

BAB II

TINJAUAN PUSTAKA

2.1 Sistem Informasi

Sistem informasi adalah sekumpulan subsistem yang saling berhubungan, berkumpul bersama-sama dan membentuk satu kesatuan, saling berintegrasi dan bekerjasama antara bagian satu dengan yang lainnya dengan cara tertentu untuk melakukan fungsi pengolahan data, menerima masukan (*input*) berupa data-data, kemudian mengolahnya (*processing*), dan menghasilkan keluaran (*output*) berupa informasi sebagai dasar pengambilan keputusan yang berguna dan mempunyai nilai nyata yang dapat dirasakan akibatnya baik pada saat itu juga maupun disaat mendatang, mendukung kegiatan operasional, manajerial, dan strategis organisasi, dan memanfaatkan berbagai sumber daya yang ada dan tersedia bagi fungsi tersebut guna mencapai tujuan [4].

2.2 SDN

Sekolah Dasar Negeri (SDN) adalah jenis sekolah dasar yang dikelola oleh pemerintah daerah. SDN memberikan pendidikan dasar bagi anak usia 6-12 tahun melalui kurikulum nasional yang meliputi mata pelajaran inti seperti Bahasa Indonesia, Matematika, IPA, dan IPS. Sekolah dasar negeri dilengkapi dengan fasilitas seperti ruang kelas, perpustakaan, dan lapangan olahraga. Evaluasi dan penilaian teratur dilakukan untuk mengukur kemajuan siswa. Peran orangtua dalam pendidikan anak sangat penting, dengan adanya partisipasi dalam kegiatan sekolah dan dukungan serta pemantauan rutin dalam proses belajar anaknya [5].

2.3 Peran Orangtua Terhadap Pemantauan Proses Belajar Siswa

Pendidikan merupakan komponen yang paling esensial dalam kehidupan manusia. Karakter baik setiap individu ditumbuhkan melalui pendidikan. Pendidikan juga dapat digambarkan sebagai suatu kegiatan yang terjadi dalam kehidupan manusia atau sebagai suatu tindakan yang terjadi dalam keberadaan manusia. Seperti pertumbuhan kepribadian seseorang, kecerdasan yang mencakup berbagai fakta, dan perkembangan emosional dan spiritual, agar memiliki karakter moral. Maka orangtua memiliki peran penting dalam pendidikan anak karena mereka memiliki pengaruh yang signifikan terhadap perilaku anak dan pengembangan keterampilan akademik anak-anak mereka. Dalam mengasuh anak, pendidik utama di rumah adalah orangtua (ayah dan ibu). Orangtua, baik ibu dan ayah, memainkan peran penting dalam ekspansi dan pematangan anak-anak mereka. Agar anak-anak berhasil dalam hidup, orangtua bertanggung jawab untuk menyediakan, mendidik, dan membimbing serta memantau proses belajar selama di sekolah, sehingga orangtua memiliki pengaruh besar pada tingkat motivasi yang ditunjukkan anak-anak mereka dalam proses belajar di sekolah [6].

2.4 Absensi

Absensi, dalam konteks manajemen sumber daya manusia dan pendidikan merujuk pada daftar hadir atau pencatatan kehadiran seseorang atau kelompok orang dalam suatu kegiatan atau institusi. Absensi berfungsi untuk memantau kehadiran dan memastikan kepatuhan terhadap aturan dan jadwal yang berlaku. **Pencatatan ketidakhadiran** dalam beberapa kasus, absensi juga digunakan untuk mencatat siapa saja yang tidak hadir atau tidak masuk. **Tujuan** absensi untuk memastikan kedisiplinan, mengetahui tingkat kehadiran, dan sebagai dasar untuk melakukan evaluasi dan laporan. Implementasi absensi dapat dilakukan secara manual dengan daftar hadir atau menggunakan sistem otomatis seperti mesin absensi [7].

