

BAB II

KAJIAN LITERATUR

Dalam bab ini akan dijelaskan mengenai alur penelitian, bagaimana penelitian tersebut terkait dengan penelitian-penelitian sebelumnya dan mengumpulkan teori/definisi terkait dengan penelitian yang dilakukan.

2.1 Tinjauan pustaka

2.1.1 Rancang

Rancangan adalah sebuah proses untuk mendefinisikan sesuatu yang akan dikerjakan dengan menggunakan teknik yang bervariasi serta di dalamnya melibatkan deskripsi mengenai arsitektur serta detail komponen dan juga keterbatasan yang akan dialami dalam proses pengerjaannya. Perancangan merupakan serangkaian prosedur untuk menjelaskan hasil analisis dan sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan detail bagaimana komponen-komponen sistem diimplementasikan [2].

2.1.2 Bangun

Bangun sistem adalah suatu kegiatan untuk menciptakan sistem baru atau memperbaiki sistem yang telah ada. Sehingga dapat disimpulkan bahwa rancang bangun adalah penggambaran, perencanaan, dan pembuatan sketsa dari beberapa elemen yang terpisah ke dalam suatu kesatuan yang utuh dan berfungsi. Dengan demikian pengertian rancang bangun merupakan kegiatan menerjemahkan hasil dari analisis ke dalam bentuk paket perangkat lunak kemudian menciptakan sebuah sistem atau memperbaiki sistem yang sudah ada [2].

2.1.3 Sistem Informasi

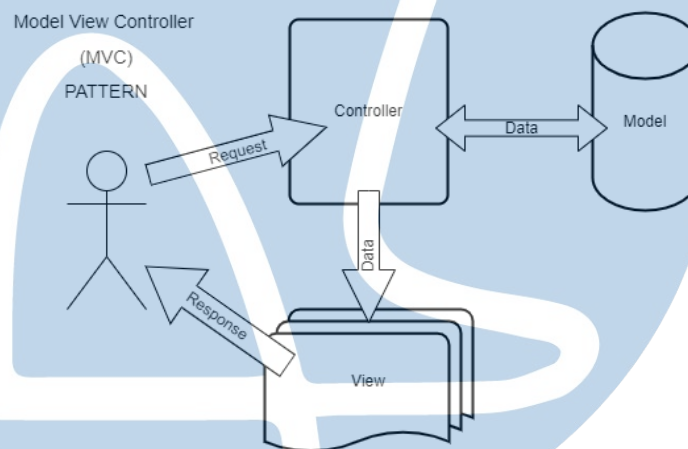
Sistem informasi adalah gabungan dari dua kata, yaitu “sistem” dan “informasi”. Sistem merupakan gabungan dari beberapa komponen yang bekerja sama untuk mencapai suatu tujuan. Informasi merupakan suatu data yang telah diproses. Sistem informasi memiliki makna yang bertujuan untuk menampilkan informasi [5].

2.1.4 Website

Website merupakan sebuah kumpulan dari halaman-halaman yang berhubungan dengan file-file lain yang saling terkait. Dalam sebuah website terdapat satu halaman yang dikenal sebagai home-page. *Homepage* adalah sebuah halaman atau tampilan yang pertama kali dilihat ketika seseorang mengunjungi sebuah website [4].

2.1.5 Model View Controller (MVC)

Model View Controller (MVC) merupakan suatu konsep yang digunakan dalam membangun sebuah aplikasi dengan memisahkan data (*Model*) dari tampilan (*View*) dan cara bagaimana memprosesnya (*Controller*) [14].



Gambar 2.1 Pola Desain MVC

Model View Controller (MVC) memisahkan pengembangan aplikasi menjadi tiga komponen yaitu :

- Model

Model merupakan bagian penanganan yang berhubungan dengan pengolahan atau manipulasi *database* (*insert, update, delete, search*), menangani validasi dari bagian *controller*.

