melakukan pengembalian akan muncul pada tabel.

8.2.9 User Interface Galeri Buku Upload Bukti Kirim Peminjaman

The screenshot shows a web application interface for managing book lending and returning. The main title is "Proses Input Data Pengiriman Peminjam". It contains several input fields: "ID Invest Buku", "ID Pengirim", and "Upload Bukti Foto" which includes a "Choose Image" button. A large red 'X' is overlaid on the "Upload Bukti Foto" input field. Below these fields is a "Submit" button. To the right of the form, there is a link labeled "Proses Invest buku". On the left side, there is a sidebar with icons and labels: "Pemilik", "Data Buku", "Invest Buku", "Cek Peminjaman", "Cek Pengembalian", and "Upload Bukti Pengiriman".

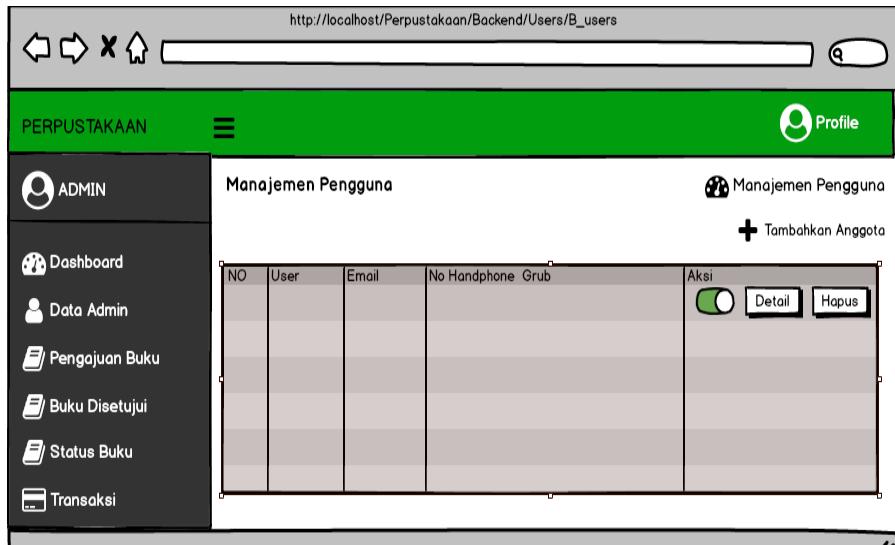
Gambar 8.9 User Interface Galeri Buku Upload Bukti Kirim Kembali

Penjabaran dari bagian pada *frame upload* bukti kirim kembali diatas adalah sebagai berikut:

- *Form upload* bukti kirim kembali pengembalian di bagian atas terdiri dari *Icon Tachometer*, *Icon Book*, *Icon User*, *Text Input*, *Image*, dan *Button*.
- *Form* diatas terdapat *input text* sebagai tempay untuk melakukan *inputan* data yang diminta oleh sistem, yang nantinya hasil *inputan* tersebut akan di proses oleh sistem setelah pemilik mengklik *button* *Submit*.

- Untuk *image* sebagai tempat untuk meng gambar sesuai dengan yang dibutuhkan.

8.2.10 User Interface Galeri Buku Kelola Data User



Gambar 8.10 User Interface Galeri Buku Kelola Data User

Penjabaran dari bagian pada frame view kelola data *user* diatas adalah sebagai berikut:

- Form* kelola data *user* di bagian atas terdiri dari *Icon Tachometer*, *Icon Book*, *Icon User*, *Data Grid*, *Icon Toggle*, *Icon Plus*, dan *Button*.
- Form* diatas terdapat *data grid* sebagai tempat untuk menampilkan hasil *query database* kita dalam bentuk tabel.
- Kemudian untuk *icon plus* diatas sebagai tempat admin untuk dapat menambahkan *user* baru.
- Untuk *icon toggle* pada *form* diatas berfungsi untuk mengaktifkan dan menon-aktifkan *user*.

- Selanjutnya terdapat dua *button* yang jika diklik maka dapat menampilkan detail dari data *user* dan juga dapat menghapus data *user*.

8.2.11 User Interface Galeri Buku Kelola Data Peminjaman

The screenshot shows a web-based application for managing book loans. On the left, a sidebar for 'ADMIN' lists several menu items: Dashboard, Data Admin, Pengajuan Buku, Buku Disetujui, Status Buku, and Transaksi. The main content area is titled 'Check Buku' and contains input fields for 'Tanggal Mulai' (start date) and 'Tanggal Akhir' (end date), along with 'ID Invest Buku' and 'Nama Pemilik'. Below these is a 'Data Grid' table with columns: NO, ID Peminjam, Tgl Peminjaman, Nama Peminjam, Nama Buku, ID Pemilik, Cost, Status, and Action. At the bottom of the grid, there are 'Total' and 'Cost' labels. A 'Check Buku' button is located at the top right of the input area, and a 'Detail' button is in the bottom right corner of the grid.

Gambar 8.11 User Interface Galeri Buku Kelola Data
Peminjaman

Penjabaran dari bagian pada frame kelola data peminjaman diatas adalah sebagai berikut:

- *Form* kelola data peminjaman di bagian atas terdiri dari *Icon Tachometer*, *Icon Book*, *Icon User*, *Date Chooser*, *Data Grid*, *Text Input*, dan *Button*.
- *Form* di atas terdapat *data grid* untuk menampilkan hasil *query database* dalam bentuk tabel.
- *Form* di atas ini akan muncul secara otomatis, yang mana datanya berasal dari data yang *din* oleh *user* yang melakukan peminjaman.

- Terdapat *button* detail di *form* ini, untuk memudahkan admin melihat detail data buku.

8.2.12 User Interface Galeri Buku Kelola Data Pengembalian

NO	ID Peminjam	Tgl Peminjaman	Nama Peminjam	Nama Buku	ID Pemilik	Cost	Status	Action
1								
2								
3								
4								
5								

Gambar 8.12 User Interface Galeri Buku Kelola Data Pengembalian

Penjabaran dari bagian pada *frame* kelola data pengembalian diatas adalah sebagai berikut:

- *Form* kelola data pengembalian di bagian atas terdiri dari *Icon Tachometer*, *Icon Book*, *Icon User*, *Date Chooser*, *Data Grid*, *Text Input*, dan *Button*.
- *Form* di atas terdapat *data grid* untuk menampilkan hasil *query database* dalam bentuk tabel.
- *Form* di atas ini akan muncul secara otomatis, yang mana datanya berasal dari data yang *dinputkan* oleh *user* yang telah melakukan pengembalian.

- *Form* ini terdapat *button* detail, untuk memudahkan admin melihat detail data buku yang telah dikembalikan oleh *user* peminjam.

8.2.13 User Interface Galeri Buku Kelola Data Buku

The screenshot shows a web-based application for managing book data. At the top, there's a header bar with a green background. On the left side of the header is the text 'PERPUSTAKAAN'. To the right of the header is a user profile icon labeled 'Profile'. Below the header, on the left, is a vertical sidebar menu with the following items:

- ADMIN** (highlighted)
- Dashboard
- Data Admin
- Pengajuan Buku
- Buku Disetujui
- Status Buku
- Transaksi

The main content area has a title 'Penyerahan Buku' and two input fields: 'Tanggal Mulai' (with a date picker icon) and 'Tanggal Akhir'. Below these are four text input fields: 'ID Invest Buku', 'ID Pemilik', 'Nama Pemilik', and 'Total' (under the heading 'Cost'). At the bottom of the main area is a data grid table with the following columns:

No	Owner ID	Date Submissi	Owner Narr	Book Nam	Author	Publisher	Publication Ye	Categor	Cost	Status	Action
											<input type="button" value="Detail"/>

Gambar 8.13 User Interface Galeri Buku Kelola Data Buku

Penjabaran dari bagian pada *frame* kelola data buku diatas adalah sebagai berikut:

- *Form* kelola data buku di bagian atas terdiri dari *Icon Tachometer*, *Icon Book*, *Icon User*, *Date Chooser*, *Data Grid*, *Text Input*, dan *Button*.
- *Form* diatas terdapat *data grid* untuk menampilkan hasil *query database* dalam bentuk tabel.
- *Form* diatas ini akan muncul secara otomatis, yang mana datanya berasal dari data yang *inputkan* oleh *user* pemilik yang melakukan *invest* buku.
- *Form* ini terdapat *button* detail, untuk memudahkan admin melihat detail data buku yang masuk dari *user* pemilik.

8.2.14 User Interface Galeri Buku Konfirmasi Bukti Kirim

The screenshot shows a web-based application for library management. The URL in the address bar is http://localhost/Perpustakaan/Backend/Transaksi/B_transaksi/pengiriman_buku. The page has a green header with the text 'PERPUSTAKAAN' and a profile icon labeled 'Profile'. On the left, there is a sidebar titled 'ADMIN' containing links: 'Dashboard', 'Data Admin', 'Pengajuan Buku', 'Buku Disetujui', 'Status Buku', and 'Transaksi'. The main content area is titled 'Check Pengiriman Buku' and includes fields for 'Tanggal Mulai' (with a date picker icon), 'Tanggal Akhir' (with a date picker icon), 'ID Invest Buku' (text input), 'ID Pemilik' (text input), and 'Nama pemilik' (text input). Below these fields is a data grid table with columns: NO, ID Peminjam, Tgl Peminjaman, Nama Peminjam, Nama Buku, ID Pemilik, Cost, Status, and Action. At the bottom of the grid, there are buttons for 'Total' and 'Cost'.

Gambar 8.14 User Interface Galeri Konfirmasi Bukti Kirim

Penjabaran dari bagian pada *frame* konfirmasi Bukti kirim diatas adalah sebagai berikut:

- *Form* kelola data peminjaman di bagian atas terdiri dari *Icon Tachometer*, *Icon Book*, *Icon User*, *Date Chooser*, *Data Grid*, *Text Input*, dan *Button*.
- *Form* diatas terdapat *data grid* untuk menampilkan hasil *query database* dalam bentuk tabel.
- *Form* diatas ini akan muncul secara otomatis, yang mana datanya berasal dari data yang *inputkan* oleh *user* pemilik yang melakukan pengiriman buku ke peminjam.
- *Form* ini terdapat *button* detail, untuk memudahkan admin melihat detail data bukti pengiriman yang dilakukan oleh *user* pemilik.

8.2.15 User Interface Perpustakaan Data Buku



Gambar 8.15 User Interface Perpustakaan Data Buku

Penjabaran dari bagian pada *frame* data buku perpustakaan diatas adalah sebagai berikut:

- *Form* data buku perpustakaan di bagian atas terdiri dari *Icon Tachometer*, *Icon List-Alt*, *Icon Pencil*, *Icon User*, *Image*, dan *Button*.
- Pada *form* diatas ini adalah tempat dimana admin dapat melihat data buku yang ada, dan juga terdapat *button* tambah untuk admin dapat menambahkan buku yang akan di masukkan kedalam sistem.
- Terdapat juga *button* Hapus dan *button* Update. Untuk *button* Hapus sebagai tempat admin dapat melakukan penghapusan jika buku tidak lagi tersedia di perpustakaan, sedangkan *button* Update adalah tempat untuk mengubah data buku jika dalam penginputan sebelumnya terdapat kekeliruan isi data.

8.2.16 User Interface Perpustakaan Data Anggota

The screenshot shows a web-based library management system. At the top, there's a header bar with a search bar containing 'http://localhost/perpustakaan_ci/admin/anggota'. To the right of the search bar is a user icon labeled 'Super Admin'. Below the header, the title 'SISTEM INFORMASI PERPUSTAKAAN SMPN 41 BULUKUMBA' is displayed, followed by a breadcrumb navigation '/ Data Anggota'. On the left side, there's a sidebar menu with icons and labels: 'Dashboard', 'Data User', 'Data Anggota' (which is currently selected), 'Kategori Buku', 'Data Buku', 'Peminjaman', and 'Report'. The main content area is titled 'Data Anggota' and contains a table with member data. The table has columns: NO, ID, NIS, Nama, JK, Kelas, Alamat, No Telepon, and Aksi. There are two rows of data:

NO	ID	NIS	Nama	JK	Kelas	Alamat	No Telepon	Aksi
1	KA0001	11562	Nurul Izza H	3C	Perempuan	Sariasih II	08971363198	HAP UPDA
2	KA0002	11678	Aulyardha	3C	Peremouan	Sarijadi	08937293718	HAP UPDA

At the top right of the main content area, there's a blue 'TAMBAH' button. The bottom right corner of the page has a small logo.

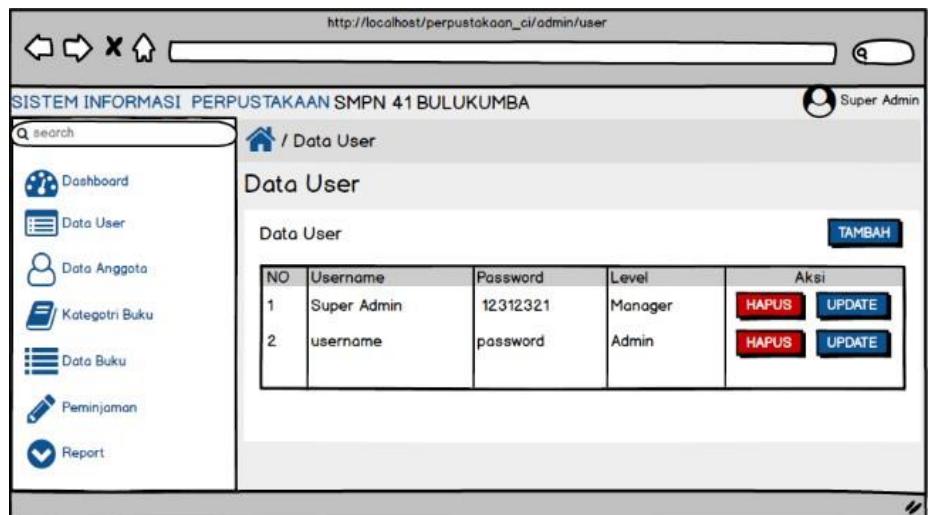
Gambar 8.16 User Interface Perpustakaan Data Anggota

Penjabaran dari bagian pada frame data anggota perpustakaan diatas adalah sebagai berikut:

- *Form* data buku perpustakaan di bagian atas terdiri dari *Icon Tachometer*, *Icon List-Alt*, *Icon Pencil*, *Icon User*, *Data Grid*, dan *Button*.
- Pada *form* diatas ini adalah tempat dimana admin dapat melihat data anggota / siswa, dan juga terdapat *button* tambah untuk admin dapat menambahkan anggota / siswa yang akan di masukkan kedalam sistem.
- Untuk *data grid* diatas sebagai tempat untuk menampilkan *query database* dalam bentuk tabel, yang mana isi dari *data grid* diatas berupa data anggota yang telah terdaftar kedalam sistem.
- Terdapat juga *button* Hapus dan *button* Update. Untuk *button* Hapus sebagai tempat admin dapat melakukan

penghapusan jika anggota / siswa tidak lagi aktif, sedangkan *button Update* adalah tempat untuk mengubah data anggota / siswa jika dalam *penginputan* sebelumnya terdapat kekeliruan isi data.

8.2.17 User Interface Perpustakaan Data User



Gambar 8.17 User Interface Perpustakaan Data User

Penjabaran dari bagian pada *frame* data *user* perpustakaan diatas adalah sebagai berikut:

- *Form* data buku perpustakaan di bagian atas terdiri dari *Icon Tachometer*, *Icon List-Alt*, *Icon Pencil*, *Icon User*, *Data Grid*, dan *Button*.
- *Form user* diatas yang di maksudkan adalah khusus untuk admin perpustakaan.
- Pada *form* diatas ini adalah tempat dimana admin dapat melihat data *user*, dan juga terdapat *button tambah* untuk admin dapat menambahkan admin baru untuk di masukkan kedalam sistem.

- Untuk *data grid* diatas sebagai tempat untuk menampilkan *query database* dalam bentuk tabel, yang mana isi dari *data grid* diatas berupa data *user* yang telah terdaftar kedalam sistem.
- Terdapat juga *button Hapus* dan *button Update*. Untuk *button Hapus* sebagai tempat admin dapat melakukan penghapusan jika *user* tidak lagi aktif, sedangkan *button Update* adalah tempat untuk mengubah data *user* jika dalam penginputan sebelumnya terdapat kekeliruan isi data.

8.2.18 User Interface Perpustakaan Kategori Buku

The screenshot shows a web-based application for library management. The URL in the browser is http://localhost/perpustakaan_ci/admin/kategori. The page title is "SISTEM INFORMASI PERPUSTAKAAN SMPN 41 BULUKUMBA". On the right, there is a user profile for "Super Admin". The left sidebar contains navigation links: Dashboard (with a house icon), Data User (with a person icon), Data Anggota (with a person icon), Kategori Buku (with a book icon, currently selected), Data Buku (with a book icon), Peminjaman (with a pen icon), and Report (with a checkmark icon). The main content area is titled "Kategori" and displays a table titled "Data Kategori". The table has columns: NO, Nama Kategori, Jumlah, and Aksi. The data shown is:

NO	Nama Kategori	Jumlah	Aksi
1	Reality	0	HAPUS UPDATE
2	Fantasy	0	HAPUS UPDATE
3	Komputer	1	HAPUS UPDATE

A "TAMBAH" button is located at the top right of the table area.

Gambar 8.18 User Interface Perpustakaan Kategori Buku

Penjabaran dari bagian pada *frame* kategori buku perpustakaan diatas adalah sebagai berikut:

- Form* kategori buku perpustakaan di bagian atas terdiri dari *Icon Tachometer*, *Icon List-Alt*, *Icon Pencil*, *Icon User*, *Data Grid*, dan *Button*.

- Pada *form* diatas ini adalah tempat dimana admin dapat melihat kategori apa saja yang ada di telah di *inputkan* kedalam sistem, dan juga terdapat *button* tambah untuk admin dapat menambahkan kategori baru buku yang ada.
- Untuk *data grid* diatas sebagai tempat untuk menampilkan *query database* dalam bentuk tabel, yang mana isi dari *data grid* diatas berupa data kategori buku yang telah berhasil *diinput* kedalam sistem.
- Terdapat juga *button* Hapus dan *button* Update. Untuk *button* Hapus sebagai tempat admin dapat melakukan penghapusan jika kategori yang di maksud tidak lagi tersedia, sedangkan *button* Update adalah tempat untuk mengubah kategori jika dalam *penginputan* sebelumnya terdapat kekeliruan isi data.

8.2.19 User Interface Perpustakaan Peminjaman

NO	N. Peminjamn	N. Buku	Tgl Pinjam	Tgl Kembali	Lama Pinjam	Denda	Aksi
1	Aulya(KA0001)	Buku Mahir Web	2019-12-14	2019-12-14	0 Hari(0)	-	Pengembalian Tambah Kembali Perpanj Hapus

Gambar 8.15 User Interface Perpustakaan Peminjaman

Penjabaran dari bagian pada *frame* peminjaman perpustakaan diatas adalah sebagai berikut:

- *Form* peminjaman perpustakaan di bagian atas terdiri dari *Icon Tachometer*, *Icon List-Alt*, *Icon Pencil*, *Icon User*, *Data Grid*, dan *Button*.
- Pada *form* diatas ini adalah tempat dimana admin dapat melihat data anggota / siswa yang melakukan peminjaman pada perpustakaan tersebut, dan juga terdapat *button* tambah untuk admin dapat menambahkan anggota / siswa yang akan melakukan peminjaman.
- Untuk *data grid* diatas sebagai tempat untuk menampilkan *query database* dalam bentuk tabel, yang mana isi dari *data grid* diatas berupa data anggota yang melakukan peminjaman buku.
- Terdapat juga *button* Hapus sebagai tempat admin dapat melakukan penghapusan jika anggota / siswa telah melakukan pengembalian buku. Selain itu terdapat *button* perpanjang jika anggota / siswa ingin melanjutkan peminjaman buku yang sama.
- Untuk *button* pengembalian kanan atas diatas adalah untuk anggota / siswa yang akan melakukan pengembalian buku ke sistem galeri.

Itulah detail dari penjelasan untuk setiap form yang ada di galeri buku dan juga yang ada di perpustakaan yang telah dirancangan dengan sebaik mungkin.

BAGIAN 9

PENULISAN SYNTAX PADA SISTEM YANG TELAH DIRANCANG

Dalam membuat sebuah sistem pasti membutuhkan bahasa pemrograman yang mendukungnya, pada sistem yang telah kami rancang ini, kami menggunakan bahasa pemrograman PHP, CSS, HTML dan JavaScript. Berikut ada beberapa source code dalam membangun sistem yang telah kami rancang.

9.1 Membuat Form *Login*

Form *login* adalah form yang pertama kali tampil ketika program dijalankan. Dengan adanya form *login* ini kita dapat memastikan bahwa *user* yang menggunakan program kita tentunya adalah *user* yang terdaftar dan bukan orang lain yang mungkin akan melakukan kekacauan dalam program yang telah kita buat. Untuk membuat form *login* agar berjalan dengan baik, kita harus membuat *controller*, model dan *view*.

9.1.1 Membuat *Controller*

Berikut adalah syntax pada file *F_login.php*

```
<?php  
defined('BASEPATH') OR exit('No direct script access allowed');  
class F_Login extends CI_Controller {  
    function __construct()  
    {  
        parent::__construct();  
        $this->load->model('Auth_model');  
        $this->load->model('crud_model');  
        $this->load->library('form_validation');  
        date_default_timezone_set("Asia/Jakarta");
```

```

        $url = site_url();
    }

    public function auth()
    {
        $this->form_validation-
>set_rules("reg_code", "User ID", "trim|required");
        $this->form_validation-
>set_rules("reg_password", "Password", "trim|required");
        if ($this->form_validation->run() == true){
            $db = get_instance()->db->conn_id;
            $reg_code = mysqli_real_escape_string($db, $this->input-
>post('reg_code'));
            $reg_password = mysqli_real_escape_string($db, sha1($this->input-
>post('reg_password')));
            $check_validation = $this->Auth_model-
>check_validate($reg_code, $reg_password)->num_rows();
            if($check_validation > 0){
                $data = $this->Auth_model-
>data_validate($reg_code, $reg_password);
                if ($data[0]->status_register == 'Pemilik'){
                    $data_session = array(
                        'reg_code' => $data[0]->register_id,
                        'reg_password' => $data[0]->register_password,
                        'status_register' => $data[0]->status_register
                    );
                } else if ($data[0]->status_register == 'Peminjam'){
                    $data_session = array(
                        'reg_code' => $data[0]->register_id,
                        'reg_password' => $data[0]->register_password,
                        'status_register' => $data[0]->status_register
                    );
                } else {
                    $this->session-
>unset_userdata("reg_code","reg_password","status_register");
                    $this->session-
>set_flashdata('alert_error', 'Anda tidak memiliki akses !!!');
                    redirect(base_url().'Frontend/Website/F_Login/Login');
                    die();
                }
            }
            $this->session->set_userdata($data_session);
        }
    }
}

```

```

$reg_code = $data[0]->register_id;
if($data[0]->status_register == 'Pemilik'){
    redirect(base_url().'Frontend/Owner/F_owner');
    die();
} else {
    redirect(base_url().'Frontend/Borrower/F_borrower');
    die();
}
} else {
    $this->session-
>unset_userdata("reg_code","reg_password","status_register");
    $this->session-
>set_flashdata('alert_error', 'Akun yang anda masukan salah, silakan coba k
embali !');
    redirect(base_url().'Frontend/Website/F_Login/Login');
    die();
}
} else {
    $this->session-
>unset_userdata("reg_code","reg_password","status_register");
    $this->session-
>set_flashdata('alert_error', 'Akun yang anda masukan salah, silakan coba k
embali !');
    redirect(base_url().'Frontend/Website/F_Login/Login');
    die();
}
}

public function Login()
{
    $data['lender_active'] = "";
    $data['form_url'] = site_url('Frontend/Website/F_Login/auth');
    $data['forgot_url'] = site_url('Frontend/Website/F_Login/forgot_passwor
d');
    $this->load->view('Frontend/partial/home', $data);
}

public function signout()
{
    $reg_code = $this->session->userdata('reg_code');
    $reg_password = $this->session->userdata('reg_password');
    $status_register = $this->session->userdata('status_register');
}

```

```

        if (isset($reg_code) AND isset($reg_password) AND isset($reg_access_status)){
            $data = $this->Auth_model->data_validate($reg_code, $status_register);
        }
        $this->session->unset_userdata("reg_code","reg_password","status_register");
        $this->session->sess_destroy();
        redirect(base_url().'Frontend/Website/F_Login/Login');
        die();
    }
}
?>

```

9.1.2 Membuat View *signin.php*

Berikut adalah syntax *signin.php*

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1.0, user-scalable=no">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="msapplication-tap-highlight" content="no">
    <meta name="description" content="Author: Heatcliffs31 - achmad.azs.dz@gmail.com">
    <meta name="keywords" content="heatcliffs31">
    <title><?php echo $this->config->item('site_title') ?></title>
    <!-- Favicons-->
    <link rel="icon" href="<?php echo base_url() ?>uploads/base-img/favicon/favicon-32x32.png" sizes="32x32">
    <!-- Favicons-->
    <link rel="apple-touch-icon-precomposed" href="<?php echo base_url() ?>uploads/base-img/favicon/apple-touch-icon-152x152.png">
    <!-- For iPhone -->
    <meta name="msapplication-TileColor" content="#00bcd4">

```

```

<meta name="msapplication-
TileImage" content="<?php echo base_url() ?>uploads/base-
img/favicon/mstile-144x144.png">
<!-- For Windows Phone -->
<!-- CORE CSS-->
<link href="<?php echo base_url();?>assets/materialize/css/materialize.cs
s" type="text/css" rel="stylesheet" media="screen,projection">
<link href="<?php echo base_url();?>assets/materialize/css/style.css" typ
e="text/css" rel="stylesheet" media="screen,projection">
<!-- Custome CSS-->
<link href="<?php echo base_url();?>assets/materialize/css/custom/custo
m.css" type="text/css" rel="stylesheet" media="screen,projection">
<link href="<?php echo base_url();?>assets/materialize/css/layouts/page-
center.css" type="text/css" rel="stylesheet" media="screen,projection">
<!-- INCLUDED PLUGIN CSS ON THIS PAGE -->
<link href="<?php echo base_url();?>assets/materialize/js/plugins/sweetalert
/dist/sweetalert.css" type="text/css" rel="stylesheet" media="screen,pro
jection">
<link href="<?php echo base_url();?>assets/materialize/js/plugins/prism/p
rism.css" type="text/css" rel="stylesheet" media="screen,projection">
<link href="<?php echo base_url();?>assets/materialize/js/plugins/perfect-
scrollbar/perfect-
scrollbar.css" type="text/css" rel="stylesheet" media="screen,projection">
</head>
<body class="darkbrown">
<!-- Start Page Loading -->
<div id="loader-wrapper">
    <div id="loader"></div>
    <div class="loader-section section-left"></div>
    <div class="loader-section section-right"></div>
</div>
<!-- End Page Loading -->
<div id="Login-page" class="row">
    <div id="default-validation" class="col s12 z-depth-4 card-panel">
        <form class="Login-
form" action="<?php echo base_url(); ?>Auth/signin" method="post" enctype="
multipart/form-data">
            <div class="row">
                <div class="input-field col s12 center">

```

```

        <p class="center login-form-text" style="font-size: 14px;font-weight: bold;font-family: serif;color: #482302;">Sistem Galeri Buku </p>
        
    </div>
</div>
<?php if ($this->session->flashdata('alert_success')) { ?>
<div id="card-alert" class="card green">
    <div class="card-content white-text">
        <p><?php echo $this->session->flashdata('alert_success'); ?></p>
    </div>
    <button type="button" class="close white-text" data-dismiss="alert" aria-label="Close">
        <span aria-hidden="true">x</span>
    </button>
</div>
<?php } ?>
<?php if ($this->session->flashdata('alert_error')) { ?>
<div id="card-alert" class="card red">
    <div class="card-content white-text">
        <p><?php echo $this->session->flashdata('alert_error'); ?></p>
    </div>
    <button type="button" class="close white-text" data-dismiss="alert" aria-label="Close">
        <span aria-hidden="true">x</span>
    </button>
</div>
<?php } ?>
<div class="row">
    <div class="input-field col s12">
        <i class="mdi-social-person-outline prefix"></i>
        <input id="username" name="users_username" type="text" class="validate" minlength="4" required="true">
        <label for="username" data-error="Minimal 4 karakter">Username</label>
    </div>
</div>
<div class="row">
    <div class="input-field col s12">

```

```

        <i class="mdi-action-lock-outline prefix"></i>
        <input id="password" name="users_password" type="password" class="validate" minlength="8" required="true">
            <label for="password" data-
error="Minimal 8 karakter">Password</label>
        </div>
    </div>
    <div class="row">
        <div class="input-field col s12">
            <button type="submit" class="btn waves-effect waves-light col s12" style="background-color: #00872b ;">Login</button>
        </div>
    </div>
    <div class="row">
        <div class="input-field col s12 m12 l12">
            <p class="margin center medium-small"><a href="=php echo site_url('auth/forgot_password') ?&gt;" style="color: #482302;"&gt;Forgot Password ?&lt;/a&gt;&lt;/p&gt;
        &lt;/div&gt;
    &lt;/div&gt;
    &lt;/form&gt;
&lt;/div&gt;
&lt;!-- jQuery Library --&gt;
&lt;script type="text/javascript" src="<?=php echo base_url();?&gt;assets/materialize/js/plugins/jquery-1.11.2.min.js"&gt;&lt;/script&gt;
&lt;!--materialize js--&gt;
&lt;script type="text/javascript" src="<?=php echo base_url();?&gt;assets/materialize/js/materialize.js"&gt;&lt;/script&gt;
&lt;!--prism--&gt;
&lt;script type="text/javascript" src="<?=php echo base_url();?&gt;assets/materialize/js/plugins/prism/prism.js"&gt;&lt;/script&gt;
&lt;!--scrollbar--&gt;
&lt;script type="text/javascript" src="<?=php echo base_url();?&gt;assets/materialize/js/plugins/perfect-scrollbar/perfect-scrollbar.min.js"&gt;&lt;/script&gt;
&lt;!--plugins.js - Some Specific JS codes for Plugin Settings--&gt;
&lt;script type="text/javascript" src="<?=php echo base_url();?&gt;assets/materialize/js/plugins.js"&gt;&lt;/script&gt;
&lt;script type="text/javascript" src="<?=php echo base_url();?&gt;assets/materialize/js/plugins/sweetalert/dist/sweetalert.min.js"&gt;&lt;/script&gt;
</pre

```

```

<!--custom-script.js - Add your own theme custom JS-->
<script type="text/javascript" src=<?php echo base_url();?>assets/mate
rialize/js/custom-script.js"></script>
<script type="text/javascript" src=<?php echo base_url();?>assets/LTE/
plugins/bootbox.min.js"></script>
</body>
</html>

```

9.1.3 Membuat Modul *Auth_model.php*

Berikut adalah syntax Auth.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class Auth_model extends CI_Model {
    public function get_access($users_username, $users_password)
    {
        $this->db-
        >select("tbu.id_backend_users, tbu.users_name, tbg.groups_name, tbu.id_back
end_groups, tbu.users_access_status")
        ->from("tb_users_backend tbu")
        -
        >join("tb_backend_groups tbg", "tbг.id_backend_groups = tbu.id_backend_grou
ps");
        $this->db->where("tbu.users_username", $users_username);
        $this->db->where("tbu.users_password", $users_password);
        return $this->db->get()->result();
    }
    public function get_access_users($reg_code, $reg_password, $reg_status)
    {
        $this->db->select("tfr.*")
        ->from("tb_users_frontend tfr");
        $this->db->where("tfr.register_id", $reg_code);
        $this->db->where("tfr.register_password", $reg_password);
        $this->db->where("tfr.status_register", $reg_status);
        return $this->db->get()->result();
    }
    public function check_validate($reg_code, $reg_password)
    {
        $this->db->select("tfr.*")
        ->from("tb_users_frontend tfr");
        $this->db->where("tfr.register_password", $reg_password);
    }
}