2.5 Nilai Raport

Nilai raport adalah laporan hasil belajar siswa yang disusun secara sistematis dan memberikan gambaran tentang perkembangan dan kemajuan belajar siswa dalam satu semester. rapor ini berfungsi sebagai alat ukur keberhasilan siswa, sarana komunikasi antara sekolah dan orang tua, serta bahan evaluasi untuk meningkatkan kualitas pembelajaran **laporan hasil belajar** nilai rapor merupakan dokumentasi resmi tentang pencapaian belajar siswa dalam satu semester. **Informasi Mendalam** Selain nilai, rapor juga berisi catatan perkembangan siswa, termasuk aspek sikap, pengetahuan, dan keterampilan **komunikasi antara sekolah dan orang tua/wali murid** nilai rapor menjadi sarana bagi sekolah untuk menyampaikan informasi tentang perkembangan siswa kepada orang tua atau wali murid. **tolak ukur pencapaian belajar** nilai raport digunakan sebagai dasar untuk menilai pencapaian belajar siswa dan sebagai bahan evaluasi bagi guru dan pihak sekolah. [8].

2.6 Visual Code

Visual code adalah Perangkat lunak penyunting kode sumber buatan *Microsoft* untuk *linux macOS* dan *windows*. *Visual code* editor kode sumber yang populer dan gratis yang dikembangkan oleh *Microsoft* *VS code* digunakan untuk menulis, mengedit, men-*debug*, dan membangun code ini adalah platform lintas sistem operasi yang mendukung berbagai bahasa pemrograman dan memiliki banyak fitur seperti penyorotan sintaks, penyelesaian kode, *debug*, integrasi Git [9].

2.7 PHP (*Personal Home Page*)

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* [10].

2.8 Framework Laravel

Framework laravel adalah sebuah framework php open source yang populer untuk membangun aplikasi web. Framework ini dirancang untuk menyederhanakan proses pengembangan web dengan menyediakan berbagai fitur dan alat yang mendukung pengembangan aplikasi yang terstruktur mudah dipelihara, dan lebih cepat. laravel menggunakan pola desain *model-view-controller* (MVC) untuk memisahkan logika aplikasi, tampilan, dan interaksi dengan *database* [11].

2.9 Xampp

Xampp adalah paket perangkat lunak (software) *open source* yang mengintegrasikan berbagai komponen penting untuk pengembangan web, seperti server web *apache*, database *mysql/mariadb*, bahasa pemrograman php dan perl, sehingga memudahkan pengembang untuk membuat dan menguji aplikasi web secara lokal di perangkat mereka [12].

2.10 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 dan digunakan kembali dengan cara yang lebih efisien [13].

2.11 *Extreme Programming*

Extreme programming dikenal dengan metode *XP* yaitu bentuk dari model pengembangan perangkat lunak yang memiliki tahapan pengembangan sistem menjadi lebih efisien, adaptif dan fleksibel. *XP* bukan hanya berfokus pada coding akan tetapi meliputi bagian dari seluruh area pengembangan perangkat lunak.

Tahapan dalam metode pengembangan sistem *Extreme Programming* yaitu:

1) *Planning* (Perencanaan)

Tahapan ini merupakan langkah awal dalam pembangunan sistem dimana dalam tahapan ini dilakukan beberapa kegiatan perencanaan yaitu, identifikasi permasalahan, menganalisa kebutuhan sampai dengan penetapan jadwal pelaksanaan pembangunan sistem. Pada tahapan *planning* dapat dimulai dengan mendengarkan kumpulan kebutuhan aktifitas dari suatu sistem yang memungkinkan pengguna dapat memahami proses bisnis untuk sistem dan mendapatkan gambaran yang jelas dalam mengenai fitur utama, fungsionalitas dan keluaran yang diinginkan.

2) *Design* (Perancangan)

Tahapan berikutnya adalah perancangan dimana pada tahapan ini dilakukan kegiatan pemodelan yang dimulai dari pemodelan sistem, pemodelan arsitektur sampai dengan pemodelan basis data. Pemodelan sistem dan arsitektur menggunakan diagram *Unified Modelling Language (UML)* sedangkan pemodelan basis data menggunakan *Entity Relationship Diagram (ERD)*.

