

BAB II TINJAUAN PUSTAKA

2.1 Lembaga Sertifikasi Profesi

Lembaga Sertifikasi Profesi (LSP) adalah lembaga pelaksanaan kegiatan sertifikasi profesi yang memperoleh lisensi dari Badan Nasional Sertifikasi Profesi (BNSP). LSP adalah lembaga yang ditunjuk oleh BNSP untuk melakukan sertifikasi profesi sesuai dengan standar kompetensi yang telah ditetapkan oleh BNSP. Dengan kata lain, LSP adalah lembaga yang bertanggung jawab untuk melakukan proses penilaian dan pemberian sertifikat kepada seseorang yang memenuhi kriteria kompetensi yang telah ditetapkan oleh BNSP (Alamsyah *et al.*, 2020).

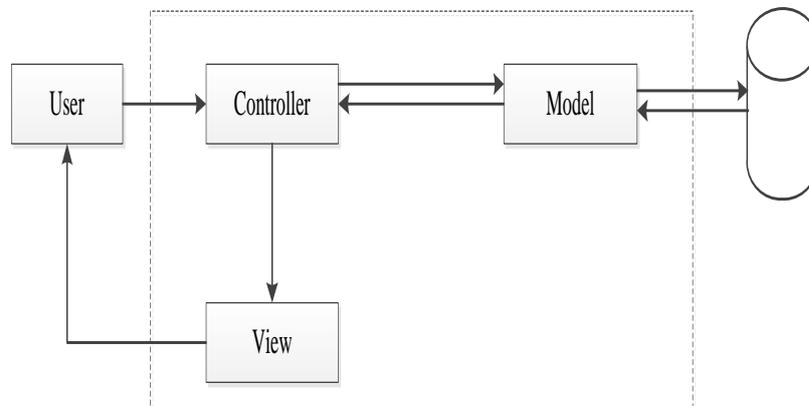
LSP membantu industri dalam melakukan seleksi calon tenaga kerja berbasis kompetensi; Membantu industri meningkatkan produktivitas kerja bagi para pekerjanya; Membantu meningkatkan kredibilitas/tingkat kepercayaan masyarakat terhadap industri (Akhmad, Sibuea and Nurdianita, 2022).

2.2 CodeIgniter

CodeIgniter adalah *Framework* untuk bahasa pemrograman PHP, yang dibuat Rick Ellis pada tahun 2006. *CodeIgniter* memiliki banyak fitur yang membantu para pengembang PHP untuk dapat membuat aplikasi secara mudah dan cepat serta memiliki sifat yang fleksibel dapat dikembangkan dalam perangkat *web*, *desktop* maupun *mobile* (Raharjo, 2018).

Codeigniter adalah sebuah aplikasi gratis yang berupa kerangka kerja untuk membangun website menggunakan bahasa pemrograman PHP (Heru, 2018). *Framework* merupakan kumpulan dari fungsi atau prosedur dan class untuk tujuan tertentu yang sudah siap untuk digunakan untuk membuat program tanpa harus membuat fungsi atau class dari awal. *Framework codeIgniter* dipilih untuk meningkatkan kualitas dari pengembangan dan kode sistematisasi (Widaningsih dan Suheri, 2019).

CodeIgniter memiliki konsep atau pola *Model-View-Controller* (MVC) sehingga kode-kode dapat di sederhanakan.



Gambar 2.1 Arsitektur MVC

Arsitektur MVC dapat dijelaskan sebagai berikut:

1. Model merupakan model adalah kode yang bertugas untuk membuat pemodelan data menuju penyimpanan data.
2. View merupakan struktur bagian yang menangani presentasi logika dan berfungsi untuk menampilkan informasi form dan konten.
3. Controller merupakan penghubung model dan view yang memproses permintaan dari sisi klien ke server dan perintah atau fungsi.

2.3 Web Mobile

Mobile web adalah halaman HTML berbasis *browser* yang diakses menggunakan perangkat *portable* (*smartphone* atau *tablet*) melalui jaringan seluler seperti 3G, 4G maupun *Wifi*. *Mobile Web* dirancang untuk menampilkan data seperti teks, gambar, dan video dari sebuah *website* kedalam sebuah tampilan yang lebih kecil yakni perangkat *mobile*. *Mobile Web* merupakan *web* yang dapat berjalan di *browser* pada telepon genggam (Alam and Rusdiyanto, 2020).

