

## BAB II

### TINJAUAN PUSTAKA DAN LANDASAN TEORI

#### 2.1 Tinjauan Pustaka

Penelitian sebelumnya pada tahun 2021 tentang sistem informasi pemantauan aktivitas dengan judul “Perancangan Sistem Monitoring Perkembangan Anak TK Islam Sunan Gunung Jati” dengan permasalahan yaitu orang tua tidak mengetahui capaian anak saat di sekolah. Sistem tersebut menggunakan model pengembangan sistem *prototype* dan menghasilkan sistem informasi monitoring dengan aktor admin, guru dan orang tua. Orang tua dapat melihat laporan perkembangan anak berupa laporan harian, bulanan, dan semester[6].

Penelitian lain dengan judul “Rancang Bangun Sistem Monitoring Perkembangan Anak di TKIT Taamasa Menggunakan Metode Spiral” tahun 2021 dengan permasalahan yaitu pada TKIT Taamasa orang tua murid hanya mengetahui mengenai informasi putra-putrinya dalam jangka waktu yang cukup lama sehingga membuat orang tua tidak mengetahui apabila terjadi penurunan pada anaknya. Sistem ini menghasilkan Aplikasi sistem monitoring perkembangan anak di TK IT Taamasa yang dapat memberi informasi perkembangan anak dengan mudah kepada orang tua yang saat ini masih dilakukan secara manual[7].

Penelitian lain dengan judul “Perancangan Sistem Informasi Monitoring Perkembangan Anak Berbasis Website Pada Rumah Pintar Indonesia (RPI) Yogyakarta” tahun 2019. Permasalahannya yaitu sistem monitoring perkembangan anak di Rumah Pintar Indonesia masih dituliskan ke dalam sebuah buku penghubung orang tua dan orang tua seringkali lalai dalam mengembalikannya. Sistem yang dibuat ini menggunakan model pengembangan *prototype* dan menghasilkan sistem informasi sebagai media untuk memantau perkembangan anak secara *real time* yang dapat diakses kapan dan dimana saja [2].

Penelitian lain dengan judul “Sistem Informasi Monitoring Kehadiran Siswa dalam Pengawasan Orang Tua pada Sekolah Menengah Kejuruan (SMK) Negeri 4 Banda Aceh” tahun 2021 menghasilkan sistem informasi dimana guru dapat melakukan absensi ke siswa siswinya berdasarkan mata pelajaran dan jam masuk kelas dan Orang tua siswa akan memperoleh SMS jika anaknya tidak masuk

sekolah pada jam guru yang melakukan absensi sebagai upaya pencegahan terhadap perilaku kenakalan siswa yang tidak diinginkan[7]

Penelitian lain dengan judul “*Developing Application Programming Interface (API) for Student Academic Activity Monitoring using Firebase Cloud Messaging (FCM)*” tahun 2018. Hasil dari penelitian tersebut adalah aplikasi pemantauan kegiatan perkuliahan yang dapat diakses langsung oleh orang tua, sehingga ketika ada perubahan kegiatan perkuliahan, orang tua dapat langsung mengetahuinya. Penelitian tersebut menggunakan *firebase cloud messaging* untuk pengiriman notifikasi ke orang tua[8].

Pada penelitian ini dibuat sebuah Sistem Informasi Pemantauan Perkembangan Aktivitas dan Belajar Anak Berbasis Website pada TK Aisyiyah 2 Kracak. Terdapat pengembangan fitur seperti laporan harian, laporan bulanan, dan laporan hasil belajar per semester. Adapun informasi yang akan diberikan yaitu dalam lingkup aktivitas harian dan dan nilai-nilai bidang pengembangan, seperti agama dan moral, bahasa, kognitif, fisik *motoric*, seni dan sosial. Perbedaan dengan penelitian sebelumnya yaitu akan ada fitur dimana guru dan orang tua murid dapat berdiskusi mengenai perkembangan anaknya dan juga guru dapat memberikan saran atas permasalahan yang dialami oleh muridnya kepada orang tua murid. Sistem ini memberikan kemudahan bagi guru dan orang tua untuk memberikan tindakan lanjut dari setiap perkembangan murid dan anaknya.