```

```

        $this->db-
>where("(tfr.register_id = '$reg_code' OR tfr.register_username = '$reg_cod
e')");
        return $this->db->get();
    }
    public function data_validate($reg_code, $reg_password)
    {
        $this->db->select("tfr.*")
        ->from("tb_users_frontend tfr");
        $this->db->where("tfr.register_password", $reg_password);
        $this->db-
>where("(tfr.register_id = '$reg_code' OR tfr.register_username = '$reg_cod
e')");
        return $this->db->get()->result();
    }
}
?>

```

9.1.4 Membuat Model *Crud_model.php*

Berikut adalah syntax *Crud_model.php*

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class Crud_model extends CI_Model {
    public function get_data($table, $where)
    {
        return $this->db->get_where($table,$where);
    }
    public function get_data_table($table)
    {
        return $this->db->get($table);
    }
    public function insert($table, $data)
    {
        $this->db->insert($table, $data);
    }
    public function update($table, $field, $id, $data)
    {
        $this->db->where($field, $id);
        $this->db->update($table, $data);
    }
}

```

```

public function delete($table, $field, $id)
{
    $this->db->where($field, $id);
    $this->db->delete($table);
}

public function deleteMany($table, $field, $id)
{
    $this->db->where_in($field, $id);
    $this->db->delete($table);
}

public function get_setting()
{
    $this->db->select("ts.*")
        ->from("tb_setting ts");
    $this->db->order_by("ts.id_setting", "ASC");
    return $this->db->get()->result();
}

public function get_setting_by_id($id_setting)
{
    $this->db->select("tam.*")
        ->from("tb_setting tam");
    $this->db->where("tam.id_setting", $id_setting);
    return $this->db->get()->result();
}

?>

```

9.2 Membuat Form Registrasi

Form registrasi adalah sebuah form yang menyimpan informasi orang yang ingin mendaftar untuk bergabung sebagai anggota dalam *web* yang kita buat. Untuk membuat form ini, sama seperti saat membuat form *login* sebelumnya, kita harus membuat file *controller*, *model* dan *view*.

9.2.1 Membuat Controller *F_register.php*

Berikut adalah syntax dalam file *F_register.php*

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');
class F_register extends CI_Controller {
    function __construct()
    {
        parent::__construct();
        $this->load->library('form_validation');
        $this->load->helper('url');
        $this->load->model('crud_model');
        $this->load->model('Front_data/register_model');
        date_default_timezone_set("Asia/Jakarta");
    }
    public function register()
    {
        $navigation['lender_active'] = "";
        $navigation['style_register'] = "orange";
        $this->load->helper(array('form', 'url'));
        $this->load->helper('file');
        $data['form_url'] = site_url('Frontend/Website/F_Login/auth');
        $this->load->view('Frontend/partial_register/header');
        $this->load->view('Frontend/partial_register/navigation');
        $this->load->view('Frontend/register/register', $data);
        $this->load->view('Frontend/partial_register/footer_register');
        $this->load->view('Frontend/partial_register/sidebar');
    }
    public function register_peminjam()
    {
        $this->form_validation->set_rules("register_email", "", "required");
        $this->form_validation-
        >set_rules("register_username1", "", "required");
        $this->form_validation-
        >set_rules("register_password1", "", "trim|required");
        $this->form_validation-
        >set_rules("register_re_password1", "", "trim|required");
        $this->form_validation-
        >set_rules("register_status1", "", "trim|required");
    }
}
```

```

    $this->form_validation-
>set_rules("bio_fullname1", "", "trim|required");
    $this->form_validation-
>set_rules("bio_birthplace1", "", "trim|required");
    $this->form_validation-
>set_rules("bio_birthdate1", "", "trim|required");
    $this->form_validation-
>set_rules("bio_gender1", "", "trim|required");
    $this->form_validation->set_rules("bio_phone1", "", "required");
    $this->form_validation-
>set_rules("bio_address1", "", "trim|required");
    $this->form_validation-
>set_rules("bio_post_code1", "", "trim|required");
    if ($this->form_validation->run() == true){
        // post
        $register_email = $this->input->post('register_email');
        $register_username = $this->input->post('register_username1');
        $register_password = sha1($this->input-
>post('register_password1'));
        $register_re_password = sha1($this->input-
>post('register_re_password1'));
        $register_status = $this->input->post('register_status1');
        $bio_fullname = $this->input->post('bio_fullname1');
        $bio_birthplace = $this->input->post('bio_birthplace1');
        $bio_birthdate = $this->input->post('bio_birthdate1');
        $bio_birthdate = date("Y-m-d", strtotime($bio_birthdate));
        $bio_gender = $this->input->post('bio_gender1');
        $bio_phone = $this->input->post('bio_phone1');
        $bio_address = $this->input->post('bio_address1');
        $bio_post_code = $this->input->post('bio_post_code1');
        $where = array('register_email' => $register_email );
        $check_validation_email = $this->crud_model-
>get_data("tb_users_frontend", $where)->num_rows();
        // parameter awal
        $date = date('Y-m-d H:i:s');
        $date_birth = date('Y-m-d');
        $reg_ID = $this->crud_model->get_setting();
        $idMax = $this->register_model->register_code($reg_ID[0]-
>reg_ID_bor);
        $noUrut = (int) substr($idMax[0]->maxID,4,8);
    }
}

```

```

    $noUrut++;
    $newID = $reg_ID[0]->reg_ID_bor sprintf("%08s",$noUrut);
    $random_ref = mt_rand(10000000, 99999999);
    $newref = "REF-".$random_ref;
    $random = rand(100000,99999999);
    $where = array('register_email' => $register_email );
    $check_validation_email = $this->crud_model-
>get_data("tb_users_frontend",$where)->num_rows();
    $where = array('register_username' => $register_username );
    $check_validation_username = $this->crud_model-
>get_data("tb_users_frontend",$where)->num_rows();
    if($check_validation_email != 0){
        $this->session-
>set_flashdata('cache_email', $register_email);
        $this->session-
>set_flashdata('cache_username', $register_username);
        $this->session-
>set_flashdata('cache_status', $register_status);
        $this->session-
>set_flashdata('cache_bio_fullname', $bio_fullname);
        $this->session-
>set_flashdata('cache_bio_birthplace', $bio_birthplace);
        $this->session-
>set_flashdata('cache_bio_birthdate', $bio_birthdate);
        $this->session-
>set_flashdata('cache_bio_gender', $bio_gender);
        $this->session-
>set_flashdata('cache_bio_phone', $bio_phone);
        $this->session-
>set_flashdata('cache_bio_address', $bio_address);
        $this->session-
>set_flashdata('cache_bio_post_code', $bio_post_code);
        $this->session-
>set_flashdata('alert_error', 'Email anda tidak tersedia !');
        redirect(base_url().'Frontend/Website/F_register/register');
        die;
    }
    if($check_validation_username != 0){
        $this->session-
>set_flashdata('cache_email', $register_email);

```

```

        $this->session-
>set_flashdata('cache_username', $register_username);
        $this->session-
>set_flashdata('cache_status', $register_status);
        $this->session-
>set_flashdata('cache_bio_fullname', $bio_fullname);
        $this->session-
>set_flashdata('cache_bio_birthplace', $bio_birthplace);
        $this->session-
>set_flashdata('cache_bio_birthdate', $bio_birthdate);
        $this->session-
>set_flashdata('cache_bio_gender', $bio_gender);
        $this->session-
>set_flashdata('cache_bio_phone', $bio_phone);
        $this->session-
>set_flashdata('cache_bio_address', $bio_address);
        $this->session-
>set_flashdata('cache_bio_post_code', $bio_post_code);
        $this->session-
>set_flashdata('alert_error', 'Username anda sudah terpakai !');
        redirect(base_url().'Frontend/Website/F_register/register');
        die;
    }

    $name_file = $bio_fullname.".".time();
    $config['upload_path'] = './uploads/ktp_peminjam';
    $config['allowed_types'] = 'gif|jpg|png|jpeg';
    $config['max_size']      = '5120';
    $config['file_name']     = $name_file;
    $this->load->library('upload', $config);
    if(!$this->upload->do_upload('bio_upload1')){
        $this->session-
>set_flashdata('alert_error', 'Upload KTP/KTS gagal, ukuran file terlalu besar atau format file salah !');
        redirect(base_url().'Frontend/Website/F_register/register');
        die();
    } else {
        $data_register = array(
            'register_id' => $newID,
            'register_email' => $register_email,
            'register_username' => $register_username,

```

```

        'register_password' => $register_password,
        'register_date' => $date,
        'register_activation_code' => $random,
        'register_status' => 'Activated',
        'status_register' => $register_status
    );
    $insert_register = $this->crud_model-
>insert('tb_users_frontend',$data_register);
    if($insert_register === false){
        $this->session-
>set_flashdata('alert_error', 'Gagal menyimpan data');
        redirect(base_url().'Frontend/Website/F_register/register');
        die();
    } else {
        if (empty($bio_ref_code)){
            $bio_ref_code = "-";
        } else {
            $date = date('Y-m-d');
        }
        $file_data = $this->upload->data();
        $data_bio = array(
            'register_id' => $newID,
            'bio_fullname' => $bio_fullname,
            'bio_place_birth_date' => $bio_birthplace,
            'bio_birth_date' => date('Y-m-
d', strtotime($bio_birthdate)), // date()format
            'bio_gender' => $bio_gender,
            'bio_phone' => $bio_phone,
            'bio_address' => $bio_address,
            'bio_post_code' => $bio_post_code,
            'bio_photo_ktp' => $file_data['file_name'],
        );
        $insert_bio = $this->crud_model-
>insert('tb_borrower_bio',$data_bio);
    }
    if($insert_bio === false){
        $this->session-
>set_flashdata('alert_error', 'Gagal menyimpan data');
        redirect(base_url().'Frontend/Website/F_register/register');
        die();
    }
}

```

```

} else {
    $config = array();
    $config['charset'] = 'iso-8859-1'; //yy
    $config['useragent'] = 'Codeigniter';
    $config['protocol']= 'smtp';
    $config['mailtype']= 'html';
    $config['smtp_host']= 'ssl://smtp.gmail.com';
    $config['smtp_port']= 465;
    $config['smtp_timeout']= 400;
    $config['smtp_user']= 'aulyardhaanindita@gmail.com';
    $config['smtp_pass']= ''; // isi dengan password kamu
    $config['crlf']="\\r\\n";
    $config['newline']="\\r\\n";
    $config['wordwrap'] = TRUE;
    $this->load->library('email',$config);
    $this->email->initialize($config);
    $where_check = array('register_activation_code' => $random);
        //konfigurasi pengiriman
    $this->email->from($config['smtp_user']);
    $this->email->to($register_email);
    $this->email->subject("Verifikasi Akun");
    //body = $this->load-
>view('Frontend/register/email_verify.php',$where_check,TRUE);
    $this->email-
>message("Silahkan verifikasi akun anda untuk masuk ke Sistem Galeri Buku!.<br><br> Klik <strong><a href='http://localhost/bismillah/' target='_blank' rel='noopener'>disini</a></strong> untuk verifikasi akun.");
    $this->email->send();
    $this->session-
>set_flashdata('alert_success', 'Registrasi berhasil cek email anda untuk melakukan aktivasi.');
    redirect(base_url().'Frontend/Website/F_register/register');
    die();
}
}
} else {
    $this->session-
>set_flashdata('alert_error', 'Registrasi gagal !');
    redirect(base_url().'Frontend/Website/F_register/register');
    die();
}

```

```

        }
    }

    public function cek_email(){
        $email = $this->input->post('register_email');
        $where = array('register_email' => $email );
        $check_validation_email = $this->crud_model-
>get_data("tb_users_frontend",$where)->num_rows();
        if($check_validation_email == 0){
            $message = '';
            $is_success = true;
        }else{
            $message = 'E-mail tidak tersedia';
            $is_success = false;
        }
        $result = array(
            'message' => $message,
            'is_success' => $is_success
        );
        echo json_encode($result);
    }

    public function register_owner()
    {
        $this->form_validation->set_rules("register_email", "", "required");
        $this->form_validation-
>set_rules("register_username2", "", "required");
        $this->form_validation-
>set_rules("register_password2", "", "trim|required");
        $this->form_validation-
>set_rules("register_re_password2", "", "trim|required");
        $this->form_validation-
>set_rules("register_status2", "", "trim|required");
        $this->form_validation-
>set_rules("bio_fullname2", "", "trim|required");
        $this->form_validation-
>set_rules("bio_birthplace2", "", "trim|required");
        $this->form_validation-
>set_rules("bio_birthdate2", "", "trim|required");
        $this->form_validation-
>set_rules("bio_gender2", "", "trim|required");
        $this->form_validation->set_rules("bio_phone2", "", "required");
    }
}

```

```

$this->form_validation-
>set_rules("bio_address2", "", "trim|required");
    $this->form_validation-
>set_rules("bio_post_code2", "", "trim|required");
    if ($this->form_validation->run() == true){
        // post
        $register_email = $this->input->post('register_email');
        $register_username = $this->input->post('register_username2');
        $register_password = sha1($this->input-
>post('register_password2'));
        $register_re_password = sha1($this->input-
>post('register_re_password2'));
        $register_status = $this->input->post('register_status2');
        $bio_fullname = $this->input->post('bio_fullname2');
        $bio_birthplace = $this->input->post('bio_birthplace2');
        $bio_birthdate = $this->input->post('bio_birthdate2');
        $bio_birthdate = date("Y-m-d", strtotime($bio_birthdate));
        $bio_gender = $this->input->post('bio_gender2');
        $bio_phone = $this->input->post('bio_phone2');
        $bio_address = $this->input->post('bio_address2');
        $bio_post_code = $this->input->post('bio_post_code2');
        $where = array('register_email' => $register_email );
        $check_validation_email = $this->crud_model-
>get_data("tb_users_frontend",$where)->num_rows();
        // parameter awal
        $date = date('Y-m-d H:i:s');
        $date_birth = date('Y-m-d');
        $reg_ID = $this->crud_model->get_setting();
        $idMax = $this->register_model->register_code($reg_ID[0]->reg_ID);
        $noUrut = (int) substr($idMax[0]->maxID,4,8);
        $noUrut++;
        $newID = $reg_ID[0]->reg_ID.sprintf("%08s",$noUrut);
        $random_ref = mt_rand(10000000, 99999999);
        $newref = "REF-".$random_ref;
        $random = rand(100000,99999999);
        $where = array('register_email' => $register_email );
        $check_validation_email = $this->crud_model-
>get_data("tb_users_frontend",$where)->num_rows();
        $where = array('register_username' => $register_username );

```

```

        $check_validation_username = $this->crud_model-
>get_data("tb_users_frontend", $where)->num_rows();
        $date_birth_cek = date("Y-m-d", strtotime($bio_birthdate));
        $check_validation_age = $date_birth - $date_birth_cek;
        if($check_validation_email != 0){
            $this->session-
>set_flashdata('cache_email', $register_email);
            $this->session-
>set_flashdata('cache_username', $register_username);
            $this->session-
>set_flashdata('cache_status', $register_status);
            $this->session-
>set_flashdata('cache_bio_fullname', $bio_fullname);
            $this->session-
>set_flashdata('cache_bio_birthplace', $bio_birthplace);
            $this->session-
>set_flashdata('cache_bio_birthdate', $bio_birthdate);
            $this->session-
>set_flashdata('cache_bio_gender', $bio_gender);
            $this->session-
>set_flashdata('cache_bio_phone', $bio_phone);
            $this->session-
>set_flashdata('cache_bio_address', $bio_address);
            $this->session-
>set_flashdata('cache_bio_post_code', $bio_post_code);
            $this->session-
>set_flashdata('alert_error', 'Username anda sudah terpakai !');
            redirect(base_url().'Frontend/Website/F_register/register');
            die;
        }
        if($check_validation_username != 0){
            $this->session-
>set_flashdata('cache_email', $register_email);
            $this->session-
>set_flashdata('cache_username', $register_username);
            $this->session-
>set_flashdata('cache_status', $register_status);
            $this->session-
>set_flashdata('cache_bio_fullname', $bio_fullname);

```

```

        $this->session-
>set_flashdata('cache_bio_birthplace', $bio_birthplace);
        $this->session-
>set_flashdata('cache_bio_birthdate', $bio_birthdate);
        $this->session-
>set_flashdata('cache_bio_gender', $bio_gender);
        $this->session-
>set_flashdata('cache_bio_phone', $bio_phone);
        $this->session-
>set_flashdata('cache_bio_address', $bio_address);
        $this->session-
>set_flashdata('cache_bio_post_code', $bio_post_code);
        $this->session-
>set_flashdata('alert_error', 'Username anda sudah terpakai !');
        redirect(base_url().'Frontend/Website/F_register/register');
        die;
    }

    $name_file = $bio_fullname.".".time();
    $config['upload_path'] = './uploads/ktp_owner';
    $config['allowed_types'] = 'gif|jpg|png|jpeg';
    $config['max_size']      = '5120';
    $config['file_name']     = $name_file;
    $this->load->library('upload', $config);
    if(!$this->upload->do_upload('bio_upload2')){
        $this->session-
>set_flashdata('alert_error', 'Upload KTP/KTS gagal, ukuran file terlalu besar atau format file salah !');
        redirect(base_url().'Frontend/Website/F_register/register');
        die();
    } else {
        $data_register = array(
            'register_id' => $newID,
            'register_email' => $register_email,
            'register_username' => $register_username,
            'register_password' => $register_password,
            'register_date' => $date,
            'register_status' => 'Activated',
            'status_register' => $register_status
        );
    }
}

```

```

$insert_register = $this->crud_model-
>insert('tb_users_frontend',$data_register);
if($insert_register === false){
    $this->session-
>set_flashdata('alert_error', 'Gagal menyimpan data');
    redirect(base_url().'Frontend/Website/F_register/register');
    die();
} else {
    if (empty($bio_ref_code)){
        $bio_ref_code = "-";
    } else {
        $date = date('Y-m-d');
    }
    $file_data = $this->upload->data();
    $data_bio = array(
        'register_id' => $newID,
        'bio_fullname' => $bio_fullname,
        'bio_place_birth_date' => $bio_birthplace,
        'bio_birth_date' => date('Y-m-
d', strtotime($bio_birthdate)), /// date()format
        'bio_gender' => $bio_gender,
        'bio_phone' => $bio_phone,
        'bio_address' => $bio_address,
        'bio_post_code' => $bio_post_code,
        'bio_photo_ktp' => $file_data['file_name'],
    );
    $insert_bio = $this->crud_model-
>insert('tb_lender_bio',$data_bio);
}
if($insert_bio === false){
    $this->session-
>set_flashdata('alert_error', 'Gagal menyimpan data');
    redirect(base_url().'Frontend/Website/F_register/register');
    die();
} else {
    $config = array();
    $config['charset'] = 'iso-8859-1'; //yy
    $config['useragent'] = 'Codeigniter';
    $config['protocol']= 'smtp';
    $config['mailtype']= 'html';
}

```

```

        $config['smtp_host']= 'ssl://smtp.gmail.com';
        $config['smtp_port']= 465;
        $config['smtp_timeout']= 400;
        $config['smtp_user']= 'aulyardhaanindita@gmail.com';
        $config['smtp_pass']= ''; //isi dengan password kamu
        $config['crlf']="\\r\\n";
        $config['newline']="\\r\\n";
        $config['wordwrap'] = TRUE;
        $this->load->library('email',$config);
        $this->email->initialize($config);
        $where_check = array('register_activation_code' => $random);
        //konfigurasi pengiriman
        $this->email->from($config['smtp_user']);
        $this->email->to($register_email);
        $this->email->subject("Verifikasi Akun");
        //body = $this->load-
        >view('Frontend/register/email_verify.php',$where_check,TRUE);
        $this->email-
        >message("Silahkan verifikasi akun anda untuk masuk ke Sistem Galeri Buku!.<br><br> Klik <strong><a href='http://localhost/bismillah/' target='_blank' rel='noopener'>disini</a></strong> untuk verifikasi akun.");
        $this->email->send();
        $this->session-
        >set_flashdata('alert_success', 'Registrasi berhasil, cek email anda untuk melakukan aktivasi.');
        redirect(base_url().'Frontend/Website/F_register/register');
        die();
    }
}
} else {
    $this->session-
    >set_flashdata('alert_error', 'Registrasi gagal !');
    redirect(base_url().'Frontend/Website/F_register/register');
    die();
}
}

public function cek_username_peminjam(){
$username = $this->input->post('register_username1');
$where = array('register_username' => $username );

```

```

    $check_validation_username = $this->crud_model-
>get_data("tb_users_frontend",$where)->num_rows();
    if($check_validation_username==0){
        $message = '';
        $is_success = true;
    } else {
        $message = 'Username telah terdaftar !';
        $is_success = false;
    }
    $result = array(
        'message' => $message,
        'is_success' => $is_success
    );
    echo json_encode($result);
}

public function cek_username_owner(){
    $username = $this->input->post('register_username2');
    $where = array('register_username' => $username );
    $check_validation_username = $this->crud_model-
>get_data("tb_users_frontend",$where)->num_rows();
    if($check_validation_username==0){
        $message = '';
        $is_success = true;
    } else {
        $message = 'Username telah terdaftar !';
        $is_success = false;
    }
    $result = array(
        'message' => $message,
        'is_success' => $is_success
    );
    echo json_encode($result);
}

public function upload_ocr()
{
    $fileName = $_FILES['bio_upload']['name'];
    $fileType = $_FILES['bio_upload']['type'];
    $fileSize = $_FILES['bio_upload']['size'];
    if (!empty($fileName) AND $fileSize <= 5242880){
        $random = rand(100000,99999999);

```

```

        $name_file = $random."_".time();
        $config['upload_path'] = './assets/OCR';
        $config['allowed_types'] = 'jpg|png|jpeg';
        $config['max_size']      = '5120';
        $config['file_name'] = $name_file;
        $this->load->library('upload', $config);
        if(!$this->upload->do_upload('bio_upload')){
            $namefile = "";
            $message = "Upload Gagal / Format File Salah !";
            $output = "";
            $message_ocr = "OCR Gagal / Format File Salah !";
            $is_success = false;
        } else {
            $file_data = $this->upload->data();
            $namefile = $file_data['file_name'];
            $message = "Upload Sukses";
            if (!empty($namefile)){
                $target_file= '~/Tools/test12_ok -
i /home/sandersco/public_html/assets/OCR/'.$namefile;
                $output = shell_exec($target_file);
                if(!empty($output) AND strlen($output) <= 17){
                    $message_ocr = "OCR Sukses";
                    $is_success = true;
                } else {
                    $message_ocr = "OCR Gagal / Format File Salah !";
                    $is_success = false;
                }
            } else {
                $output = "";
                $message_ocr = "OCR Gagal / Format File Salah !";
                $is_success = false;
            }
        }
        $result = array(
            'is_success' => $is_success,
            'namefile' => $namefile,
            'nik_ocr' => trim($output),
            'message_ocr' => $message_ocr,
            'message_upload' => $message
        );
    }
}

```

```
        echo json_encode($result);
    }
}
?>
```

9.2.2 Membuat View *register.php*

Berikut adalah syntax dari file *register.php*

```
<?php date_default_timezone_set("Asia/Jakarta"); ?>
<style>
/* The container */
.label_radio {
    display: inline;
    position: relative;
    padding-left: 35px;
    padding-right: 10px;
    margin-bottom: 12px;
    cursor: pointer;
    font-size: 22px;
    -webkit-user-select: none;
    -moz-user-select: none;
    -ms-user-select: none;
    user-select: none;
}
/* Hide the browser's default radio button */
.label_radio input {
    position: absolute;
    opacity: 0;
    cursor: pointer;
}
/* Create a custom radio button */
.radio_style {
    position: absolute;
    top: 0;
    left: 0;
    height: 25px;
    width: 25px;
    background-color: #c1c1c1;
    border-radius: 50%;
```

```

}

/* On mouse-over, add a grey background color */
.label_radio:hover input ~ .radio_style {
    background-color: #bbb;
}

/* When the radio button is checked, add a blue background */
.label_radio input:checked ~ .radio_style {
    background-color: #22839c;
}

/* Create the indicator (the dot/circle - hidden when not checked) */
.radio_style:after {
    content: "";
    position: absolute;
    display: none;
}

/* Show the indicator (dot/circle) when checked */
.label_radio input:checked ~ .radio_style:after {
    display: block;
}

/* Style the indicator (dot/circle) */
.label_radio .radio_style:after {
    top: 9px;
    left: 9px;
    width: 8px;
    height: 8px;
    border-radius: 50%;
    background: white;
}

.success_pass {
    color: green;
}

.error_pass {
    color: red;
}

.btn-circle {
    width: 64px;
    height: 34px;
    text-align: center;
    padding: 3px 0;
    font-size: 12px;
}

```

```
line-height: 26px;
border-radius: 8px;
}

.btn_modal_ok {
height: 27px;
width: 65px;
}

.btn_modal_cancel {
height: 27px;
width: 70px;
}

.check_custom {
position: relative;
padding-left: 35px;
margin-bottom: 12px;
margin-right: 10px;
cursor: pointer;
font-size: 15px;
-webkit-user-select: none;
-moz-user-select: none;
-ms-user-select: none;
user-select: none;
}

/* Hide the browser's default checkbox */
.check_custom input {
position: absolute;
opacity: 0;
cursor: pointer;
}

/* Create a custom checkbox */
.check_mark {
position: absolute;
top: 0;
left: 0;
height: 25px;
width: 25px;
background-color: #eee;
}

/* On mouse-over, add a grey background color */
.check_custom:hover input ~ .check_mark {
```

```

background-color: #ccc;
}
/* When the checkbox is checked, add a blue background */
.check_custom input:checked ~ .check_mark {
background-color: #ffd100;
}
/* Create the check_mark/indicator (hidden when not checked) */
.check_mark:after {
content: "";
position: absolute;
display: none;
border-radius: 10%;
}
/* Show the check_mark when checked */
.check_custom input:checked ~ .check_mark:after {
display: block;
}
/* Style the check_mark/indicator */
.check_custom .check_mark:after {
left: 10px;
top: 7px;
width: 7px;
height: 10px;
border: solid white;
border-width: 0 3px 3px 0;
-webkit-transform: rotate(45deg);
-ms-transform: rotate(45deg);
transform: rotate(45deg);
}
</style>
<style>
.panel.with-nav-tabs .panel-heading{
padding: 5px 5px 0 5px;
border-right: 1px solid;
border-top: 1px solid;
border-left: 1px solid;
border-color: #22839c;
background-color: #22839c;
font-size: 15px;
font-weight: bold;

```

```

}

.panel.with-nav-tabs .panel-body{
    border-right: 1px solid;
    border-bottom: 1px solid;
    border-left: 1px solid;
    border-color: #22839c;
}

.panel.with-nav-tabs .nav-tabs{
    border-bottom: none;
}

.panel.with-nav-tabs .nav-justified{
    margin-bottom: -1px;
}

.with-nav-tabs.panel-default .nav-tabs > li > a,
.with-nav-tabs.panel-default .nav-tabs > li > a:hover,
.with-nav-tabs.panel-default .nav-tabs > li > a:focus {
    color: #6d6d6d;
}

.with-nav-tabs.panel-default .nav-tabs > .open > a,
.with-nav-tabs.panel-default .nav-tabs > .open > a:hover,
.with-nav-tabs.panel-default .nav-tabs > .open > a:focus,
.with-nav-tabs.panel-default .nav-tabs > li > a:hover,
.with-nav-tabs.panel-default .nav-tabs > li > a:focus {
    color: #6d6d6d;
    background-color: #fff;
    border-color: #22839c;
}

.with-nav-tabs.panel-default .nav-tabs > li.active > a,
.with-nav-tabs.panel-default .nav-tabs > li.active > a:hover,
.with-nav-tabs.panel-default .nav-tabs > li.active > a:focus {
    color: #6d6d6d;
    background-color: #fff;
    border-color: #22839c;
    border-bottom-color: transparent;
}

.with-nav-tabs.panel-default .nav-tabs > li.dropdown .dropdown-menu {
    background-color: #22839c;
    border-color: #ddd;
}

```

```
.with-nav-tabs.panel-default .nav-tabs > li.dropdown .dropdown-
menu > li > a {
    color: #777;
}
.with-nav-tabs.panel-default .nav-tabs > li.dropdown .dropdown-
menu > li > a:hover,
.with-nav-tabs.panel-default .nav-tabs > li.dropdown .dropdown-
menu > li > a:focus {
    background-color: #ddd;
}
.with-nav-tabs.panel-default .nav-tabs > li.dropdown .dropdown-
menu > .active > a,
.with-nav-tabs.panel-default .nav-tabs > li.dropdown .dropdown-
menu > .active > a:hover,
.with-nav-tabs.panel-default .nav-tabs > li.dropdown .dropdown-
menu > .active > a:focus {
    color: #fff;
    background-color: #555;
}
.page-mask {
    position: fixed;
    left: 0;
    right: 0;
    bottom: 0;
    top: 0;
    background-color: rgba(0,0,0,0.6);
    z-index: 10;
}
.centered_image {
    position: fixed;
    top: 50%;
    left: 50%;
    -webkit-transform: translate(-50%, -50%);
    -moz-transform: translate(-50%, -50%);
    -ms-transform: translate(-50%, -50%);
    -o-transform: translate(-50%, -50%);
    transform: translate(-50%, -50%);
    z-index: 12;
}

```

```

<!-- <div id="page_mask" class="page-mask" style="display: none"></div>
<div id="page_image" class="centered_image" style="display: none"><img src=
"<?php echo base_url(); ?>uploads/base-img/loading_icon.gif"></div> -->
<section class="container home">
    <div class="row" style="margin-bottom: 15px;">
        <div class="container div-feedback">
            <div class="row">
                <?php if ($this->session->flashdata('alert_success')) { ?>
                    <div class="alert alert-success alert-
dismissible" style="background-color: #11a7119c !important; border-
radius: 0px !important;padding-right: 35px;">
                        <button type="button" class="close" data-dismiss="alert" aria-
hidden="true">&times;</button>
                        <p><i class="icon fa fa-check" style="font-size: 20px;"></i>
                        <?php echo $this->session->flashdata('alert_success'); ?>
                    </p>
                </div>
                <?php } ?>
                <?php if ($this->session->flashdata('alert_error')) { ?>
                    <div class="alert alert-danger alert-
dismissible" style="background-color: #ff6b6bba !important; border-
radius: 0px !important;padding-right: 35px;">
                        <button type="button" class="close" data-dismiss="alert" aria-
hidden="true">&times;</button>
                        <p><i class="icon fa fa-ban" style="font-size: 20px;"></i>
                        <?php echo $this->session->flashdata('alert_error'); ?>
                    </p>
                </div>
                <?php } ?>
            <?>
                <!-- <div class="col-md-12 col-sm-12 col-xs-
12" id="left2" style="margin-top: 2%"> -->
                <br>
                <h3 align="center"><strong>Form Registrasi</strong></h3>
                <br>
                <HR style="display: block; border-width: 5px; border-
color: black;">
                <br>
                <?php
                    if ($this->session->flashdata('cache_status') == "Owner") {