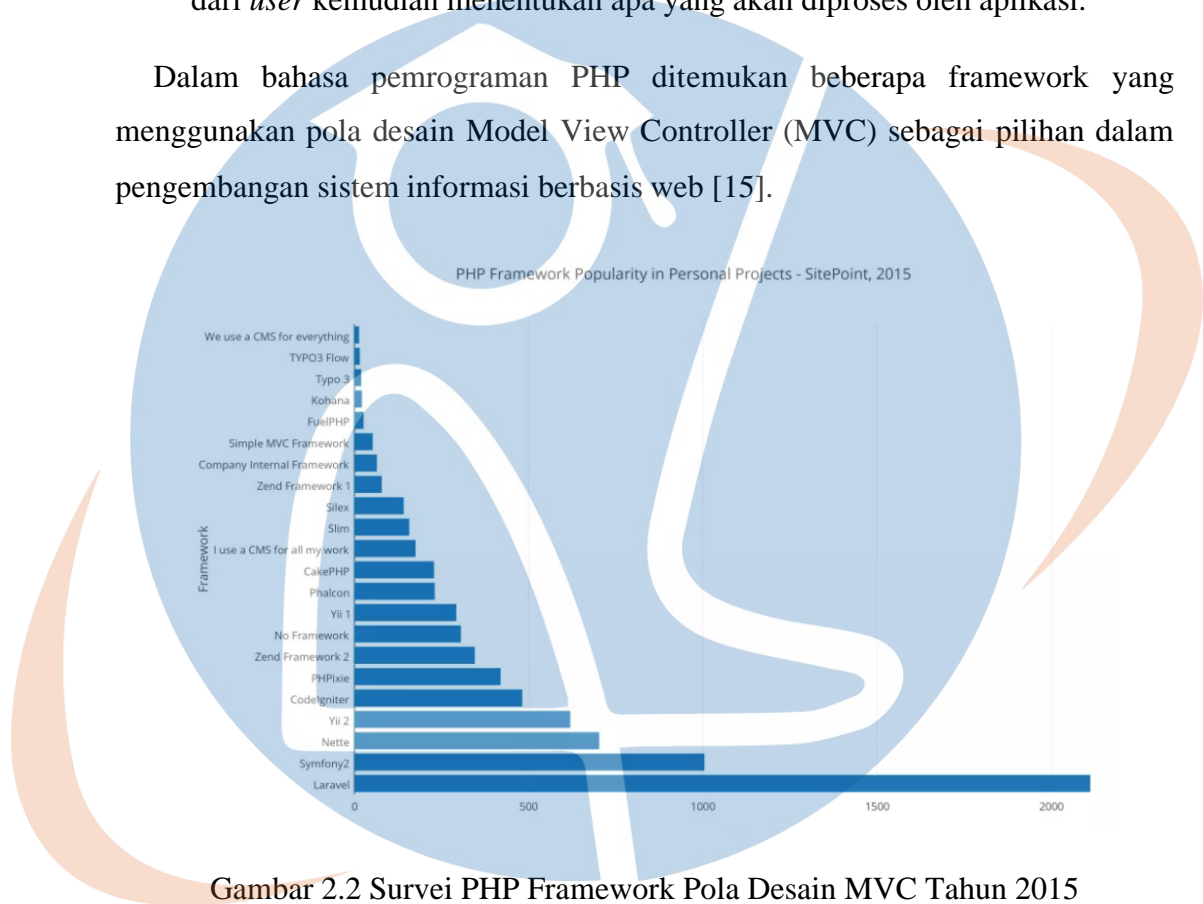
- View

View merupakan bagian yang menampilkan halaman *user interface* atau halaman yang muncul pada user. *View* memiliki fungsi untuk menerima dan merepresentasikan data kepada *user*.

- Controller

Controller merupakan bagian pengatur hubungan antara bagian *model* dan bagian *view*. *Controller* memiliki fungsi untuk menerima *request* dan data dari *user* kemudian menentukan apa yang akan diproses oleh aplikasi.

Dalam bahasa pemrograman PHP ditemukan beberapa framework yang menggunakan pola desain Model View Controller (MVC) sebagai pilihan dalam pengembangan sistem informasi berbasis web [15].



