

BAB II

TINJAUAN PUSTAKA

2.1. Pengertian Aplikasi

Aplikasi adalah suatu perangkat lunak atau program yang dapat membantu pengguna dalam menyelesaikan tugas-tugas tertentu. Aplikasi dapat berupa software desktop, seperti aplikasi pengolah teks, spreadsheet, atau database, atau aplikasi web yang dapat diakses melalui internet. Aplikasi juga dapat berupa aplikasi *mobile* yang diinstall pada perangkat seluler seperti smartphone atau tablet[1].

Aplikasi memiliki beragam fungsi, mulai dari tugas sehari-hari seperti mengirim email, membuat dokumen, atau mengatur jadwal, hingga tugas yang lebih kompleks seperti analisis data, pemodelan sistem, dan lain-lain. Aplikasi juga dapat membantu mempermudah akses dan pengelolaan data, sehingga mempermudah pekerjaan dan meningkatkan efisiensi.

Aplikasi dapat dikembangkan oleh pengembang aplikasi independen atau perusahaan teknologi informasi, dan dapat diunduh atau dibeli melalui toko aplikasi pada perangkat elektronik. Dalam era digital saat ini, aplikasi menjadi bagian yang tidak terpisahkan dalam kehidupan sehari-hari dan memegang peran penting dalam mempermudah akses dan pengolahan informasi.

Berdasarkan pengertian diatas aplikasi merupakan bantu kerja yang dapat mempermudah atau mempercepat pekerjaan secara efektif dan efisien, sehingga sangat membantu dalam menyelesaikan sebuah masalah dengan cepat dan tepat.

2.2. Buku Induk

Buku Induk Siswa adalah suatu catatan resmi yang berisi data-data penting tentang siswa, seperti nama, tanggal lahir, alamat, orang tua, riwayat pendidikan, dan informasi-informasi lain yang berkaitan dengan siswa tersebut. Buku Induk Siswa sering digunakan sebagai acuan bagi sekolah atau institusi pendidikan untuk mengelola dan mencatat sejarah pendidikan siswa[2].

Buku Induk Siswa berfungsi sebagai dokumen resmi yang memuat informasi penting tentang siswa dan merupakan sumber informasi yang dapat dipercaya bagi sekolah atau institusi pendidikan. Informasi yang terdapat dalam Buku Induk Siswa dapat digunakan untuk membuat keputusan-keputusan yang berkaitan dengan pendidikan siswa, seperti penetapan kelas, pemberian beasiswa, dan lain-lain.

Secara umum, Buku Induk Siswa memegang peran penting dalam mengelola data siswa dan membantu sekolah atau institusi pendidikan dalam menyediakan layanan pendidikan yang baik bagi siswa. Buku Induk Siswa juga dapat menjadi sumber informasi yang berguna bagi siswa dan orang tua saat mencari informasi tentang riwayat pendidikan siswa.

2.3. Website

Website dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang disediakan melalui jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang di seluruh dunia [9].

World Wide Web atau WWW atau juga dikenal dengan WEB adalah salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke internet. Web ini menyediakan informasi bagi pemakai komputer yang terhubung ke internet dari sekedar informasi “sampah” atau informasi yang tidak berguna sama sekali sampai informasi yang serius dari informasi yang gratisan sampai informasi yang komersial.

Jenis-Jenis Website Secara umum, website dibagi menjadi 3 jenis, yaitu website statis, dinamis, dan interaktif [9].

1) Website Statis

Website statis yaitu website yang isinya tidak diperbaharui secara berkala, sehingga isinya dari waktu ke waktu akan selalu tetap.

2) Website Dinamis

Website dinamis yaitu jenis website yang isinya terus diperbaharui secara berkala oleh pengelola web atau pemilik website. Website jenis ini

banyak dimiliki oleh perusahaan atau perorangan yang aktifitas bisnisnya memang berkaitan dengan internet.

3) Website Interaktif

Website interaktif pada dasarnya termasuk dalam kategori website dinamis, dimana isinya selalu diperbaharui dari waktu ke waktu. Hanya saja, isi informasi tidak hanya diubah oleh pengelola website tetapi lebih banyak dilakukan oleh pengguna website itu sendiri.

2.4. PHP

PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan *web* dan dapat ditanamkan pada skripsi HTML [3].