2.4 Website

Website merupakan halaman yang menampilkan informasi data teks, gambar, suara, video atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis. Halaman pertama sebuah *website* disebut *homepage*. *Website/Situs* merupakan kumpulan informasi atau kumpulan halaman/*page* yang bisa diakses

lewat jalur *internet*. Setiap orang di berbagai tempat dan segala waktu bisa menggunakannya selama terhubung secara *online* (Harianto, *et al.*, 2019).

World Wide Web (WWW) atau sering disebut Web merupakan salah satu sumber daya internet yang berkembang pesat. Informasi Web didistribusikan dengan pendekatan hyperlink yang memungkinkan seseorang memperoleh informasi dengan meloncat dari satu halaman ke halaman lain (Juliany, *et al.*, 2018).

2.5 PHP

PHP (*Personal Home Page*) adalah pemrograman (interpreter) yang melakukan proses penerjemahan baris sumber menjadi kode mesin yang dimengerti oleh komputer secara dinamis. Pengertian PHP juga merupakan singkatan dari *Hypertext Preprocessor* dengan Bahasa yang berbentuk skrip yang bersifat *server side* yang dimana proses pengerjaan kode program dilakukan di *server*, dan hasilnya akan ditampilkan di *browser* (Sihombing, 2022).

2.6 MySql

MySQL adalah sebuah *database management system* (manajemen basis data) menggunakan perintah dasar SQL (*Structured Query Language*) yang cukup terkenal. *Database management system* (DBMS) MySQL multi pengguna dan bersifat gratis. Mysql digunakan sebagai wadah dalam mengelola data yang dapat disimpan didigunakan kembali dengan cara yang lebih efisien (Setyawan and Pratiwi, 2019).

2.7 Penjadwalan

Penjadwalan adalah aktivitas perencanaan untuk menentukan kapan dan di mana setiap operasi sebagai bagian dari pekerjaan secara keseluruhan harus dilakukan pada sumber daya yang terbatas. Penjadwalan adalah tindakan menetapkan sumber daya untuk melakukan tugas. Sumber daya dapat berupa prosesor, tautan jaringan, atau kartu ekspansi. Tugas dapat berupa utas, proses, atau aliran data. Kegiatan penjadwalan dilakukan oleh proses yang disebut scheduler.

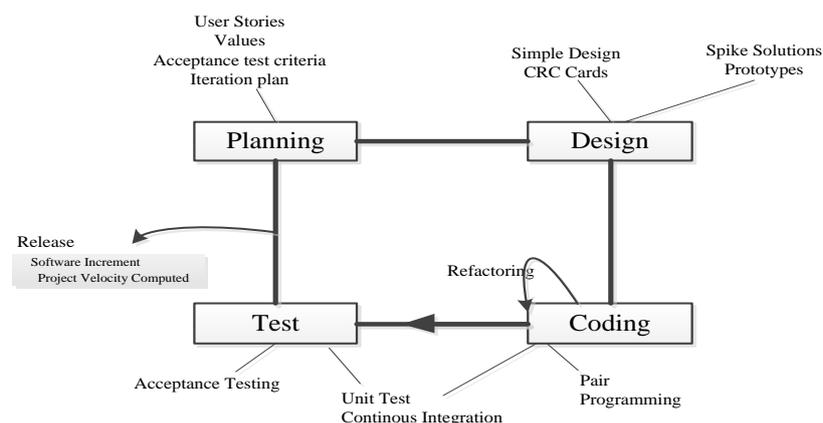
penjadwalan adalah sebuah proses perencanaan pengalokasian sumber daya untuk mengerjakan aktivitas pekerjaan didalam suatu waktu tertentu. Kegiatan penjadwalan adalah proses dimana pengalokasian sumber-sumber atau mesin-mesin yang ada untuk menjalankan sekumpulan tugas dalam jangka waktu tertentu.