## **2.2 Landasan Teori**

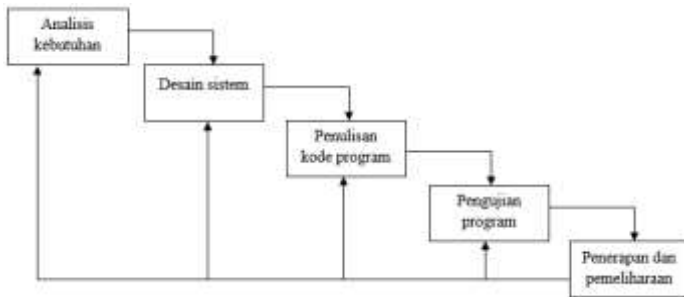
### **2.2.1 Sistem Informasi**

Sistem informasi adalah suatu sistem pada suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi yang menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan[9]. Sedangkan pemantauan merupakan kegiatan yang dimaksudkan untuk mengamati perkembangan pelaksanaan rencana pembangunan; mengidentifikasi serta mengantisipasi permasalahan yang timbul dan atau akan timbul untuk dapat diambil tindakan sedini mungkin[10].

### **2.2.2 Rekayasa Perangkat Lunak**

Rekayasa perangkat lunak adalah suatu pengembangan perangkat lunak yang meliputi berbagai tahapan seperti analisis, desain

pengkodean, pengujian dan pemeliharaan, di mana semua tahapan ini dilakukan oleh para ahli [11].



**Gambar 2. 1** Metode *waterfall* menurut Ian Sommerville [5]

Pada gambar 2.1 merupakan tahapan metode pengembangan sistem menurut Ian Sommerville. Metode *waterfall* ini terdiri dari 5 tahapan dengan penjelasan sebagai berikut:

1. Analisis Kebutuhan  
Pada tahap analisis kebutuhan dilakukan riset untuk mengetahui apa saja kebutuhan dari sistem yang dibangun untuk menentukan fitur dan layanan pada sistem. Riset yang dilakukan diantaranya ada wawancara, survey, maupun diskusi dengan pihak terkait.
2. Desain Sistem  
Pada tahap desain sistem berfokus pada desain *interface* sistem berdasarkan dari kebutuhan yang sudah dianalisis pada tahap sebelumnya.
3. Penulisan Kode Program  
Pada tahap penulisan kode program ini merupakan bentuk implementasi dengan menuliskan kode atau bahasa pemrograman tertentu dari analisis dan desain sistem dari tahap sebelumnya
4. Pengujian Program  
Setelah penulisan kode selesai, kemudian dilakukan pengujian pada program. Pengujian ini dilakukan untuk memastikan sistem sudah berjalan sempurna tanpa adanya kesalahan pada program dan juga memastikan bahwa masukan sudah sesuai.
5. Penerapan dan Pemeliharaan  
Pada tahap terakhir metode *waterfall*, sistem yang telah dibuat selanjutnya dioperasikan pengguna dan dilakukan pemeliharaan.

Pemeliharaan meliputi perbaikan kesalahan, perbaikan implementasi unit sistem, dan kesalahan lain yang tidak terdeteksi pada tahap-tahap sebelumnya.

Selain metode pengembangan sistem, ada juga metode pengujian sistem. Metode pengujian sistem merupakan pengujian dari sistem yang telah dibuat apakah fungsi – fungsinya sudah benar atau perlu diperbaiki. Metode yang digunakan pada pengujian pada penelitian ini yaitu Metode *Black Box*. Metode *Black Box* merupakan pengujian sistem tanpa menguji setiap kodenya. Pengujian hanya pada bagian fungsi, antarmuka dan alur sistem.[5]

### 2.2.3 Pemrograman Berorientasi Objek

Pemrograman Berorientasi Objek adalah suatu cara baru dalam berpikir serta berlogika dalam menghadapi masalah - masalah yang akan dicoba atasi dengan bantuan komputer, dimana setiap objek adalah entitas tunggal yang memiliki kombinasi struktur data dan fungsi tertentu. Sedangkan objek adalah orang, tempat, benda, kejadian, objek sebuah kejadian seperti pembayaran uang pendidikan, registrasi biodata siswa, membaca buku dan sebagainya [12].

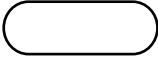
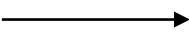
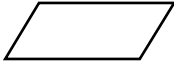

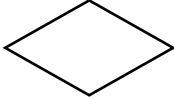
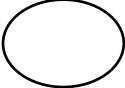
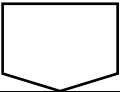


#### A. Unified Modelling Language (UML)



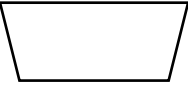
*Unified Modelling Language* (UML) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks - teks pendukung[13]. Jenis – jenis diagram UML adalah sebagai berikut:

##### 1. Flowchart

*Flowchart* adalah langkah – langkah penyelesaian masalah yang dituliskan dalam suatu simbol – simbol tertentu. Dengan *flowchart* dapat menggambarkan langkah atau pemecahan dari suatu masalah secara sederhana, mudah dimengerti, dan tidak ambigu dengan menggunakan standart simbol – simbol *flowchart* [14]. Simbol – simbol *flowchart* dapat dilihat pada Tabel 2.1.