```

```

        $cache_owner = 'checked';
        $display_panel_peminjam = 'none';
        $display_panel_owner = 'show';
    } else {
        $cache_peminjam = 'checked';
        $display_panel_peminjam = 'show';
        $display_panel_owner = 'none';
    }
?>
<div class="col-xs-12 col-sm-12 col-md-6 col-lg-6" style="text-align: center;">
    <label class="label_radio">Peminjam Buku
        <input type="radio" <?php echo @$cache_peminjam; ?> value="Peminjam" name="register_status_radio">
            <span class="radio_style"></span>
        </label>
    </div>
    <div class="col-xs-12 col-sm-12 col-md-6 col-lg-6" style="text-align: center;">
        <label class="label_radio">Pemilik Buku
        <input type="radio" <?php echo @$cache_owner; ?> value="Owner" name="register_status_radio">
            <span class="radio_style"></span>
        </label>
    </div>
    <div id="display_panel_peminjam" class="col-xs-12 col-sm-12 col-md-12 col-lg-12" style="display: <?php echo $display_panel_peminjam; ?>;">
        <br>
        <div class="panel-body">
            <div class="tab-content">
                <div class="tab-pane fade in active" id="tab1peminjam">
                    <form action="<?php echo site_url('Frontend/Website/F_register/register_peminjam');?>" method="post" enctype="multipart/form-data" onsubmit="return validateForm_peminjam()">
                        <input type="hidden" name="register_status1" value="Peminjam" required="required">
                        <div class="col-xs-12 col-sm-6 col-md-6 col-lg-6">
                            <div class="form-group">
                                <label>Email</label>

```

```

        <input type="text" id="email" class="form-control" name="register_email" required="required" onchange="check_email()">
            <p id="error_email" style="font-size: 12px"></p>
        </div>
        <div class="form-group">
            <label>Username</label>
            <input type="text" id="username1" class="form-control" name="register_username1" required="required" onchange="check_username1()">
                <p id="error_username1" style="font-size: 12px"></p>
            </div>
            <div class="form-group">
                <label>Nama (Sesuai KTP/KTS)</label>
                <input type="text" id="id_name" class="form-control" name="bio_fullname1" required="required" onkeyup="validhuruf(this)">
                    <p id="error_nik_name" style="font-size: 12px"></p>
                </div>
            </div>
            <div class="col-xs-12 col-sm-6 col-md-6 col-lg-6">
                <div class="form-group">
                    <label>Password</label>
                    <input type="password" id="id_password1" class="form-control" name="register_password1" required="required" onchange="check_pass_reg1()">
                        <p id="error_password1" style="font-size: 12px"></p>
                    </div>
                    <div class="form-group">
                        <label>Masukan Kembali Password</label>
                        <input type="password" id="id_re_password1" class="form-control" name="register_re_password1" required="required" onchange="check_pass_re_reg1()">
                            <p id="error_html1" style="font-size: 12px"></p>
                        </div>
                    </div>
                    <br>
                    <div class="col-xs-12 col-sm-12 col-md-12 col-lg-12">
                        <hr style="border: 2px solid; border-color: blue;">
                        <p>
                    </div>
    
```

```

<div class="col-xs-12 col-sm-6 col-md-6 col-lg-6">
    <div class="form-group">
        <label>Tempat Lahir</label>
        <input type="text" id="bio_birthplace1" class="form-control" name="bio_birthplace1" required="required">
    </div>
    <div class="form-group">
        <label>Tanggal Lahir</label>
        <div class="input-group">
            <div class="input-group-addon">
                <i class="fa fa-calendar" style="display: inline-block; font: normal normal normal 14px/1 FontAwesome; font-size: inherit; text-rendering: auto; -webkit-font-smoothing: antialiased; -moz-osx-font-smoothing: grayscale; padding: 0; width: 11px; margin: 0;"/></i>
            </div>
            <input type="text" name="bio_birthdate1" class="form-control pull-right datepicker_date" id="id_birthdate1" onchange="birthdate_len1()" required="required">
        </div>
        <span id="alert-age1" style="font-size: 12px"></span>
    </div>
    <div class="form-group">
        <label>Jenis Kelamin</label>
        <select id="id_gender1" class="form-control select2" name="bio_gender1" style="width: 100%;" required="required">
            <option value="">- Pilih Data -</option>
            <option value="Male">Pria</option>
            <option value="Female">Wanita</option>
        </select>
    </div>
    <div class="form-group">
        <label>Nomor Telepon (Handphone)</label>
        <input type="text" id="id_phone1" maxlength="13" minlength="5" onkeyup="this.value=this.value.replace(/[^0-9]/g,'');" class="form-control" name="bio_phone1" required="required">
    </div>
    <div class="form-group">
        <label>Alamat Lengkap (Tempat Tinggal Saat Ini)</label>
    </div>

```

```

        <textarea id="id_address1" rows="2" name="bio_address1" class="form-control" required="required"></textarea>
    </div>
</div>
<div class="col-xs-12 col-sm-6 col-md-6 col-lg-6">
    <div class="form-group">
        <label>Upload KTP/KTS</label>
        <input type="file" onchange="Validate_file1(this);>
        id="id_upload1" name="bio_upload1" class="form-control" required="required">
        <br>
        <span style="font-size: 12px;">* tunggu beberapa saat hingga file/gambar muncul. <br> * Format file gambar harus berekstensi JPG, JPEG atau PNG dan maksimal berukuran 5MB</span>
        </img>
        <span id="error_ocr1" style="font-size: 12px;"></span>
    </div>
    <br>
    <div class="form-group">
        <label>Zip Code</label>
        <input type="text" id="id_post_code1" maxlength="5" minlength="5" onkeyup="this.value=this.value.replace(/[^0-9]/g,'');">
        class="form-control" name="bio_post_code1" required="required">
    </div>
    <div class="form-group">
        <button class="btn btn-success btn-lg btn-block" type="submit" style="height: 35px">Register</button>
    </div>
</div>
</form>
</div>
</div>
<!-- </div>
</div> -->
<div id="display_panel_owner" class="col-xs-12 col-sm-12 col-md-12 col-lg-12" style="display: <?php echo $display_panel_owner; ?>;">

```

```

<br>
<div class="panel-body">
    <div class="tab-content">
        <div class="tab-pane fade in active" id="tabowner">
            <form action="php echo site_url('Frontend/Website/F_register/register_owner');?" method="post" enctype="multipart/form-data" onsubmit="return validateForm_owner()">
                <input type="hidden" name="register_status2" value="Pemilik" required="required">
                <div class="col-xs-12 col-sm-6 col-md-6 col-lg-6">
                    <div class="form-group">
                        <label>Email</label>
                        <input type="email" id="email" class="form-control" name="register_email" required="required" onchange="check_email()">
                        <p id="error_email" style="font-size: 12px"></p>
                    </div>
                    <div class="form-group">
                        <label>Username</label>
                        <input type="text" id="username2" class="form-control" name="register_username2" required="required" onchange="check_username2()">
                        <p id="error_username2" style="font-size: 12px"></p>
                    </div>
                    <div class="form-group">
                        <label>Nama (Sesuai KTP/KTS)</label>
                        <input type="text" id="id_name2" class="form-control" name="bio_fullname2" required="required" onkeyup="validhuruf(this)">
                    </div>
                </div>
                <div class="col-xs-12 col-sm-6 col-md-6 col-lg-6">
                    <div class="form-group">
                        <label>Password</label>
                        <input type="password" id="id_password2" class="form-control" name="register_password2" required="required" onchange="check_passreg2()">
                        <p id="error_password2" style="font-size: 12px"></p>
                    </div>
                    <div class="form-group">
                        <label>Masukan Kembali Password</label>

```

```

        <input type="password" id="id_re_password2" class="form-control" name="register_re_password2" required="required" onchange="check_passre_reg2()">
            <p id="error_html2" style="font-size: 12px"></p>
        </div>
    </div>
    <br>
    <div class="col-xs-12 col-sm-12 col-md-12 col-lg-12">
        <hr style="border: 2px solid; border-color: blue;">
        <p>
    </div>
    <div class="col-xs-12 col-sm-6 col-md-6 col-lg-6">
        <div class="form-group">
            <label>Tempat Lahir</label>
            <input type="text" id="bio_birthplace2" class="form-control" name="bio_birthplace2" required="required">
        </div>
        <div class="form-group">
            <label>Tanggal Lahir</label>
            <div class="input-group">
                <div class="input-group-addon">
                    <i class="fa fa-calendar" style="display: inline-block; font: normal normal normal 14px/1 FontAwesome; font-size: inherit; text-rendering: auto; -webkit-font-smoothing: antialiased; -moz-osx-font-smoothing: grayscale; padding: 0; width: 11px; margin: 0;"></i>
                </div>
                <input type="text" name="bio_birthdate2" class="form-control pull-right datepicker_date" id="id_birthdate2" onchange="birthdate_len2()" required="required">
            </div>
            <span id="alert-age2" style="font-size: 12px"></span>
        </div>
        <div class="form-group">
            <label>Jenis Kelamin</label>
            <select id="id_gender2" class="form-control select2" name="bio_gender2" style="width: 100%;" required="required">
                <option value="">- Pilih Data -</option>
                <option value="Male">Pria</option>
                <option value="Female">Wanita</option>
            </select>
        </div>
    </div>

```

```

        </select>
    </div>
    <div class="form-group">
        <label>Nomor Telepon (Handphone)</label>
        <input type="text" id="id_phone2" maxlength="13" minlength="5" onkeyup="this.value=this.value.replace(/[^0-9]/g,'');" class="form-control" name="bio_phone2" required="required">
    </div>
    <div class="form-group">
        <label>Alamat Lengkap (Tempat Tinggal Saat Ini)</label>
        <textarea id="id_address2" rows="2" name="bio_address2" class="form-control" required="required"></textarea>
    </div>
    <div class="col-xs-12 col-sm-6 col-md-6 col-lg-6">
        <div class="form-group">
            <label>Upload KTP/KTS</label>
            <input type="file" onchange="Validate_file2(this);"
id="id_upload2" name="bio_upload2" class="form-control" required="required">
            <br>
            <span style="font-size: 12px;">* tunggu beberapa saat hingga file/gambar muncul. <br> * Format file gambar harus berekstensi JPG, JPEG atau PNG dan maksimal berukuran 5MB</span>
            </img>
            <span id="error_ocr2" style="font-size: 12px;"></span>
        </div>
        <br>
        <div class="form-group">
            <label>Zip Code</label>
            <input type="text" id="id_phost_code2" maxlength="5" minlength="5" onkeyup="this.value=this.value.replace(/[^0-9]/g,'');" class="form-control" name="bio_post_code2" required="required">
        </div>
        <div class="form-group">
            <button class="btn btn-success btn-lg btn-block" type="submit" style="height: 35px">Register</button>
        </div>
    
```

```

        </div>
    </form>
</div>
</div>
</div>
</section>
<script type="text/javascript">
    $('input[type="radio"]').on('change', function(e) {
        var value_radio = $(this).val();
        if (value_radio == 'Peminjam') {
            document.getElementById('display_panel_owner').style.display = 'none';
            document.getElementById('display_panel_peminjam').style.display = 'block';
        } else {
            document.getElementById('display_panel_owner').style.display = 'block';
            document.getElementById('display_panel_peminjam').style.display = 'none';
        }
    });
    function validateForm_peminjam(){
        var is_return = "";
        var is_success1 = check_username1();
        var is_success2 = check_pass_reg1();
        var is_success3 = birthdate_len1();
        var is_success4 = check_passre_reg1();
        var is_success5 = check_email();
        if(is_success1 && is_success2 && is_success3 && is_success4 && is_success5){
            is_return = true;
        } else {
            is_return = false;
        }
        console.log(is_return);
        return is_return;
    }
    function validateForm_owner(){
        var is_return = "";
        var is_success1 = check_username2();
        var is_success2 = check_pass_reg2();
        var is_success3 = birthdate_len2();
        var is_success4 = check_passre_reg2();
        var is_success5 = check_email();
        if(is_success1 && is_success2 && is_success3 && is_success4 && is_success5){
    
```

```

        is_return = true;
    } else {
        is_return = false;
    }
    console.log(is_return);
    return is_return;
}

function check_email(){
    var result = false;
    var message = "";
    var email = $("#email").val();
    $.ajax({
        url: <?php echo "" . site_url("Frontend/Website/F_register/cek_email") . "?>",
        type: 'POST',
        dataType: 'json',
        data: $('#email').serialize(),
        timeout: 180000,
        async:false,
        success: function(data) {
            // console.log(data);
            if(data.is_success==true){
                result = true;
                message = data.message;
            }else{
                result = false;
                message = data.message;
            }
        },
    });
    if (result == true){
        document.getElementById("error_username1").style.color = "green";
        document.getElementById('error_username1').innerHTML = "";
        return result;
    } else {
        document.getElementById("error_username1").style.color = "red";
        document.getElementById('error_username1').innerHTML = message;
        return result;
    }
}

```

```

function check_username1(){
    var result = false;
    var message = "";
    var username = $("#username1").val();
    $.ajax({
        url: <?php echo "" . site_url("Frontend/Website/F_register/cek_username_peminjam") . "?>,
        type: 'POST',
        dataType: 'json',
        data: $('#username1').serialize(),
        timeout: 180000,
        async:false,
        success: function(data) {
            // console.log(data);
            if(data.is_success==true){
                result = true;
                message = data.message;
            }else{
                result = false;
                message = data.message;
            }
        },
    });
    if (result == true){
        document.getElementById("error_username1").style.color = "green";
        document.getElementById('error_username1').innerHTML = "";
        return result;
    } else {
        document.getElementById("error_username1").style.color = "red";
        document.getElementById('error_username1').innerHTML = message;
        return result;
    }
}
function check_username2(){
    var result = false;
    var message = "";
    var username = $("#username2").val();
    $.ajax({
        url: <?php echo "" . site_url("Frontend/Website/F_register/cek_username_owner") . "?>,

```

```

        type: 'POST',
        dataType: 'json',
        data: $('#username2').serialize(),
        timeout: 180000,
        async:false,
        success: function(data) {
            // console.log(data);
            if(data.is_success==true){
                result = true;
                message = data.message;
            }else{
                result = false;
                message = data.message;
            }
        },
    ));
    if (result == true){
        document.getElementById("error_username2").style.color = "green";
        document.getElementById('error_username2').innerHTML = "";
        return result;
    } else {
        document.getElementById("error_username2").style.color = "red";
        document.getElementById('error_username2').innerHTML = message;
        return result;
    }
}

function birthdate_len1() {
    var val_birthdate = $("#id_birthdate1").val();
    var convert = val_birthdate.split("-").reverse().join("-");
    var birthdate = new Date(convert);
    // console.log(birthdate);
    var date_now = new Date();
    var age = Math.floor((date_now - birthdate) / (365.25 * 24 * 60 * 60 * 1000));
    if (age < 12 || age > 65) {
        document.getElementById("alert-age1").style.color = "red";
        document.getElementById('alert-
age1').innerHTML = 'Anda harus berumur lebih dari 12 tahun untuk bisa mendaftar !';
        is_error = false;
    }
}

```

```

    } else {
        document.getElementById("alert-age1").style.color = "green";
        document.getElementById('alert-age1').innerHTML = '';
        is_error = true;
    }
    return is_error;
}

function birthdate_len2() {
    var val_birthdate = $("#id_birthdate2").val();
    var convert = val_birthdate.split("-").reverse().join("-");
    var birthdate = new Date(convert);
    // console.log(birthdate);
    var date_now = new Date();
    var age = Math.floor((date_now - birthdate) / (365.25 * 24 * 60 * 60 * 1000));
    if (age < 12 || age > 65) {
        document.getElementById("alert-age2").style.color = "red";
        document.getElementById('alert-age2').innerHTML = 'Anda harus berumur lebih dari 12 tahun untuk bisa mendaftar !';
        is_error = false;
    } else {
        document.getElementById("alert-age2").style.color = "green";
        document.getElementById('alert-age2').innerHTML = '';
        is_error = true;
    }
    return is_error;
}

function check_pass_reg1(){
    var is_error = "";
    var error_pass = $("#id_password1").val();
    var error_pass_valid = $("#id_re_password1").val();
    if (error_pass.length < 4) {
        document.getElementById("error_password1").style.color = "red";
        document.getElementById('error_password1').innerHTML = 'password anda terlalu pendek.';
        is_error = false;
    } else if (error_pass.length > 50) {
        document.getElementById("error_password1").style.color = "red";

```

```

        document.getElementById('error_password1').innerHTML = 'password anda terlalu panjang.';
        is_error = false;
    } else {
        document.getElementById("error_password1").style.color = "green";
        document.getElementById('error_password1').innerHTML = '';
        is_error = true;
    }
    return is_error;
}

function check_pass_reg2(){
    var is_error = "";
    var error_pass = $("#id_password2").val();
    var error_pass_valid = $("#id_re_password2").val();
    if (error_pass.length < 4) {
        document.getElementById("error_password2").style.color = "red";
        document.getElementById('error_password2').innerHTML = 'password anda terlalu pendek.';
        is_error = false;
    } else if (error_pass.length > 50) {
        document.getElementById("error_password2").style.color = "red";
        document.getElementById('error_password2').innerHTML = 'password anda terlalu panjang. ';
        is_error = false;
    } else {
        document.getElementById("error_password2").style.color = "green";
        document.getElementById('error_password2').innerHTML = '';
        is_error = true;
    }
    return is_error;
}

function check_passre_reg1(){
    var is_error = "";
    var error_pass = $("#id_password1").val();
    var error_pass_valid = $("#id_re_password1").val();
    if (error_pass == error_pass_valid) {
        document.getElementById("error_html1").style.color = "green";
        document.getElementById('error_html1').innerHTML = 'password anda cocok';
        is_error = true;
    } else {

```

```

document.getElementById("error_html1").style.color = "red";
document.getElementById('error_html1').innerHTML = 'password anda tidak cocok !!!';
is_error = false;
}
return is_error;
}

function check_passre_reg2(){
var is_error = "";
var error_pass = $("#id_password2").val();
var error_pass_valid = $("#id_re_password2").val();
if (error_pass == error_pass_valid) {
    document.getElementById("error_html2").style.color = "green";
    document.getElementById('error_html2').innerHTML = 'password anda cocok';
    is_error = true;
} else {
    document.getElementById("error_html2").style.color = "red";
    document.getElementById('error_html2').innerHTML = 'password anda tidak cocok !!!';
    is_error = false;
}
return is_error;
}

function preview_ktp_peminjam(){
var input = document.getElementById('id_upload1');
var file = input.files[0];
var reader = new FileReader();
var noimage = "<?php echo base_url();?>uploads/noimage.jpg";
reader.onloadend = function(e) {
    $('#ktp_upload1').attr('src', reader.result);
}
if (file) {
    reader.readAsDataURL(file);
} else {
    $('#ktp_upload1').attr('src', noimage);
}
}

function preview_ktp_owner(){
var input = document.getElementById('id_upload2');
var file = input.files[0];

```

```

var reader = new FileReader();
var noimage = "<?php echo base_url();?>uploads/noimage.jpg";
reader.onloadend = function(e) {
    $('#ktp_upload2').attr('src', reader.result);
}
if (file) {
    reader.readAsDataURL(file);
} else {
    $('#ktp_upload2').attr('src', noimage);
}
}

function clear_file1(){
    var noimage = "<?php echo base_url();?>uploads/noimage.jpg";
    document.getElementById("id_upload1").value = null;
    $('#ktp_upload1').attr('src', noimage);
    document.getElementById("error_ocr1").style.color = "black";
    document.getElementById('error_ocr1').innerHTML = "";
}

function clear_file2(){
    var noimage = "<?php echo base_url();?>uploads/noimage.jpg";
    document.getElementById("id_upload2").value = null;
    $('#ktp_upload2').attr('src', noimage);
    document.getElementById("error_ocr2").style.color = "black";
    document.getElementById('error_ocr2').innerHTML = "";
}

function Validate_file1(oInput) {
    var _validFileExtensions = [".jpg", ".jpeg", ".png"];
    var input = document.getElementById('id_upload1');
    var picsize = input.files[0].size;
    if (oInput.type == "file") {
        var sFileName = oInput.value;
        if (sFileName.length > 0) {
            var blnValid = false;
            for (var j = 0; j < _validFileExtensions.length; j++) {
                var sCurExtension = _validFileExtensions[j];
                if (sFileName.substr(sFileName.length - sCurExtension.length, sCurExtension.length).toLowerCase() == sCurExtension.toLowerCase()) {
                    blnValid = true;
                    break;
                }
            }
        }
    }
}

```

```

    }
    if (picsize > 5000000){
        alert("Ukuran Data yang di upload tidak sesuai, ukuran yang sesuai adalah dibawah 5 MB");
        clear_file1();
        return false;
    } else if (!blnValid) {
        alert("Data yang di upload tidak di izinkan, yang di izinkan adalah: " + _validFileExtensions.join(", "));
        oInput.value = "";
        clear_file1();
        return false;
    } else {
        preview_ktp_peminjam();
        //OCR_FILE();
    }
}
return true;
}

function Validate_file2(oInput) {
    var _validFileExtensions = [".jpg", ".jpeg", ".png"];
    var input = document.getElementById('id_upload2');
    var picsize = input.files[0].size;
    if (oInput.type == "file") {
        var sFileName = oInput.value;
        if (sFileName.length > 0) {
            var blnValid = false;
            for (var j = 0; j < _validFileExtensions.length; j++) {
                var sCurExtension = _validFileExtensions[j];
                if (sFileName.substr(sFileName.length - sCurExtension.length, sCurExtension.length).toLowerCase() == sCurExtension.toLowerCase()) {
                    blnValid = true;
                    break;
                }
            }
            if (picsize > 5000000){
                alert("Ukuran Data yang di upload tidak sesuai, ukuran yang sesuai adalah dibawah 5 MB");
                clear_file2();
            }
        }
    }
}

```

```

        return false;
    } else if (!blnValid) {
        alert("Data yang di upload tidak di izinkan, yang di izinkan
adalah: " + _validFileExtensions.join(", "));
        oInput.value = "";
        clear_file2();
        return false;
    } else {
        preview_ktp_owner();
        //OCR_FILE2();
    }
}
}

return true;
}

function validhurst(a) {
if (!/^[a-zA-Z ]+$/ .test(a.value)){
    a.value = a.value.substring(0,a.value.length-1000);
}
}

</script>

```

9.2.3 Membuat Model *register_model.php*

Berikut adalah syntax dari file *register_model.php*

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class Register_model extends CI_Model {
    public function register_code($reg_ID){
        return $this->db->query("SELECT max(register_id) as maxID
        FROM tb_users_frontend WHERE register_id LIKE '".$reg_ID."%"')-
>result();
    }
    public function get_register_by_id($register_code){
        $this->db->select("tam.*")
        ->from("tb_users_frontend tam");
        $this->db->where("tam.register_id", $register_code);
        return $this->db->get()->result();
    }
}
?>

```

9.3 Membuat Proses Pada Bagian Pemilik

Ada beberapa proses yang terdapat pada pemilik, diantaranya *upload* buku, *view* peminjaman, *view* pengembalian dan *upload* bukti kirim peminjaman. Nah dalam pembuatan proses-proses tersebut, kami hanya membuat satu *controller* dalam mengelola proses tersebut.

9.3.1 Membuat *Controller F_owner.php*

Berikut adalah syntax dalam file *F_owner.php*

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');
class F_owner extends CI_Controller {
    function __construct(){
        parent::__construct();
        $this->load->library('form_validation');
        $this->load->model('crud_model');
        $this->load->model('Auth_model');
        $this->load->model('Front_data/personal_info_model');
        $this->load->model('Front_data/invest_book');
        $this->load->model('Website/users_model');
        $this->load->model('Front_data/register_model');
        $this->load->model('Front_data/loan_model');
        date_default_timezone_set("Asia/Jakarta");
        $reg_code = $this->session->userdata('reg_code');
        $reg_password = $this->session->userdata('reg_password');
        $reg_status = $this->session->userdata('reg_status');
        $check_access = $this->Auth_model-
>get_access_users($reg_code, $reg_password, $reg_status);
        if (!isset($reg_code) && !isset($reg_password) && !isset($reg_status)){
            $this->session-
>unset_userdata("reg_code","reg_password","reg_status");
            $this->session-
>set_flashdata('alert_error', 'Anda belum Login atau anda meninggalkan halaman terlalu lama !');
            redirect(base_url().'Frontend/Website/F_Login/Login');
            die();
        }
    }
    public function index(){
```

```

$reg_code = $this->session->userdata('reg_code');
$reg_password = $this->session->userdata('reg_password');
$reg_status = $this->session->userdata('reg_status');
$buku_id = $this->uri->segment(5);
$register_code = $reg_code;
$data['get_code'] = $register_code;
$where = array('register_id'=>$register_code);
$header['bio_fullname'] = $this->crud_model-
>get_data('tb_lender_bio', $where)->result();
$data['title'] = "My Book";
$data['brd_title_main'] = "My Book";
$data['brd_title_url'] = site_url('Frontend/Owner/F_owner');
$data['register_code'] = $reg_code;
$data['delete_url'] = site_url('Frontend/Owner/F_owner/delete_buku');
$data['info_url'] = site_url('Frontend/Owner/F_owner/detail_buku');
$data['edit_url'] = site_url('Frontend/Owner/F_owner/edit_buku');
$this->load->view('Frontend/website/partial/metadata');
$this->load->view('Frontend/website/partial/header', $header);
$this->load->view('Frontend/website/partial/navigation');
$this->load->view('Frontend/owner/dashboard', $data);
$this->load->view('Frontend/website/partial/footer');
}
public function detail_buku()
{
    $reg_code = $this->session->userdata('reg_code');
    $reg_password = $this->session->userdata('reg_password');
    $reg_status = $this->session->userdata('reg_status');
    $buku_id = $this->uri->segment(5);
    $register_code = $reg_code;
    $data['get_code'] = $register_code;
    $where = array('register_id'=>$register_code);
    $header['bio_fullname'] = $this->crud_model-
>get_data('tb_lender_bio', $where)->result();
    $data['title'] = "Detail Buku";
    $data['brd_title_url'] = site_url('Frontend/Owner/F_owner/detail_cek
_peminjaman')."/".$buku_id;
    $data['brd_title_main'] = "Detail Buku";
    $data['data_book'] = $this->personal_info_model-
>get_personal_info_book($buku_id);
}

```

```

    $data['data_cost'] = $this->personal_info_model-
>get_personal_info_invest($buku_id);
    $this->load->view('Frontend/website/partial/metadata');
    $this->load->view('Frontend/website/partial/header', $header);
    $this->load->view('Frontend/website/partial/navigation');
    $this->load->view('Frontend/owner/cek_detail_buku', $data);
    $this->load->view('Frontend/website/partial/footer');
}
public function edit_buku()
{
    $reg_code = $this->session->userdata('reg_code');
    $reg_password = $this->session->userdata('reg_password');
    $reg_status = $this->session->userdata('reg_status');
    $buku_id = $this->uri->segment(5);
    $id_invest_book = $this->uri->segment(6);
    $register_code = $reg_code;
    $data['get_code'] = $register_code;
    $where = array('register_id'=>$register_code);
    $header['bio_fullname'] = $this->crud_model-
>get_data('tb_lender_bio', $where)->result();
    $data['title'] = "Detail Buku";
    $data['brd_title_url'] = site_url('Frontend/Owner/F_owner/detail_cek
_peminjaman')."/".$buku_id;
    $data['brd_title_main'] = "Detail Buku";
    $data['data_book'] = $this->personal_info_model-
>get_personal_info_book($buku_id);
    $data['data_cost'] = $this->personal_info_model-
>get_personal_info_invest($buku_id);
    $data['update_url'] = site_url('Frontend/Owner/F_owner/invest_buku_u
pdate')."/".$buku_id."/". $id_invest_book;
    $this->load->view('Frontend/website/partial/metadata');
    $this->load->view('Frontend/website/partial/header', $header);
    $this->load->view('Frontend/website/partial/navigation');
    $this->load->view('Frontend/owner/edit_buku', $data);
    $this->load->view('Frontend/website/partial/footer');
}
public function invest_buku(){
    $reg_code = $this->session->userdata('reg_code');
    $reg_password = $this->session->userdata('reg_password');
    $reg_status = $this->session->userdata('reg_status');

```

```

$register_code = $reg_code;
$where = array('register_id'=>$register_code);
$data['get_code'] = $register_code;
$data['data_code'] = $this->personal_info_model-
>get_personal_info_lender($register_code);
$header['bio_fullname'] = $this->crud_model-
>get_data('tb_lender_bio',$where)->result();
$header['akun_owner'] = "active";
$data['back_url'] = site_url('Frontend/Owner/F_owner');
$data['insert_url'] = site_url('Frontend/Owner/F_owner/invest_buku_i
nsert');
$this->load->view('Frontend/website/partial/metadata');
$this->load->view('Frontend/website/partial/header', $header);
$this->load->view('Frontend/owner/invest_buku', $data);
$this->load->view('Frontend/website/partial/navigation');
$this->load->view('Frontend/website/partial/footer');
}

public function cek_peminjaman(){
$reg_code = $this->session->userdata('reg_code');
$reg_password = $this->session->userdata('reg_password');
$reg_status = $this->session->userdata('reg_status');
$register_code = $reg_code;
$data['get_code'] = $register_code;
$where = array('register_id'=>$register_code);
$header['bio_fullname'] = $this->crud_model-
>get_data('tb_lender_bio',$where)->result();
$data['title'] = "Cek Peminjaman";
$data['brd_title_main'] = "Cek Peminjaman";
$data['brd_title_url'] = site_url('Frontend/Owner/F_owner');
$data['register_code'] = $reg_code;
$data['info_url'] = site_url('Frontend/Owner/F_owner/detail_cek_pemi
njaman');
$data['delete_url'] = site_url('Frontend/Owner/F_owner/delete_buku');
$this->load->view('Frontend/website/partial/metadata');
$this->load->view('Frontend/website/partial/header', $header);
$this->load->view('Frontend/website/partial/navigation');
$this->load->view('Frontend/owner/cek_peminjaman', $data);
$this->load->view('Frontend/website/partial/footer');
}

public function approved_book()

```

```

{
    $reg_code = $this->session->userdata('reg_code');
    $loan_id = $this->uri->segment(5);
    $register_code1 = $this->uri->segment(6);
    $id_invest_book = $this->uri->segment(7);
    $register_code = $reg_code;
    $data_invest = array(
        'status' => "CheckingSend",
    );
    $update_invest= $this->crud_model-
>update('tb_invest_book','id_invest_book',$id_invest_book,$data_invest);
    $this->session-
>set_flashdata('alert_success', 'Anda telah berhasil menyetujui pinjaman ini
!');
    redirect(base_url().'Frontend/Owner/F_owner/cek_peminjaman');
    die();
}
public function detail_cek_peminjaman()
{
    $reg_code = $this->session->userdata('reg_code');
    $reg_password = $this->session->userdata('reg_password');
    $reg_status = $this->session->userdata('reg_status');
    $loan_id = $this->uri->segment(5);
    $register_code1 = $this->uri->segment(6);
    $invest_book = $this->uri->segment(7);
    $register_code = $reg_code;
    $data['get_code'] = $register_code;
    $where = array('register_id'=>$register_code);
    $header['bio_fullname'] = $this->crud_model-
>get_data('tb_lender_bio',$where)->result();
    $data['title'] = "Cek Peminjaman Buku";
    $data['brd_title_url'] = site_url('Frontend/Owner/F_owner/detail_cek
_peminjaman')."/".$loan_id."/". $register_code1;
    $data['brd_title_main'] = "Checking Buku";
    $data['data_register'] = $this->register_model-
>get_register_by_id($register_code1);
    $data['get_code1'] = $register_code1;
    $data['loan_id'] = $loan_id;
    $data['id_invest_book'] = $invest_book;
}