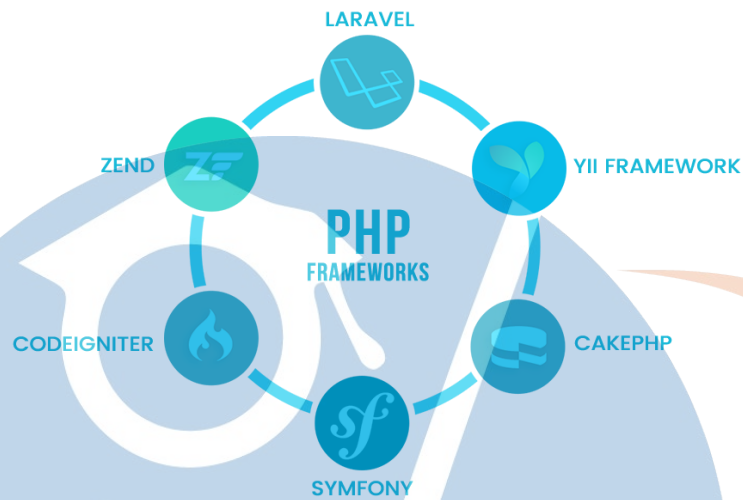
Gambar 2.2 Survei PHP Framework Pola Desain MVC Tahun 2015

2.1.6 Web Framework

Web Framework merupakan kerangka kerja perangkat lunak yang dirancang untuk mendukung pengembangan aplikasi web termasuk web *service*, web *resources*, dan web *API*. Web framework menyediakan cara standar untuk membangun dan mengembangkan aplikasi web, sehingga pekerjaan *developer* lebih fokus dan lebih cepat dalam membangun aplikasi [3].

Web framework merupakan kumpulan fungsi yang memudahkan seorang *developer* aplikasi tidak perlu lagi membuat fungsi-fungsi dari awal. *Developer* cukup memanggil kumpulan fungsi yang sudah ada di dalam framework. Berikut

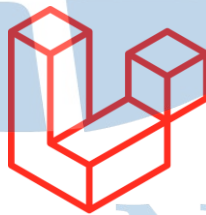
beberapa contoh dari Framework PHP, yaitu Laravel, CodeIgniter, Yii Framework, CakePHP, Symphony, dan Zend [14].



Gambar 2.3 Framework PHP yang Populer

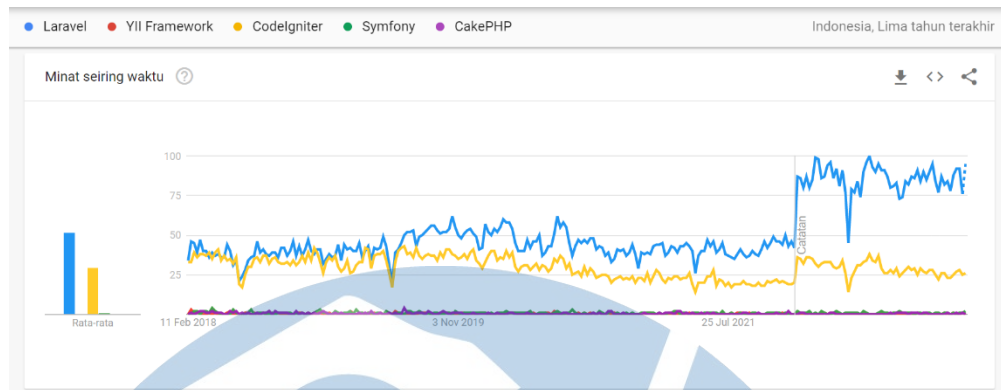
2.1.7 Laravel

Laravel adalah framework berbasis bahasa pemrograman PHP yang dapat digunakan untuk membangun aplikasi web mulai dari proyek berskala kecil (sederhana) hingga besar (kompleks). Laravel pertama kali diciptakan oleh Taylor Otwell pada Juli 2011 [14].



