

BAB II LANDASAN TEORI

2.1 Tinjauan Pustaka

Untuk mendukung penelitian ini, Penulis menggunakan beberapa *Literature* yang berkaitan dengan judul dan pokok bahasan pada penelitian. Adapun *Literature* yang dipergunakan dapat ditinjau pada Tabel 2.1.

Tabel 2.1 *Literature*

No	Penulis, Tahun	Judul	Metode	Hasil
01	(Firmansyah Maulana Sugiartana Nursuwars, 2018)	<i>API (Application Programming Interface) Mikrotik Untuk Otentikasi Sistem Akademik Universitas Siliwangi</i>	Metode penelitian yang digunakan yaitu: Metode XP (<i>Extreme Programming</i>)	Sistem <i>otentikasi</i> dengan memanfaatkan <i>API MikroTik</i> untuk dapat mengakses Sistem Akademik Universitas Siliwangi pengguna tidak perlu memasukan kembali <i>username</i> dan <i>password</i> .
02	(Reyhan Nafis Ibrahim, Aziz Musthafa dan Oddy Virgantara Putra, 2021)	Rancang Bangun Sistem Monitoring Aktivitas Pengguna <i>Hotspot</i> UNIDA Gontor Menggunakan <i>Rsyslog</i> dan <i>Mikrotik API</i>	Menggunakan metode NDLC (<i>Network Development Life Cycle</i>)	Sistem ini dapat melihat dan merekam riwayat penelusuran pengguna <i>Hotspot</i> dan menyaring pengguna yang mengakses situs terlarang

Tabel 2.1 (Lanjutan)

No	Penulis, Tahun	Judul	Metode	Hasil
03	(Yoga Hilmi Tasanah Assakur, Mohamad Seh Fahrudin dan Ferdiansyah, 2020)	Implementasi <i>API Mikrotik</i> untuk Management Router Berbasis <i>Android</i>	Menggunakan Metode Eksperimental dan Metode Demonstrasi	Aplikasi ini dapat digunakan untuk membuat tabel <i>routing</i> , <i>monitoring traffic</i> , menampilkan <i>resource</i> dari <i>Mikrotik</i> serta <i>memonitoring</i> jaringan menggunakan <i>ping</i> .
04	(Kadek Juni Arta, Nyoman Bagus Suweta Nugraha, 2020)	Implementasi Aplikasi User Management <i>Hotspot</i> <i>Mikrotik</i> Berbasis <i>PHP</i> Dengan <i>Application</i> <i>Programming</i> <i>Interface (API)</i> Dan Framework <i>Bootstrap</i>	Menggunakan metode NDLC (<i>Network</i> <i>Development Life</i> <i>Cycle</i>)	Aplikasi ini dapat mempermudah admin yang tidak bisa mengoperasikan <i>Mikrotik</i> secara langsung, sehingga siapa saja pemegang hak akses dapat menambah dan menghapus data pada Aplikasi.

Tabel 2.1 (Lanjutan)

No	Penulis, Tahun	Judul	Metode	Hasil
05	(Andriansyah Zakaria, Andesita Prihantara dan Antonius Agung Hartono, 2019)	Integrasi <i>Application Programming Interface, PHP, dan MySQL</i> untuk Otomatisasi <i>Verifikasi dan Aktivasi Pengguna Layanan Hotspot MikroTik</i>	Metode penelitian yang digunakan adalah Deskriptif Kualitatif	Pendaftaran secara mandiri dan otomatisasi aktivasi dapat mempermudah dan mempercepat proses pelayanan dalam upaya peningkatan pelayanan <i>Hotspot</i> internet kampus untuk <i>Mahasiswa</i>
06	(Assyahid Hasan Albana, Ernawati dan Funny Farady Coastera, 2020)	Sistem Informasi <i>Monitoring User Manager Dan Voucher Payment Mikrotik Router</i> Melalui <i>Routers Api Class</i> Berbasis <i>Web Mobile</i>	Menggunakan metode NDLC (<i>Network Development Life Cycle</i>)	Penelitian ini telah berhasil mengimplementasikan sistem jaringan untuk melakukan <i>Monitoring</i> , membagi <i>bandwidth</i> dan usaha <i>voucher</i> menggunakan <i>API Router Mikrotik</i> melalui <i>Smartphone</i> berbasis <i>Android</i>