PHP merupakan *server-side scripting* maka sintaks dan perintah-perintah PHP akan dieksekusi di *server* kemudian hasilnya dikirimkan ke *browser* dalam *format* HTML. Dengan demikian kode program yang ditulis dalam PHP tidak akan terlihat oleh *user* sehingga keamanan halaman *website* lebih terjamin. PHP dirancang untuk membentuk halaman *website* yang dinamis, yaitu halaman *website* yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke halaman *website*.

PHP juga dapat berjalan pada berbagai *web server* seperti IIS (*Internet Information Server*), PWS (*Personal Web server*), Apache, Xitami. PHP juga mampu lintas *platform*. Artinya *PHP* dapat berjalan di banyak sistem operasi yang beredar saat ini, diantaranya: Sistem Operasi Microsoft Windows (semua versi), Linux, Mac OS, Solaris. *PHP* dapat dibangun sebagai modul pada *web server* Apache dan sebagai binary yang dapat berjalan sebagai CGI (*Common Gateway Interface*). PHP dapat mengirim HTTP header, dapat mengatur *cookies*, mengatur *authentication* dan *redirect users*.

Salah satu keunggulan yang dimiliki oleh *PHP* adalah kemampuannya untuk melakukan koneksi ke berbagai macam *software* sistem manajemen basis data atau *Database Management System* (DBMS), sehingga dapat menciptakan suatu halaman *web* yang dinamis. *PHP* mempunyai koneksitas yang baik dengan

beberapa DBMS antara lain Oracle, Sybase, mSQL, *MYSQL*, Unix dbm, dan tak terkecuali semua *database* ber-interface ODBC. Kemampuan utamanya adalah konektivitas basis data dengan *web*. Dengan kemampuan ini kita akan mempunyai suatu sistem basis data yang dapat diakses dari *web*.



Gambar. 2.1 Logo PHP

2.5. PostgreSQL

PostgreSQL merupakan aplikasi yang digunakan untuk mengolah basis data yang banyak digunakan untuk membangun aplikasi yang menggunakan *database* [7].

PostgreSQL dikembangkan oleh *University of California di Berkeley Computer Science Department*. Dengan sifatnya yang *open source* menjadikan database ini dapat dikembangkan sesuai kebutuhan

Sebagai ORDBMS (*Object Relational Database Management System*) yang ada saat ini, *PostgreSQL* memiliki berbagai macam kemampuan yang dimiliki oleh database komersil umum lainnya, seperti dukungan perintah-perintah SQL, dimana dengan menggunakan perintah-perintah SQL memungkinkan database administrator lebih mudah berinteraksi dengan database *PostgreSQL*, baik dalam manipulasi data seperti : *insert*, *update*, ataupun *delet*.

Berdasarkan pengertian diatas disimpulkan bahwa *PostgreSQL* adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan pada *web app*, aplikasi *mobile*, dan aplikasi *analytics* yang menggunakan *database* sebagai sumber dan pengelolaan datanya.

2.6. XAMPP

XAMPP adalah perangkat pembantu yang menyediakan alat untuk sebagai jembatan pembuatan sebuah program [2]. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP *server*, MySQL

database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. Nama XAMPP merupakan singkatan dari:

X program ini dapat dijalankan di banyak sistem operasi, seperti Windows, Linux, Mac OS, dan Solaris.

A merupakan *Apache* aplikasi *web server*. Tugas utama Apache adalah menghasilkan halaman *web* yang benar kepada user berdasarkan kode *PHP* yang dituliskan oleh pembuat halaman *web*.

M merupakan MySQL aplikasi *database server*. *MYSQL* dapat digunakan untuk membuat dan mengelola *database* beserta isinya. Kita dapat memanfaatkan *MYSQL* untuk menambahkan, mengubah, dan menghapus data yang berada dalam *database*.

P merupakan PHP bahasa pemrograman *web*. Bahasa pemrograman *PHP* merupakan bahasa pemrograman untuk membuat *web* yang bersifat *server-side scripting*. *PHP* memungkinkan kita untuk membuat halaman *web* yang bersifat dinamis.

Perl merupakan bahasa pemrograman untuk segala keperluan, dikembangkan pertama kali oleh Larry Wall di mesin *Unix*. *Perl* dirilis pertama kali pada tanggal 18 Desember 1987 ditandai dengan keluarnya .

Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis.

Berdasarkan pengertian diatas, xampp merupakan aplikasi *server localhost* yang biasa digunakan untuk membangun sebuah *web* dinamis dan sudah dilengkapi dengan *database*.