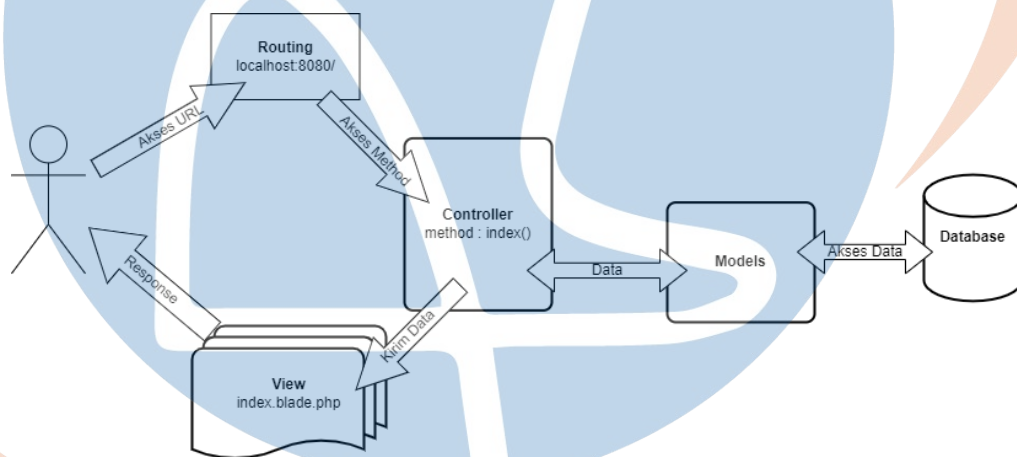
Gambar 2.4 Logo Framework Laravel

Laravel adalah framework *Open Source*. Memiliki seperangkat fitur yang sangat banyak yang akan mempercepat pengembangan website. Dari data Google Trends, menunjukkan bahwa Laravel yang paling populer dicari dan dibaca di Indonesia khususnya.



Gambar 2.5 Survei Pencarian Framework PHP

Laravel merupakan web framework berbasis pemrograman PHP yang menggunakan pendekatan Object Oriented Programming (OOP). Laravel menggunakan pola desain Model View Controller (MVC) [14].



Gambar 2.6 Pola Desain MVC pada Laravel

Pada Framework PHP dengan pola desain *Model View Controller* (MVC) memiliki susunan direktori yang berbeda dari Framework PHP lainnya. Berikut beberapa skema direktori yang dimiliki Framework Laravel [14].

A. App

Direktori yang digunakan Framework Laravel untuk *logical business*, berikut subfolder yang dikenal secara umum.

1. Console

Direktori ini berisi tentang perintah artisan yang dideklarasikan oleh Laravel. Artisan membantu mengelola website, seperti membuat tabel *database*, atau melihat *route* website. Berikut perintah artisan yang umum digunakan.

- php artisan migrate

Perintah ini berfungsi untuk mengeksekusi file *migrations* supaya disimpan ke *database*.

- php artisan serve

Perintah ini berfungsi untuk mengaktifkan *web server* yang disediakan oleh Framework Laravel.

- php artisan make:auth

Perintah ini berfungsi untuk mempermudah dalam pembuatan register dan login di Framework Laravel.

- php artisan make:controller

Perintah ini berfungsi untuk membuat file *controller* di dalam Framework Laravel.

- php artisan make:migration

Perintah ini berfungsi untuk membuat file migration yang digunakan untuk membuat tabel pada *database* menggunakan bahasa pemrograman PHP.

- php artisan make:model

Perintah ini berfungsi untuk membuat *model* di dalam Framework Laravel.

- php artisan make:seeder

Perintah ini berfungsi untuk mengisi data pada suatu tabel.

2. HTTP

Direktori App/Http yang berfungsi untuk menyimpan seluruh proses request dan response. Direktori ini berisi Controllers, Middleware, dan Requests.

- App/Http/Controllers

Direktori ini berfungsi untuk menyimpan seluruh *class Controller* yang dibuat.

- App/Http/Middleware

Direktori ini berfungsi untuk menyimpan class yang berhubungan dengan middleware PHP. Middleware adalah penengah atau sebagai filter request dari user.

- App/Http/Requests

Direktori ini berfungsi untuk berinteraksi dengan permintaan HTTP (GET, POST, Delete dan lain-lain).

3. Models

Direktori yang berfungsi untuk *logical database* yang menghubungkan antara *controller* dengan *database*.

4. Providers

Direktori yang berfungsi sebagai penyedia layanan pada Framework Laravel.

B. Config

Direktori ini berisi file konfigurasi yang dijalankan oleh Framework Laravel.

C. Database

Direktori ini berfungsi untuk membuat database *migrations*, *factories*, dan *seeders*.

- Factories

Direktori yang berfungsi untuk mengisi data dummy suatu tabel dalam *database* yang telah dipilih berdasarkan model.

- Migrations

Direktori yang berfungsi untuk melakukan perubahan pada *database*, baik itu penambahan tabel, penambahan kolom, dan setiap perubahan pada tabel.