Tabel 2.1 (Lanjutan)

No	Penulis, Tahun	Judul	Metode	Hasil
07	(Muhamad Fuat Asnawi, 2018)	Aplikasi Konfigurasi <i>Mikrotik</i> Sebagai Manajemen <i>Bandwidth</i> Dan <i>Internet Gateway</i> Berbasis <i>Web</i>	Metode penelitian yang digunakan yaitu menggunakan metode <i>Prototyping</i>	Sistem mudah digunakan dalam melakukan konfigurasi router sesuai dengan permasalahan yang sering dihadapi mengenai <i>router mikrotik</i> yang digunakan sebagai manajemen <i>bandwidth</i> dan <i>Internet Gateway</i> .
08	(Muhammad Syahal Setiawan, Bayu Adhi Prakosa dan Ade Hendri Hendrawan, 2018)	Sistem Pendaftaran <i>Hotspot</i> Menggunakan <i>Router Mikrotik</i> Berbasis <i>Web</i> (Studi Kasus Fakultas Teknik)	Metode penelitian yang digunakan yaitu metode <i>Waterfall</i>	Pendaftaran <i>hotspot</i> secara mandiri berbasis <i>web</i> memberikan kemudahan kepada civitas akademi selaku pengguna layanan <i>hotspot</i> dalam proses mendapatkan akun akses layanan <i>hotspot</i> .

Tabel 2.1 (Lanjutan)

No	Penulis, Tahun	Judul	Metode	Hasil
09	(Muhammad Hanif, 2021)	Sistem <i>Monitoring</i> Trafik Pada <i>Mikrotik</i> berbasis <i>App Mobile</i> Dengan Notifikasi <i>Telegram</i>	Menggunakan metode NDLC (<i>Network Development Life Cycle</i>)	Sistem <i>monitoring</i> trafik pada <i>mikrotik</i> , membantu admin jaringan, <i>web server</i> dalam pemantauan infrastruktur pada OPD/SKPD. Sehingga dalam proses penanganan gangguan dapat dengan mudah dilakukan
10	(Rinanza Zulmy Alhamri, Toga Aldila Cinderatama, Kunti Eliyen dan Agustono Heriadi, 2021)	Pengembangan Aplikasi <i>Monitoring</i> Jaringan Berbasis <i>Android</i> Studi Kasus Puskesmas PSDKU Polinema di Kota Kediri	Metode yang digunakan <i>Prototyping</i>	Adapun fungsi-fungsi yang telah dikembangkan untuk memudahkan <i>istrator</i> dalam memantau jaringan meliputi <i>istrator</i> dapat mengaktifkan aplikasi <i>agen</i> berbasis <i>web</i> , dapat melakukan <i>login</i> pada aplikasi <i>monitoring</i> jaringan berbasis <i>android</i>

2.1.1 Tinjauan Terhadap *Literature*

Nursuwars, (2018) Pada penelitian ini yang berjudul *API Mikrotik Untuk Otentikasi Sistem Akademik Universitas Siliwangi*. Penelitian yang dilakukan penulis untuk mengatasi permasalahan banyaknya *username* dan *password* untuk satu orang pengguna dan menghindarkan tertukarnya *username* dan *password* maka dibuat sistem *otentikasi* dengan memanfaatkan *API MikroTik*, Dengan demikian penulis menggunakan metode *XP (Extreme Programming)* dan hasil kesimpulan sistem yang telah dibuat pada penelitian ini adalah sistem dengan memanfaatkan *API MikroTik* sehingga pengguna tidak memiliki tambahan *username* dan *password* untuk sistem *one access* ini. Pengguna hanya perlu melakukan aktivasi pada Sistem Akademik Universitas Siliwangi untuk dapat menggunakan layanan.