```

```

        $data['data_code'] = $this->personal_info_model-
>get_personal_info_borrower($register_code1);
        $data['data_book'] = $this->personal_info_model-
>get_personal_info_invest_book($register_code);
        $data['data_cost'] = $this->personal_info_model-
>get_personal_info_invest_owner($register_code);
        $data['data_pembayaran'] = $this->personal_info_model-
>get_personal_info_pembayaran_borrower($register_code1);
        $data['approved_url'] = site_url('Frontend/Owner/F_owner/approved_book');
        $data['unapproved_url'] = site_url('Frontend/Owner/F_owner/unapprov
d_book');
        $this->load->view('Frontend/website/partial/metadata');
        $this->load->view('Frontend/website/partial/header', $header);
        $this->load->view('Frontend/website/partial/navigation');
        $this->load->view('Frontend/owner/cek_detail_peminjaman', $data);
        $this->load->view('Frontend/website/partial/footer');
    }
    public function cek_pengembalian(){
        $reg_code = $this->session->userdata('reg_code');
        $reg_password = $this->session->userdata('reg_password');
        $reg_status = $this->session->userdata('reg_status');
        $register_code = $reg_code;
        $data['get_code'] = $register_code;
        $where = array('register_id'=>$register_code);
        $header['bio_fullname'] = $this->crud_model-
>get_data('tb_lender_bio',$where)->result();
        $data['title'] = "Cek Pengembalian";
        $data['brd_title_main'] = "Cek Pengembalian";
        $data['brd_title_url'] = site_url('Frontend/Owner/F_owner');
        $data['register_code'] = $reg_code;
        $data['info_url'] = site_url('Frontend/Owner/F_owner/detail_cek_peng
embalian');
        $data['delete_url'] = site_url('Frontend/Owner/F_owner/delete_buku');
        $this->load->view('Frontend/website/partial/metadata');
        $this->load->view('Frontend/website/partial/header', $header);
        $this->load->view('Frontend/website/partial/navigation');
        $this->load->view('Frontend/owner/cek_pengembalian', $data);
        $this->load->view('Frontend/website/partial/footer');
    }
    public function approved_pengembalian()

```

```

{
    $reg_code = $this->session->userdata('reg_code');
    $loan_id = $this->uri->segment(5);
    $register_code1 = $this->uri->segment(6);
    $id_invest_book = $this->uri->segment(7);
    $register_code = $reg_code;
    $data_invest = array(
        'status' => "ApprovedReturn",
    );
    $update_invest= $this->crud_model-
>update('tb_invest_book','id_invest_book',$id_invest_book,$data_invest);
    $this->session-
>set_flashdata('alert_success', 'Anda telah berhasil menyetujui pinjaman ini
!');
    redirect(base_url().'Frontend/Owner/F_owner/cek_pengembalian');
    die();
}
public function detail_cek_pengembalian()
{
    $reg_code = $this->session->userdata('reg_code');
    $reg_password = $this->session->userdata('reg_password');
    $reg_status = $this->session->userdata('reg_status');
    $loan_id = $this->uri->segment(5);
    $register_code1 = $this->uri->segment(6);
    $invest_book = $this->uri->segment(7);
    $register_code = $reg_code;
    $data['get_code'] = $register_code;
    $where = array('register_id'=>$register_code);
    $header['bio_fullname'] = $this->crud_model-
>get_data('tb_lender_bio',$where)->result();
    $data['title'] = "Cek Pengembalian Buku";
    $data['brd_title_url'] = site_url('Frontend/Owner/F_owner/detail_cek
_pengembalian')."/".$loan_id."/". $register_code1;
    $data['brd_title_main'] = "Checking Buku";
    $data['data_register'] = $this->register_model-
>get_register_by_id($register_code1);
    $data['get_code1'] = $register_code1;
    $data['loan_id'] = $loan_id;
    $data['id_invest_book'] = $invest_book;
}

```

```

        $data['data_code'] = $this->personal_info_model-
>get_personal_info_borrower($register_code1);
        $data['data_book'] = $this->personal_info_model-
>get_personal_info_invest_book($register_code);
        $data['data_cost'] = $this->personal_info_model-
>get_personal_info_invest_owner($register_code);
        $data['data_pengembalian'] = $this->personal_info_model-
>get_personal_info_pengembalian_borrower($register_code1);
        $data['data_loan'] = $this->personal_info_model-
>get_personal_info_loan($loan_id);
        $data['approved_url'] = site_url('Frontend/Owner/F_owner/approved_pe
ngembalian');
        $data['unapproved_url'] = site_url('Frontend/Owner/F_owner/unapprov
ed_pengembalian');
        $this->load->view('Frontend/website/partial/metadata');
        $this->load->view('Frontend/website/partial/header', $header);
        $this->load->view('Frontend/website/partial/navigation');
        $this->load->view('Frontend/owner/cek_detail_pengembalian', $data);
        $this->load->view('Frontend/website/partial/footer');
    }
    public function invest_buku_insert(){
        $reg_code = $this->session->userdata('reg_code');
        $register_code = $reg_code;
        $idMax_invest = $this->invest_book->invest_code();
        $noUrut_invest =(int) substr($idMax_invest[0]->maxID,3,9);
        $idMax_buku = $this->invest_book->invest_buku();
        $noUrut_buku =(int) substr($idMax_buku[0]->maxID,3,9);
        $noUrut_invest++;
        $noUrut_buku++;
        $newID_Invest="I".sprintf("%09s",$noUrut_invest);
        $newID_Buku="B".sprintf("%09s",$noUrut_buku);
        $this->form_validation->set_rules("judul_buku", "", "required");
        $this->form_validation->set_rules("pengarang", "", "required");
        $this->form_validation->set_rules("penerbit", "", "required");
        $this->form_validation->set_rules("tahun", "", "required");
        $this->form_validation->set_rules("kategori", "", "required");
        $this->form_validation->set_rules("cost", "", "required");
        $this->form_validation->set_rules("harga_buku", "", "required");
        $this->form_validation->set_rules("biaya_admin", "", "required");
        if ($this->form_validation->run() == true){

```

```

$judul_buku = $this->input->post('judul_buku');
$pengarang = $this->input->post('pengarang');
$penerbit = $this->input->post('penerbit');
$tahun = $this->input->post('tahun');
$tahun = date("Y-m-d", strtotime($tahun));
$kategori = $this->input->post('kategori');
$sinopsis = $this->input->post('sinopsis');
$cost = $this->input->post('cost');
$harga_buku = $this->input->post('harga_buku');
$biaya_admin = $this->input->post('biaya_admin');
if(!empty($_FILES['id_upload']['name'])){
$file_upload_name = $this->input->post('foto_buku');
$name_file = $file_upload_name." ".time()." ".rand(0,1000);
$config['upload_path'] = './uploads/buku';
$config['allowed_types'] = 'gif|jpg|png|jpeg';
$config['max_size'] = '5120';
$config['file_name'] = $name_file;
$this->load->library('upload', $config);
if(!$this->upload->do_upload('id_upload')){
$this->session-
>set_flashdata('alert_error', 'Upload foto buku gagal, ukuran file terlalu besar atau format file salah !');
redirect(base_url().'Frontend/Owner/F_owner/invest_buku');
die();
} else {
$file_upload = $this->upload->data();
$data_buku = array(
'buku_id' => $newID_Buku,
'register_id' => $register_code,
'judul_buku' => $judul_buku,
'pengarang' => $pengarang,
'penerbit' => $penerbit,
'tahun' => $tahun,
'kategori' => $kategori,
'sinopsis' => $sinopsis,
'date' => date("Y-m-d"),
'foto_buku' => $file_upload['file_name'],
'harga_buku' => $harga_buku,
);
}
}

```

```

        $insert_buku = $this->crud_model-
>insert('tb_buku', $data_buku);
        $data_invest = array(
            'id_invest_book' => $newID_Invest,
            'register_id' => $register_code,
            'buku_id' => $newID_Buku,
            'invest_date' => date("Y-m-d"),
            'status' => 'Checking',
            'cost' => $cost,
            'biaya_admin' => $biaya_admin,
        );
        $insert_invest = $this->crud_model-
>insert('tb_invest_book', $data_invest);
    }
    $this->session-
>set_flashdata('alert_success', 'Proses Invest Buku Berhasil, Silahkan menu-
ggu proses persetujuan oleh Admin!');
    redirect(base_url().'Frontend/Owner/F_owner/invest_buku');
    die();
}
} else {
    $this->session-
>set_flashdata('alert_error', 'Proses insert data gagal !');
    redirect(base_url().'Frontend/Owner/F_owner/invest_buku');
    die();
}
}

public function invest_buku_update(){
    $reg_code = $this->session->userdata('reg_code');
    $register_code = $reg_code;
    $buku_id = $this->uri->segment(5);
    $id_invest_book = $this->uri->segment(6);
    $judul_buku = $this->input->post('judul_buku');
    $pengarang = $this->input->post('pengarang');
    $penerbit = $this->input->post('penerbit');
    $tahun = $this->input->post('tahun');
    $tahun = date("Y-m-d", strtotime($tahun));
    $kategori = $this->input->post('kategori');
    $sinopsis = $this->input->post('sinopsis');
    $cost = $this->input->post('cost');
}

```

```

    $data_buku = array(
        'judul_buku' => $judul_buku,
        'pengarang' => $pengarang,
        'penerbit' => $penerbit,
        'tahun' => $tahun,
        'kategori' => $kategori,
        'sinopsis' => $sinopsis,
        'date' => date("Y-m-d"),
        'foto_buku' => $file_upload['file_name'],
    );
    $update_buku = $this->crud_model-
>update('tb_buku','buku_id',$buku_id, $data_buku);
    $data_invest = array(
        'cost' => $cost,
    );
    $update_invest = $this->crud_model-
>update('tb_invest_book','id_invest_book',$id_invest_book, $data_invest);
    $this->session-
>set_flashdata('alert_success', 'Proses Invest Buku Berhasil, Silahkan menunggu proses persetujuan oleh Admin!');
    redirect(base_url().'Frontend/Owner/F_owner');
    die();
}
public function pengiriman(){
    $reg_code = $this->session->userdata('reg_code');
    $reg_password = $this->session->userdata('reg_password');
    $reg_status = $this->session->userdata('reg_status');
    $register_code = $reg_code;
    $data['get_code'] = $register_code;
    $where = array('register_id'=>$register_code);
    $header['bio_fullname'] = $this->crud_model-
>get_data('tb_lender_bio',$where)->result();
    $data['title'] = "My Book";
    $data['brd_title_main'] = "List Book";
    $data['brd_title_url'] = site_url('Frontend/Owner/F_owner');
    $data['register_code'] = $reg_code;
    $data['info_url'] = site_url('Frontend/Owner/F_owner/detail_pengiriman');
    $this->load->view('Frontend/website/partial/metadata');
    $this->load->view('Frontend/website/partial/header', $header);
    $this->load->view('Frontend/website/partial/navigation');
}

```

```

        $this->load->view('Frontend/owner/pr_a_pengiriman', $data);
        $this->load->view('Frontend/website/partial/footer');
    }

    public function detail_pengiriman(){
        $reg_code = $this->session->userdata('reg_code');
        $reg_password = $this->session->userdata('reg_password');
        $reg_status = $this->session->userdata('reg_status');
        $id_invest_book = $this->uri->segment(5);
        $data['id_invest_book'] = $id_invest_book;
        $register_code = $reg_code;
        $where = array('register_id'=>$register_code);
        $data['get_code'] = $register_code;
        $data['data_code'] = $this->personal_info_model-
>get_personal_info_lender($register_code);
        $header['bio_fullname'] = $this->crud_model-
>get_data('tb_lender_bio',$where)->result();
        $header['akun_owner'] = "active";
        $data['back_url'] = site_url('Frontend/Owner/F_owner');
        $data['insert_url'] = site_url('Frontend/Owner/F_owner/insert_pengiriman');
        $this->load->view('Frontend/website/partial/metadata');
        $this->load->view('Frontend/website/partial/header', $header);
        $this->load->view('Frontend/owner/pengiriman', $data);
        $this->load->view('Frontend/website/partial/navigation');
        $this->load->view('Frontend/website/partial/footer');
    }

    public function insert_pengiriman(){
        $reg_code = $this->session->userdata('reg_code');
        $register_code = $reg_code;
        $idMax_pengiriman = $this->loan_model->pengiriman_code();
        $noUrut_pengiriman=(int) substr($idMax_pengiriman[0]->maxID,3,9);
        $noUrut_pengiriman++;
        $newID_pengiriman="S".sprintf("%09s",$noUrut_pengiriman);
        $this->form_validation->set_rules("id_invest_book", "", "required");
        if ($this->form_validation->run() == true){
            $id_invest = $this->input->post('id_invest_book');
            if(!empty($_FILES['id_upload']['name'])){
                $file_upload_name = $this->input->post('bukti');
                $name_file = $file_upload_name.".".time().".".rand(0,1000);
                $config['upload_path'] = './uploads/pengiriman';

```

```

$config['allowed_types'] = 'gif|jpg|png|jpeg';
$config['max_size']      = '5120';
$config['file_name']    = $name_file;
$this->load->library('upload', $config);
if(!$this->upload->do_upload('id_upload')){
    $this->session-
>set_flashdata('alert_error', 'Upload foto buku gagal, ukuran file terlalu b
esar atau format file salah !');
    redirect(base_url().'Frontend/Owner/F_owner/pengiriman');
    die();
} else {
    $file_upload = $this->upload->data();
    $data_pengiriman = array(
        'pengiriman_id' => $newID_pengiriman,
        'register_id' => $register_code,
        'bukti' => $file_upload['file_name'],
    );
    $insert_pengiriman = $this->crud_model-
>insert('tb_pengiriman_buku', $data_pengiriman);
    $data_invest = array(
        'status' => "SubmitSend",
    );
    $update_invest= $this->crud_model-
>update('tb_invest_book','id_invest_book',$id_invest,$data_invest);
}
$this->session-
>set_flashdata('alert_success', 'Proses Pembayaran Buku Berhasil, Silahkan m
enunggu proses persetujuan oleh Admin!');
    redirect(base_url().'Frontend/Owner/F_owner/pengiriman');
    die();
}
} else {
$this->session-
>set_flashdata('alert_error', 'Proses insert data gagal !');
    redirect(base_url().'Frontend/Owner/F_owner/pengiriman');
    die();
}
}

public function invest_buku1(){
$reg_code = $this->session->userdata('reg_code');

```

```

$reg_password = $this->session->userdata('reg_password');
$reg_status = $this->session->userdata('reg_status');
$register_code = $reg_code;
$where = array('register_id'=>$register_code);
$data['get_code'] = $register_code;
$data['data_code'] = $this->personal_info_model-
>get_personal_info_lender($register_code);
$header['bio_fullname'] = $this->crud_model-
>get_data('tb_lender_bio',$where)->result();
$header['akun_owner'] = "active";
$data['back_url'] = site_url('Frontend/Owner/F_owner');
$data['form_url'] = site_url('Frontend/Owner/F_owner/invest_buku');
$this->load->view('Frontend/partial/header1');
$this->load->view('Frontend/partial/navigation', $navigation);
$this->load->view('Frontend/owner/akun_owner', $data);
}

public function account_owner(){
$reg_code = $this->session->userdata('reg_code');
$reg_password = $this->session->userdata('reg_password');
$reg_status = $this->session->userdata('reg_status');
$register_code = $reg_code;
$where = array('register_id'=>$register_code);
$data['get_code'] = $register_code;
$data['data_code'] = $this->personal_info_model-
>get_personal_info_lender($register_code);
$navigation['bio_fullname'] = $this->crud_model-
>get_data('tb_lender_bio',$where)->result();
$navigation['account_owner'] = "active";
$data['back_url'] = site_url('Frontend/Owner/F_owner');
$data['form_url'] = site_url('Frontend/Owner/F_owner/invest_buku');
$this->load->view('Frontend/partial/header1');
$this->load->view('Frontend/partial/navigation', $navigation);
$this->load->view('Frontend/owner/akun_owner', $data);
}

public function delete_buku()
{
$buku_id = $this->uri->segment(5);
if (!empty($buku_id)){
$this->crud_model->delete('tb_buku','buku_id',$buku_id);
}
}

```

```

        $this->session-
>set_flashdata('alert_success', 'Data berhasil dihapus');
        redirect(base_url().'Frontend/Owner/F_owner');
        die();
    } else {
        $this->session-
>set_flashdata('alert_error', 'Data gagal untuk dihapus !');
        redirect(base_url().'Frontend/Owner/F_owner');
        die();
    }
}
?>

```

9.3.2 Membuat Model *loan_model.php*

Berikut adalah syntax dari file *loan_model.php*

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class loan_model extends CI_Model {
    public function loan_buku()
    {
        return $this->db->query("SELECT max(loan_id) as maxID
        FROM tb_loan WHERE loan_id LIKE 'LN%'")->result();
    }
    public function pembayaran_code()
    {
        return $this->db->query("SELECT max(pembayaran_id) as maxID
        FROM tb_pembayaran WHERE pembayaran_id LIKE 'C%'")->result();
    }
    public function pengembalian_code()
    {
        return $this->db->query("SELECT max(pengembalian_id) as maxID
        FROM tb_pengembalian_buku WHERE pengembalian_id LIKE 'R%'")-
>result();
    }
    public function pengiriman_code()
    {
        return $this->db->query("SELECT max(pengiriman_id) as maxID
        FROM tb_pengiriman_buku WHERE pengiriman_id LIKE 'S%'")->result();
    }
}

```

```

public function get_book_loan_list($limit = null, $position = null, $id_invest_book = null, $register_code = null, $owner_name = null)
{
    $this->db->select("twb.* , ta.bio_fullname, ta.bio_phone, tc.*")
        ->from("tb_users_frontend twb");
    $this->db-
>join('tb_borrower_bio ta', 'ta.register_id = twb.register_id');
    $this->db->join('tb_loan tc', 'tc.register_id = twb.register_id');
    if (!empty($limit)){
        $this->db->limit($limit, $position);
    }
    if (!empty($id_invest_book)){
        $this->db->where("tc.invest_book", $id_invest_book);
    }
    if (!empty($register_code)){
        $this->db->where("tc.register_id", $register_code);
    }
    if (!empty($owner_name)){
        $this->db->like('ta.bio_fullname', $owner_name, 'both');
    }
    return $this->db->get();
}
}

```

9.3.3 Membuat View *invest_buku.php*

Berikut adalah syntax dari file *invest_buku.php*

```

<div class="content-wrapper">
    <!-- Content Header (Page header) -->
    <section class="content-header">
        <h1>Proses Upload Buku</h1>
        <ol class="breadcrumb">
            <li class="active"><a href="php echo site_url('Frontend/Owner/f_owner/invest_buku') ?&gt;"&gt;&lt;i class="fa fa-dashboar
</pre

```

```

<div class="col-xs-12">
    <?php if ($this->session->flashdata('alert_success')) { ?>
        <div class="alert alert-success alert-dismissible">
            <button type="button" class="close" data-
dismiss="alert" aria-hidden="true">&times;</button>
            <p><i class="icon fa fa-check"></i>
                <?php echo $this->session-
flashdata('alert_success'); ?></p>
        </div>
    <?php } ?>
    <?php if ($this->session->flashdata('alert_error')) { ?>
        <div class="alert alert-danger alert-dismissible">
            <button type="button" class="close" data-
dismiss="alert" aria-hidden="true">&times;</button>
            <p><i class="icon fa fa-ban"></i>
                <?php echo $this->session-
flashdata('alert_error'); ?></p>
        </div>
    <?php } ?>
<div class="box">
    <h3 style="text-align: left; margin-left: 2%;">Data Buku</h3>
    <div class="box-body">
        <div class="row">
            <div class="col-xs-12 col-lg-12" >
                <div class="col-xs-12 col-lg-12" style="overflow-y: auto;">
                    <div class="col-md-12 col-sm-12 col-xs-12" style="text-
align: right;">
                        </div>
                    <script>
                        function sum() {
                            var txtFirstNumberValue = document.getElementById('txt1').value;
                            var txtSecondNumberValue = document.getElementById('txt2')>
                                .value;
                            var result = parseInt(txtFirstNumberValue) + parseInt(txt
SecondNumberValue);
                            if (!isNaN(result)) {
                                document.getElementById('txt3').value = result;
                            }
                        }
                    </script>
                </div>
            </div>
        </div>
    </div>
</div>

```

```

        </script>
        <table id="" class="table table-bordered table-striped">
            <thead>
            </thead>
            <tbody>
                <div class="col-md-6 col-sm-12 col-xs-12" id="left3">
                    <form action="php echo $insert_url; ?" method="post" enctype="multipart/form-data">
                        <br>
                        <div class="form-group">
                            <label>Nama Buku</label>
                            <input type="text" class="form-control" id="judul_buku" name="judul_buku" minlength="1" maxlength="200" placeholder="Nama Buku" required="true">
                        </div>
                        <div class="form-group">
                            <label>Pengarang Buku</label>
                            <input type="text" class="form-control" id="pengarang" name="pengarang" minlength="1" maxlength="200" placeholder="Pengarang Buku" required="true">
                        </div>
                        <div class="form-group">
                            <label>Penerbit Buku</label>
                            <input type="text" class="form-control" id="penerbit" name="penerbit" minlength="1" maxlength="200" placeholder="Penerbit Buku" required="true">
                        </div>
                        <div class="form-group">
                            <label>Tahun Terbit Buku</label>
                            <div class="input-group">
                                <div class="input-group-addon">
                                    <i class="fa fa-calendar" style="display: inline-block;font: normal normal normal 14px/1 FontAwesome;font-size: inherit;text-rendering: auto; -webkit-font-smoothing: antialiased; -moz-osx-font-smoothing: grayscale;padding: 0; width: 11px; margin: 0;"></i>
                                </div>
                                <input type="text" name="tahun" class="form-control pull-right datepicker_date" id="tahun" required="required">
                            </div>
                        </div>
                    </form>
                </div>
            </tbody>
        </table>
    
```

```

        </div>
        <div class="form-group">
            <label>Kategori</label>
            <select class="form-control" style="width: 100%;" name="kategori">
                <option value="">Select Kategori</option>
                <option value="Anak">Anak-Anak</option>
                <option value="Novel">Novel</option>
                <option value="Dongeng">Dongeng</option>
                <option value="Biografi">Biografi</option>
                <option value="Karya Ilmiah">Karya ilmiah</option>
                <option value="Ilmiah">Ilmiah</option>
                <option value="Ensiklopedia">Ensiklopedia</option>
                <option value="Sejarah">Sejarah</option>
                <option value="Komputer">Komputer</option>
            </select>
        </div>
        <div class="form-group">
            <label>Sinopsis Buku</label>
            <textarea id="sinopsis" rows="3" name="sinopsis" class="form-control" minlength="1" maxlength="225" required="required" placeholder="Sinopsis Buku Minimal 225 Karakter"></textarea>
        </div>
    </div>
    <div class="col-md-6 col-sm-12 col-xs-12">
        <br>
        <div class="form-group">
            <label>Upload Foto Buku</label>
            <input type="file" onchange="Validate_file(this); id="id_upload" name="id_upload" class="form-control" required="required">
            <span style="font-size: 12px;">* tunggu beberapa saat hingga file/gambar muncul.<br> * Format file gambar harus berekstensi JPG, JPEG atau PNG dan maksimal berukuran 5MB</span>
            Harga Buku + Ongkir</label>
        <input type="text" class="form-
control" id="txt1" onkeyup="sum();" name="harga_buku" minlength="1" maxlength
="200" placeholder="Harga Buku + Ongkir" required="true">
    </div>
    <div class="form-group">
        <label>Biaya Admin</label>
        <input type="text" class="form-
control" name="biaya_admin" id="txt2" onkeyup="sum(); value=<?php echo "1
0000"; ?>" readonly>
    </div>
    <div class="form-group">
        <label>Total Bayar</label>
        <input type="text" class="form-
control" id="txt3" name="cost" minlength="1" maxlength="200" placeholder="Jum
lah Bayar Yang Harus Dibayar Oleh Peminjam" required="true">
        <span style="font-
size: 12px;">* Jumlah Bayar yang harus dibayarkan untuk setiap peminjam minim
al Rp.150.000. Biaya tersebut sudah termasuk harga buku sebenarnya, biaya adm
inistrasi dan ongkos kirim</span>
    </div>
    <br><br><br>
    <div class="form-group" style="text-align: center">
        <button type="submit" class="btn btn-warning btn-
sm btn-block">Submit</button>
    </div>
</form>
</tbody>
</table>
</div>
</div>
<!-- /.box-body -->
</div></div></div></div></div>
</section>
<script type="text/javascript">
    function clear_file(){
        var noimage = "<?php echo base_url();?>uploads/buku/noimage.jpg";
        document.getElementById("id_upload").value = null;

```

```

        $('#ktp_upload').attr('src', noimage);
        document.getElementById("error_ocr").style.color = "black";
        document.getElementById('error_ocr').innerHTML = "";
    }

    function preview_foto(){
        var input = document.getElementById('id_upload');
        var file = input.files[0];
        var reader = new FileReader();
        var noimage = "<?php echo base_url();?>uploads/buku/noimage.jpg";
        reader.onloadend = function(e) {
            $('#ktp_upload').attr('src', reader.result);
        }
        if (file) {
            reader.readAsDataURL(file);
        } else {
            $('#ktp_upload').attr('src', noimage);
        }
    }

    function Validate_file(oInput) {
        var _validFileExtensions = [".jpg", ".jpeg", ".png"];
        var input = document.getElementById('id_upload');
        var picsize = input.files[0].size;
        if (oInput.type == "file") {
            var sFileName = oInput.value;
            if (sFileName.length > 0) {
                var blnValid = false;
                for (var j = 0; j < _validFileExtensions.length; j++) {
                    var sCurExtension = _validFileExtensions[j];
                    if (sFileName.substr(sFileName.length - sCurExtension.length,
sCurExtension.length).toLowerCase() == sCurExtension.toLowerCase()) {
                        blnValid = true;
                        break;
                    }
                }
                if (picsize > 5000000){
                    alert("Ukuran Data yang di upload tidak sesuai, ukuran yang sesuai adalah dibawah 5 MB");
                    clear_file();
                    return false;
                } else if (!blnValid) {

```

```

        alert("Data yang di upload tidak di izinkan, yang di izinkan
adalah: " + _validFileExtensions.join(", "));
        oInput.value = "";
        clear_file();
        return false;
    } else {
        preview_foto();
    }
}
return true;
}
</script>

```

9.3.4 Membuat Model *invest_book.php*

Berikut adalah syntax dari file *invest_book.php*

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class invest_book extends CI_Model {
    public function get_book_owner_list($limit = null, $position = null, $sta
tus, $id_invest_book = null, $register_code = null, $start_date = null, $end_
date = null, $owner_name = null)
    {
        $this->db->select("twb.* , ta.bio_fullname, ta.bio_phone, tb.*")
            ->from("tb_users_frontend twb");
        $this->db-
>join('tb_lender_bio ta', 'ta.register_id = twb.register_id');
        $this->db-
>join('tb_invest_book tb', 'tb.register_id = twb.register_id');
        $this->db->where("tb.status", $status);
        if (!empty($limit)){
            $this->db->limit($limit, $position);
        }
        if (!empty($id_invest_book)){
            $this->db->where("tb.id_invest_book", $id_invest_book);
        }
        if (!empty($register_code)){
            $this->db->where("tb.register_id", $register_code);
        }
        if (!empty($start_date)){

```

```

        $this->db->where("tb.invest_date >= '".$start_date."'");
    }
    if (!empty($end_date)){
        $this->db->where("tb.invest_date <= '".$end_date."'");
    }
    if (!empty($owner_name)){
        $this->db->like('ta.bio_fullname', $owner_name, 'both');
    }
    $this->db->order_by("tb.invest_date", "DESC");
    return $this->db->get();
}

public function get_book($limit = null, $position = null, $id_invest_book
= null, $register_code = null, $start_date = null, $end_date = null, $owner_
name = null)
{
    $this->db->select("twb.* , ta.bio_fullname, ta.bio_phone, tb.*")
        ->from("tb_users_frontend twb");
    $this->db-
>join('tb_lender_bio ta', 'ta.register_id = twb.register_id');
    $this->db-
>join('tb_invest_book tb', 'tb.register_id = twb.register_id');
    if (!empty($limit)){
        $this->db->limit($limit, $position);
    }
    if (!empty($id_invest_book)){
        $this->db->where("tb.id_invest_book", $id_invest_book);
    }
    if (!empty($register_code)){
        $this->db->where("tb.register_code", $register_code);
    }
    if (!empty($start_date)){
        $this->db->where("tb.invest_date >= '".$start_date."'");
    }
    if (!empty($end_date)){
        $this->db->where("tb.invest_date <= '".$end_date."'");
    }
    if (!empty($owner_name)){
        $this->db->like('ta.bio_fullname', $owner_name, 'both');
    }
    $this->db->order_by("tb.invest_date", "DESC");
}

```

```

        return $this->db->get();
    }

    public function get_book_owner_list2($limit = null, $position = null, $status,
        $id_invest_book = null, $register_code = null, $start_date = null, $end_
        _date = null, $owner_name = null)
    {
        $this->db->select("twb.* , ta.bio_fullname, ta.bio_phone, tb.*")
            ->from("tb_users_frontend twb");
        $this->db-
>join('tb_lender_bio ta', 'ta.register_id = twb.register_id');
        $this->db-
>join('tb_invest_book tb', 'tb.register_id = twb.register_id');
        $this->db->where("tb.status" == 'Checking Loan');
        if (!empty($limit)){
            $this->db->limit($limit, $position);
        }
        if (!empty($id_invest_book)){
            $this->db->where("tb.id_invest_book", $id_invest_book);
        }
        if (!empty($register_code)){
            $this->db->where("tb.register_id", $register_code);
        }
        if (!empty($start_date)){
            $this->db->where("tb.invest_date >= '". $start_date . "'");
        }
        if (!empty($end_date)){
            $this->db->where("tb.invest_date <= '". $end_date . "'");
        }
        if (!empty($owner_name)){
            $this->db->like('ta.bio_fullname', $owner_name, 'both');
        }
        $this->db->order_by("tb.invest_date", "DESC");
        return $this->db->get();
    }

    public function get_book_owner()
    {
        $this->db->select("twb.* , ta.bio_fullname, ta.bio_phone, tb.*")
            ->from("tb_users_frontend twb");
        $this->db-
>join('tb_lender_bio ta', 'ta.register_id = twb.register_id');

```

```

    $this->db-
>join('tb_invest_book tb', 'tb.register_id = twb.register_id');
    return $this->db->get();
}
public function get_book_owner_list3($limit = null, $position = null, $status,
$id_invest_book = null, $register_code = null, $start_date = null, $end_
date = null, $owner_name = null)
{
    $this->db->select("twb.* , ta.bio_fullname, ta.bio_phone, tb.*")
    ->from("tb_users_frontend twb");
    $this->db-
>join('tb_lender_bio ta', 'ta.register_id = twb.register_id');
    $this->db-
>join('tb_invest_book tb', 'tb.register_id = twb.register_id');
    $this->db->where("tb.status", $status);
    if (!empty($limit)){
        $this->db->limit($limit, $position);
    }
    if (!empty($id_invest_book)){
        $this->db->where("tb.id_invest_book", $id_invest_book);
    }
    if (!empty($register_code)){
        $this->db->where("tb.register_code", $register_code);
    }
    if (!empty($start_date)){
        $this->db->where("tb.invest_date >= '". $start_date . "'");
    }
    if (!empty($end_date)){
        $this->db->where("tb.invest_date <= '". $end_date . "'");
    }
    if (!empty($owner_name)){
        $this->db->like('ta.bio_fullname', $owner_name, 'both');
    }
    $this->db->order_by("tb.invest_date", "DESC");
    return $this->db->get();
}
public function get_book_loan_list($limit = null, $position = null, $stat
us, $id_invest_book = null, $register_code = null, $owner_name = null)
{
    $this->db->select("twb.* , ta.bio_fullname, ta.bio_phone, tc.*")