Tabel 2. 1 Simbol *Flowchart*



No.	Simbol	Keterangan
1.		<b>Terminal</b> Awal atau akhir suatu program
2.		<b>Arus/flow</b> Menyatakan jalannya arus suatu proses
3.		<b>Input dan Output</b> Menyatakan proses input dan output
4.		<b>Process</b> menunjukkan proses operasional komputer
5.		<b>Decision</b> menunjukkan suatu kondisi tertentu mengarah pada dua kemungkinan, ya/tidak.
6.		<b>Connector (on-page)</b> koneksi penghubung proses ke proses lain pada halaman yang sama
7.		<b>Connector (off-page)</b> koneksi penghubung dari satu proses ke proses lain di halaman lain
8.		<b>Preparation</b> mewakili ketentuan penyimpanan untuk diproses untuk memberikan awal harga.
9.		<b>Document</b> Mencetak <i>output</i> dalam dokumen melalui printer




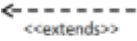
No.	Simbol	Keterangan
10.		<b>Punched Card</b> Input berasal dari kartu atau output ditulis ke kartu
11.		<b>Manual Input</b> Simbol yang digunakan untuk memasukan data secara manual dengan menggunakan <i>online keyboard</i>
12.		<b>Manual Operation</b> menunjukkan operasi yang tidak dilakukan oleh komputer

## 2. Use Case Diagram

*Use case diagram* adalah rangkaian atau uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor[15]. Simbol – simbol *use case diagram* dapat dilihat pada Tabel 2.2 [16].

**Tabel 2. 2** Simbol *Use case Diagram*

No.	Simbol	Keterangan
1.		<b>Use Case</b> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktif
2.		<b>Actor</b> merupakan orang yang terlibat dalam mengaktifkan fungsi dari target sistem.







No.	Simbol	Keterangan
3.		<b>Asosiasi</b> antara aktor dan use case, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan data
4.		<b>Asosiasi</b> antara aktor dan use case yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem
5.		<b>Include</b> , merupakan di dalam use case lain ( <i>required</i> ) atau pemanggilan sebuah fungsi program
6.		<b>Extend</b> , merupakan perluasan dari use case lain jika kondisi atau syarat terpenuhi.

### 3. Sequence Diagram

*Sequence* diagram adalah diagram yang menjelaskan interaksi objek yang berdasarkan urutan waktu, *sequence* diagram juga dapat menggambarkan urutan atau tahapan yang harus dilakukan untuk dapat menghasilkan sesuatu seperti pada use case diagram[17]. Simbol – simbol pada *sequence* diagram dapat dilihat pada Tabel 2.3.

**Tabel 2. 3** Simbol *Sequence* Diagram

No.	Simbol	Keterangan
-----	--------	------------


No.	Simbol	Keterangan
1.		<b>Actor</b> menggambarkan orang yang sedang berinteraksi dengan sistem
2.		<b>Entity Class</b> menggambarkan hubungan yang sedang dilakukan
3.		<b>Boundary Class</b> Menggambarkan sebuah gambaran dari <i>form</i>
4.		<b>Control Class</b> menggambarkan penghubung antara boundary dengan tabel
5.		<b>A focus of control &amp; a life line</b> menggambarkan tempat muladi dan berakhirnya message
6.		<b>A message</b> menggambarkan pengiriman pesan

### 2.2.4 Basis Data

Basis Data adalah struktur penyimpanan data untuk menambah, mengakses dan memproses data yang disimpan dalam sebuah komputer, yang diperlukan sistem manajemen *database* seperti MySQL server[7]. Untuk memodelkan suatu basis data diperlukan *Entity Relationship Diagram* (ERD). *Entity Relationship Diagram* (ERD) adalah model teknik pendekatan yang menyatakan atau menggambarkan hubungan suatu model. Didalam hubungan tersebut dinyatakan dengan objek data (*Entity*) dan hubungan (*Relationship*) yang ada pada *entity* berikutnya[18]. Dalam pembuatan ERD menggunakan simbol – simbol yang dapat dilihat pada Tabel 2.4.