Ibrahim et al., (2021) Penelitian yang berjudul *Bangun Sistem Monitoring Aktivitas Pengguna Hotspot UNIDA Gontor Menggunakan Rsyslog dan Mikrotik API*, Penelitian yang dilakukan penulis untuk membaangun sistem *monitoring* untuk pemantauan yang lebih *intens* terkait aktivitas *internet* mahasiswa selama menggunakan layanan internet kampus. Dalam menganalisis dan mengimplementasikan sistem, penelitian ini menggunakan metode *NDLC (Network Development Life Cycle)* dan hasil kesimpulan pembuatan sistem ini adalah dapat melihat dan merekam riwayat penelusuran pengguna *hotspot* dan menyaring pengguna yang mengakses situs terlarang. Sehingga pengguna yang mengakses situs yang dilarang akan masuk ke dalam daftar *log* pengguna yang mengakses situs terlarang dan mendapatkan peringatan. agar dapat melakukan perbaikan diri.

Assakur et al., (2020) Pada penelitian yang berjudul *Implementasi API Mikrotik untuk Management Router Berbasis Android*. Penelitian yang dilakukan penulis untuk mencoba melakukan implementasi API mikrotik yang didapat dari *library* API milik *mikrotik*. Sehingga dapat digunakan untuk mengakses dan melakukan konfigurasi pada *mikrotik* menggunakan perangkat *device mobile*. Dengan demikian penulis menggunakan metode *Eksperimental* dan metode *Demonstrasi*. Kesimpulan hasil akhir dari penelitian yaitu aplikasi ini mencakup status dari *mikrotik router* yang sedang diakses, konfigurasi dasar *mikrotik*, manajemen bandwidth, dan fungsi ping untuk memeriksa kualitas jaringan berdasarkan IP *address*. dengan fitur-fitur utama yang telah tersedia di aplikasi *mobile*, konfigurasi dan *troubleshooting* router *mikrotik* akan memudahkan pekerjaan dari *istrator* jaringan pada perusahaan.

Arta & Nugraha, (2020) Pada penelitian ini yang berjudul *Implementasi Aplikasi User Management Hotspot Mikrotik Berbasis PHP Dengan Application Programing Interface (API) Dan Framework Bootstrap*. Penelitian yang dilakukan penulis bertujuan untuk mempermudah proses belajar mengajar dan dikarenakan pesatnya perkembangan teknologi *hotspot* maka diperlukan sebuah alat bernama *router* untuk mengatur jalanya *hotspot* dan aplikasi *interfacing* agar lebih mudah dalam pengoprasian sistem *router mikrotik*. Dengan demikian penulis menggunakan metode yang digunakan dalam perancangan dan pembuatan perangkat lunak ini adalah metode NDLC (*Network Development Life Cycle*) serta menghasilkan kesimpulan dari penelitian ini adalah dapat menanggulangi permasalahan pada *hotspot* yang biasanya internetnya lambat ketika jumlah pengguna yang terkoneksi banyak, serta dapat memberikan kemudahan kepada *admin* yang tidak bisa mengoperasikan

mikrotik secara langsung, sehingga siapa saja pemegang hak akses dapat menambah dan menghapus data pada aplikasi.

Zakaria et al., (2019) Pada penelitian yang berjudul *Integrasi Application Programming Interface, PHP, dan MySQL untuk Otomatisasi Verifikasi dan Aktivasi Pengguna Layanan Hotspot MikroTik*. Penelitian yang dilakukan penulis untuk mengatasi permasalahan pengelola *hotspot* sering disibukan dengan pendaftaran, *verifikasi*, dan aktivasi akun pengguna *hotspot* secara manual, hal ini tentunya akan mengakibatkan terhambatnya proses layanan kepada mahasiswa dengan semakin banyaknya pendaftar, Dengan demikian penulis menggunakan metode *deskriptif kualitatif* serta hasil kesimpulan yang didapat dari penelitian ini adalah mempermudah pekerjaan pengelola internet kampus, sehingga pengelola bisa melakukan aktivitas lain tanpa disibukan dengan urusan *verifikasi* dan *aktivasi* manual pengguna *hotspot*. Albana et al., (2020) Pada penelitian yang berjudul *Sistem Informasi Monitoring User Manager Dan Voucher Payment Mikrotik Router Melalui Routeros Api Class Berbasis Web Mobile*. Penelitian yang dilakukan penulis bertujuan untuk pemilik usaha ruko queen yang berencana untuk menyediakan fasilitas internet yang berukuran besar dan cepat sebagai usaha peningkatan pengunjung. Dengan demikian penulis menggunakan metode NDLC (*Network Development Life Cycle*) serta kesimpulan hasil dari penelitian ini adalah telah berhasil mengimplementasikan sistem jaringan untuk melakukan *monitoring*, membagi *bandwidth* dan usaha *voucher* menggunakan API *router mikrotik* melalui *smartphone* berbasis *android*, serta dapat melakukan *payment transaction* melalui aplikasi *telegram*. Dengan adanya transaksi melalui *telegram*, dapat dengan mudah memeriksa pengunjung untuk membeli *voucher* dapat lebih efisien.