- Seeders

Direktori yang berfungsi untuk membuat dummy data pada *database*.

D. Public

Direktori yang seluruh resource aplikasi yang dapat diakses secara penuh oleh *user* melalui *web browser*.

E. Resources

- Views

Direktori ini berfungsi untuk menyimpan seluruh file HTML/Blade yang dibuat pada Framework Laravel.

F. Routes

Direktori ini berfungsi sebagai alur yang akan dilalui oleh user dalam menggunakan Framework Laravel.

- Web.php

Route untuk web didaftarkan melalui file ini, seluruh rute yang digunakan pada Framework Laravel berada pada file *Web.php*.

G. Tests

Direktori yang berfungsi untuk menyimpan seluruh file test yang digunakan untuk menguji Framework Laravel.

H. Vendor

Ketika menginstal *library* menggunakan *composer*, maka akan di install pada vendor.

2.1.8 MySQL

SQL atau singkatan dari “*Structured Query Language*” adalah bahasa standar yang digunakan untuk mengakses *server database*. MySQL merupakan sebuah *server database SQL multi user* dan *multi-thread*. SQL juga salah satu bahasa *database* yang paling populer di dunia. *Database* ini dikembangkan untuk keperluan sistem *database* yang cepat, andal, dan mudah digunakan. Beberapa kelebihan yang dimiliki MySQL sebagai berikut [6] :

1. Portability

Database MySQL memiliki fungsi yang stabil, *database* MySQL juga *support* pada berbagai sistem operasi seperti *Windows, Linux, FreeBSD, Mac OS X Server, Solaris, Amiga, Hp-Unix*, dan lain-lain.

2. Open Source

MySQL merupakan *database open source* (gratis), di bawah lisensi GPL sehingga anda dapat memperoleh dan menggunakannya secara Cuma-Cuma tanpa membayar sepeserpun.

3. Multi User

MySQL juga dapat digunakan untuk menangani beberapa *user* dalam waktu bersamaan tanpa mengalami masalah atau konflik. Sehingga memungkinkan sebuah *database server* MySQL dapat diakses *client* secara bersamaan pula.

4. Column Type

Database MySQL memiliki tipe data yang sangat kompleks, seperti *signed/unsigned integer, float, double, char, varchar, text, blob, data, time, datetime, timestamp, year, set* serta *enum*.

5. Command dan Functions

MySQL *server* memiliki operator dan fungsi yang mendukung perintah *SELECT* dan *WHERE* dalam *query*.

6. Security

Sistem *security* pada MySQL memiliki beberapa lapisan sekuritas seperti level *subnetmask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang mendetail beserta *password* terenkripsi.

7. Scalability dan Limits

MySQL memiliki kemampuan menangani *database* dalam skala cukup besar, dengan jumlah *record* lebih dari 50 juta dan 60 ribu tabel serta 5 miliar baris. MySQL juga dapat menampung indeks sampai 32 indeks pada tiap tabelnya.

8. Connectivity

Adanya kemampuan MySQL melakukan koneksi dengan *client* menggunakan protokol *TCP/IP, Unix Socket (Unix)*, atau *Named Pipes (NT)*.

2.2 Model Pengembangan

2.2.1 Unified Modelling Language (UML)

Unified Modelling Language (UML) adalah sebuah bahasa yang telah menjadi standar dalam industri bahkan dalam pengembang *software* untuk visualisasi, merancang, dan mendokumentasikan sistem *software*.

UML dapat membuat model untuk semua jenis aplikasi *software*, di mana aplikasi tersebut dapat berjalan di *hardware*, sistem operasi, jaringan. UML menggunakan konsep *class* dan *operation* dalam konsep dasarnya sehingga UML lebih cocok untuk bahasa pemrograman berorientasi objek (*Object Oriented Programming*). Walau begitu, UML masih bisa digunakan untuk *modelling* bahasa pemrograman berbasis prosedural.

Dalam *modelling* suatu *software*, maka dibutuhkan diagram yang menggambarkan rancangan dari suatu *software*. UML telah mendefinisikan diagram-diagram yang dibutuhkan sebagai berikut [9]:

1. Use Case Diagram

Use Case Diagram menggambarkan fungsionalitas dari sebuah sistem. *Use Case Diagram* dapat menjelaskan apa yang dilakukan oleh sistem. *Use Case Diagram* membantu dalam menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan *client*, dan merancang *test case* untuk semua fitur yang ada pada sistem.