2.8 Metode Penjadwalan

Metode *First Come First Served* (FCFS) adalah sistem antrian penjadwalan yang mengutamakan proses yang pertama kali diinput akan dilayani terlebih dahulu sampai selesai. Proses tersebut diberi jarak waktu yang diurutkan berdasarkan waktu kedatangan. Algoritma FCFS merupakan penjadwalan non-preemptive, sebuah proses yang sedang dikerjakan harus diselesaikan terlebih dulu barulah proses lainnya dapat dilayani (Purwanto *et al.*, 2022).

2.9 Metode *Extreme Programming*

Extreme programming berdasarkan sejarah singkat bahwa pengembangan perangkat lunak banyak digunakan untuk pengembangan yang lebih cepat dengan meliputi tahapan *planning*, *design*, *coding* dan *testing*. Berikut merupakan konsep *Extreme programming* (Suryantara, 2017).



Gambar 2.2 *Extreme Programming*

2.9.1 Sejarah Singkat *Extreme Programming*

Extreme Programming merupakan salah satu metodologi rekayasa perangkat lunak yang banyak digunakan untuk pengembangan aplikasi oleh para developer, diperkenalkan oleh kent beck yang ditunjukkan untuk menangani sebuah proyek

penggajian yang dikenal sebagai C3 (*Chrysler Comprehensive Compensation*). Proyek tersebut dimulai pada maret 1996 yang terancam gagal karena rumitnya sistem mengalami kegagalan pada proses testing, kemudian pihak Chrysler menyewa Kent Beck sebagai konsultan di bidang *software engineering* yang kemudian disebut sebagai pencetus XP.

2.9.2 Kerangka Kerja *Extreme Programming*

Pengembangan yang dilakukan menggunakan XP dengan proses yang lebih cepat dengan tahapan seperti *planning*, *design*, *coding* dan *testing*.

1. *Planning*/Perencanaan

Tahap ini dimulain dengan pemahaman konteks bisnis dari aplikasi dengan mendefinisikan keluaran seperti fitur, fungsi, penentuan waktu dan biaya serta alur pengembangan.

2. *Design*/Perancangan

Tahap perencanaan secara sederhana dengan alat mendesain kartu CRC (*Class Responsibility Collaborator*) yang digunakan untuk pemetaan kelas-kelas yang akan digunakan pada diagram UML.

3. *Coding*/Pengkodean

Hal utama dalam pengembangan menggunakan XP yaitu *pari programming* (Proses pembuatan program melibatkan 2 atau lebih programmer).

4. *Testing*/Pengujian

Tahap ini fokus pada pengujian fitur pada aplikasi sehingga tidak ada kesalahan dan sesuai dengan proses bisnisnya.

2.9.3 Nilai Inti *Extreme Programming*

Nilai inti pada pengembangan sistem yaitu :

1. *Communication* (Komunikasi)

Komunikasi antar tim yang digunakan untuk saling berbagi pengetahuan dalam pengembangan.

2. *Simplicity* (Kesederhanaan)

Melakukan semua proses dengan sederhana dan mencoba mencari solusi yang paling sederhana.

3. *Feedback* (Masukan)
Masukan untuk mengetahui kemajuan dari proses dan kualitas perangkat lunak yang dibuat.
4. *Courage* (Kesalahan)
Kesuksesan pengembangan aplikasi harus memiliki keberanian, keyakinan dan integritas dalam pekerjaan.
5. *Respect* (Menghormati)
Menerapkan siklus pendek dan integrasi *continue*.

2.9.4 Aspek Dasar *Extreme Programming*

Aspek dasar pada penerapan metode *extreme programming* yaitu :