Asnawi, (2018) Pada penelitian yang berjudul Aplikasi Konfigurasi *Mikrotik* Sebagai Manajemen *Bandwidth* Dan *Internet Gateway* Berbasis *Web*. Penelitian yang dilakukan penulis bertujuan untuk mempermudah dalam mengkonfigurasi sebuah *router mikrotik* tanpa harus hafal dan paham dengan perintah-perintah CLI *mikrotik* tersebut. Sehingga diharap dapat menggunakannya untuk mengkonfigurasi *router mikrotik* sebagai manajemen *bandwidth client* dan *internet gateway* untuk keperluan share koneksi internet. Dengan demikian penulis menggunakan metode *Prototyping* serta hasil kesimpulan dari penelitian ini adalah *istrator* jaringan yang menggunakan *router mikrotik* lebih mudah menggunakan aplikasi ini dalam melakukan konfigurasi *router* sesuai dengan permasalahan yang sering dihadapi mengenai *router mikrotik* yang digunakan sebagai manajemen *bandwidth* dan *internet gateway* pada Jaringan Komputer yang sudah ada. Setiawan et al., (2018) Pada penelitian yang berjudul Sistem Pendaftaran *Hotspot* Menggunakan *Router Mikrotik* Berbasis *Web*. Penelitian yang dilakukan penulis bertujuan untuk memberikan solusi *web* pendaftaran *hotspot* untuk memberi kemudahan kepada pengguna pada saat yang akan membuat akun *login* layanan *hotspot*, serta mempermudah tugas *istrator* untuk mengatur jaringan *hotspot*. Dengan demikian penulis menggunakan metode *Waterfall* serta hasil kesimpulan yang didapat dari penelitian ini adalah Dengan adanya aplikasi pendaftaran *hotspot* secara mandiri dan aplikasi *backend mikrotik* berbasis *web*, memudahkan kerja *istrator* dalam verifikasi data pengguna yang telah mendaftar akun *hotspot* serta memudahkan proses pembuatan akun *hotspot* pada *server hotspot mikrotik* tanpa harus melakukan konfigurasi di perangkat *router mikrotik*.

Hanif & Kamisutara, (2021) Pada penelitian yang berjudul Sistem *Monitoring* trafik Pada *Mikrotik* berbasis App *Mobile* Dengan Notifikasi *Telegram*. Penelitian yang dilakukan penulis bertujuan untuk menghasilkan sistem *monitoring trafik* jaringan dan *web server* secara *realtime* dengan notifikasi *telegram*, untuk mencegah terjadinya kendala serta mempercepat proses penanganan kendala pelayanan *publik* pada OPD/SKPD. Dengan demikian penulis menggunakan Metode NDLC (*Network Development Life Cycle*) serta hasil kesimpulan yang didapat dari penelitian ini adalah Sistem *monitoring trafik* pada *mikrotik*, membantu jaringan, *admin web server* dalam pemantauan infrastruktur pada OPD/SKPD. Sehingga dalam proses penanganan gangguan dapat dengan mudah dilakukan. Notifikasi yang diterima oleh jaringan dan *server*, berupa notifikasi *telegram* dan notifikasi pada *website* berupa *trafik* jaringan dan *web server*. Alhamri et al., (2021) Pada penelitian yang berjudul Pengembangan Aplikasi *Monitoring* Jaringan Berbasis *Android* Studi Kasus Puskom PSDKU Polinema di Kota Kediri. Penelitian yang dilakukan penulis bertujuan untuk mengembangkan aplikasi *monitoring* jaringan berbasis *android* yang dapat memanfaatkan *mikroTik* API serta basis data berbasis *cloud firebase* sehingga kegiatan *monitoring* bisa dilakukan secara fleksibel. Dengan demikian penulis menerapkan metode pengembangan perangkat lunak *prototyping* serta hasil kesimpulan dari penelitian ini adalah adapun fungsi-fungsi yang telah dikembangkan untuk memudahkan *istrator* dalam memantau jaringan meliputi *istrator* dapat mengaktifkan aplikasi *agen* berbasis *web*, dapat melakukan *login* pada aplikasi *monitoring* jaringan berbasis *android*, dapat melihat data *log* aktivitas *router*, melihat *traffic* pada *interface router*, serta dapat melihat kondisi internet dan notifikasi.