**Tabel 2. 4** Simbol ERD



No.	Simbol	Keterangan
1.		<p><b>Entitas</b> Entitas merupakan kumpulan objek yang dapat diidentifikasi secara unik. Nama entitas adalah kata benda tunggal.</p>
2.		<p><b>Relasi</b> Relasi merupakan kejadian atau transaksi yang terjadi antara dua entitas yang perlu menutup informasi dalam database.</p>
3.		<p><b>Atribut</b> Atribut merupakan potongan informasi tentang entitas yang perlu disimpan sebagai database.</p>
4.		<p><b>Link</b> Garis penghubung atribut dengan kumpulan entitas dan kumpulan entitas dengan relasi</p>

### 2.2.5 Pendidikan

Pendidikan nasional berfungsi mengembangkan kemampuan dan bertujuan untuk mengembangkan potensi peserta didik yang bertakwa kepada Tuhan yang Maha Esa, berakhlak mulia, sehat, berilmu, cakap, kreatif dan bertanggung jawab [19]. Dalam mewujudkan potensi tersebut diperlukan lembaga prasekolah yaitu Pendidikan Taman Kanak – Kanak. Pendidikan Taman Kanak – Kanak merupakan periode awal yang paling penting dan mendasar disepanjang rentang pertumbuhan dan perkembangan kehidupan manusia. Pada masa tersebut merupakan salah satu periode yang menjadi ciri masa anak – anak atau disebut *The Golden Age* dimana semua potensi anak berkembang sangat cepat. Oleh karena itu, keterlibatan orang tua sangat diperlukan untuk memulai pembentukan katakter anak melalui

pengembangan sikap moral, agama, sosial, dan emosional. Pengembangan karakter tersebut hanya dapat dicapai secara maksimal dengan adanya kesinambungan antara pendidikan di rumah dan di sekolah, yang tentunya tidak dapat terlepas dari peran serta orang tua. Hal tersebut yang menjadi dasar pentingnya Pendidikan Taman Kanak – Kanak[20].

### 2.2.6 Codeigniter (CI)

CodeIgniter merupakan *framework* PHP yang dibuat berdasarkan *Model View Controller* (MVC). Codeigniter 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.). 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 [21].

### 2.2.7 Usability Testing

*Usability* adalah atribut kualitas yang menjelaskan atau mengukur seberapa mudah penggunaan suatu antar muka (*interface*). Kata “usability” juga merujuk pada suatu metode untuk meningkatkan kemudahan pemakaian selama proses desain [22]. Pertanyaan kuesioner pada uji usability harus memuat komponen-komponen berikut:

1. *Learnability*, yaitu tingkat kemudahan pengguna sistem yang dapat diukur berdasarkan pemakaian fitur dan fungsi pada sistem
2. *Efficiency*, yaitu tingkat kecepatan pengguna dalam melakukan fungsi-fungsi pada sistem
3. *Memorability*, yaitu tingkat kemampuan pengguna untuk mengingat dan mempertahankan pengetahuannya setelah beberapa waktu tertentu tidak memakai sistem
4. *Errors*, yaitu tingkat/jumlah error atau kesalahan yang dilakukan pengguna sistem, tingkat kebosanan dalam menghadapi *error*, dan cara memperbaikinya
5. *Satisfaction*, yaitu tingkat kepuasan pengguna setelah berinteraksi dengan sistem

Setiap pertanyaan pada pengujian usability menggunakan skala Likert. Skala *likert* adalah skala yang digunakan untuk mengukur persepsi, sikap atau pendapat seseorang atau kelompok mengenai sebuah peristiwa atau fenomena sosial [23]. Adapun jawaban setiap pertanyaan yang digunakan dapat dinilai dengan skor seperti Tabel 2.5.

**Tabel 2. 5** Skala Likert

Jawaban	Skor
Sangat Tidak Setuju	1
Tidak Setuju	2
Setuju	3
Sangat Setuju	4

Kemudian untuk menghitung pengujian usability dihitung menggunakan rumus berikut:

$$U = \frac{np}{nm} \times 100\%$$

keterangan:

U = Nilai uji *usability*

np = Nilai perolehan, jumlah seluruh hasil kuesioner/jawaban responden

nm = Nilai maksimal, jumlah maksimal yang dapat diperoleh (jumlah responden x nilai maksimal skala Likert yang dipakai)

Hasil uji usability yang didapat kemudian dicocokkan dengan kategori interpretasi peringkat sistem. Kategori interpretasi peringkat sistem dapat dilihat pada Tabel 2.6 [24] :

**Tabel 2. 6** Kategori Interpretasi Predikat Sistem

Presentase	Predikat
81 – 100%	Sangat baik
61 – 80%	Baik
41 – 60%	Cukup baik
21 – 40%	Kurang baik
0 – 20 %	Tidak baik