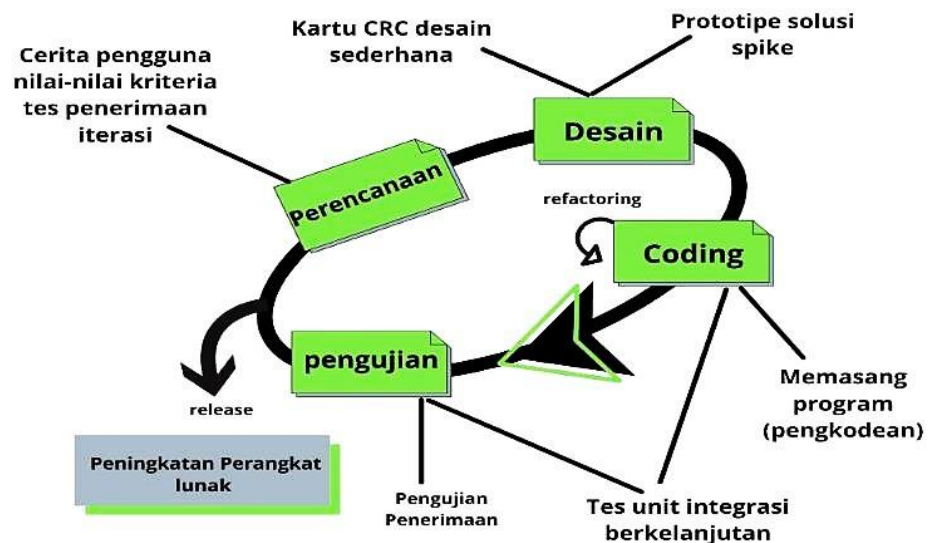
3) *Coding* (Pengkodean)

Tahapan ini merupakan kegiatan penerapan pemodelan yang sudah dibuat kedalam bentuk user interface dengan menggunakan bahasa pemrograman. Adapun bahasa pemrograman yang digunakan adalah

PHP dengan metode terstruktur. Untuk sistem manajemen basis data menggunakan piranti lunak *MySQL*.

4) *Testing* (Pengujian)

Setelah tahapan pengkodean selesai, kemudian dilakukan tahapan pengujian sistem untuk mengetahui kesalahan apa saja yang timbul saat aplikasi sedang berjalan serta mengetahui apakah sistem yang dibangun sudah sesuai dengan kebutuhan pengguna. Metode pengujian yang digunakan pada tahapan ini adalah metode *blackbox testing*, dimana pengujian yang dilakukan terhadap form beberapa masukkan apakah sudah berjalan sesuai dengan fungsinya masing-masing.



Gambar 2.1 Metode *Extreme programming*

Berikut adalah tahapan penelitian menggunakan *Extreme programming*.

- **Analisis Kebutuhan:** Tahap awal ini melibatkan pengumpulan informasi tentang kebutuhan pengguna, tantangan yang dihadapi oleh sekolah, dan tujuan dari sistem informasi yang akan dikembangkan. Ini juga melibatkan identifikasi fitur-fitur utama yang diperlukan dalam sistem informasi monitoring.

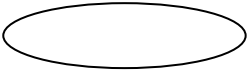
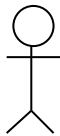
- **Perencanaan Proyek:** Tim pengembang akan merencanakan proyek secara keseluruhan, termasuk menetapkan waktu, sumber daya, dan anggaran yang dibutuhkan. Rencana ini akan berfokus pada pembagian tugas, jadwal iterasi, dan strategi pengujian.
- **Pengujian:** Setiap fitur yang dikembangkan akan diuji secara menyeluruh untuk memastikan keandalan, kinerja, dan kepatuhan terhadap kebutuhan pengguna. Pengujian akan dilakukan baik secara otomatis maupun manual, dan umpan balik dari pengguna akan digunakan untuk melakukan perbaikan jika diperlukan

2.12 Unified Modelling Language (UML)



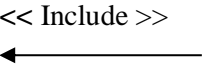
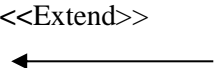
2.12.1 Use Case Diagram

Use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut [14]. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Use Case Diagram* dapat dilihat pada Tabel 2.1

Tabel 2.1 Simbol Use Case Diagram

No.	Simbol	Deskripsi
1.		Use Case Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i> .
2.		Aktor Aktor seseorang/sesuatu yang berinteraksi dengan yang akan dibuat. diluar sistem informasi. Biasanya dinyatakan menggunakan kata benda.