2.2 Mikrotik

Mikrotik merupakan sistem operasi *RouterOS* berlisensi dengan kernel *Linux* yang dapat menjalankan hampir semua konfigurasi dan manajemen jaringan internet. Sistem operasi *RouterOS* dapat diinstal pada komputer yang kemudian dapat difungsikan sebagai router. *Mikrotik* didesain untuk mudah digunakan dan sangat baik digunakan untuk keperluan *istrasi* jaringan komputer seperti merancang dan membangun sebuah sistem jaringan *komputer* skala kecil hingga yang kompleks sekalipun (Jayanto, 2019).

2.3 API Mikrotik

(*Application Programing Interface*) API *mikrotik* merupakan perintah pemrograman yang memungkinkan perangkat lunak pengguna berkomunikasi dengan *RouterOS* untuk mendapatkan informasi, konfigurasi dan manajemen perangkat tersebut. Pada dasarnya *RouterOS* bisa dikonfigurasi melalui perintah langsung berupa *CLI (command line interface)* atau melalui *GUI (graphical user interface)* dengan bantuan aplikasi *Winbox*. *MikroTik API* mendukung bahasa pemrograman seperti *PHP, Python, C*, dan lain-lainnya (Zakaria et al., 2019).

2.4 PPPOE

PPPoE digunakan untuk membangun *virtual private network (VPN)* dimana koneksinya menggunakan *tunnel point-to-point*. *PPPoE* Merupakan *protokol tunneling*, memiliki keamanan yang sangat baik, memerlukan beberapa otentikasi untuk terhubung, keamanan data dalam hal ini adalah prioritas utama, proses enkapsulasi dan otentikasi membuat *tunneling* menjadi salah satu dari banyak cara untuk mengamankan data di jaringan. Selain keamanan, kinerja jaringan merupakan

faktor yang harus diperhatikan agar dapat memberikan layanan yang baik (Slameto & Hidayat, 2019).

2.5 Monitoring

Monitoring adalah suatu proses pengumpulan dan menganalisis informasi dari penerapan suatu program termasuk mengecek secara reguler untuk melihat apakah kegiatan (*program*) itu berjalan sesuai rencana sehingga masalah yang dilihat (ditemui) dapat diatasi. merupakan tahapan yang berkaitan erat dengan kegiatan *monitoring*, karena kegiatan *evaluasi* dapat menggunakan data yang disediakan melalui kegiatan *monitoring*. Dalam merencanakan suatu kegiatan hendaknya *evaluasi* merupakan bagian yang tidak terpisahkan, sehingga dapat dikatakan sebagai kegiatan yang lengkap. Evaluasi diarahkan untuk mengendalikan dan mengontrol ketercapaian tujuan (Samsir & Sitorus, 2021).

2.6 Web Server

Web Server sebuah aplikasi yang berfungsi sebagai *server* untuk memberikan layanan berbasis data, menggunakan protokol *HTTP* atau *HTTPS*. *Request* data dari *client* menggunakan aplikasi *web browser* dan *server* akan mengirim data dalam bentuk halaman *web* yang pada umumnya berbentuk dokumen *HTML* merupakan definisi dari *web browser* halaman web yang dikirim dari *web server* bisa terdiri dari berkas teks berupa file *HTML* dan *CSS*, video, gambar, dan banyak jenis file yang lainnya yang nantinya di-parsing atau ditata oleh *web browser* sehingga menjadi halaman *web* yang bisa dimengerti oleh pengguna (Nursuwars, 2018).