1. *The Planning Game*
Proses pendek dan cepat, mengutamakan aspek teknik, memisahkan unsur bisnis dengan unsur teknis dan pertemuan intensif antara klien dengan developer. Pada XP proses ini menggunakan terminologi “game” karena Beck menyarankan untuk menggunakan teknik *score card* dalam menentukan *requirements*.
2. *Small Releases*
Menyelesaikan bagian - bagian aplikasi dan melakukan persentasi kepada *client*, setelah mendapatkan persetujuan maka dilakukan penerapan keaplikasi.
3. *Metaphor*
Menggambarkan visi yang luas terhadap tujuan dari pengembangan perangkat lunak. Dengan Tujuan diharapkan komunikasi antara klien dengan developer akan berlangsung lebih baik.
4. *Simple Design*
Menghindari desain yang rumit dalam sebuah pengembangan perangkat lunak. Dengan desain yang simpel apabila terjadi perubahan dapat meminimalkan kesalahan.
5. *Refactoring*
Melakukan perubahan pada kode program dari perangkat lunak dengan tujuan meningkatkan kualitas dari struktur program tersebut tanpa mengubah cara program tersebut bekerja

6. *Testing*
Membuat test terhadap aplikasi berdasarkan model test yang telah ditentukan.
7. *Pair Programming*
Dua orang programmer saling bekerjasama di komputer yang sama untuk menyelesaikan sebuah unit .
8. *Colletive Ownership*
Saling berbagi pengetahuan agar tidak saling ketergantungan pada programmer tertentu ataupun berbagai hambatan akibat perbedaan gaya menulis program dapat diperkecil.
9. *Coding Standard*
Dengan adanya coding standard yang telah disepakati terlebih dahulu maka pemahaman terhadap program akan menjadi mudah untuk semua programmer dalam tim.
10. *Continuous Integration*
Melakukan *build* sesering mungkin berbagai kesalahan pada program dapat dideteksi dan diperbaiki secepat mungkin.
11. *40-hours Week*
Beck berpendapat bekerja 8 jam sehari dan 5 hari seminggu adalah maksimal untuk tiap programmer.
12. *On-Site Customer*
XP menganjurkan bahwa ada anggota dari klien yang terlibat pada proses pengembangan perangkat lunak. Apabila ada kesalahan dalam pengembangan diharapkan klien dapat segera memberikan masukan untuk koreksinya.

2.9.5 Tujuan *Extreme Programming*

Tujuan metode *extreme programming* untuk menghasilkan perangkat lunak yang berkualitas tinggi dan lebih produktif dan mengurangi biaya selama ada perubahan dalam pengembangan perangkat lunak menggunakan siklus pengembangan perangkat lunak singkat.

2.9.6 Daur Hidup Metodologi *Extreme Programming*

Metode XP dapat diterapkan bila:

1. Adanya perubahan yang sangat cepat
2. Memiliki resiko yang tinggi pada pembuatan aplikasi
3. Dalam tim pengembangan aplikasi dengan sedikit programmer
4. Mampu mengotomatisasikan uji sistem
5. Keterlibatan peran serta klien secara langsung
6. Harus ada komunikasi yang baik

2.9.7 Keuntungan dan Kerugian *Extreme Programming*

Keuntungan pada penerapan metode XP yaitu:

1. Dalam hal XP menjalin komunikasi yang baik dengan klien pada pengembangan aplikasi
2. Saling menghargai antar developer dan meningkatkan komunikasi
3. Dapat menjadi pembelajaran bagi orang lain
4. Klien mendapatkan umpan balik yang akurat mengenai aplikasi yang dibuat
5. Dengan XP dapat mengubah pemikiran pelanggan terhadap aplikasi yang dibuat
6. Developer tidak berkerja secara berlebihan
7. Dengan XP dapat membuat keputusan yang bersifat teknikal

2.10 *Unified Modelling Language (UML)*

Alat pengembang sistem merupakan konsep desain yang digunakan untuk menggambarkan sistem dengan menggunakan diagram. Penyesuaian alat yang digunakan harus sesuai dengan metode pengembangan yang dilakukan salah satunya adalah penerapan *Unified Modelling Language*. Menurut (Rosa dan Salahuddin, 2019), *Unified Modelling Language* adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. Berikut ini merupakan penjelasan tentang masing-masing diagram yang ada pada *Unified Modelling Language*.

2.10.1 Use Case Diagram

Use Case adalah *use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut (Rosa and Shalahuddin, 2019).

2.10.2 Activity Diagram

Activity diagram adalah *activity* diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Salahuddin, 2019).

2.10.3 Class Diagram

Class Diagram adalah *Class diagram* mengembangkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem (Rosa dan Salahuddin, 2019).