Gambar 2.2 Logo XAMPP

2.7. Codeigniter

CodeIgniter merupakan *framework* PHP yang dibuat berdasarkan *model view Controlleer* (MVC). CI memiliki *library* yang lengkap untuk mengerjakan operasi-operasi yang umum dibutuhkan oleh aplikasi berbasis web misalnya mengakses *database*, memvalidasi *form* sehingga sistem yang dikembangkan mudah. CI juga menjadi satu-satunya *Framework* dengan dokumentasi yang lengkap dan jelas. *Source code* CI yang dilengkapi dengan *comment* didalamnya sehingga lebih memperjelas fungsi sebuah kode program dan CI yang dihasilkan sangat Bersih (*clean*) dan *search Engine Friendly* (SEF). *Codeigniter* juga dapat memudahkan *developer* dalam membuat aplikasi web berbasis PHP, karena *framework* sudah memiliki kerangka kerja sehingga tidak perlu menulis semua kode program dari awal. Selain itu, struktur dan susunan logis dari *codeigniter* membuat aplikasi menjadi semakin teratur dan dapat fokus pada fitur-fitur apa yang akan dibutuhkan dalam pembuatan aplikasi tersebut. [5].

Berdasarkan kesimpulan di atas *Framework Codeigniter* adalah kerangka kerja yang memiliki *library* yang lengkap untuk mengerjakan aplikasi buku induk ini, diharapkan dengan menggunakan kerangka kerja ini aplikasi buku induk lebih tersusun dengan sistematis.

2.8.Bootstrap

Bootstrap merupakan sebuah aplikasi yang dijadikan sebagai alat bantu untuk mendesain halaman web yang menarik [2].

Bootstrap adalah *framework* web development berbasis HTML, CSS dan *JavaScript* yang dirancang untuk mempercepat proses pengembangan web *responsive*. Bootstrap merupakan *framework* gratis yang bersifat *open-source*.

2.9.HTML

HTML (*Hyper Text Markup Language*) merupakan bahasa standar yang digunakan oleh protocol HTTP (*Hyper Text Transfer Protocol*). HTML memiliki ciri-ciri sebagai berikut:

- a. Tersusun oleh *tag-tag* (sebagai penanda, karena inilah maka HTML disebut sebagai *markup language*), misalnya `<html>...</html>`.
- b. Pada umumnya tag selalu mempunyai tag pembuka dan tag penutup, namun ada beberapa *tag* yang tidak mempunyai tag penutup misalnya `
`, `<hr>`, dan sebagainya.
- c. Tidak *case sensitive* (huruf kecil dan kapital dianggap sama).
- d. Nama *file* berupa `*.html` .

2.10. VS Code

Visual Studio Code (VS Code) ini adalah sebuah teks editor ringan dan handal yang dibuat oleh *microsoft* untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi linux, Mac, dan Windows. Teks editor ini secara langsung mendukung bahasa pemrograman *JavaScript*, *Typescript*, dan *Node.js* serta bahasa pemrograman lainnya dengan bantuan *plugin* yang dapat dipasang *via marketplace Visual Studio Code* (seperti C++, C#, Python, Go, Java, dst)[4].

Banyak sekali fitur-fitur yang disediakan oleh *Visual Studio Code*. Diantaranya *Intellesense*, *Git Integraton*, *Debuging*, dan fitur ekstensi yang menambah kemampuan teks editor. Fitur-fitur tersebut akan terus bertambah seiring dengan bertambahnya versi *Visual Studio Code*. Pembaharuan versi *Visual Studio Code* ini juga dilakukan berkala setiap bulan, inilah yang membedakan VS Code dengan teks editor yang lain.

Berdasarkan kesimpulan diatas *visual studio code* adalah sebuah teks editor bersifat *open source* ringan dan handal dibuat oleh *Microsoft* untuk sistem operasi Linux, Mac dan Windows.



Gambar 2.3 Logo *Visual Code Studio*

2.11. Java script

Java script merupakan bahasa *scripting* pada client dimana kita dapat menambah program pada *web page* yang beroperasi secara langsung pada client (misalnya, pada mesin yang menjalankan *web browser*). *Java script* sering digunakan untuk tipe komputasi pada hal-hal berikut:

- a. Deteksi *browser* : digunakan untuk mendeteksi tipe *web browser* dan me-load halaman khusus pada *web browser*.
- b. Validasi *Form* : melakukan pemeriksaan konsisten yang sederhana pada *field-field form*.