2. Class Diagram

Class Diagram menggambarkan atribut atau properti dari suatu sistem, sekaligus menyediakan layanan untuk memanipulasi fungsi atau metode tersebut. *Class Diagram* menjelaskan struktur dan deskripsi *class*, *package*, dan objek serta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class* jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek.

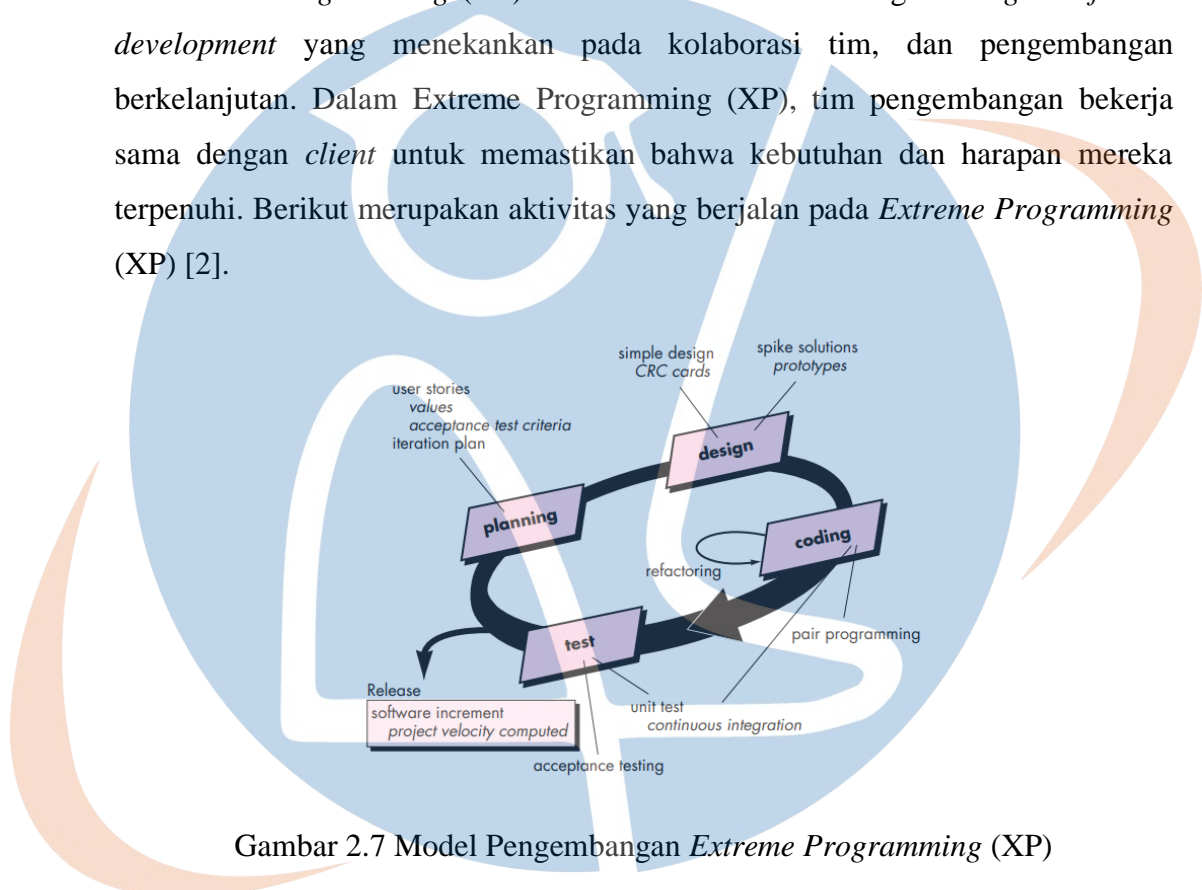
3. Activity Diagram

Activity Diagram menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, *decision* yang terjadi, dan bagaimana sistem berakhir. *Activity Diagram* yang terjadi pada beberapa eksekusi digambarkan dalam proses paralel. *Activity Diagram* menjelaskan proses-proses dan jalur-jalur aktivitas dari level secara umum. Sebuah aktivitas yang direalisasikan oleh satu *use case* atau lebih. Aktivitas

menggambarkan proses yang sedang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem melakukan aktivitas.