```

```

        ->from("tb_users_frontend twb");
        $this->db-
>join('tb_borrower_bio ta', 'ta.register_id = twb.register_id');
        $this->db->join('tb_loan tc', 'tc.register_id = twb.register_id');
        $this->db->where("tc.status", $status);
        if (!empty($limit)){
            $this->db->limit($limit, $position);
        }
        if (!empty($id_invest_book)){
            $this->db->where("tc.invest_book", $id_invest_book);
        }
        if (!empty($register_code)){
            $this->db->where("tc.register_id", $register_code);
        }
        if (!empty($owner_name)){
            $this->db->like('ta.bio_fullname', $owner_name, 'both');
        }
        $this->db->order_by("tc.loan_date", "DESC");
        return $this->db->get();
    }

    public function get_book_loan_list2($limit = null, $position = null, $status, $loan_id = null, $id_invest_book = null, $register_code = null, $owner_name = null)
    {
        $this->db->select("twb.* , ta.bio_fullname, ta.bio_phone, tc.*")
        ->from("tb_users_frontend twb");
        $this->db-
>join('tb_borrower_bio ta', 'ta.register_id = twb.register_id');
        $this->db->join('tb_loan tc', 'tc.register_id = twb.register_id');
        $this->db-
>join('tb_invest_book td', 'tc.id_invest_book = td.id_invest_book');
        $this->db->where("td.status", $status);
        if (!empty($limit)){
            $this->db->limit($limit, $position);
        }
        if (!empty($id_invest_book)){
            $this->db->where("tc.invest_book", $id_invest_book);
        }
        if (!empty($register_code)){
            $this->db->where("tc.register_id", $register_code);
        }
    }
}

```

```

        }
        if (!empty($loan_id)){
            $this->db->where("tc.loan_id", $loan_id);
        }
        if (!empty($owner_name)){
            $this->db->like('ta.bio_fullname', $owner_name, 'both');
        }
        $this->db->order_by("tc.loan_date", "DESC");
        return $this->db->get();
    }

    public function invest_code()
    {
        return $this->db->query("SELECT max(id_invest_book) as maxID
        FROM tb_invest_book WHERE id_invest_book LIKE 'I%'")->result();
    }

    public function invest_buku()
    {
        return $this->db->query("SELECT max(buku_id) as maxID
        FROM tb_buku WHERE buku_id LIKE 'B%'")->result();
    }

    public function data_buku($limit = null, $position = null, $register_code)
    {
        $this->db->select("twb.*", ta.*")
        ->from("tb_buku twb");
        $this->db->join('tb_invest_book ta', 'ta.buku_id = twb.buku_id');
        if (!empty($limit)){
            $this->db->limit($limit, $position);
        }
        $this->db->where("twb.register_id", $register_code);
        return $this->db->get();
    }

    public function data_buku2($register_code)
    {
        $this->db->select("twb.*")
        ->from("tb_buku twb");
        if (!empty($limit)){
            $this->db->limit($limit, $position);
        }
        $this->db->where("twb.register_id", $register_code);
    }
}

```

```

        return $this->db->get();
    }

    public function data_buku_owner($status)
    {
        $this->db->select("twb.* , ta.bio_fullname, ta.bio_phone, tb.* , tbk.*")
        ->from("tb_users_frontend twb");
        $this->db->join('tb_lender_bio ta', 'ta.register_id = twb.register_id');
        $this->db-
        >join('tb_invest_book tb', 'tb.register_id = twb.register_id');
        $this->db->join('tb_buku tbk', 'tb.buku_id = tbk.buku_id');
        $this->db->where("tb.status", $status);
        return $this->db->get();
    }

    public function get_book_owner_list1($limit = null, $position = null, $st
atus, $id_invest_book = null, $register_code = null, $start_date = null, $end
_date = null, $owner_name = null)
{
    $this->db->select("twb.* , ta.bio_fullname, ta.bio_phone, tb.* , tc.*")
        ->from("tb_users_frontend twb");
    $this->db-
    >join('tb_lender_bio ta', 'ta.register_id = twb.register_id');
    $this->db-
    >join('tb_invest_book tb', 'tb.register_id = twb.register_id');
    $this->db->join('tb_buku tc', 'tc.register_id = twb.register_id');
    $this->db->where("tb.status", $status);
    if (!empty($limit)){
        $this->db->limit($limit, $position);
    }
    if (!empty($id_invest_book)){
        $this->db->where("tb.id_invest_book", $id_invest_book);
    }
    if (!empty($register_code)){
        $this->db->where("tb.register_code", $register_code);
    }
    if (!empty($start_date)){
        $this->db->where("tb.invest_date >= '". $start_date . "'");
    }
    if (!empty($end_date)){
        $this->db->where("tb.invest_date <= '". $end_date . "'");
    }
}

```

```

        if (!empty($owner_name)){
            $this->db->like('ta.bio_fullname', $owner_name, 'both');
        }
        $this->db->order_by("tb.invest_date", "DESC");
        return $this->db->get();
    }

    public function data_buku_1($limit = null, $position = null, $status = null)
    {
        $this->db->select("twb.*")
        ->from("tb_buku twb");
        if (!empty($limit)){
            $this->db->limit($limit, $position);
        }
        return $this->db->get();
    }

```

9.3.5 Membuat View cek_peminjaman.php

Berikut adalah syntax dari file cek_peminjaman.php

```

<?php date_default_timezone_set("Asia/Jakarta"); ?>
<!-- Contains page content -->
<div class="content-wrapper">
    <!-- Content Header (Page header) -->
    <section class="content-header">
        <h1>
            <?php echo $title; ?>
            <small></small>
        </h1>
        <ol class="breadcrumb">
            <li class="active"><a href="<?php echo $brd_title_url; ?>"><i class="fa fa-dashboard"></i> <?php echo $brd_title_main; ?></a></li>
        </ol>
    </section>
    <!-- Main content -->
    <section class="content">
        <div class="row">
            <div class="col-xs-12">
                <?php if ($this->session->flashdata('alert_success')) { ?>
                    <div class="alert alert-success alert-dismissible">

```

```

        <button type="button" class="close" data-
dismiss="alert" aria-hidden="true">&times;</button>
        <p><i class="icon fa fa-check"></i>
        <?php echo $this->session-
>flashdata('alert_success'); ?></p>
        </div>
        <?php } ?>
        <?php if ($this->session->flashdata('alert_error')) { ?>
        <div class="alert alert-danger alert-dismissible">
        <button type="button" class="close" data-
dismiss="alert" aria-hidden="true">&times;</button>
        <p><i class="icon fa fa-ban"></i>
        <?php echo $this->session->flashdata('alert_error'); ?></p>
        </div>
        <?php }

?>
<div class="box">
    <div class="box-header with-border">
        <div class="row">
            <div class="col-lg-12 col-xs-12">
                <form method="get" action="">
                    <div class="col-lg-6 col-xs-12">
                        <?php
                            $start_date = isset($_REQUEST['start_date']) ? $_REQUEST['start_date'] : date('01-01-2019');
                            $end_date = isset($_REQUEST['end_date']) ? $_REQUEST['end_date'] : date('d-m-Y');
                            $id_invest_book = isset($_REQUEST['id_invest_book']) ? $_REQUEST['id_invest_book'] : null;
                            $register_code_page = isset($_REQUEST['register_code']) ? $_REQUEST['register_code'] : null;
                            $owner_name = isset($_REQUEST['owner_name']) ? $_REQUEST['owner_name'] : null;
                        ?>
                    <div class="form-group">
                        <label for="">Start Date </label>
                        <div class="input-group">
                            <div class="input-group-addon">
                                <i class="fa fa-calendar"></i>
                            </div>
                        </div>
                    </div>
                </form>
            </div>
        </div>
    </div>
</div>

```

```

        <input type="text" name="start_date" class="form-control pull-right datepicker_date" value="= $start_date; ?" placeholder="Start Date" id="start_date">
            </div>
        </div>
        </div>
        <div class="col-lg-6 col-xs-12">
            <div class="form-group">
                <label for="">End Date</label>
                <div class="input-group">
                    <div class="input-group-addon">
                        <i class="fa fa-calendar"></i>
                    </div>
                    <input type="text" name="end_date" class="form-control pull-right datepicker_date" value="= $end_date; ?" placeholder="End Date" id="end_date">
                </div>
            </div>
            <br>
            <button type="submit" class="btn btn-warning">Search !</button>
        </div>
    </form>
</div>
</div>
</div>
<div class="box-body">
    <div class="row">
        <div class="col-xs-12 col-lg-12" >
            <div class="col-xs-12 col-lg-12" style="overflow-y: auto;">
                <table id="" class="table table-bordered table-striped">
                    <thead>
                        <tr>
                            <th style="text-align: center;">No</th>
                            <th style="text-align: center;">ID Peminjam</th>
                            <th style="text-align: center;">Tanggal Peminjaman</th>
                            <th style="text-align: center;">Nama Peminjam</th>
                            <th style="text-align: center;">Nama Buku</th>

```

```

        <th style="text-align: center;">Jumlah Bayar</th>
        <th style="text-align: center;">Status</th>
        <th width="80" style="text-align: center;">Aksi</th>
    </tr>
</thead>
<tbody>
<?php
$this->load->model("crud_model");
$data_setting = $this->crud_model->get_setting();
$limit      = $data_setting[0]->setting_pagination;
$page       = isset($_REQUEST['page']) ? $_REQUEST['page'] : null;
$start_date = date("Y-m-d", strtotime($start_date));
$end_date   = date("Y-m-d", strtotime($end_date));
$start_date_page = date("d-m-Y", strtotime($start_date));
$end_date_page = date("d-m-Y", strtotime($end_date));
if(empty($page)){
    $position  = 0;
    $page     = 1;
} else {
    $position = ($page-1) * $limit;
}
$this->load->model('Front_data/invest_book');
$data_loan = $this->invest_book-
>get_book_loan_list2(@$limit, @$position, 'ApprovedLoan', @$id_invest_book,
@$register_code_page, @$owner_name)->result();
$no = $position+1;
$total_loan_amount=0;
foreach ($data_loan as $loan_entry){
    $register_code_borrower = $loan_entry->register_id;
    $data_bio = $this->personal_info_model-
>get_personal_info_borrower($register_code_borrower);
    $data_book = $this->personal_info_model-
>get_personal_info_invest_owner3($loan_entry-
>id_invest_book);
    ?>
    <tr>
        <td style="text-align: center;"><?php echo $no; ?></td>
        <td style="text-align: center;"><?php echo $loan_entry-
>register_id; ?></td>

```

```

        <td style="text-align: center;"><?php echo date('d/m/Y', strtotime($loan_entry->loan_date)); ?></td>
            <td style="text-align: center;"><?php echo $data_bio[0]->bio_fullname; ?></td>
            <td style="text-align: center;"><?php echo $data_book[0]->judul_buku; ?></td>
            <td style="text-align: center;"><?php echo $data_book[0]->cost; ?></td>
            <td style="text-align: center;"><?php echo $data_book[0]->status; ?></td>
            <td class="text-center">
                <a href=<?php echo $info_url; ?>/<?php echo $loan_entry->loan_id; ?>/<?php echo $loan_entry->register_id; ?>/<?php echo $data_book[0]->id_invest_book; ?>" id="info" class="btn btn-info btn-sm btnwrdt">Detail</a>
            </td>
        </tr>
    </?php
    $total_loan_amount = $total_loan_amount + $data_book[0]->cost;
    $no++;
}
?>
<tr>
    <td class="text-center" colspan="5"><strong>TOTAL</strong></td>
    <td class="text-right"><strong><?php echo number_format($total_loan_amount,0,".", "."); ?></strong></td>
        <td></td>
        <td></td>
    </tr>
</tbody>
</table>
<?php
$data_rows = $this->invest_book->get_book_owner_list(null, null, 'Checking', $id_invest_book, $register_code_page, $start_date, $end_date, $owner_name)->num_rows();
$all_page = ceil($data_rows/$limit);
?>
```

```

<center>
    <ul class="pagination">
        <li>
            <?php
                if($page > 1){
                    $prev = $page-1;
                    echo "<a href='".$base_url()."Backend/Book/B_submission
_book?page=$prev&start_date=".$start_date_page."&end_date=".$end_date_page
."&id_invest_book=".$id_invest_book."&register_code=".$register_code_page.
"&owner_name=".$owner_name."'>Previous</a>";
                } else {
                    echo "<a href='#' style='color:black;'>Previous</a>";
                }
            ?>
        </li>
        <li>
            <?php
                $showPage = $page;
                for($i=1;$i<=$all_page;$i++){
                    if (((($i >= $page - 3) && ($i <= $page + 3)) || ($i ==
1) || ($i == $all_page))
                    {
                        if ($all_page > 1){
                            if (($page != ($all_page - 1)) && ($i == $all_page
))
                                echo "<a href='#'>...</a>";
                        }
                        if ($i == $page) echo " <a href='#' style='background
d-color:gray; color:white;'">".$i."</a>";
                        else echo "<a href='".$base_url()."Backend/Book/B_sub
mission_book?page=$i&start_date=".$start_date_page."&end_date=".$end_date_
page."&id_invest_book=".$id_invest_book."&register_code=".$register_code_p
age."&owner_name=".$owner_name."'>".$i."</a>";
                    }
                }
            ?>
        </li>
        <li>
            <?php
                if($page < $all_page){
                    $next=$page+1;

```

```

        echo "<a href='".$base_url()."Backend/Book/B_submission
_book?page=$next&start_date=".@$start_date_page."&end_date=".@$end_date_page
."&id_invest_book=".@$id_invest_book."&register_code=".@$register_code_page.
"&owner_name=".@$owner_name."'>Next</a>";
    } else {
        echo "<a href='#" style='color:black;'>Next</a>";
    }
?
</li>
</ul>
</center>
</div></div></div> </div>
<!-- /.box-body -->
</div></div></div>
</section>
</div>

```

9.3.6 Membuat View cek_pengembalian.php

Berikut adalah syntax dari file cek_pengembalian.php

```

<?php date_default_timezone_set("Asia/Jakarta"); ?>
<!-- Contains page content -->
<div class="content-wrapper">
    <!-- Content Header (Page header) -->
    <section class="content-header">
        <h1>
            <?php echo $title; ?>
            <small></small>
        </h1>
        <ol class="breadcrumb">
            <li class="active"><a href=<?php echo $brd_title_url; ?>><i class
="fa fa-dashboard"></i> <?php echo $brd_title_main; ?></a></li>
        </ol>
    </section>
    <!-- Main content -->
    <section class="content">
        <div class="row">
            <div class="col-xs-12">
                <?php if ($this->session->flashdata('alert_success')) { ?>
                    <div class="alert alert-success alert-dismissible">

```

```

        <button type="button" class="close" data-
dismiss="alert" aria-hidden="true">&times;</button>
        <p><i class="icon fa fa-check"></i>
        <?php echo $this->session-
>flashdata('alert_success'); ?></p>
        </div>
        <?php } ?>
        <?php if ($this->session->flashdata('alert_error')) { ?>
        <div class="alert alert-danger alert-dismissible">
            <button type="button" class="close" data-
dismiss="alert" aria-hidden="true">&times;</button>
            <p><i class="icon fa fa-ban"></i>
            <?php echo $this->session->flashdata('alert_error'); ?></p>
            </div>
            <?php } ?>
<div class="box">
    <div class="box-header with-border">
        <div class="row">
            <div class="col-lg-12 col-xs-12">
                <form method="get" action="">
                    <div class="col-lg-6 col-xs-12">
                        <?php
                            $start_date = isset($_REQUEST['start_date']) ? $_REQUEST['start_date'] : date('01-01-2019');
                            $end_date = isset($_REQUEST['end_date']) ? $_REQUEST['end_date'] : date('d-m-Y');
                            $id_invest_book = isset($_REQUEST['id_invest_book']) ? $_REQUEST['id_invest_book'] : null;
                            $register_code_page = isset($_REQUEST['register_code']) ? $_REQUEST['register_code'] : null;
                            $owner_name = isset($_REQUEST['owner_name']) ? $_REQUEST['owner_name'] : null;
                        ?>
                    <div class="form-group">
                        <label for="">Start Date </label>
                        <div class="input-group">
                            <div class="input-group-addon">
                                <i class="fa fa-calendar"></i>
                            </div>

```

```

        <input type="text" name="start_date" class="form-control pull-right datepicker_date" value="php echo @$start_date; ?" placeholder="Start Date" id="start_date">
            </div></div> </div>
        <div class="col-lg-6 col-xs-12">
            <div class="form-group">
                <label for="">End Date</label>
                <div class="input-group">
                    <div class="input-group-addon">
                        <i class="fa fa-calendar"></i>
                    </div>
                <input type="text" name="end_date" class="form-control pull-right datepicker_date" value="php echo @$end_date; ?" placeholder="End Date" id="end_date">
                    </div><div></div>
                <div class="col-lg-12 col-xs-12" align="right">
                    <br>
                    <button type="submit" class="btn btn-warning">Search !</button>
                </div>
            </form>
        </div></div></div>
        <div class="box-body">
            <div class="row">
                <div class="col-xs-12 col-lg-12" >
                    <div class="col-xs-12 col-lg-12" style="overflow-y: auto;">
                        <table id="" class="table table-bordered table-striped">
                            <thead>
                                <tr>
                                    <th style="text-align: center;">No</th>
                                    <th style="text-align: center;">ID Peminjam</th>
                                    <th style="text-align: center;">Tanggal Peminjaman</th>
                                    <th style="text-align: center;">Nama Peminjam</th>
                                    <th style="text-align: center;">Nama Buku</th>
                                    <th style="text-align: center;">Jumlah Bayar</th>
                                    <th style="text-align: center;">Status</th>
                                    <th width="80" style="text-align: center;">Aksi</th>
                                </tr>
                            </thead>
                            <tbody>

```

```

<?php
$this->load->model("crud_model");
$data_setting = $this->crud_model->get_setting();
$limit      = $data_setting[0]->setting_pagination;
$page = isset($_REQUEST['page']) ? $_REQUEST['page'] : null;
$start_date = date("Y-m-d", strtotime($start_date));
$end_date   = date("Y-m-d", strtotime($end_date));
$start_date_page = date("d-m-Y", strtotime($start_date));
$end_date_page = date("d-m-Y", strtotime($end_date));
if(empty($page)){
    $position  = 0;
    $page = 1;
} else {
    $position = ($page-1) * $limit;
}
$this->load->model('Front_data/invest_book');
$data_loan = $this->invest_book-
>get_book_loan_list2(@$limit, @$position, 'SendReturn', @$id_invest_book, @
$register_code_page, @$owner_name)->result();
$no = $position+1;
$total_loan_amount=0;
foreach ($data_loan as $loan_entry){
    $register_code_borrower = $loan_entry->register_id;
    $data_bio = $this->personal_info_model-
>get_personal_info_borrower($register_code_borrower);
    $data_book = $this->personal_info_model-
>get_personal_info_invest_owner3($loan_entry->id_invest_book);
    ?
    <tr>
        <td style="text-align: center;"><?php echo $no; ?></td>
        <td style="text-align: center;"><?php echo $loan_entry-
>register_id; ?></td>
        <td style="text-
align: center;"><?php echo date('d/m/Y', strtotime($loan_entry-
>loan_date)); ?></td>
        <td style="text-align: center;"><?php echo $data_bio[0]-
>bio_fullname; ?></td>
        <td style="text-align: center;"><?php echo $data_book[0]-
>judul_buku; ?></td>

```

```

        <td style="text-align: center;"><?php echo $data_book[0]-
>cost; ?></td>
        <td style="text-align: center;"><?php echo $data_book[0]-
>status; ?></td>
        <td class="text-center">
            <a href=<?php echo $info_url; ?>/<?php echo $loan_entr
y->loan_id; ?>/<?php echo $loan_entry-
>register_id; ?>/<?php echo $data_book[0]-
>id_invest_book; ?>" id="info" class="btn btn-info btn-
sm btnwrdt">Detail</a>
        </td>
    </tr>
    <?php
        $total_loan_amount = $total_loan_amount + $data_book[0]->cost;
        $no++;
    }
    ?>
    <tr>
        <td class="text-
center" colspan="5"><strong>TOTAL</strong></td>
        <td class="text-
right"><strong><?php echo number_format($total_loan_amount,0,".",".");?></s
trong></td>
        <td></td>
        <td></td>
    </tr>
    </tbody>
</table>
<?php
    $data_rows = $this->invest_book-
>get_book_owner_list(null, null, 'SendReturn', $id_invest_book, $register_c
ode_page, $start_date, $end_date, $owner_name)->num_rows();
    $all_page = ceil($data_rows/$limit);
?>
<center>
    <ul class="pagination">
        <li>
            <?php
                if($page > 1){
                    $prev = $page-1;

```

```

        echo "<a href='".$base_url()."Backend/Book/B_submissio
n_book?page=$prev&start_date=".@$start_date_page."&end_date=".@$end_date_pa
ge."&id_invest_book=".@$id_invest_book."&register_code=".@$register_code_pa
ge."&owner_name=".@$owner_name."'>Previous</a>";
    } else {
        echo "<a href='#' style='color:black;'>Previous</a>";
    }
?>
</li>
<li>
<?php
$showPage = $page;
for($i=1;$i<=$all_page;$i++){
    if (((($i >= $page - 3) && ($i <= $page + 3)) || ($i =
= 1) || ($i == $all_page))
    {
        if ($all_page > 1){
            if (($page != ($all_page - 1)) && ($i == $all_pag
e)) echo "<a href='#'"...</a>";
        }
        if ($i == $page) echo " <a href='#" style='background-color:gray; color:whiteSmoke;'>".$i."</a>";
            else echo "<a href='".$base_url()."Backend/Book/B_su
bmission_book?page=$i&start_date=".@$start_date_page."&end_date=".@$end_dat
e_page."&id_invest_book=".@$id_invest_book."&register_code=".@$register_cod
e_page."&owner_name=".@$owner_name."'>".$i."</a>";
        }
    }
?>
</li>
<li>
<?php
if($page < $all_page){
    $next=$page+1;
    echo "<a href='".$base_url()."Backend/Book/B_submissio
n_book?page=$next&start_date=".@$start_date_page."&end_date=".@$end_date_pa
ge."&id_invest_book=".@$id_invest_book."&register_code=".@$register_code_pa
ge."&owner_name=".@$owner_name."'>Next</a>";
} else {
    echo "<a href='#' style='color:black;'>Next</a>";
}

```

```

        }
    ?>
    </li>
</ul>
</center>
</div></div> </div></div>
<!-- /.box-body -->
</div></div></div>
</section>
</div>

```

9.4 Membuat Proses Pada Bagian Peminjam

Ada beberapa proses yang terdapat pada peminjam, diantaranya Meminjam buku, mengembalikan buku dan *upload* bukti kirim kembali. Nah dalam pembuatan proses-proses tersebut, kami hanya membuat satu *controller* dalam mengelola proses tersebut.

9.4.1 Membuat Controller *F_Borrower.php*

Berikut adalah syntax dari file *F_Borrower.php*

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');
class F_borrower extends CI_Controller {
    function __construct(){
        parent::__construct();
        $this->load->library('form_validation');
        $this->load->model('crud_model');
        $this->load->model('Auth_model');
        $this->load->model('Front_data/personal_info_model');
        $this->load->model('Front_data/invest_book');
        $this->load->model('Front_data/loan_model');
        $this->load->model('Front_data/register_model');
        $this->load->model('Website/users_model');
        $this->load->model('search_model');
        date_default_timezone_set("Asia/Jakarta");
        $reg_code = $this->session->userdata('reg_code');
        $reg_password = $this->session->userdata('reg_password');
    }
}

```

```

$reg_status = $this->session->userdata('reg_status');
$check_access = $this->Auth_model-
>get_access_users($reg_code, $reg_password, $reg_status);
if (!isset($reg_code) && !isset($reg_password) && !isset($reg_status)){
    $this->session-
>unset_userdata("reg_code","reg_password","reg_status");
    $this->session-
>set_flashdata('alert_error', 'Anda belum Login atau anda meninggalkan halam
an terlalu lama !');
    redirect(base_url().'Frontend/Website/F_Login/Login');
    die();
}
public function index(){
    $reg_code = $this->session->userdata('reg_code');
    $reg_password = $this->session->userdata('reg_password');
    $reg_status = $this->session->userdata('reg_status');
    $register_code = $reg_code;
    $data['get_code'] = $register_code;
    $where = array('register_id'=>$register_code);
    $header['bio_fullname'] = $this->crud_model-
>get_data('tb_borrower_bio',$where)->result();
    $data['title'] = "My Book";
    $data['brd_title_main'] = "List Book";
    $data['brd_title_url'] = site_url('Frontend/Borrower/F_borrower');
    $data['register_code'] = $reg_code;
    $data['info_url'] = site_url('Frontend/Borrower/F_borrower/detail_buku');
    $data['delete_url'] = site_url('Frontend/Borrower/F_borrower/delete_buku');
    $this->load->view('Frontend/website/partial2/metadata');
    $this->load->view('Frontend/website/partial2/header', $header);
    $this->load->view('Frontend/website/partial2/navigation');
    $this->load->view('Frontend/peminjam/dashboard', $data);
    $this->load->view('Frontend/website/partial2/footer');
}
public function approved_book()
{
    $reg_code = $this->session->userdata('reg_code');
    $register_code1 = $this->uri->segment(5);
    $id_invest_book = $this->uri->segment(6);
    $id_buku = $this->uri->segment(7);
}

```

```

$register_code = $reg_code;
$idMax_loan = $this->loan_model->loan_buku();
$noUrut_loan =(int) substr($idMax_loan[0]->maxID,3,9);
$noUrut_loan++;
$newID_Loan="LN".sprintf("%09s",$noUrut_loan);
$data_loan = array(
    'loan_id' => $newID_Loan,
    'register_id' => $register_code,
    'id_invest_book' => $id_invest_book,
    'loan_date' => date("Y-m-d"),
    'status' => "Checking",
);
$insert_loan = $this->crud_model->insert('tb_loan', $data_loan);
$data_invest = array(
    'status' => "PraLoan",
);
$update_invest= $this->crud_model-
>update('tb_invest_book','id_invest_book',$id_invest_book,$data_invest);
$this->session-
>set_flashdata('alert_success', 'Anda telah berhasil menyetujui pinjaman ini
!');
redirect(base_url().'Frontend/Borrower/F_borrower/peminjaman');
die();
}
public function detail_buku()
{
    $reg_code = $this->session->userdata('reg_code');
    $reg_password = $this->session->userdata('reg_password');
    $reg_status = $this->session->userdata('reg_status');
    $register_code1 = $this->uri->segment(5);
    $id_invest_book = $this->uri->segment(6);
    $id_buku = $this->uri->segment(7);
    $register_code = $reg_code;
    $data['get_code'] = $register_code;
    $where = array('register_id'=>$register_code);
    $header['bio_fullname'] = $this->crud_model-
>get_data('tb_borrower_bio',$where)->result();
    $data['title'] = "Peminjaman Buku";
    $data['brd_title_url'] = site_url('Frontend/Borrower/F_borrower/deta
il_book')."/".$id_invest_book;
}

```

```

        $data['brd_title_main'] = "Checking Buku";
        $data['data_register'] = $this->register_model-
>get_register_by_id($register_code1);
        $data['get_code'] = $register_code1;
        $data['id_invest_book'] = $id_invest_book;
        $data['buku_id'] = $id_buku;
        $data['data_code'] = $this->personal_info_model-
>get_personal_info_investment($register_code1);
        $data['data_book'] = $this->personal_info_model-
>get_personal_info_invest_book3($id_buku);
        $data['data_cost'] = $this->personal_info_model-
>get_personal_info_invest_owner2($id_invest_book);
        $data['approved_url'] = site_url('Frontend/Borrower/F_borrower/approved_book');
        $data['unapproved_url'] = site_url('Frontend/Borrower/F_borrower');
        $this->load->view('Frontend/website/partial2/metadata');
        $this->load->view('Frontend/website/partial2/header', $header);
        $this->load->view('Frontend/website/partial2/navigation');
        $this->load->view('Frontend/peminjam/buku_detail', $data);
        $this->load->view('Frontend/website/partial2/footer');
    }
    public function peminjaman(){
        $reg_code = $this->session->userdata('reg_code');
        $reg_password = $this->session->userdata('reg_password');
        $reg_status = $this->session->userdata('reg_status');
        $register_code = $reg_code;
        $data['get_code'] = $register_code;
        $where = array('register_id'=>$register_code);
        $header['bio_fullname'] = $this->crud_model-
>get_data('tb_borrower_bio',$where)->result();
        $data['title'] = "My Book";
        $data['brd_title_main'] = "List Book";
        $data['brd_title_url'] = site_url('Frontend/Borrower/F_borrower');
        $data['register_code'] = $reg_code;
        $data['info_url'] = site_url('Frontend/Borrower/F_borrower/pembayaran');
        $this->load->view('Frontend/website/partial2/metadata');
        $this->load->view('Frontend/website/partial2/header', $header);
        $this->load->view('Frontend/website/partial2/navigation');
        $this->load->view('Frontend/peminjam/peminjaman', $data);
        $this->load->view('Frontend/website/partial2/footer');
    }
}

```

```

    }

    public function pembayaran(){
        $reg_code = $this->session->userdata('reg_code');
        $reg_password = $this->session->userdata('reg_password');
        $reg_status = $this->session->userdata('reg_status');
        $id_invest_book = $this->uri->segment(5);
        $data['id_invest_book'] = $id_invest_book;
        $register_code = $reg_code;
        $where = array('register_id'=>$register_code);
        $data['get_code'] = $register_code;
        $data['data_code'] = $this->personal_info_model-
>get_personal_info_lender($register_code);
        $header['bio_fullname'] = $this->crud_model-
>get_data('tb_borrower_bio',$where)->result();
        $header['akun_owner'] = "active";
        $data['back_url'] = site_url('Frontend/Borrower/F_owner');
        $data['insert_url'] = site_url('Frontend/Borrower/F_borrower/insert_
pembayaran');
        $this->load->view('Frontend/website/partial2/metadata');
        $this->load->view('Frontend/website/partial2/header', $header);
        $this->load->view('Frontend/peminjam/pembayaran', $data);
        $this->load->view('Frontend/website/partial2/navigation');
        $this->load->view('Frontend/website/partial2/footer');

    }

    public function insert_pembayaran(){
        $reg_code = $this->session->userdata('reg_code');
        $register_code = $reg_code;
        $idMax_pembayaran = $this->loan_model->pembayaran_code();
        $noUrut_pembayaran=(int) substr($idMax_pembayaran[0]->maxID,3,9);
        $noUrut_pembayaran++;
        $newID_Pembayaran="C".sprintf("%09s",$noUrut_pembayaran);
        $this->form_validation->set_rules("amount", "", "required");
        if ($this->form_validation->run() == true){
            $amount = $this->input->post('amount');
            $id_invest = $this->input->post('id_invest_book');
            if(!empty($_FILES['id_upload']['name'])){
                $file_upload_name = $this->input->post('bukti');
                $name_file = $file_upload_name."_" .time()."_" .rand(0,1000);
                $config['upload_path'] = './uploads/pembayaran';
                $config['allowed_types'] = 'gif|jpg|png|jpeg';

```