2.7 PHP

PHP merupakan secara umum dikenal sebagai bahasa pemrograman *script* yang membuat dokumen *HTML* secara *on the fly* yang dieksekusi di *server web*. Dokumen *HTML* yang dihasilkan dari suatu aplikasi bukan dokumen *HTML* yang dibuat dengan menggunakan *editor* teks atau *editor HTML*. Dikenal juga sebagai bahasa pemrograman *server side* (Arta & Nugraha, 2020).

2.8 Aplikasi

Aplikasi merupakan penerapan, menyimpan sesuatu hal, data, permasalahan, pekerjaan ke dalam suatu sarana atau media yang dapat digunakan untuk diterapkan menjadi sebuah bentuk yang baru. Pengertian aplikasi secara umum adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya aplikasi merupakan suatu perangkat komputer yang siap pakai bagi *user* (Siregar et al., 2018).

2.9 XAMPP

Xampp merupakan perangkat lunak bebas yang mendukung banyak sistem operasi, merupakan kumpulan dari beberapa *program*. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*) yang terdiri atas *Apache HTTP Server*, *Mysql database* dan penerjemahan bahasa yang ditulis dengan bahasa pemrograman *PHP*. Nama *xampp* merupakan singkatan dari empat sistem operasi yaitu *Apache*, *Mysql*, *PHP*, dan *Perl*. Program ini tersedia dalam *GNU* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis (Sarwindah, 2018).

2.10 CodeIgniter

CodeIgniter adalah Sebuah *framework php* yang bersifat *open source* dan menggunakan metode *MVC (Model, View, Controller)* untuk memudahkan *developer* atau *programmer* dalam membangun sebuah aplikasi berbasis web tanpa harus membuatnya dari awal. *Framework* merupakan kumpulan intruksi-intruksi yang dikumpulkan dalam *class* dan *function-function* dengan fungsi masing-masing untuk memudahkan *developer* dalam memanggilnya tanpa harus menuliskan *syntax* program yang sama berulang-ulang serta dapat menghemat waktu (Sallaby & Kanedi, 2020).

Kelebihan *Codeigniter*:

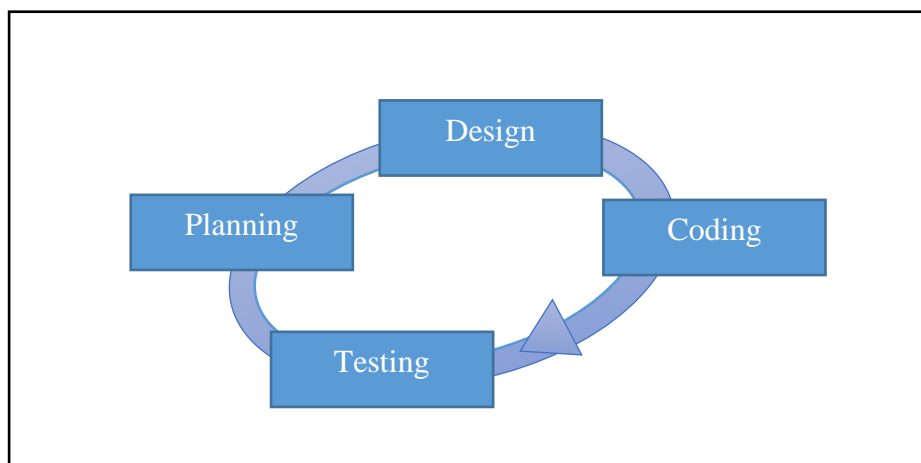
1. Performa yang cepat *Framework* ini diklaim memiliki performa yang cukup cepat dibandingkan *framework-framework* seperti *Cake*, *Symfony*, *Yii*, dan *Laravel*. Oleh sebab itu dari keunggulan ini, para *web developer* memilih untuk menggunakan *codeIgniter* dibandingkan dengan *framework PHP* lain.
2. Konfigurasi yang sederhana pada *codeIgniter*, para *web developer* dapat melakukan konfigurasi file dengan lebih sederhana. Konfigurasi pada *CodeIgniter* yang sederhana merupakan keunggulan lain yang dimiliki oleh *framework PHP* ini. Hanya perlu melakukan perubahan *simpel* pada *config* jika ingin melakukan konfigurasi.
3. Banyak komunitas dari sekian banyaknya komunitas *codeIgniter*, para *web developer* dapat bertukar pikiran, berbagi informasi, serta mendapatkan wawasan baru terkait dengan *framework PHP* yang satu ini.