2.2.2 Extreme Programming (XP)

Extreme Programming (XP) adalah salah satu metodologi dari *agile software development* yang menekankan pada kolaborasi tim, dan pengembangan berkelanjutan. Dalam *Extreme Programming (XP)*, tim pengembangan bekerja sama dengan *client* untuk memastikan bahwa kebutuhan dan harapan mereka terpenuhi. Berikut merupakan aktivitas yang berjalan pada *Extreme Programming (XP)* [2].



Gambar 2.7 Model Pengembangan *Extreme Programming (XP)*

A. Planning (Perencanaan)

Aktivitas *Extreme Programming (XP)* yang dimulai dengan mendengarkan kebutuhan *client*, sehingga dapat mengumpulkan data dan memahami proses bisnis dari suatu aplikasi. Dan juga dapat memberi pemahaman tentang output yang diperlukan serta fitur dan fungsionalitas dari *software* yang akan dirancang.

B. Design (Perancangan)

Aktivitas *Extreme Programming (XP)* yang memberi gambaran pemodelan sistem, pemodelan arsitektur sampai dengan pemodelan *database* serta prototipe pada aplikasi yang akan dirancang.

C. Coding (Pengkodean)

Aktivitas Extreme Programming (XP) yang mengimplementasikan ke dalam bahasa pemrograman dari hasil desain yang telah dirancang sebelumnya.

D. Testing (Pengujian)

Aktivitas Extreme Programming (XP) yang berfungsi untuk menguji apakah fitur dan fungsionalitas aplikasi yang dirancang sudah sesuai dengan kebutuhan *client*.

2.3 Metode Pengujian

2.3.1 Black Box Testing

Black Box Testing merupakan pengujian yang berfokus pada spesifikasi fungsional *software*, tester dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengetesan pada spesifikasi fungsional program [10].

Pengujian *software* sangat diperlukan untuk memastikan *software* yang sudah/sedang dibuat dapat berjalan sesuai dengan fungsionalitas yang diharapkan. Pengujian *software* merupakan elemen kritis dari jaminan kualitas *software* dan merupakan bagian yang tak terpisahkan dari siklus hidup *software* yang lain seperti analisis, desain, dan pengkodean.

2.3.2 User Acceptance Test

User Acceptance Test atau dikenal dengan Uji Penerimaan Pengguna adalah suatu proses pengujian oleh pengguna yang berfungsi untuk menghasilkan dokumen yang dijadikan bukti bahwa *software* yang telah dikembangkan telah dapat diterima oleh pengguna. Hasil pengujian (*testing*) sudah bisa dianggap apabila memenuhi kebutuhan dari pengguna. Proses *User Acceptance Test* berdasarkan pada dokumen requirement yang disepakati bersama. Dokumen *requirement* adalah dokumen yang berisi lingkup pekerjaan *software* yang harus dikembangkan, sehingga dokumen ini semestinya menjadi acuan untuk pengujian. Proses *User Acceptance Test* dalam pemeriksaan dan pengujian terhadap hasil pekerjaan, diperiksa apakah item-item yang ada dalam dokumen requirement sudah

ada dalam *software* yang diuji atau tidak. Dan diuji apakah semua item yang telah ada, dapat memenuhi kebutuhan penggunanya [12].

2.3.3 Skala Likert

Skala Likert merupakan skala yang berfungsi dalam mengukur sikap, pendapat, dan persepsi seseorang atau orang tentang fenomena social. Dengan Skala Likert, sehingga variabel yang akan diukur bisa dijabarkan menjadi indikator variabel. Sehingga indikator tersebut dijadikan sebagai titik tolak untuk menyusun item-item instrumen berupa pernyataan atau pertanyaan [11].

Dalam skala likert terdapat dua bentuk pertanyaan, yaitu bentuk pertanyaan positif untuk mengukur skala positif, dan bentuk pertanyaan negatif untuk mengukur skala negatif. Jika pertanyaan positif diberi skor 5, 4, 3, 2, dan 1. Dan pertanyaan negatif diberi skor 1, 2, 3, 4, dan 5 [13].