```

        $config['max_size']      = '5120';
        $config['file_name'] = $name_file;
        $this->load->library('upload', $config);
        if(!$this->upload->do_upload('id_upload')){
            $this->session-
>set_flashdata('alert_error', 'Upload foto buku gagal, ukuran file terlalu b
esar atau format file salah !');
            redirect(base_url().'Frontend/Borrower/F_borrower/peminjaman');
            die();
        } else {
            $file_upload = $this->upload->data();
            $data_pembayaran = array(
                'pembayaran_id' => $newID_Pembayaran,
                'register_id' => $register_code,
                'id_invest_book' => $id_invest,
                'amount' => $amount,
                'date' => date('Y-m-d'),
                'bukti' => $file_upload['file_name'],
            );
            $insert_pembayaran = $this->crud_model-
>insert('tb_pembayaran', $data_pembayaran);
            $data_invest = array(
                'status' => "CheckingLoan",
            );
            $update_invest= $this->crud_model-
>update('tb_invest_book','id_invest_book',$id_invest,$data_invest);
        }
        $this->session-
>set_flashdata('alert_success', 'Proses Pembayaran Buku Berhasil, Silahkan m
enunggu proses persetujuan oleh Admin!');
        redirect(base_url().'Frontend/Borrower/F_borrower/peminjaman');
        die();
    }
} else {
    $this->session-
>set_flashdata('alert_error', 'Proses insert data gagal !');
    redirect(base_url().'Frontend/Borrower/F_borrower/peminjaman');
    die();
}
}

```

```

public function pengembalian(){
    $reg_code = $this->session->userdata('reg_code');
    $reg_password = $this->session->userdata('reg_password');
    $reg_status = $this->session->userdata('reg_status');
    $register_code = $reg_code;
    $data['get_code'] = $register_code;
    $where = array('register_id'=>$register_code);
    $header['bio_fullname'] = $this->crud_model-
>get_data('tb_borrower_bio',$where)->result();
    $data['title'] = "My Book";
    $data['brd_title_main'] = "List Book";
    $data['brd_title_url'] = site_url('Frontend/Borrower/F_borrower');
    $data['register_code'] = $reg_code;
    $data['info_url'] = site_url('Frontend/Borrower/F_borrower/detail_pe
ngembalian');
    $this->load->view('Frontend/website/partial2/metadata');
    $this->load->view('Frontend/website/partial2/header', $header);
    $this->load->view('Frontend/website/partial2/navigation');
    $this->load->view('Frontend/peminjam/pengembalian', $data);
    $this->load->view('Frontend/website/partial2/footer');
}
public function approved_pengembalian()
{
    $reg_code = $this->session->userdata('reg_code');
    $register_code1 = $this->uri->segment(5);
    $id_invest_book = $this->uri->segment(6);
    $register_code = $reg_code;
    $data_invest = array(
        'status' => "CheckingReturn",
    );
    $update_invest= $this->crud_model-
>update('tb_invest_book','id_invest_book',$id_invest_book,$data_invest);
    $this->session-
>set_flashdata('alert_success', 'Anda telah berhasil menyetujui pengembalian
ini!');
    redirect(base_url().'Frontend/Borrower/F_borrower/cek_pengembalian');
    die();
}
public function detail_pengembalian()
{

```

```

$reg_code = $this->session->userdata('reg_code');
$reg_password = $this->session->userdata('reg_password');
$reg_status = $this->session->userdata('reg_status');
$register_code1 = $this->uri->segment(5);
$id_invest_book = $this->uri->segment(6);
$loan_id = $this->uri->segment(7);
$register_code = $reg_code;
$data['get_code'] = $register_code;
$where = array('register_id'=>$register_code);
$header['bio_fullname'] = $this->crud_model-
>get_data('tb_borrower_bio',$where)->result();
$data['title'] = "Pengembalian Buku";
$data['brd_title_url'] = site_url('Frontend/Borrower/F_borrower/deta
il_pengembalian')."/".$register_code1."/". $id_invest_book;
$data['brd_title_main'] = "Pengembalian Buku";
$data['data_register'] = $this->register_model-
>get_register_by_id($register_code1);
$data['get_code'] = $register_code1;
$data['id_invest_book'] = $id_invest_book;
$data['data_code'] = $this->personal_info_model-
>get_personal_info_investment($register_code1);
$data['data_book'] = $this->personal_info_model-
>get_personal_info_invest_book($register_code1);
$data['data_cost'] = $this->personal_info_model-
>get_personal_info_invest_owner($register_code1);
$data['data_loan'] = $this->personal_info_model-
>get_personal_info_loan($loan_id);
$data['approved_url'] = site_url('Frontend/Borrower/F_borrower/appro
ved_pengembalian');
$data['unapproved_url'] = site_url('Frontend/Borrower/F_borrower');
$this->load->view('Frontend/website/partial2/metadata');
$this->load->view('Frontend/website/partial2/header', $header);
$this->load->view('Frontend/website/partial2/navigation');
$this->load->view('Frontend/peminjam/pengembalian_detail', $data);
$this->load->view('Frontend/website/partial2/footer');
}

public function cek_pengembalian(){
$reg_code = $this->session->userdata('reg_code');
$reg_password = $this->session->userdata('reg_password');
$reg_status = $this->session->userdata('reg_status');
}

```

```

$register_code = $reg_code;
$data['get_code'] = $register_code;
$where = array('register_id'=>$register_code);
$header['bio_fullname'] = $this->crud_model-
>get_data('tb_borrower_bio',$where)->result();
$data['title'] = "My Book";
$data['brd_title_main'] = "List Book";
$data['brd_title_url'] = site_url('Frontend/Borrower/F_borrower');
$data['register_code'] = $reg_code;
$data['info_url'] = site_url('Frontend/Borrower/F_borrower/upload_pe-
ngembalian');
$this->load->view('Frontend/website/partial2/metadata');
$this->load->view('Frontend/website/partial2/header', $header);
$this->load->view('Frontend/website/partial2/navigation');
$this->load->view('Frontend/peminjam/cek_pengembalian', $data);
$this->load->view('Frontend/website/partial2/footer');
}
public function upload_pengembalian(){
$reg_code = $this->session->userdata('reg_code');
$reg_password = $this->session->userdata('reg_password');
$reg_status = $this->session->userdata('reg_status');
$id_invest_book = $this->uri->segment(6);
$data['id_invest_book'] = $id_invest_book;
$register_code = $reg_code;
$where = array('register_id'=>$register_code);
$data['get_code'] = $register_code;
$data['data_code'] = $this->personal_info_model-
>get_personal_info_lender($register_code);
$header['bio_fullname'] = $this->crud_model-
>get_data('tb_borrower_bio',$where)->result();
$header['akun_owner'] = "active";
$data['back_url'] = site_url('Frontend/Borrower/F_owner');
$data['insert_url'] = site_url('Frontend/Borrower/F_borrower/insert_-
pengembalian_buku');
$this->load->view('Frontend/website/partial2/metadata');
$this->load->view('Frontend/website/partial2/header', $header);
$this->load->view('Frontend/peminjam/upload_pengembalian', $data);
$this->load->view('Frontend/website/partial2/navigation');
$this->load->view('Frontend/website/partial2/footer');
}

```

```

public function insert_pengembalian_buku(){
    $reg_code = $this->session->userdata('reg_code');
    $register_code = $reg_code;
    $idMax_pengembalian = $this->loan_model->pengembalian_code();
    $noUrut_pengembalian=(int) substr($idMax_pengembalian[0]-
>maxID,3,9);
    $noUrut_pengembalian++;
    $newID_pengembalian="R".sprintf("%09s",$noUrut_pengembalian);
    $this->form_validation->set_rules("id_invest_book", "", "required");
    if ($this->form_validation->run() == true){
        $id_invest = $this->input->post('id_invest_book');
        if(!empty($_FILES['id_upload']['name'])){
            $file_upload_name = $this->input->post('bukti');
            $name_file = $file_upload_name."_" .time()."_" .rand(0,1000);
            $config['upload_path'] = './uploads/pengembalian';
            $config['allowed_types'] = 'gif|jpg|png|jpeg';
            $config['max_size']      = '5120';
            $config['file_name']     = $name_file;
            $this->load->library('upload', $config);
            if(!$this->upload->do_upload('id_upload')){
                $this->session-
>set_flashdata('alert_error', 'Upload foto buku gagal, ukuran file terlalu b
esar atau format file salah !');
                redirect(base_url().'Frontend/Borrower/F_borrower/cek_pengembalian');
                die();
            } else {
                $file_upload = $this->upload->data();
                $data_pengembalian = array(
                    'pengembalian_id' => $newID_pengembalian,
                    'register_id' => $register_code,
                    'bukti' => $file_upload['file_name'],
                );
                $insert_pengembalian = $this->crud_model-
>insert('tb_pengembalian_buku', $data_pengembalian);
                $data_invest = array(
                    'status' => "SendReturn",
                );
                $update_invest= $this->crud_model-
>update('tb_invest_book','id_invest_book',$id_invest,$data_invest);
            }
        }
    }
}

```

```

        $this->session-
>set_flashdata('alert_success', 'Proses Pembayaran Buku Berhasil, Silahkan m
enunggu proses persetujuan oleh Admin!');
        redirect(base_url().'Frontend/Borrower/F_borrower/cek_pengembalian');
        die();
    }
} else {
    $this->session-
>set_flashdata('alert_error', 'Proses insert data gagal !');
    redirect(base_url().'Frontend/Borrower/F_borrower/cek_pengembalian');
    die();
}
}

public function received_book(){
    $reg_code = $this->session->userdata('reg_code');
    $reg_password = $this->session->userdata('reg_password');
    $reg_status = $this->session->userdata('reg_status');
    $register_code = $reg_code;
    $data['get_code'] = $register_code;
    $where = array('register_id'=>$register_code);
    $header['bio_fullname'] = $this->crud_model-
>get_data('tb_borrower_bio',$where)->result();
    $data['title'] = "My Book";
    $data['brd_title_main'] = "List Book";
    $data['brd_title_url'] = site_url('Frontend/Borrower/F_borrower');
    $data['register_code'] = $reg_code;
    $data['info_url'] = site_url('Frontend/Borrower/F_borrower/detail_re
ceived_book');
    $this->load->view('Frontend/website/partial2/metadata');
    $this->load->view('Frontend/website/partial2/header', $header);
    $this->load->view('Frontend/website/partial2/navigation');
    $this->load->view('Frontend/peminjam/received_book', $data);
    $this->load->view('Frontend/website/partial2/footer');
}
public function approved_received_book()
{
    $reg_code = $this->session->userdata('reg_code');
    $register_code1 = $this->uri->segment(5);
    $id_invest_book = $this->uri->segment(6);
    $buku_id = $this->uri->segment(7);
}

```

```

$id_loan = $this->uri->segment(8);
$register_code = $reg_code;
$data_invest = array(
    'status' => "ReceivedBook",
);
$update_invest= $this->crud_model-
>update('tb_invest_book','id_invest_book',$id_invest_book,$data_invest);
$this->session-
>set_flashdata('alert_success', 'Anda telah berhasil confirm menerima buku i
ni!');
redirect(base_url().'Frontend/Borrower/F_borrower/received_book');
die();
}
public function detail_received_book()
{
    $reg_code = $this->session->userdata('reg_code');
    $reg_password = $this->session->userdata('reg_password');
    $reg_status = $this->session->userdata('reg_status');
    $register_code1 = $this->uri->segment(5);
    $id_invest_book = $this->uri->segment(6);
    $buku_id = $this->uri->segment(7);
    $id_loan = $this->uri->segment(8);
    $register_code = $reg_code;
    $data['get_code'] = $register_code;
    $where = array('register_id'=>$register_code);
    $header['bio_fullname'] = $this->crud_model-
>get_data('tb_borrower_bio',$where)->result();
    $data['title'] = "Pengembalian Buku";
    $data['brd_title_url'] = site_url('Frontend/Borrower/F_borrower/deta
il_received_book')."/".$register_code1."/". $id_invest_book."/". $buku_id."/".
$id_loan;
    $data['brd_title_main'] = "Pengembalian Buku";
    $data['data_register'] = $this->register_model-
>get_register_by_id($register_code1);
    $data['get_code'] = $register_code1;
    $data['id_invest_book'] = $id_invest_book;
    $data['data_code'] = $this->personal_info_model-
>get_personal_info_investment($register_code1);
    $data['data_book'] = $this->personal_info_model-
>get_personal_info_book($buku_id);
}

```

```

        $data['data_cost'] = $this->personal_info_model-
>get_personal_info_invest_owner($register_code1);
        $data['data_loan'] = $this->personal_info_model-
>get_personal_info_loan($id_loan);
        $data['approved_url'] = site_url('Frontend/Borrower/F_borrower/appro-
ved_received_book');
        $data['unapproved_url'] = site_url('Frontend/Borrower/F_borrower');
        $this->load->view('Frontend/website/partial2/metadata');
        $this->load->view('Frontend/website/partial2/header', $header);
        $this->load->view('Frontend/website/partial2/navigation');
        $this->load->view('Frontend/peminjam/received_book_detail', $data);
        $this->load->view('Frontend/website/partial2/footer');

    }
    public function search()
    {
        $this->form_validation-
>set_rules('search_product', 'Cari Barang', 'required');
        if ($this->form_validation->run() == FALSE){
            $reg_code = $this->session->userdata('reg_code');
            $reg_password = $this->session->userdata('reg_password');
            $reg_status = $this->session->userdata('reg_status');
            $register_code = $reg_code;
            $data['get_code'] = $register_code;
            $where = array('register_id'=>$register_code);
            $header['bio_fullname'] = $this->crud_model-
>get_data('tb_borrower_bio',$where)->result();
            $data['title'] = "My Book";
            $data['brd_title_main'] = "List Book";
            $data['brd_title_url'] = site_url('Frontend/Borrower/F_borrower');
            $data['register_code'] = $reg_code;
            $data['info_url'] = site_url('Frontend/Borrower/F_borrower/detail_buku');
            $data['delete_url'] = site_url('Frontend/Borrower/F_borrower/delete_buku');
            $this->load->view('Frontend/website/partial2/metadata');
            $this->load-
>view('Frontend/website/partial2/header', $header);
            $this->load->view('Frontend/website/partial2/navigation');
            $this->load->view('Frontend/peminjam/dashboard', $data);
            $this->load->view('Frontend/website/partial2/footer');
        }else{
            $key = $this->input->post('search_product');
        }
    }
}

```

```

        $data['bukus'] = $this->search_model->get_buku();
        $data['products'] = $this->search_model-
>product_search($key);
        $this->load->view('Frontend/peminjam/dashboard1',$data);
    }
}
}
?>

```

9.4.2 Membuat View peminjaman.php

Berikut adalah syntax dari file peminjaman.php

```

<div class="content-wrapper">
    <!-- Content Header (Page header) -->
    <section class="content-header">
        <h1>Data Peminjaman</h1>
        <ol class="breadcrumb">
            <li class="active"><a href=<?php echo site_url('Frontend/Owner/F_ow-
ner/invest_buku') ?>"><i class="fa fa-
dashboard"></i>Data Peminjaman</a></li>
        </ol>
    </section>
    <!-- Main content -->
    <section class="content">
        <div class="row">
            <div class="col-xs-12">
                <?php if ($this->session->flashdata('alert_success')) { ?>
                    <div class="alert alert-success alert-dismissible">
                        <button type="button" class="close" data-
dismiss="alert" aria-hidden="true">&times;</button>
                        <p><i class="icon fa fa-check"></i>
                            <?php echo $this->session-
>flashdata('alert_success'); ?></p>
                    </div>
                <?php } ?>
                <?php if ($this->session->flashdata('alert_error')) { ?>
                    <div class="alert alert-danger alert-dismissible">
                        <button type="button" class="close" data-
dismiss="alert" aria-hidden="true">&times;</button>
                        <p><i class="icon fa fa-ban"></i>

```

```

        <?php echo $this->session-
>flashdata('alert_error'); ?></p>
        </div>
        <?php }
    ?>
<div class="box">
    <div class="box-header with-border">
    </div>
    <!-- /.box-header -->
    <div class="box-body">
        <div class="row">
            <?php
                $this->load->model("crud_model");
                $data_setting = $this->crud_model->get_setting();
                $limit      = $data_setting[0]->setting_pagination;
                $page = isset($_REQUEST['page']) ? $_REQUEST['page'] : null;
                if(empty($page)){
                    $position  = 0;
                    $page = 1;
                } else {
                    $position = ($page-1) * $limit;
                }
                $this->load->model('Front_data/invest_book');
                $data_users = $this->invest_book-
>data_buku_owner('PraLoan')->result();
                $no = $position+1;
                foreach ($data_users as $users_entry){
                    $register_code = $users_entry->register_id;
                    $data_book = $this->personal_info_model-
>get_personal_info_invest_book($register_code);
                ?>
                <div class="col-sm-6 col-md-4">
                    <div class="thumbnail">
                        
                        <div class="caption" style="text-align: center;">
                            <h3>Judul Buku : <?php echo @$users_entry-
>judul_buku; ?></h3>
                            <h4>Pengarang Buku : <?php echo @$users_entry-
>pengarang; ?></h4>

```

```

        <h4>Sinopsis : <?php echo @$users_entry-
>sinopsis; ?></h4>
        <h5>Tahun Terbit : <?php echo @$users_entry-
>tahun; ?></h5>
        <p><a href="<?php echo $info_url; ?>/<?php echo $users_e
ntry->id_invest_book; ?>" id="info" class="btn btn-info btn-
sm btnwtdt">Bayar</a></p>
        </div> </div> </div>
        <?php
    } ?>
</div>
</section>
</div>
<script type="text/javascript">
    function delete_data(buku_id)
    {
        bootbox.confirm({
            title: "Deleting Data?",
            message: "Do you want to delete this data?",
            buttons: {
                cancel: {
                    label: '<i class="fa fa-times"></i> Cancel',
                    className: 'btn-default'
                },
                confirm: {
                    label: '<i class="fa fa-check"></i> Delete',
                    className: 'btn-danger'
                }
            },
            callback: function (result) {
                if (result == true){
                    window.location = "<?php echo $delete_url; ?>/" + buku_id;
                } else {
                    event.preventDefault();
                }
            });
        });
    }
</script>

```

9.4.3 Membuat View pengembalian.php

Berikut adalah syntax dari file pengembalian.php

```
<div class="content-wrapper">
    <!-- Content Header (Page header) -->
    <section class="content-header">
        <h1>Data Buku untuk di Kembalikan</h1>
        <ol class="breadcrumb">
            <li class="active"><a href="php echo site_url('Frontend/Borrower/F_borrower/pengembalian') ??"&gt;&lt;i class="fa fa-dashboard"&gt;&lt;/i&gt;Data Pengembalian&lt;/a&gt;&lt;/li&gt;
        &lt;/ol&gt;
    &lt;/section&gt;
    &lt;!-- Main content --&gt;
    &lt;section class="content"&gt;
        &lt;div class="row"&gt;
            &lt;div class="col-xs-12"&gt;
                &lt;?php if ($this-&gt;session-&gt;flashdata('alert_success')) { ?&gt;
                    &lt;div class="alert alert-success alert-dismissible"&gt;
                        &lt;button type="button" class="close" data-
dismiss="alert" aria-hidden="true"&gt;&amp;times;&lt;/button&gt;
                        &lt;p&gt;&lt;i class="icon fa fa-check"&gt;&lt;/i&gt;
                            &lt;?php echo $this-&gt;session-
&gt;flashdata('alert_success'); ?&gt;&lt;/p&gt;
                    &lt;/div&gt;
                &lt;?php } ?&gt;
                &lt;?php if ($this-&gt;session-&gt;flashdata('alert_error')) { ?&gt;
                    &lt;div class="alert alert-danger alert-dismissible"&gt;
                        &lt;button type="button" class="close" data-
dismiss="alert" aria-hidden="true"&gt;&amp;times;&lt;/button&gt;
                        &lt;p&gt;&lt;i class="icon fa fa-ban"&gt;&lt;/i&gt;
                            &lt;?php echo $this-&gt;session-
&gt;flashdata('alert_error'); ?&gt;&lt;/p&gt;
                    &lt;/div&gt;
                &lt;?php } ?&gt;
            &lt;div class="box"&gt;
                &lt;div class="box-header with-border"&gt;
                &lt;/div&gt;
            &lt;!-- /.box-header --&gt;</pre
```

```

<div class="box-body">
    <div class="row">
        <?php
            $this->load->model("crud_model");
            $data_setting = $this->crud_model->get_setting();
            $limit      = $data_setting[0]->setting_pagination;
            $page = isset($_REQUEST['page']) ? $_REQUEST['page'] : null;
            if(empty($page)){
                $position  = 0;
                $page = 1;
            } else {
                $position = ($page-1) * $limit;
            }
            $this->load->model('Front_data/invest_book');
            $data_users = $this->invest_book-
>get_book_loan_list2(@$limit, @$position, 'ReceivedBook', @$id_invest_book,
@$register_code_page, @$owner_name)->result();
            $no = $position+1;
            foreach ($data_users as $users_entry){
                $register_code = $users_entry->register_id;
                $data_book = $this->personal_info_model-
>get_personal_info_invest_owner3($users_entry->id_invest_book);
            ?>
                <div class="col-sm-6 col-md-4">
                    <div class="thumbnail">
                        
                        <div class="caption" style="text-align: center;">
                            <h3>Judul Buku : <?php echo @$data_book[0]-
>judul_buku; ?></h3>
                            <h4>Pengarang Buku : <?php echo @$data_book[0]-
>pengarang; ?></h4>
                            <h4>Sinopsis : <?php echo @$data_book[0]-
>sinopsis; ?></h4>
                            <h5>Tahun Terbit : <?php echo @$data_book[0]-
>tahun; ?></h5>
                            <p><a href="<?php echo $info_url; ?>/<?php echo $data_bo
ok[0]->register_id; ?>/<?php echo $users_entry-
>id_invest_book; ?>/<?php echo $users_entry-
>loan_id; ?>" id="info" class="btn btn-info btn-sm btnwrdt">Detail</a></p>

```