Kontrol *Browser* : hal ini meliputi halaman pembuka dalam windows yang di kustomkan, contohnya iklan.



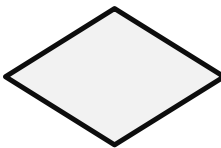


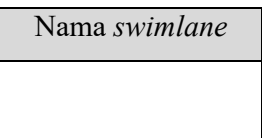
2.12. Perancangan Sistem

Dalam aplikasi ini penulis menggunakan perancangan bersifat *Object Oriented* (berorientasi objek) dengan menggunakan Unified Modeling Language (UML) yang bertujuan untuk memberi gambaran kepada pengguna mengenai aplikasi yang akan di buat [6].

UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. (Sukamto dan Shalahuddin, 2013) Bahasa pemrograman berorientasi objek yang pertama dikembangkan dikenal dengan nama Simula-67 yang dikembangkan pada tahun 1967. Perkembangan aktif dari pemrograman berorientasi objek mulai menggeliat ketika berkembangnya bahasa pemrograman Smalltalk pada awal 1980-an. Pada 1996, *Object Management Group* (OMG) mengajukan proposal agar adanya standardisasi pemodelan berorientasi objek dan pada bulan September 1997 *Unified Modeling Language* (UML) diakomodasi oleh *Object Management Group* (OMG) sehingga sampai saat ini *Unified Modeling Language* (UML) telah memberikan kontribusinya yang cukup besar dalam metodologi berorientasi objek.

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* sebuah proses bisnis dan urutan aktivitas dalam suatu proses untuk dapat memahami proses secara keseluruhan. *Activity* diagram juga sangat berguna ketika ingin menggambarkan perilaku paralel atau menjelaskan bagaimana perilaku dalam berbagai *use case* berinteraksi.

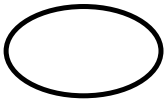
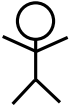

Table 2.1 Simbol *Activity Diagram*




No	Nama dan Simbol	Keterangan
1.	Status Awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.	Aktivitas / <i>activity</i> 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.	Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan / <i>Join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.	Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.	<i>Swimlane</i> 	Status awal aktivitas sistem, sebuah <i>Swimlane</i> memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

Use-case diagram merupakan suatu bentuk diagram yang menggambarkan fungsi-fungsi yang diharapkan dari sebuah sistem yang dirancang. Dalam *Use-case*

diagram penekanannya adalah “apa” yang diperbuat oleh sistem , dan bukan “bagaimana”. Sebuah use-case akan mempresentasikan sebuah interaksi antara pelaku atau *actor* dengan sistem. Use-case diagram yang digunakan dalam merancang suatu sistem dapat sangat membantu pada saat kita menyusun requirement sebuah sistem, mengkomunikasikannya dengan klien, dan merancang pengujian untuk semua fitur yang terdapat dalam sistem. Dalam suatu sistem aplikasi *Database*, *Use Case* diagram sangat membantu *requirement* apa saja yang diperlukan. *Use Case* diagram menjelaskan manfaat sistem jika dilihat menurut pandangan orang yang berada diluar sistem (*actor*). Diagram ini menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem berinteraksi dengan dunia luar. Pada tahap analisa, *Use Case* diagram sangat berperan untuk menemukan requirement-requirement dan untuk memahami bagaimana sistem seharusnya bekerja. Dalam sebuah model mungkin terdapat satu atau beberapa use-case diagram.

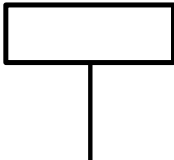

Tabel 2.2 Simbol Use Case Diagram

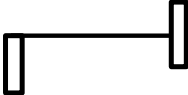
No	Nama dan Simbol	Keterangan
1.	<p><i>Use Case</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit - unit yang saling bertukar pesan antar unit atau aktor. Biasa dinyatakan dengan nama <i>use case</i> .
2.	<p>Aktor / <i>Actor</i></p> 	Mengspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>Use Case</i> .
3.	<p>Asosiasi / <i>Association</i></p> 	Apa yang menghubungkan antara objek satu dengan objek lainnya.

4.	Kebergantungan / <i>Dependdency</i> 	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung.
5.	Generalisasi / <i>Generalization</i> 	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
6.	<i>System</i> 	Menspesifikasikan paket yang menampilkan sistem secara terbatas.