2.4 Penelitian Terkait

2.4.1 Tabel Penelitian Terkait

Penulisan penelitian ini tak lepas dari berbagai inspirasi lain dari penelitian-penelitian sebelumnya yang mempunyai latar belakang yang sama.

Tabel 2.1 Penelitian terkait

No	Judul	Peneliti	Tahun	Kesimpulan
1	Rancang Bangun Aplikasi Sistem Informasi Sekolah <i>Go To School (GOS)</i> Berbasis Web Menggunakan PHP	Chatur Nugroho	2018	Perancangan aplikasi sistem informasi <i>Go To School</i> berbasis website dengan Framework <i>PHP</i> ini sangat efektif dan menjadi alternatif solusi bagi orang tua dalam mencari informasi tentang sekolah secara detail

				dan mendaftarkan anaknya menggunakan fitur pendaftaran online yang tersedia dalam aplikasi <i>Go To School</i> .
2	Analisa dan Perancangan Sistem Informasi Akademik Berbasis Web pada MI Al-Mursyidiyah Al-‘Asyirotussyafi’iyah	Khaerul Anam	2018	Sistem Informasi akademik yang dirancang dapat membantu dan mempercepat proses pencatatan data guru, siswa, kelas, pembuatan jadwal pelajaran dan penilaian menggunakan sistem akademik berbasis web.
3	Pemodelan Sistem Informasi Akademik Menggunakan <i>Extreme Programming</i> pada Madrasah Aliyah (MA) Mambaul Ulum Tanggamus	Nita Ayunandita	2021	Hasil pengujian teknologi Sistem Informasi Akademik pada Madrasah Aliyah Mambaul Ulum Tanggamus menggunakan <i>Technology Acceptance Model (TAM)</i> didapatkan hasil bahwa 85%

				<p>pengguna setuju bahwa Sistem Informasi Akademik pada Madrasah Aliyah Mambaul Ulum Tanggamus memiliki kegunaan atau bermanfaat, kemudian 85% pengguna setuju bahwa aplikasi tersebut mudah untuk digunakan, selanjutnya 86% pengguna sudah berniat atau tertarik untuk menggunakan Sistem Informasi Akademik pada Madrasah Aliyah Mambaul Ulum Tanggamus, dan 76% pengguna setuju untuk sungguh-sungguh menggunakan Sistem Informasi Akademik pada Madrasah Aliyah Mambaul Ulum Tanggamus.</p>
4	Aplikasi Penerimaan Peserta Didik Baru Berbasis Web pada	Najamudin	2019	Sistem Informasi Penerimaan Peserta Didik Baru yang

	SMK Negeri 2 Kuripan		dirancang Berbasis Web di SMKN 2 Kuripan mampu mempermudah calon siswa baru untuk melakukan pendaftaran dimana saja selama terkoneksi internet dan diakses sesuai dengan jadwal yang sudah ditentukan. Menu yang ada pada sistem ini dibuat untuk mempermudah calon siswa baru dalam melakukan pendaftaran.
--	-------------------------	--	---

STT - NF

2.4.2 Posisi Penelitian

Pada penelitian ini posisi penelitian yang dilakukan di antara penelitian yang telah dilakukan sebelumnya, tertera pada tabel 2.1.

Tabel 2.2 Posisi Penelitian

Manajemen Proyek	Web	MySQL	PHP Framework	Extreme Programming
Rancang Bangun Aplikasi Sistem Informasi Sekolah Go To School (GOS) Berbasis Web Menggunakan PHP				
Analisa dan Perancangan Sistem Informasi Akademik Berbasis Web pada MI Al-Mursyidiyah Al-'Asyirotusyafi'iyah				
Pemodelan Sistem Informasi Akademik Menggunakan Extreme Programming pada Madrasah Aliyah (MA) Mambaul Ulum Tanggamus				
Aplikasi Penerimaan Peserta Didik Baru Berbasis Web pada SMK Negeri 2 Kuripan				
Implementasi Pola Desain MVC pada Rancang Bangun Sistem Informasi Pendaftaran Mahasantri Berbasis Web Studi Kasus di Pesantren Teknologi Informasi dan Komunikasi				

STT - NF