```

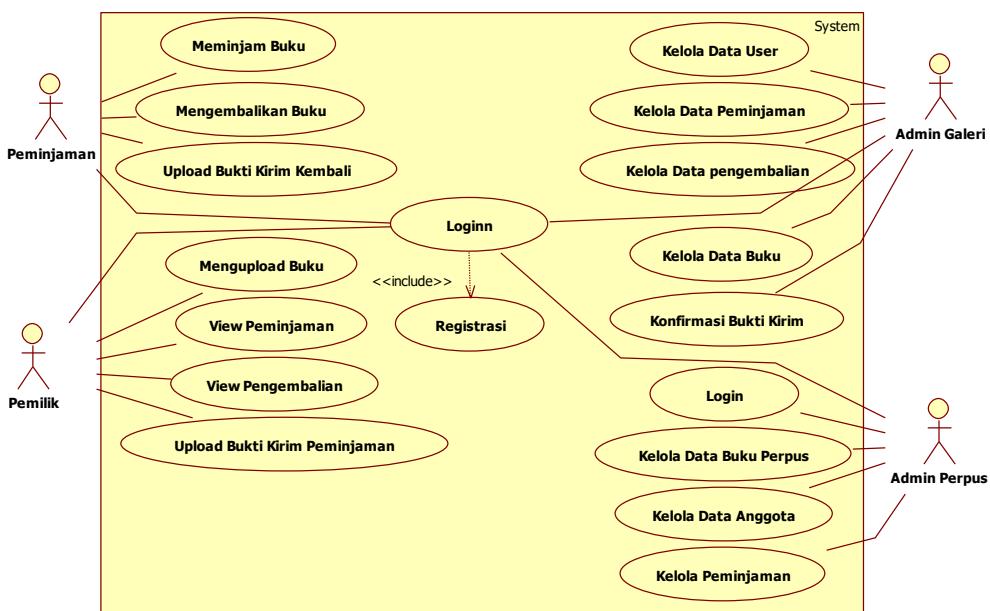
        </div></div></div>
    <?php
    } ?>
</div>
</section>
</div>
<script type="text/javascript">
function delete_data(buku_id)
{
    bootbox.confirm({
        title: "Deleting Data?",
        message: "Do you want to delete this data?",
        buttons: {
            cancel: {
                label: '<i class="fa fa-times"></i> Cancel',
                className: 'btn-default'
            },
            confirm: {
                label: '<i class="fa fa-check"></i> Delete',
                className: 'btn-danger'
            }
        },
        callback: function (result) {
            if (result == true){
                window.location = "<?php echo $delete_url; ?>/" + buku_id;
            } else {
                event.preventDefault();
            }
        }
    });
}
</script>

```

BAGIAN 10

PENGUJIAN DAN IMPLEMENTASI SISTEM YANG TELAH DIBANGUN

10.1 Perancangan Menggunakan Proses *Use Case Diagram*



Gambar 10.1 Use Case Diagram

Gamber 10.1 diatas dijelaskan *use case* masing-masing *actor*, yaitu:

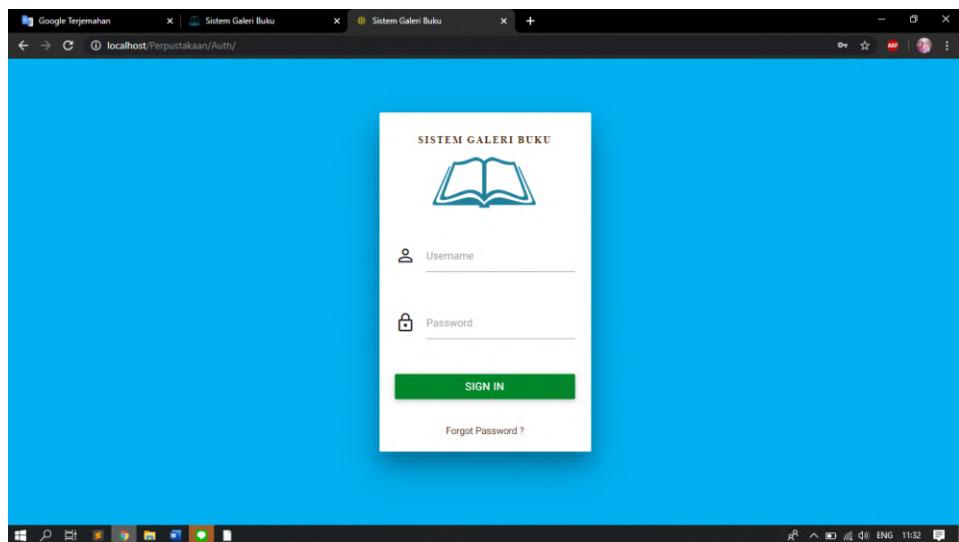
- Aktor Peminjam dapat meminjam buku, mengembalikan buku, *upload* bukti kirim kembali, dan melakukan registrasi dan *login* terlebih dahulu.
- Aktor Pemilik terlebih dahulu melakukan registrasi dan *login*. Pemilik dapat melakukan *upload* buku, melakukan *view* peminjaman artinya melihat siapa saja yang melakukan

peminjaman pada buku yang telah dia *investikan*, kemudian dia dapat mengupload bukti kirim peminjaman.

- Aktor Admin Perpus terlebih dahulu *login* kedalam sistem untuk dapat melakukan kelola data buku perpus, kelola data peminjaman dan kelola data anggota.
- Aktor Admin Galeri terlebih dahulu melakukan *login* kedalam sistem untuk dapat melakukan kelola data *user*, kelola data peminjaman, kelola data pengembalian, kelola data buku, dan konfirmasi bukti kirim.

10.2 Implementasi Sistem

10.2.1 Tampilan Halaman *Login* Admin Pada Galeri Buku



Gambar 10.2 Tampilan Halaman *Login* Admin

Task Usability Testing

No

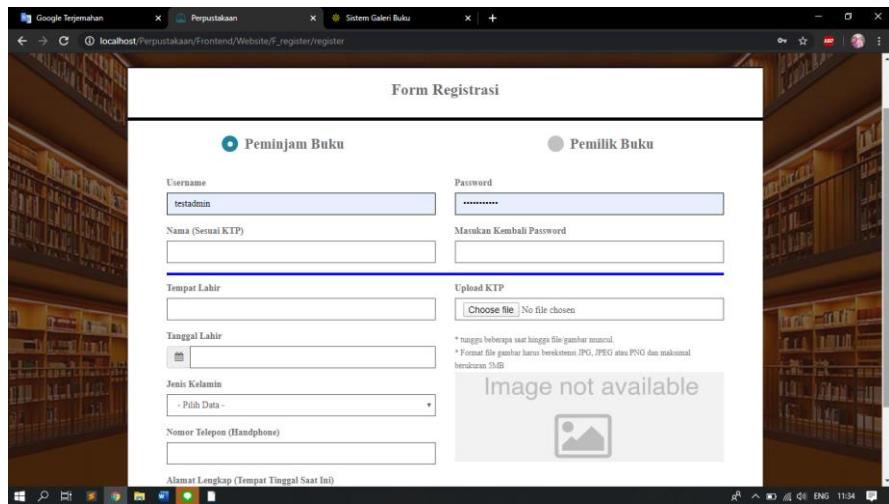
Task/Tugas

- 1 Lakukan *login* sebagai *user* Admin dalam *web* aplikasi ini

Penjelasan Dari *task* adalah:

Task 1. Lakukan *login* sebagai Admin dalam *web* aplikasi ini. *Task* ini meminta *user* Admin untuk melakukan *login*, diawali pada halaman awal *web* aplikasi dengan mengisi *form login*, yaitu *username* dan *password*, dan kemudian mengklik *Button login*.

10.2.2 Tampilan *Form Registrasi User* Pada Galeri Buku



Gambar 10.3 Tampilan Halaman Registrasi

Task Usability Testing

No	Task/Tugas
1	Lakukan registrasi sebagai peminjam
2	Lakukan registrasi sebagai pemilik

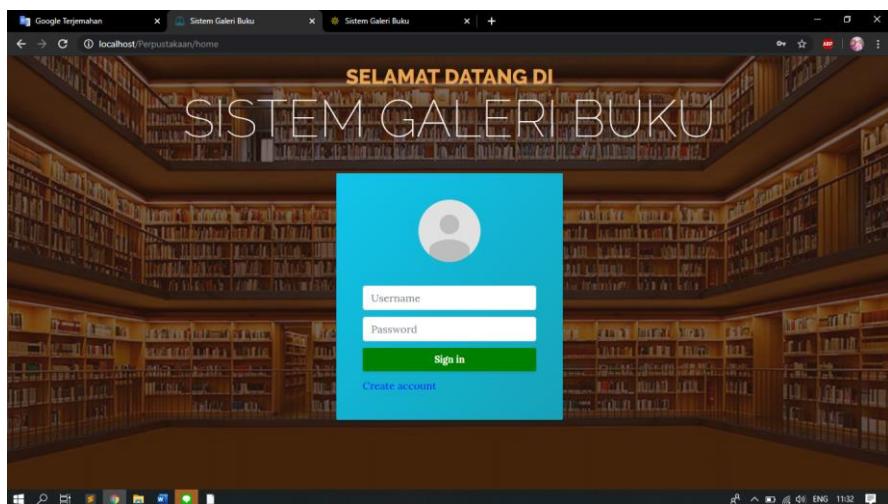
Penjelasan Dari *task* adalah:

Task 1. Lakukan registrasi sebagai peminjam dalam aplikasi ini untuk mendapatkan akun agar bisa melakukan peminjaman. *Task* ini meminta *user* untuk melakukan registrasi dengan meng-*input*-kan

data-data nya yang terdiri dari *username*, password, nama, tanggal lahir, jenis kelamin, nomor telepon, alamat lengkap serta meng-*upload* foto ktp/kts sebagai identitas peminjam. Kemudian *user* mengklik *Button* daftar. Jika data valid dan terisi dengan benar maka *user* berhasil melakukan proses registrasi. Jika data tidak valid maka *user* bisa mengisi kembali data-data dengan benar.

Task 2. Lakukan registrasi sebagai pemilik dalam aplikasi ini untuk mendapatkan akun. Task ini meminta *user* untuk melakukan registrasi dengan meng-*input*-kan data-data nya yang terdiri dari *username*, password, nama, tanggal lahir, jenis kelamin, nomor telepon, alamat lengkap serta meng-*upload* foto ktp/kts sebagai identitas pemilik. Kemudian *user* mengklik *Button* daftar. Jika data valid dan terisi dengan benar maka *user* berhasil melakukan proses registrasi. Jika data tidak valid maka *user* bisa mengisi kembali data-data dengan benar.

10.2.3 Tampilan *Form Login User* Pada Galeri Buku



Gambar 10.4 Tampilan Halaman Login User

Task Usability Testing

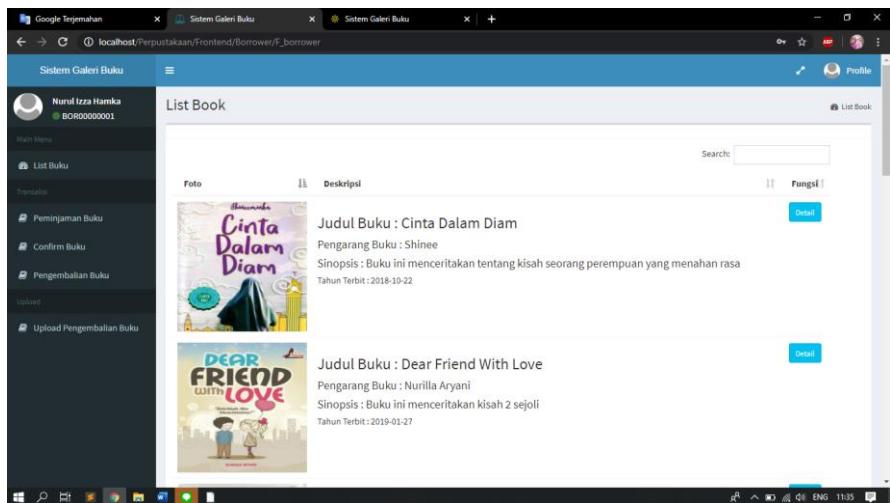
No	<i>Task/Tugas</i>
1	Lakukan <i>login</i> sebagai <i>user</i> peminjam atau pemilik dalam <i>web</i> aplikasi ini

Penjelasan Dari *task* adalah:

Task 1. Lakukan *login* sebagai peminjam atau pemilik dalam *web* aplikasi ini. *Task* ini meminta *user* peminjam atau pemilik untuk melakukan *login*, diawali pada halaman awal *web* aplikasi dengan mengisi *form login*, yaitu *username* dan *password*, dan kemudian mengklik *Button login*.

10.2.4 Tampilan *Form* Peminjaman Admin Pada Galeri Buku

a. Tampilan Halaman Galeri Buku *Page List Buku*



Gambar 10.5 Tampilan Halaman List Buku

Task Usability Testing

No	<i>Task/Tugas</i>
----	-------------------

- 1 Peminjam memilih buku yang akan dipinjam

Penjelasan Dari *task* adalah:

Task 1. Peminjam memilih buku yang akan dipinjam. Sistem akan menampilkan buku-buku yang bisa dipinjam dengan mencantumkan cover atau gambar buku, penulis, sinopsis dan tahun terbit, sehingga peminjam lebih mudah dalam mencari buku yang ingin dipinjamnya, selain itu sistem juga bisa melakukan pencarian buku, *user* hanya tinggal mengetikkan judul buku, maka otomatis sistem akan menampilkan judul buku sesuai yang dicari oleh *user*. Jika *user* sudah memilih buku, *user* hanya tinggal mengklik *Button* detail.

b. Tampilan Halaman Galeri Buku *Page Detail Peminjaman*

The screenshot shows a web-based library system interface. On the left, a sidebar menu includes 'Main Menu', 'List Buku', 'Transaksi', 'Peminjaman Buku' (selected), 'Confirm Buku', 'Pengembalian Buku', 'Upload', and 'Upload Pengembalian Buku'. The main content area is titled 'Peminjaman Buku' and shows the details of a user named 'Aulyardha Anindita'. The form fields include:

- Nama Lengkap: Aulyardha Anindita
- No. Hp: 082345123865
- Tempat Lahir: Watampone
- Alamat: Jl. Sarimanah 2
- Tanggal Lahir: 23/09/1999
- Jenis Kelamin: Perempuan
- Kode Pos: 40151

At the bottom of the form, there is a section labeled 'Data Peminjaman' with a 'Tanggal Peminjaman' field. Two buttons are visible at the top right: a green 'Ajukan Peminjaman' button and a red 'Cancel' button. The browser address bar shows the URL: localhost/Perpuslatakan/Frontend/Borrower/detail_buku/LEN00000001/00000002/8000000002.

Gambar 10.6 Tampilan Halaman Detail Peminjaman

Task Usability Testing

No

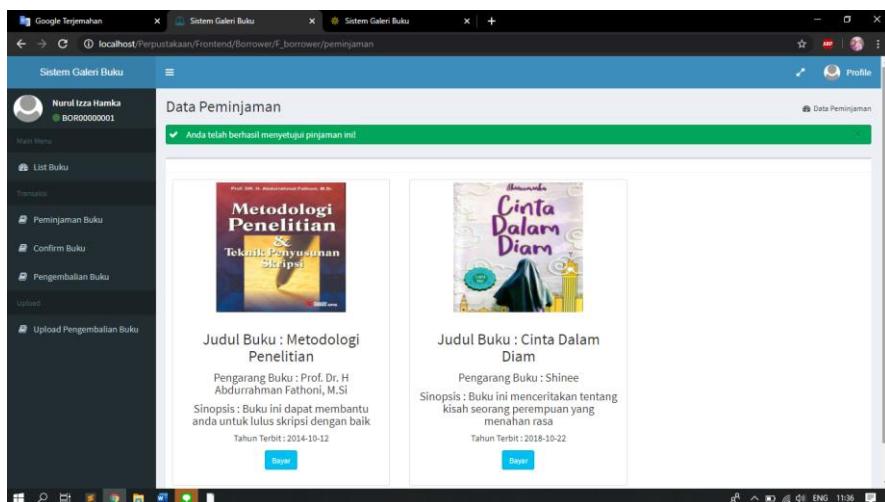
Task/Tugas

- 1 Peminjam melihat detail buku yang akan dipinjam

Penjelasan Dari *task* adalah:

Task 1. Peminjam melihat detail buku yang akan dipinjam. Pada *Task* ini sistem akan menampilkan data pemilik, data peminjaman yang terdiri dari tanggal peminjaman dan tanggal pengembalian buku, data buku yang terdiri dari jumlah yang harus dibayarkan oleh peminjam. Jika peminjam menyetujui data-data tersebut dan berniat untuk melakukan peminjaman, peminjam hanya tinggal mengklik *Button* Ajukan Peminjaman. Jika peminjam tidak menyetujui, peminjam tinggal mengklik *Button* Cancel.

c. Tampilan Halaman Galei Buku Page Peminjaman



Gambar 10.7 Tampilan Halaman Peminjaman

Task Usability Testing

No	Task/Tugas
1	Peminjam melihat data peminjaman.

Task 1. Peminjam melihat data peminjaman. Pada *Task* ini sistem akan menampilkan data peminjaman yang telah dilakukan oleh

peminjam. Data peminjaman yang ditampilkan adalah berupa buku-buku yang akan dipinjam dan menunggu untuk proses pembayaran. Jika peminjam ingin melakukan proses atau mengkonfirmasi pembayaran yang telah dilakukan, peminjam mengklik *Button Bayar*.

d. Tampilan Halaman Galei Buku Page Pembayaran

The screenshot shows a web application window titled 'Proses Input Pembayaran'. On the left, there is a sidebar with a user profile ('Nurul Izza Hamka') and several menu items: Main Menu, List Buku, Transaksi, Peminjaman Buku, Confirm Buku, Pengembalian Buku, Upload, and Upload Pengembalian Buku. The main content area has three input fields: 'Total Pembayaran (Bersua dengan nominal transfer)' containing '250.000', 'ID Upload Buku' containing 'I000000002', and an 'Upload Bukti Foto' field which displays 'Image not available' and a placeholder icon. Below these fields is a 'Submit' button.

Gambar 10.8 Tampilan Halaman Pembayaran

Task Usability Testing

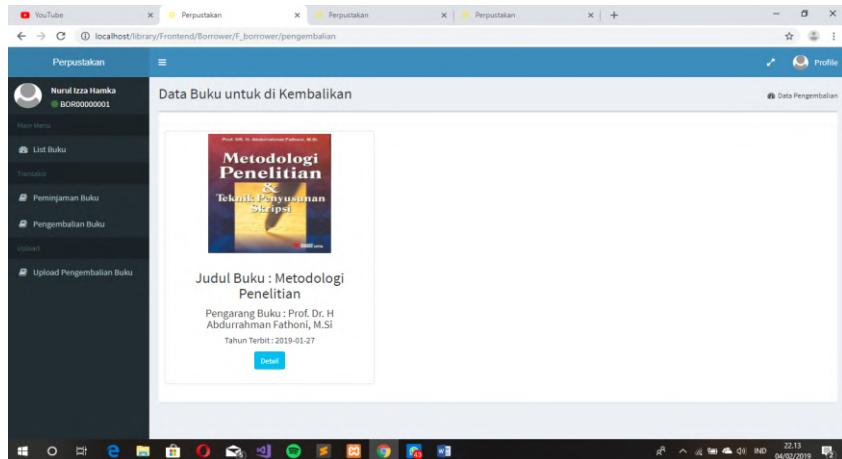
No	Task/Tugas
1	Peminjam mengisi data pembayaran

Penjelasan dari *task* adalah:

Task 1. Peminjam mengisi data pembayaran. Pada *Task* ini sistem akan menampilkan *form input-an* data pembayaran. *User* mengisi data-data pembayaran seperti jumlah yang telah dibayarkan, dan meng-*upload* bukti pembayaran. Setelah itu, *user* mengklik *Button submit*.

10.2.5 Tampilan Form Pengembalian Pada Galeri Buku

a. Tampilan Galeri Buku Page Pengembalian



Gambar 10.9 Tampilan Halaman Pengembalian

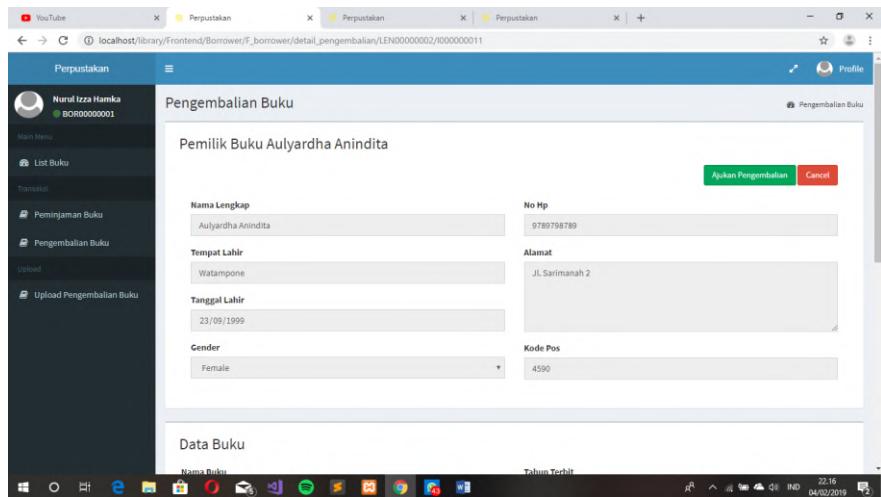
Task Usability Testing

No	Task/Tugas
1	Peminjam melihat data pengembalian

Penjelasan dari *task* adalah:

Task 1. Peminjam melihat data pengembalian. Pada *Task* ini sistem akan menampilkan data pengembalian. Data pengembalian disini berisi data buku yang telah dipinjam. Jika peminjam ingin melakukan pengembalian buku, peminjam hanya tinggal mengklik *Button Detail* untuk melihat data-data pengembalian.

b. Tampilan Galeri Buku Page Detail Pengembalian



Gambar 10.10 Tampilan Halaman Detail Pengembalian

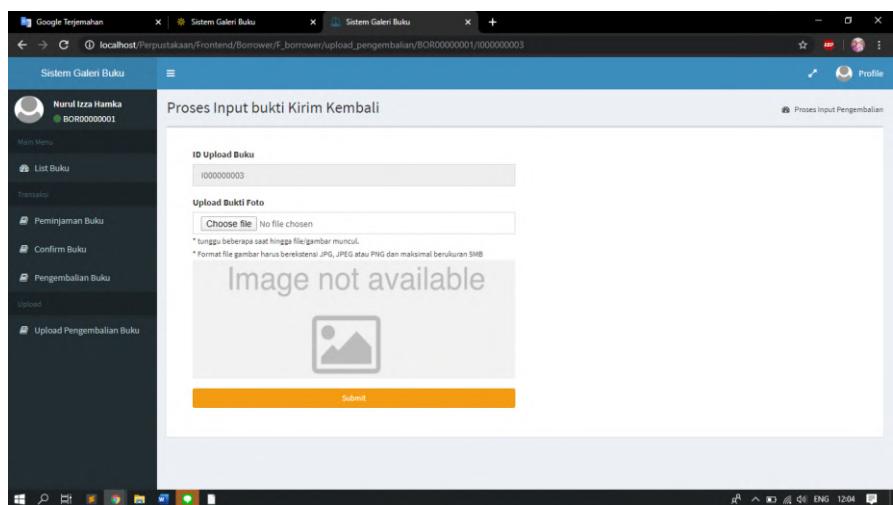
Task Usability Testing

No	Task/Tugas
1	Peminjam melihat detail data pengembalian

Penjelasan dari *task* adalah:

Task 1. Peminjam melihat detail data pengembalian. Pada *Task* ini sistem akan menampilkan data pemilik, data peminjaman yang terdiri dari tanggal peminjaman dan tanggal pengembalian buku, data buku yang terdiri dari jumlah yang harus dibayarkan oleh peminjam. Jika peminjam ingin melakukan pengembalian buku, peminjam tinggal mengklik *Button* Ajukan pengembalian. Jika peminjam belum ingin melakukan pengembalian buku, peminjam tinggal mengklik *Button* Cancel.

10.2.6 Tampilan Form Upload Bukti Kirim Kembali Pada Galeri Buku



Gambar 10.11 Tampilan Halaman Upload Bukti Kirim Kembali
Task Usability Testing

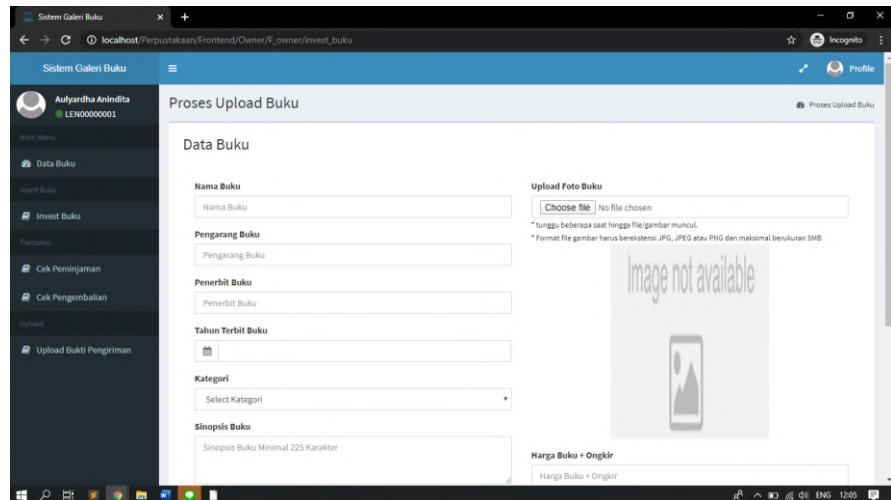
No	Task/Tugas
----	------------

1	Peminjam meng-upload data bukti kirim kembali
---	---

Penjelasan dari task adalah:

Task 1. Peminjam mengisi data bukti kirim kembali. Pada Task ini sistem akan menampilkan *form input*-an bukti kirim kembali. User meng-upload bukti kirim kembali. Jika foto atau bukti sudah berhasil ter-upload, user tinggal mengklik *Submit* untuk menyimpan data yang sudah di-upload.

10.2.7 Tampilan Form Upload Buku Pada Sistem Galeri Buku



Gambar 10.12 Tampilan Halaman Upload Buku

Task Usability Testing

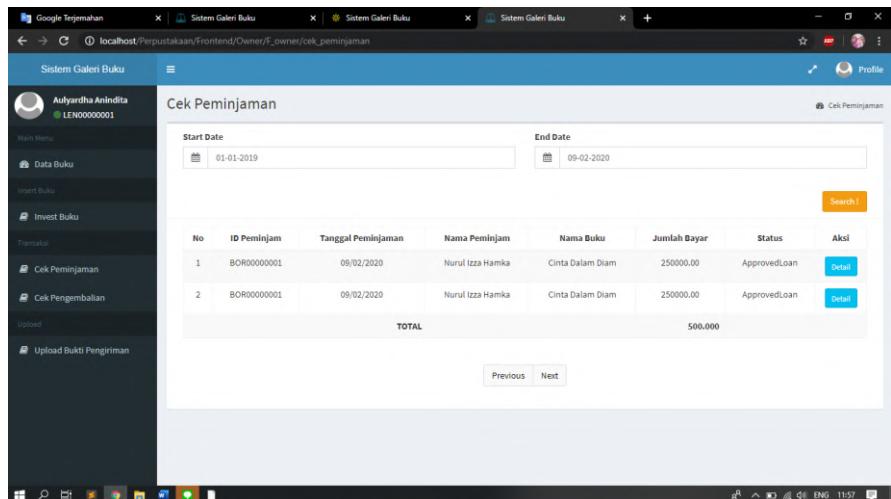
No	Task/Tugas
1	Pemilik meng-upload data buku

Penjelasan dari task adalah:

Task 1. Pemilik meng-upload data buku. Pada Task ini sistem akan menampilkan *form input-an* data buku. *User* mengisi *form input-an* tersebut yang terdiri dari nama buku, pengarang buku, pemilik buku, tahun terbit buku, kategori, sinopsis buku, dan *upload* foto buku. Untuk meng-upload foto buku, *user* mengklik *Button choose file*, lalu *user* memilih foto buku yang akan di-upload, tunggu sampai foto buku berhasil muncul di sistem. Setelah *user* mengisi *form input-an* data buku, *user* mengklik *Button submit* untuk menyimpan data buku yang sudah di-input-kan.

10.2.8 Tampilan Form View Peminjaman Pada Sistem Galeri Buku

a. Tampilan Galeri Buku Page Cek Peminjaman



Gambar 10.13 Tampilan Halaman Cek Peminjaman

Task Usability Testing

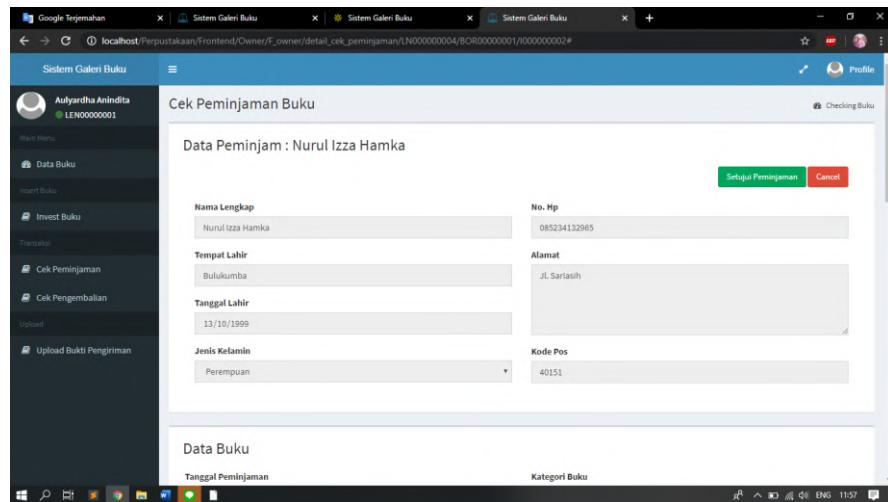
No	Task/Tugas
----	------------

1	Pemilik mengecek data peminjaman
---	----------------------------------

Penjelasan dari task adalah:

Task 1. Pemilik mengecek data peminjaman. Pada Task ini sistem akan menampilkan data peminjaman yang masuk. Data peminjaman yang dimaksud adalah data buku yang dimiliki oleh pemilik akan dipinjam oleh peminjam. Disini pemilik bisa mengetahui buku yang dia miliki akan dipinjam oleh siapa, selain bisa mengetahui dipinjam oleh siapa, pemilik juga bisa melihat tanggal peminjaman, id peminjam, dan status peminjaman. Untuk melihat lebih detail data peminjaman tersebut, pemilik mengklik Button Detail.

b. Tampilan Galeri Buku *Page Setujui Peminjaman*



Gambar 10.14 Tampilan Halaman Setujui Peminjaman

Task Usability Testing

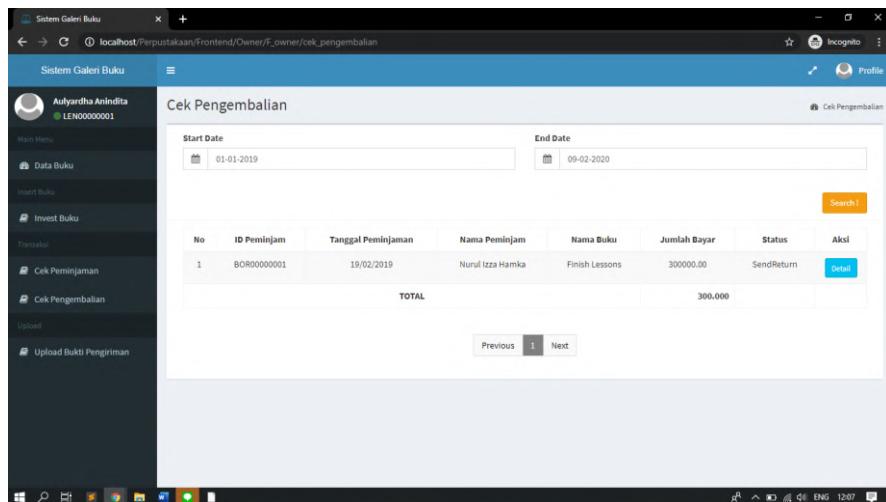
No	Task/Tugas
1	Pemilik menyetujui data peminjaman

Penjelasan dari *task* adalah:

Task 1. Pemilik menyetujui data peminjaman. Pada *Task* ini sistem akan menampilkan detail data peminjaman yang masuk. Data peminjaman disini akan menampilkan data peminjam yang terdiri dari nama peminjam, alamat peminjam, no hp, jenis kelamin, tempat dan tanggal lahir, serta kode pos peminjam. Selain data peminjam, sistem akan menampilkan data buku yang terdiri dari nama buku, kategori buku, penulis dan penerbit buku, jumlah yang telah dibayarkan oleh peminjam serta bukti transfer oleh peminjam. Jika pemilik menyetujui data-data tersebut, pemilik mengklik *Button* Setujui Peminjaman. Jika pemilik tidak menyetujui, pemilik mengklik *Button* Cancel.

10.2.9 Tampilan Form View Pengembalian Pada Sistem Galeri Buku

a. Tampilan Galeri Buku Page Cek Pengembalian



Gambar 10.15 Tampilan Halaman Cek Peminjaman

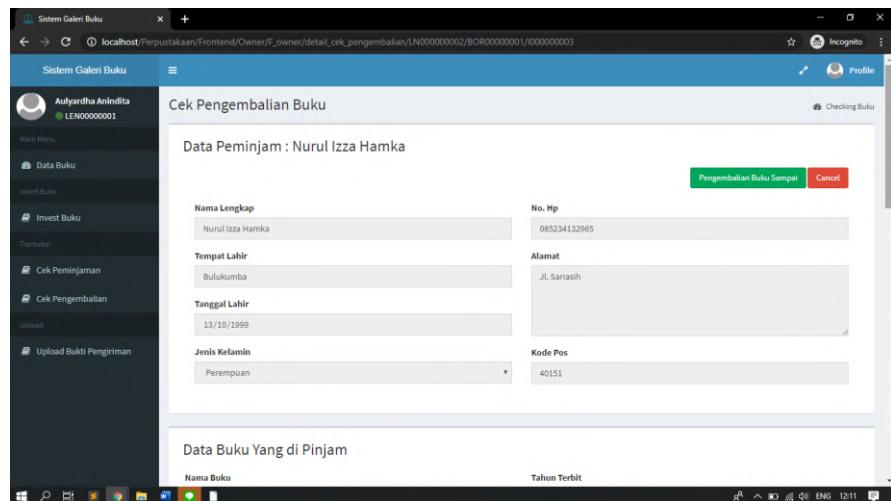
Task Usability Testing

No	Task/Tugas
1	Pemilik mengecek data pengembalian

Penjelasan dari task adalah:

Task 1. Pemilik mengecek data pengembalian. Pada Task ini sistem akan menampilkan data pengembalian yang masuk. Data pengembalian yang dimaksud adalah data peminjaman yang telah selesai dan peminjam akan mengembalikan buku . Disini pemilik bisa mengetahui bahwa buku telah selesai dipinjam dan akan dikembalikan, jadi pemilik bisa siap untuk menerima bukunya kembali. Untuk melihat detail pengembaliannya, pemilik bisa mengklik Button Detail.

b. Tampilan Galeri Buku *Page Konfirmasi Buku Sampai*



*Gambar 10.16 Tampilan Halaman Konfirmasi Buku Sampai
Task Usability Testing*

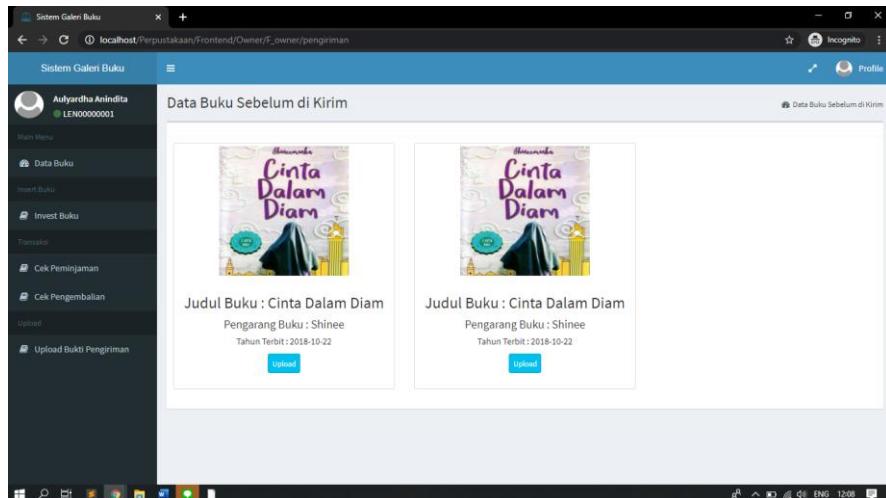
No	Task/Tugas
1	Pemilik mengkonfirmasi buku sampai

Penjelasan dari *task* adalah:

Task 1. Pemilik mengkonfirmasi buku sampai. Pada *Task* ini sistem akan menampilkan data pengembalian yang telah selesai. Jika buku telah sampai ke tangan pemilik, pemilik bisa mengkonfirmasi bahwa buku telah sampai dengan mengklik *Button Pengembalian Buku Sampai*. Jika buku belum sampai ditangan pemilik, pemilik bisa mengklik *Button Cancel*.

10.2.10 Tampilan *Form Upload* Bukti Kirim Peminjaman Pada Sistem Galeri Buku

a. Tampilan Galeri Buku *Page Data Buku Sebelum Pengiriman*



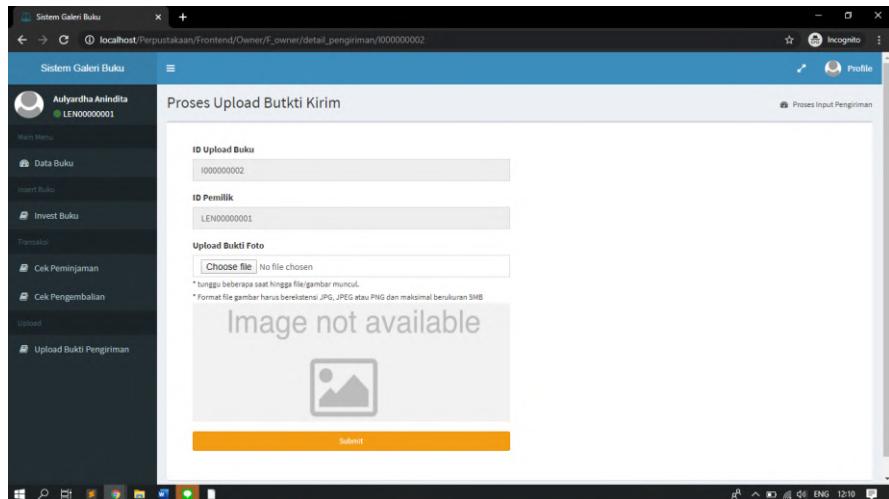
Gambar 10.17 Tampilan Halaman Data Buku Sebelum Pengiriman Task Usability Testing

No	Task/Tugas
1	Pemilik melihat data buku sebelum pengiriman

Penjelasan dari *task* adalah:

Task 1. Pemilik melihat data sebelum pengiriman. Pada *Task* ini sistem akan menampilkan data buku sebelum pengiriman. Data buku yang dimaksud disini adalah data buku yang akan dikirimkan oleh pemilik ke peminjam. Disini pemilik bisa melihat buku-buku apa saja yang belum dikirim, tapi sudah disetujui olehnya. Dengan adanya menu ini, pemilik bisa mengetahui bahwa buku yang dia punya sudah dikirim apa tidak ke peminjam. Pemilik bisa melanjutkan proses pengiriman dengan mengklik *Button Upload* guna meng-upload bukti pengiriman, jika buku telah dikirim ke peminjam.

b. Tampilan Galeri Buku *Page Upload* Bukti Kirim Peminjaman



Gambar 10.18 Tampilan Halaman *Upload* *Bukti Kirim* *Peminjaman*
Task Usability Testing

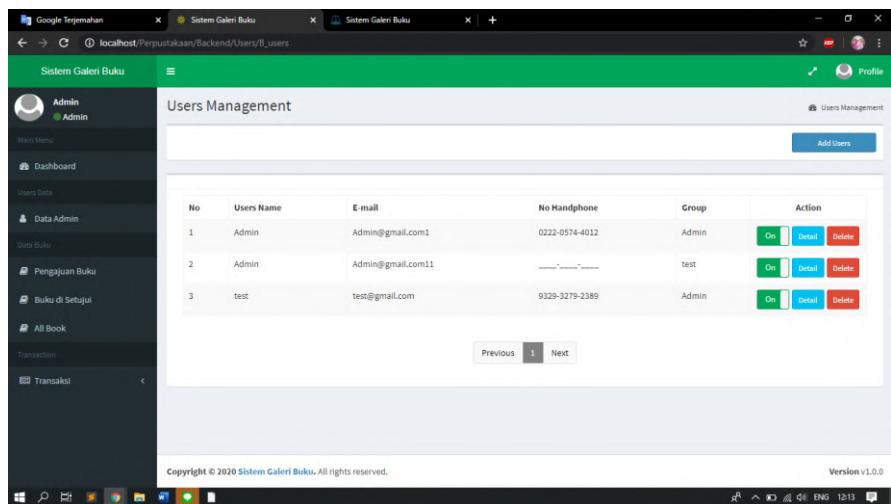
No	Task/Tugas
----	------------

1	Pemilik meng- <i>upload</i> data bukti kirim peminjaman
---	---

Penjelasan dari *task* adalah:

Task 1. Peminjam mengisi data bukti kirim peminjaman. Pada *Task* ini sistem akan menampilkan *form input*-an bukti kirim peminjaman. *User* meng-*upload* bukti kirim peminjaman. Jika foto atau bukti sudah berhasil ter-*upload*, *user* tinggal mengklik *Submit* untuk menyimpan data yang sudah di-*upload*.

10.2.11 Tampilan *Form* Kelola Data User Pada Sistem Galeri Buku



Gambar 10.19 Tampilan Halaman Kelola Data User

Task Usability Testing

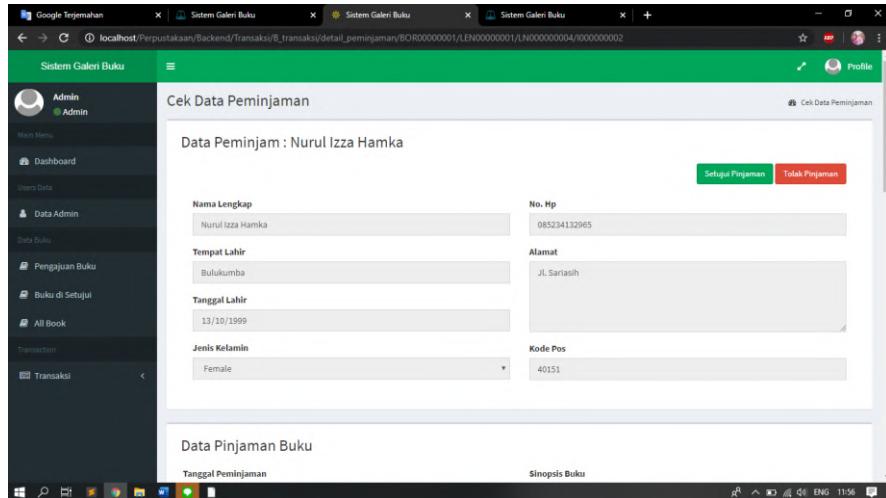
No	Task/Tugas
1	Admin mengelola data user

Penjelasan dari task adalah:

Task 1. Admin mengelola data user. Pada Task ini sistem akan menampilkan menu kelola data user. Pada menu ini, admin bisa menambahkan user, mengedit user, menghapus user dan menonaktifkan akun user. Selain itu, pada menu ini juga admin bisa melihat user yang terdaftar. Untuk menambahkan user admin bisa mengklik Button Add Users. Untuk mengedit user, admin bisa mengklik Button Edit. Dan untuk menghapus user, admin bisa mengklik Button Hapus.

10.2.12 Tampilan *Form Upload* Bukti Kirim Peminjaman Pada Sistem Galeri Buku

a. Tampilan Galeri Buku *Page Cek Peminjaman*



Gambar 10.20 Tampilan Halaman Cek Peminjaman

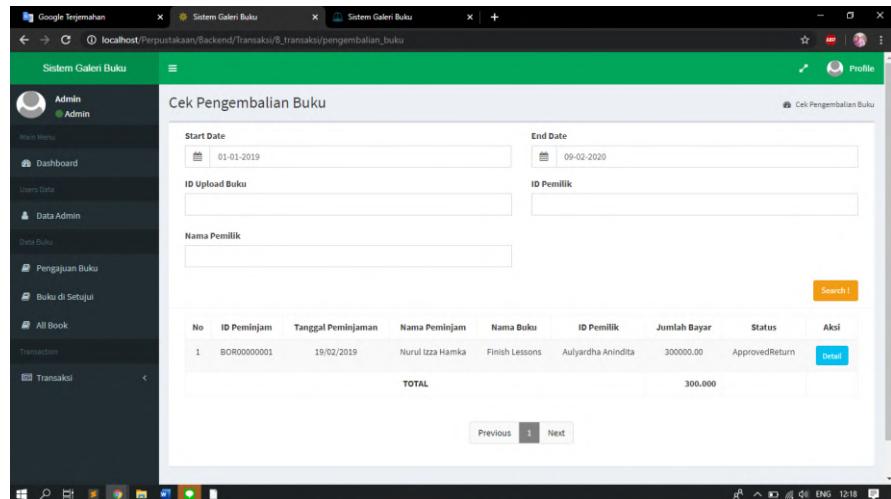
Task Usability Testing

No	Task/Tugas
1	Admin mengecek data peminjaman

Penjelasan dari task adalah:

Task 1. Admin mengecek data peminjaman. Pada Task ini sistem akan menampilkan data peminjaman yang masuk. Data peminjaman yang dimaksud adalah data buku yang dimiliki oleh pemilik akan dipinjam oleh peminjam. Disini admin bisa mengetahui buku yang akan dipinjam. Untuk melihat lebih detail data peminjaman tersebut, pemilik mengklik Button Detail. Disini admin bisa menyetujui peminjaman yang masuk.

b. Tampilan Galeri Buku Kelola Data Pengembalian



Gambar 10.21 Tampilan Halaman Kelola Data Pengembalian
Task Usability Testing

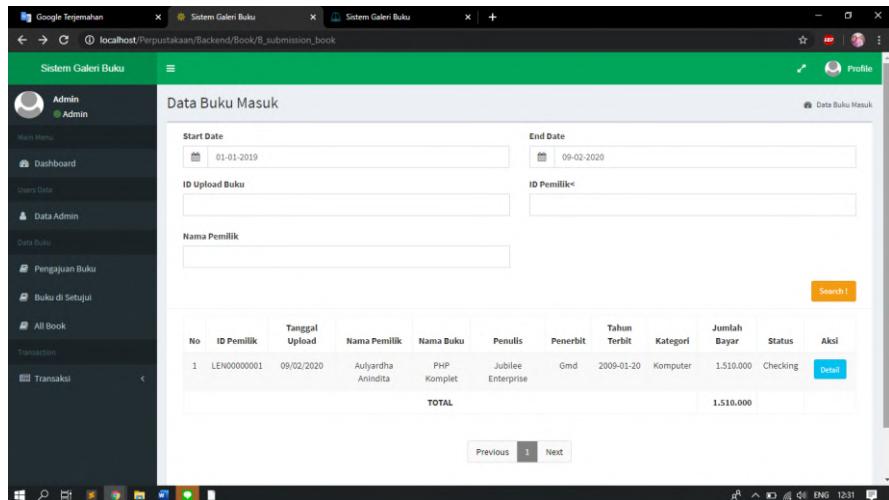
No	Task/Tugas
1	Admin mengelola data pengembalian

Penjelasan dari *task* adalah:

Task 1. Admin mengelola data pengembalian. Pada *Task* ini sistem akan menampilkan data pengembalian yang masuk. Data pengembalian yang dimaksud adalah data peminjaman yang telah selesai, dan peminjam akan melakukan proses pengembalian. Setelah peminjam melakukan proses pengembalian, admin bertugas untuk memeriksa kembali data yang telah *diinput* oleh peminjam. Untuk melihat detail data pengembalian, admin bisa mengklik *Button Detail*.data buku yang dimiliki oleh pemilik akan dipinjam oleh peminjam.

10.2.13 Tampilan *Form* Kelola Data Buku Pada Sistem Galeri Buku

a. Tampilan Buku Yang Akan Disetujui



Gambar 10.22 Tampilan Halaman Buku Yang Akan Disetujui

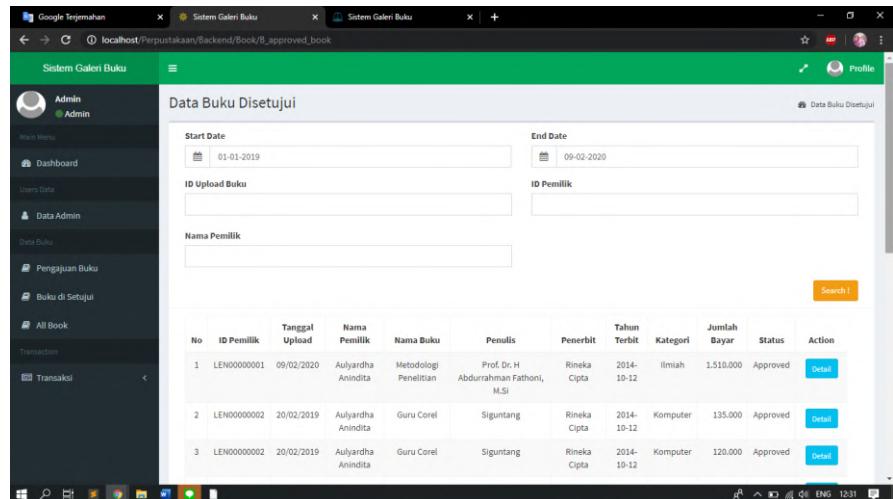
Task Usability Testing

No	Task/Tugas
1	Admin melihat data buku yang masuk

Penjelasan dari task adalah:

Task 1. Admin melihat data buku yang masuk. Pada Task ini sistem akan menampilkan buku yang masuk. Data buku yang dimaksud disini adalah data buku yang telah diinputkan oleh pemilik akan masuk disini. Admin bertugas untuk memeriksa apakah buku tersebut benar dan valid. Untuk melihat lebih detail dari rincian buku tersebut, admin bisa mengklik *Button Detail*.

b. Tampilan Buku Yang Sudah Disetujui



Gambar 10.23 Tampilan Halaman Buku Yang Sudah Disetujui
Task Usability Testing

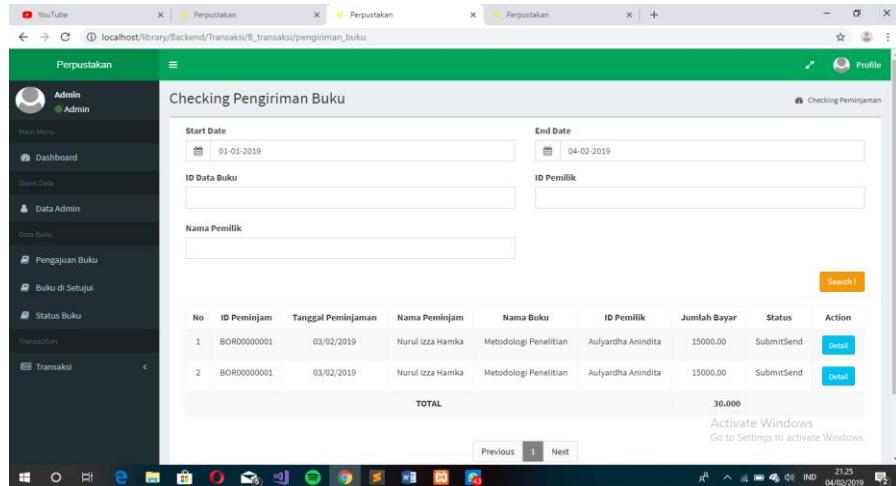
No	Task/Tugas
1	Admin melihat data buku yang sudah disetujui

Penjelasan dari *task* adalah:

Task 1. Admin melihat data buku yang sudah disetujui. Pada *Task* ini sistem akan menampilkan buku yang telah disetujui. Data buku yang dimaksud disini adalah data buku yang telah disetujui oleh Admin. Disini admin bisa melihat buku-buku yang tersedia dan bisa dipinjam oleh peminjam. Untuk melihat detail buku yang sudah disetujui admin bisa mengklik *Button Detail*.

10.2.14 Tampilan Form Konfirmasi Bukti Pengiriman Pada Sistem Galeri Buku

a. Tampilan Page Bukti Pengiriman Masuk



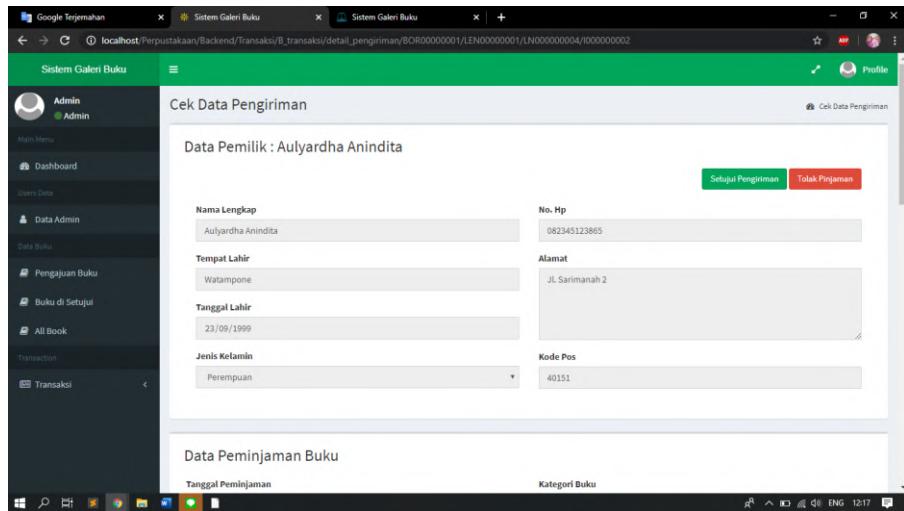
Gambar 10.24 Tampilan Halaman Bukti Pengiriman Masuk
Task Usability Testing

No	Task/Tugas
1	Admin melihat data bukti pengiriman yang masuk

Penjelasan dari task adalah:

Task 1. Admin melihat data bukti pengiriman yang masuk. Pada Task ini sistem akan menampilkan data bukti pengiriman yang masuk. Data bukti pengiriman yang dimaksud disini adalah data bukti pengiriman yang telah di-upload oleh peminjam atau pemilik. Disini admin bisa melihat bukti pengiriman yang masuk. Untuk melihat detail bukti pengiriman yang masuk admin bisa mengklik Button Detail.

b. Tampilan Page Pengiriman Masuk



Gambar 10.25 Tampilan Halaman Pengiriman Masuk

Task Usability Testing

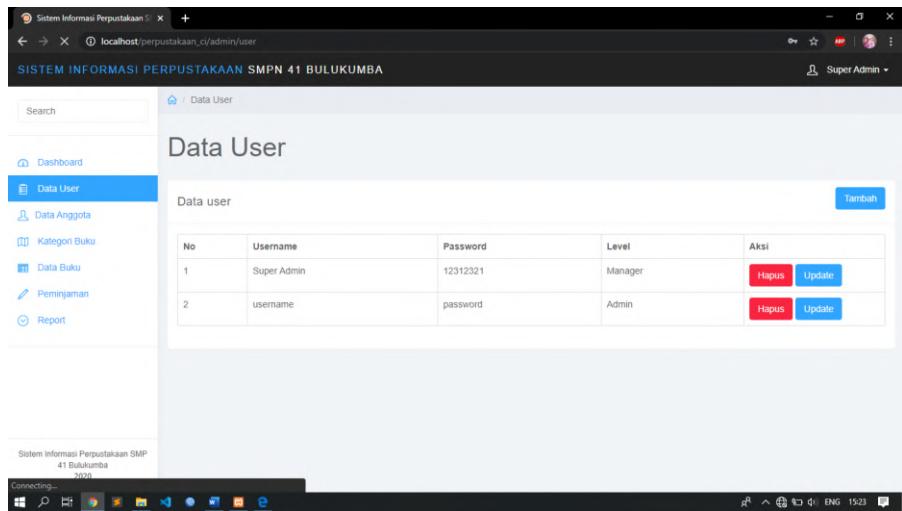
No	Task/Tugas
----	------------

1	Admin menyetujui pengiriman yang masuk
---	--

Penjelasan dari *task* adalah:

Task 1. Admin menyetujui pengiriman yang masuk. Pada *Task* ini sistem akan menampilkan data bukti pengiriman yang masuk. Data bukti pengiriman yang dimaksud disini adalah data bukti pengiriman yang telah di-upload oleh peminjam atau pemilik. Disini admin bisa melihat bukti pengiriman yang masuk. Jika data pengiriman sudah benar, maka admin bisa menyetujui bukti pengiriman tersebut dengan mengklik *Button* Setujui Pengiriman. Dan jika data tidak sesuai atau salah admin bisa mengklik *Button* Cancel.

10.2.15 Tampilan Perpustakaan *Page Data User*



Gambar 10.26 Tampilan Halaman Data User

Task Usability Testing

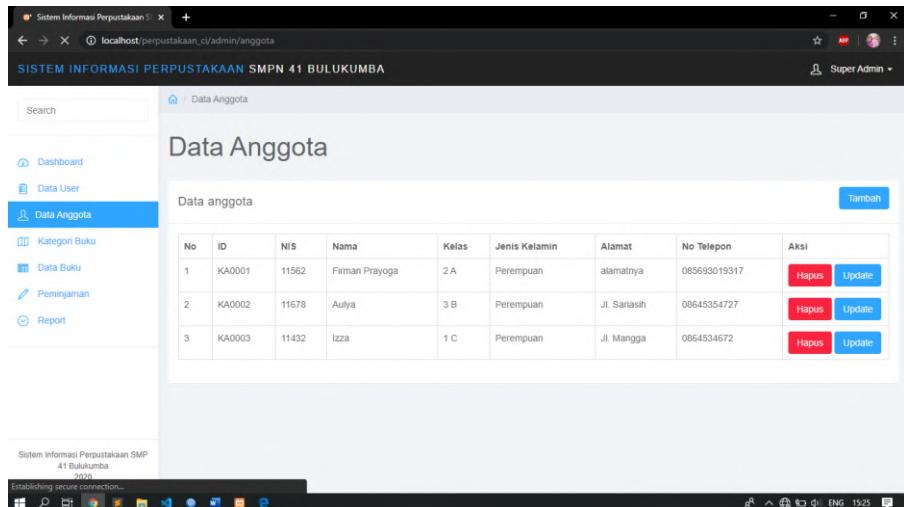
No	Task/Tugas
----	------------

1	Admin mengelola data <i>user</i>
---	----------------------------------

Penjelasan dari *task* adalah:

Task 1. Admin mengelola data *user*. Pada *Task* ini sistem akan menampilkan menu kelola data *user*. Pada menu ini, admin bisa menambahkan *user*, mengedit *user*, menghapus *user*. Selain itu, pada menu ini juga admin bisa melihat *user* yang terdaftar. Untuk menambahkan *user* admin bisa mengklik *Button* Tambah. Untuk mengedit *user*, admin bisa mengklik *Button* Edit. Dan untuk menghapus *user*, admin bisa mengklik *Button* Hapus.

10.2.16 Tampilan Perpustakaan Page Data Anggota



Gambar 10.27 Tampilan Halaman Data Anggota

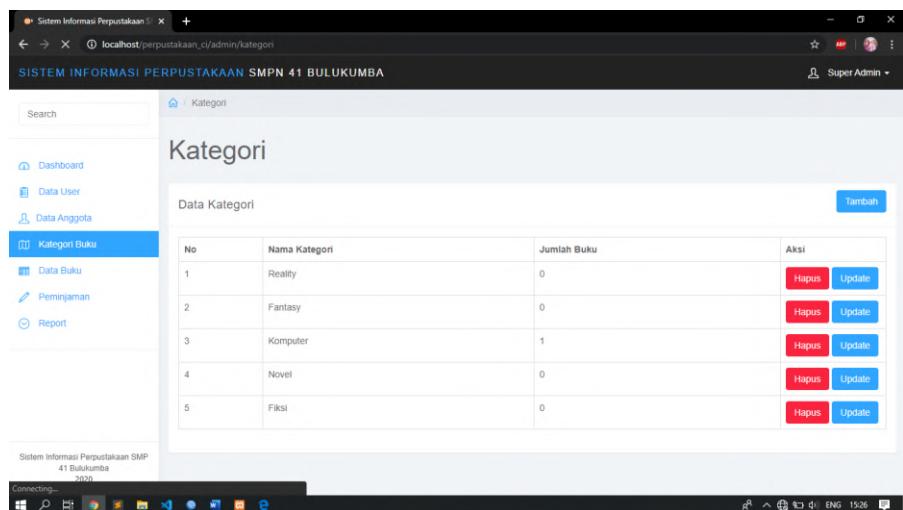
Task Usability Testing

No	Task/Tugas
1	Admin mengelola data anggota

Penjelasan dari *task* adalah:

Task 1. Admin mengelola data anggota. Pada *Task* ini sistem akan menampilkan menu kelola data anggota. Pada menu ini, admin bisa menambahkan anggota, mengedit anggota, menghapus anggota. Selain itu, pada menu ini juga admin bisa melihat anggota yang terdaftar. Untuk menambahkan anggota, admin bisa mengklik *Button* Tambah. Untuk mengedit anggota, admin bisa mengklik *Button* Edit. Dan untuk menghapus anggota, admin bisa mengklik *Button* Hapus.

10.2.17 Tampilan Perpustakaan Kategori Buku



Gambar 10.28 Tampilan Halaman Kategori Buku

Task Usability Testing

No	Task/Tugas
----	------------

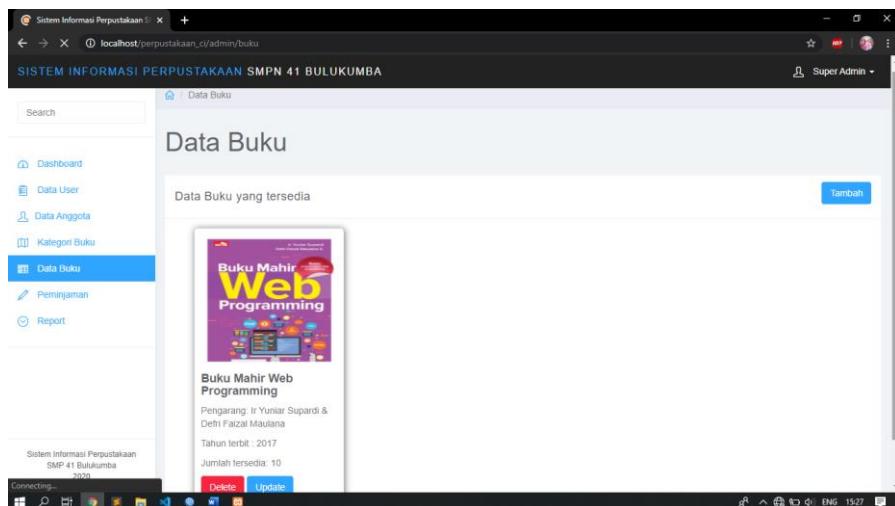
1	Admin mengelola kategori buku
---	-------------------------------

Penjelasan dari *task* adalah:

Task 1. Admin mengelola kategori buku. Pada *Task* ini sistem akan menampilkan menu kategori buku. Pada menu ini, admin bisa menambahkan kategori buku, mengedit kategori buku, menghapus kategori buku. Selain itu, pada menu ini juga admin bisa melihat kategori buku yang sudah diinputkan. Untuk menambahkan kategori buku, admin bisa mengklik *Button* Tambah. Untuk mengedit kategori

buku, admin bisa mengklik *Button* Edit. Dan untuk menghapus kategori buku, admin bisa mengklik *Button* Hapus.

10.2.18 Tampilan Perpustakaan Data Buku



Gambar 10.29 Tampilan Halaman Data Buku

Task Usability Testing

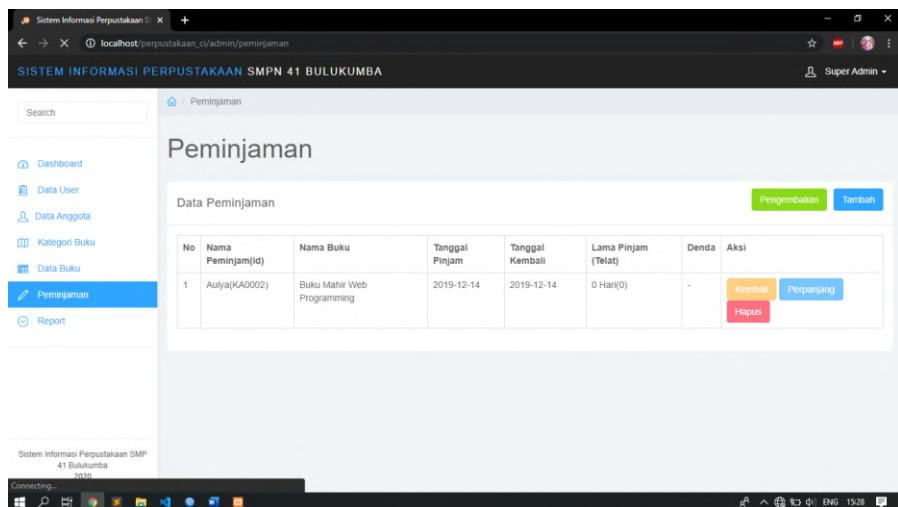
No	Task/Tugas
1	Admin mengelola data buku

Penjelasan dari *task* adalah:

Task 1. Admin mengelola data buku. Pada *Task* ini sistem akan menampilkan menu data buku. Pada menu ini, admin bisa menambahkan data buku, mengedit buku, menghapus buku. Selain itu, pada menu ini juga admin bisa melihat buku yang sudah diinputkan. Untuk menambahkan buku, admin bisa mengklik *Button* Tambah.

Untuk mengedit buku, admin bisa mengklik *Button Edit*. Dan untuk menghapus buku, admin bisa mengklik *Button Hapus*.

10.2.19 Tampilan Perpustakaan Peminjaman



Gambar 10.30 Tampilan Halaman Peminjaman

Task Usability Testing

No	Task/Tugas
1	Admin mengelola peminjaman

Penjelasan dari *task* adalah:

Task 1. Admin mengelola peminjaman. Pada *Task* ini sistem akan menampilkan menu peminjaman. Pada menu ini, admin bisa menambahkan peminjaman, mengedit peminjaman, menghapus peminjaman, pengembalian dan meminjam buku di sistem galeri

buku. Selain itu, pada menu ini juga admin bisa melihat data peminjaman yang sedang berjalan. Untuk menambahkan peminjaman, admin bisa mengklik *Button* Tambah. Untuk mengedit peminjaman, admin bisa mengklik *Button* Edit. Dan untuk menghapus peminjaman, admin bisa mengklik *Button* Hapus. Serta untuk mengembalikan buku yang telah dipinjam, admin bisa mengklik *Button* Kembali.

DAFTAR PUSTAKA

- [1] Anza, Fikri Akbarsyah., Fathmawati, Mayer. “MANAJEMEN INTEGRASI SISTEM INFORMASI PERPUSTAKAAN DI LINGKUNGAN UNIVERSITAS INDONESIA DALAM MENUJU PERPUSTAKAAN DIGITAL.” Jurnal Vokasi Indonesia Volume 7, Januari-Juni 2019
- [2] Hikmah, Agung Baitul. “MENDEFINISIKAN ENTERPRISE ARCHITECTURE PLANNING DALAM PERENCANAAN INTEGRASI SISTEM INFORMASI PERPUSTAKAAN SEKOLAH”. Jurnal Informatik Vol.1 No. 2. September 2014
- [3] Lovasz,Agneta Elisabate, Lovasz, Erwin-Christian and Gruescu, Corina Mihaela, “Digital Library of Mechanisms.” Procedia - Social and Behavioral Sciences 163. 85 – 91, 2014
- [4] Xu,Fang and Du, Jia Tina, “ Factors Influencing Users’ Satisfaction and Loyalty To Digital Libraries In Chinese Universities.” Computers In Human Behavior 83. 64-72, 2018
- [5] Sibagariang, Swono. “PENERAPAN WEB SERVICE PADA PERPUSTAKAAN BERBASIS ANDROID.” Jurnal Mahajana Inforamasi 1, 2527-8290, 2016

- [6] Ardi Binarso, Yusi., Adi Sarwoko, Eko., Bahtiar, Nurdin. (2012). “Pembangunan Sistem Infomasi Alumni Berbasis Web Pada Program Studi Teknik Informatika Universiats Diponegoro.” *Journal of Informatics and Technology*, Vol. 1: 72-84
- [7] Tata Sutabri, Christian Putri (Ed.).2012. *Analisi Sistem Informasi*:Yogyakarta : CV ANDI OFFEST
- [8] Arisma Susanto.2010 “PENGEMBANGAN SISTEM INFORMASI INVENTORY PADA PT. DWIWARNA SEJAHTERA”. Skripsi. Program Studi Sistem Informasi. Fakultas Sains Dan Teknologi. Universitas Islam Negeri Syarif Hidayatullah. Jakarta.
- [9] Hanif Al Fatta. Agnes Heni Triyuliana (Ed.). 2007.*Analisis & Perancangan Sistem Informasi untuk Keunggulan Bersaing Perusahaan & Organisai Modern*: Yogyakarta: CV ANDI OFFEST
- [10] Zadir., Ardani Andreas. “ANALISIS DAN PERANCANGAN SISTEM INFORMASI TERINTEGRASI UNTUK MANAJEMEN PRODUKSI, PERSEDIAAN DAN DISTRIBUSI BARANG (STUDI KASUS: PABRIK KEMASAN KERTAS CV. YOGYAKARTAS).” *Jurnal Teknologi Informasi* Vol.XII, 1907-2430, 35 Juli 2017
- [11] Rizal, Randi., Rahmatulloh, Alam. “RESTful Web Service untuk Integrasi Sistem Akademik dan Perpustakaan Universitas Perjuangan.” *Jurnal Ilmiah komputer* Vol.07, 2337-8379.Print. 2019
- [12]
- [13] Abdul Kadir, “Dasar Aplikasi *Database MySQL + Delphi*”: Yogyakarta: CV ANDI OFFEST
- [14] Fathansyah. Agustus 2004 “BUKU TEKS KOMPUTER SISTEM BASIS DATA LANJUTAN BUKU BASIS DATA”: Bandung: Informatika

- [15] Fathansyah. 5 Agustus 2004 “BUKU TEKS KOMPUTER SISTEM BASIS DATA”: Bandung: Informatika Bandung
- [16] Istiono, Wirawan., Hujrah, Sutarya. “Pengembangan Sistem Aplikasi Penilaian dengan Pendekatan MVC dan Menggunakan Bahasa PHP dengan *Framework* Codeigniter dan *Database* MYSQL pada Pahoa College Indonesia.” Jurnal TICOM Vol. 5. 1 September 2016
- [17] Achmad Solichin, S.Kom. Februari 2016 “Pemrograman Web dengan PHP dan MySQL”.
- [18] Lee, Sunguk. “Unified Modeling Language (UML) for *Database* Systems and Computer Applications.” International Journal of *Database* Theory and Application 5. March, 2012
- [19] Kaur, Harpreet., Singh, Pardeep. “UML (Unified Modeling Language): Standard Language for Software Architecture Development.” 2009 International Symposium on Computing, Communication, and Control 1. Singapore 2010
- [20] Wardana, S.Hut., M.Si. Septermber 2010 “Menjadi Master PHP Dengan *Framework* Codeigniter”.
- [21] Haviluddin, Agus Tri Haryono, Dwi Rachmawati, Kiswanto, Triana Fitriastuti (Ed.). Desember 2016. Aplikasi Program PHP & MySQL. Samarinda – Kalimantan Timur – INDONESIA 75123: Mulawarman University PRESS
- [22] Wuri Oktaviani, Theresia. “Perancangan *User Interface* Berbasis *Web* untuk Home Automation Gateway Berbasis IQRF TR53B.” JNTETI 03.Augustus 2014
- [23] Renaldy Suteja, Bernard., Harjoko, Agus. “*User Interface* Design for e-Learning System.” Seminar Nasional Aplikasi Teknologi Informasi. ISSN: 1907-5022. 21 Juni 2008

TENTANG PENULIS