Sequence Diagram merupakan gambaran secara grafis dari sebuah skenario yang menunjukkan interaksi objek dalam sebuah urutan waktu apa yang terjadi pertama kali dan apa yang terjadi berikutnya. Diagram ini secara khusus berasosiasi dengan *use case*. *Squence* diagram memperlihatkan tahap demi tahap apa yang seharusnya dilakukan untuk menghasilkan sesuatu didalam *use case*. Diagram ini sangat diperlukan pada tahap analisis atau tahap awal desain *system*.

Tabel 2.3 Simbol *Sequence Diagram*

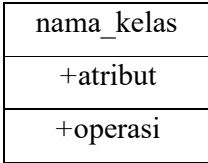

No	Nama dan Simbol	Keterangan
1.	<i>Life Line</i> 	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2.	<i>Messege</i> 	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang akitifitas yang terjadi.





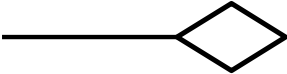
3.	<p style="text-align: center;"><i>Message</i></p> 	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.

Class diagram adalah alat perancangan terbaik untuk tim pengembang. Diagram tersebut membantu pengembang mendapatkan struktur sistem sebelum kode ditulis, dan membantu untuk memastikan bahwa *system* adalah desain terbaik. 13 *Class* diagram digunakan untuk menampilkan kelas-kelas dan paket-paket di dalam sistem. *Class* diagram memberikan gambaran sistem secara statis dan relasi antar mereka. Biasanya, dibuat beberapa *class* diagram untuk sistem tunggal. Beberapa diagram akan menampilkan *subset* dari kelas-kelas dan relasinya. Dapat dibuat beberapa diagram sesuai dengan yang diinginkan untuk mendapatkan gambaran lengkap terhadap sistem yang dibangun. *Class* diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

- a. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
- b. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Tabel 2.4 Simbol *Class Diagram*

No	Nama dan Simbol	Keterangan
1.	<p style="text-align: center;">Kelas / <i>Class</i></p> 	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
2.	<p style="text-align: center;"><i>Nary Association</i></p> 	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.

3.	Asosiasi / <i>Association</i> 	Apa yang menghubungkan antara objek satu dengan objek lainnya.
4.	Asosiasi berarah / <i>Directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.	Generalisasi / <i>Generalization</i> 	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
6.	Kebergantungan / <i>Dependency</i> 	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7.	Agregasi / <i>aggregation</i> 	Relasi antar kelas dengan makna semua bagian (<i>whole - part</i>).

2.13. Studi Literatur Sejenis

Dalam penelitian ini penyusun menggunakan dari berbagai sumber karya ilmiah yang merupakan hasil dari penelitian ilmiah yang menekankan pada kelebihan dan kekurangan dari sistem yang telah dirancang sebagai bahan acuan terhadap sistem yang akan dibuat.

Tabel 2.5 Literatur Sejenis

Judul	Tahun	Masalah	Hasil
Pengembangan sistem informasi buku induk siswa berbasis web untuk membantu	2022	Sistem pengolahan data siswa yang masih manual	Aplikasi Buku Induk Siswa di SMP Negeri 4 Ngawi

administrasi sekolah di SMP Negeri 4 Ngawi			
Sistem Informasi Buku Induk Siswa Pada SMA Negeri 1 Kelekar	2022	Bagaimana membuat Sistem Informasi Buku Induk Pada SMA Negeri 1 Kelekar	Sistem Informasi Buku Induk Siswa di SMA Negeri 1 Kelekar
Sistem Informasi Buku Induk Siswa SMK Negeri 1 Siantar berbasis Web	2022	Bagaimana membuat Sistem Informasi Buku Induk Siswa SMK Negeri 1 Siantar	Sistem Informasi Buku Induk Siswa SMK Negeri 1 Siantar

Penyusun membahas pembuatan sistem pengelolaan data siswa sehingga mempermudah kepala sekolah dan guru untuk memantau keadaan siswa. Dimana dengan aplikasi buku induk ini data semua siswa dalam satu wadah yang dapat diakses dengan mudah. Sehingga data yang tersimpan dapat menjadi referensi dalam proses pencarian data siswa jika dibutuhkan. Namun dalam sistem yang dibuat oleh penyusun di atas belum menggunakan kerangka sisetem. Dalam hal ini kerangka sistem yang dimaksud penulis adalah penggunaan framework CodeIgniter sehingga dalam susunan kerangka aplikasi dapat tersusun dengan baik sehingga diharapkan dapat memberikan perfoma yang baik dalam aplikasi buku induk yang akan penulis buat.