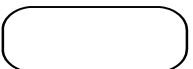
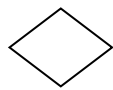

Tabel 2.1 Simbol Use Case Diagram (lanjutan)

3.		Asosiasi/association merupakan komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.
4.		Generalisasi (<i>generalization</i>) merupakan hubungan (umum – khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum
5.		Include berarti use case yang ditambahkan akan dipanggil saat use case tambahan dijalankan.
6.		Ekstensi (<i>extend</i>) merupakan use case tambahan ke sebuah use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu.

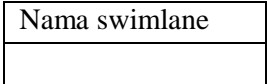

2.12.2 Activity Diagram

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 [15]. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *activity diagram* dapat dilihat pada Tabel 2.2.

Tabel 2.2 Simbol Activity Diagram

No.	Simbol	Keterangan
1.		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.		Percabangan (<i>Decision</i>) merupakan asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.		Penggabungan (<i>Join</i>) merupakan asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.

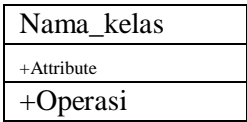


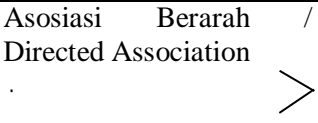
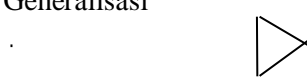
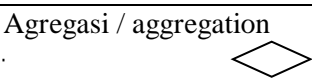
Tabel 2.2 Simbol Activity Diagram (lanjutan)

5.		Swimlane memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas.
6.		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

2.12.3 Class Diagram

Class diagram mengembangkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem [16]. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Class Diagram* dapat dilihat pada Tabel 2.3.

Tabel 2.3 Simbol Class Diagram

No.	Simbol	Deskripsi
1.		Kelas pada struktur sistem.
2.		Sama dengan konsep interface dalam pemrograman berorientasi objek.
3.		Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan simbol
4.		Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan simbol.
5.		Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
6.		Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>)

2.12.4 Sequence Diagram

Diagram rangkaian menggambarkan bagaimana objek berinteraksi dengan satu sama lain melalui pesan pada eksekusi sebuah use-case atau operasi [17]. Diagram ini mengilustrasikan bagaimana pesan terkirim dan diterima di antara objek dan dalam sekuensi pada Tabel 2.4.

Tabel 2.4 Simbol *Sequence Diagram*

No.	Simbol	Deskripsi
1.	<i>Object lifeline</i>	Menggambarkan panjang kehidupan suatu objek selama skenario sedang dibuat contohnya
2.	<i>Activation</i>	Dimana proses sedang dilakukan oleh <i>object</i> atau <i>class</i> untuk memenuhi pesan atau perintah
3.	<i>Message</i>	Sebuah anak panah yang mengindikasikan pesan diantara objek. Dan objek dapat mengirimkan pesan ke dirinya sendiri

2.12.5 Pengujian *Black Box Testing*

Black Box Testing merupakan pengujian fungsional struktur program tidak dipertimbangkan. Kasus uji diputuskan berdasarkan persyaratan atau spesifikasi program atau modul dan internal yang diuji kepada user. Metode *Black box Testing* adalah sebuah metode yang dipakai untuk menguji sebuah *software* tanpa harus memperhatikan detail *software*. Pengujian ini hanya memeriksa nilai keluaran berdasarkan nilai masukan masing-masing. Tidak ada upaya untuk mengetahui kode program apa yang output pakai. Proses *Black Box Testing* dengan cara mencoba program yang telah dibuat dengan mencoba memasukkan data pada setiap formnya. Pengujian ini diperlukan untuk mengetahui program tersebut berjalan sesuai dengan yang dibutuhkan