4. Dokumentasi yang lengkap terdapat dokumentasi yang cukup lengkap jika ingin melakukan instalasi pada *codeIgniter*. Dokumentasi tersebut berupa *user manual* yang lengkap dan dapat dengan mudah dipahami.

Kekurangan *Codeigniter*:

1. *CodeIgniter* tidak ditujukan untuk pembuatan *web* dengan skala besar.
2. *Library* yang sangat terbatas. Hal ini dikarenakan sangat sulit mencari *plugin* tambahan yang terverifikasi secara resmi, karena pada situsnya *codeIgniter* tidak menyediakan *plugin-plugin* tambahan untuk mendukung pengembangan aplikasi dengan CI.
3. Belum adanya editor khusus *codeIgniter*, sehingga dalam melakukan *create project* dan modul-modulnya harus berpindah-pindah *folder*.

2.11 Extreme Programming



Gambar 2.1 *Extreme Programming*

Extreme Programming juga dikenal sebagai XP, adalah salah satu dari banyak metodologi rekayasa perangkat lunak dan bagian dari metodologi

pengembangan perangkat lunak *agile* (Novitasari et al., 2021). Empat nilai kunci yang sangat mendasar dan menjadi ciri khas metode XP, yaitu:

1. *Planning* (Perencanaan)

Tahap ini merupakan tahap pertama dari proses pengembangan sistem dimana pada tahap ini dilakukan beberapa perencanaan yaitu identifikasi masalah, analisis kebutuhan untuk menentukan jadwal pelaksanaan pembangunan sistem.

2. *Design* (Perancangan)

Tahap selanjutnya adalah perancangan dimana pada tahap ini dilakukan kegiatan pemodelan mulai dari pemodelan *sistem*, pemodelan arsitektur hingga pemodelan *database*. Pemodelan sistem dan arsitektur menggunakan diagram *Unified Modeling Language* (UML) sedangkan pemodelan *database* menggunakan *Entity Relationship Diagram* (ERD).

3. *Coding* (Pengkodean)

Fase ini merupakan aplikasi dari operasi pemodelan yang telah diimplementasikan dalam bentuk antarmuka pengguna dengan menggunakan bahasa pemrograman. Bahasa pemrograman yang digunakan adalah *Visual Basic*. Untuk sistem pengelolaan *database* menggunakan *software MySQL*.

4. *Testing* (Pengujian)

Setelah tahap *coding* selesai dilakukan tahap pengujian sistem untuk mengetahui kesalahan apa saja yang terjadi pada saat aplikasi sedang berjalan dan apakah sistem yang dibangun sudah sesuai dengan kebutuhan pengguna.

2.12 Metode ISO/IEC 25010

Tangkudung et al., (2019) mengungkapkan bahwa metode ISO/IEC 25010 ialah komponen dari *Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – Systems and software quality models*. ISO/IEC 25010 ialah pengembangan dari ISO/IEC 9126, serta secara resmi membatalkan dan menggantikan ISO / IEC 9126:2001. Edisi pertama ISO/IEC 25010:2011 yang sudah direvisi secara teknis, menjadi standar internasional terupdate dan relevan untuk menguji sistem informasi yang akan dimaksimalkan.

ISO/IEC 25010 menjadi standar tolak ukur analisis mutu perangkat lunak yang dipakai oleh perusahaan, instansi, ataupun organisasi. Dengan ISO/IEC 25010 evaluasi mutu sistem perangkat lunak dapat dijalankan secara spesifik berdasarkan dimensi *product quality* yang terdiri dari 8 karakteristik ialah *functional suitability, performance efficiency, compatibility, usability, reliability, security, maintainability, dan portability*. (*International Organization for Standardization and International Electrotechnical Commission, 2011*).