

BAB II

LANDASAN TEORI

2.1 Mobile

Menurut (Gunawan Pribadi, 2017) Menjelaskan, Perangkat Mobile telah berubah menjadi salah satu perangkat multifungsi. Salah satunya perangkat multifungsi yang sering digunakan sekarang adalah aplikasi *mobile* sebagai media untuk mengakses informasi dengan mudah. Perkembangan aplikasi *mobile* didukung dengan semakin berkembangnya bahasa pemrograman.

2.2 Monitoring

Menurut (Dave Michael, Dian Gustina, 2019) Menjelaskan Monitoring (pemantauan) adalah kegiatan untuk mengamati perkembangan pelaksanaan program atau proyek. Dengan monitoring dapat diketahui program atau proyek berjalan sesuai atau kurang sesuai dengan rencana.

2.3 Siswa

Dalam kamus Besar Bahasa Indonesia, menjelaskan bahwa siswa adalah murid (terutama pada tingkat dasar, menengah, pelajar).

Siswa atau Peserta Didik menurut ketentuan umum Undang-undang RI No. 20 Tahun 2003 tentang Sistem Pendidikan Nasional, Peserta didik adalah anggota masyarakat yang berusaha mengembangkan potensi diri melalui proses pembelajaran yang tersedia pada jalur, jenjang dan jenis pendidikan tertentu.

2.4 SMA Islam Global Surya

SMA Islam Global Surya merupakan salah satu sekolah internasional yang bernaungan dalam Yayasan Pendidikan Taraka Surya yang mengedepankan nilai-nilai religius dengan visi misi membentuk pemimpin masa depan yang berkarakter, cerdas dan global. SMA Islam Global Surya, beralamat di Jl. Sutan Jamil No. 1 Gedung Meneng, Kec.Rajabasa, Bandar Lampung.

2.5 Android

Menurut (Yuni Puspita Sari, 2019) Menjelaskan Android adalah sebuah system operasi untuk smartpone dan talet. Sistem Operasi dapat diilustrasikan sebagai ‘jembatan’ antara piranti (*device*) dan penggunanya, sehingga pengguna bisa berinteraksi dengan *device-nya* dan menjalankan aplikasi-aplikasi yang tersedia pada *device*.

2.6 Bahasa Pemrograman Yang Digunakan

2.6.1 Android Studio

Menurut (Badri Zaki, Syahrizal Dwi Putra, 2018) Menjelaskan *Android Studio* adalah sebuah IDE (*Integrated Development Environment*) untuk *Android Development* yang dikenalkan pihak *google* pada acara *Google I/O* di tahun 2013. *Android Studio* dibuat berdasarkan IDE Java populer, yaitu *IntelliJ IDEA*. *Android Studio* merupakan IDE resmi untuk pengembangan aplikasi Android. *Android Studio* dibangun dengan tujuan mempercepat proses pembuatan maupun pengembangan aplikasi Android yang berkualitas tinggi untuk setiap *device* Android.

2.6.2 My SQL

Menurut (Yolan, Mansuri, 2015) Menjelaskan *MySQL* adalah salah satu jenis program database. Sebuah program berfungsi untuk mengolah, menyimpan data dan memanipulasi data di server.

2.7 Metode Perangkat Lunak yang digunakan

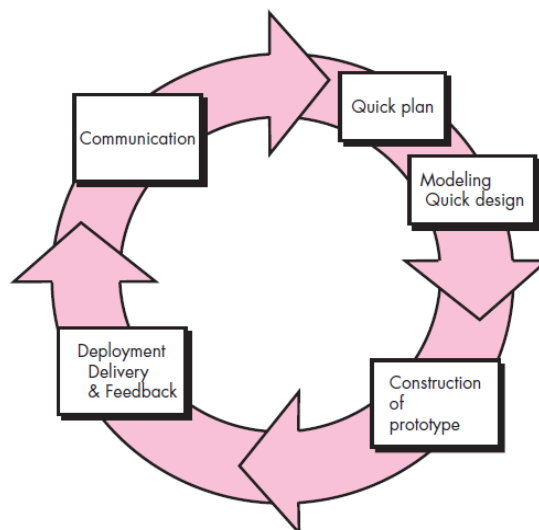
2.7.1 Prototype

Menurut (Pressman, 2017) *Prototype* adalah proses pembuatan model sederhana *software* yang memungkinkan pengguna memiliki gambaran dasar tentang program serta melakukan pengujian awal. *Prototype* memberikan fasilitas bagi pengembang dan pengguna untuk saling berinteraksi selama proses pembuatan, sehingga pengembang dapat dengan mudah memodelkan perangkat lunak yang akan di buat. Metode ini cocok digunakan untuk mengembangkan sebuah perangkat lunak yang dikembangkan kembali. Metode ini dimulai dengan pengumpulan kebutuhan pengguna. Kemudian membuat sebuah rancangan kilat yang selanjutnya akan dievaluasi kembali sbelum di produksi secara benar

Prototype, bukanlah merupakan sesuatu yang lengkap, tetapi sesuatu yang harus dievaluasi dan dimodifikasi kembali. Segala perubahan dapat terjadi pada saat *prototype* dibuat untuk memenuhi kebutuhan pengguna dan saat yang sama memungkinkan pengembangan untuk lebih memahami kebutuhan pengguna secara baik.

Tahapan-tahapan dalam metode *Prototype* :

1. Komunikasi (*Communication*) : pengumpulan data awal, yaitu komunikasi dengan klien dan *user* untuk menentukan kebutuhan.
2. Perencanaan Cepat (*Quick Plan*) : pembuaan perencanaan analisis terhadap kebutuhan pengguna.
3. Pemodelan Perancangan Cepat (*Modeling Quick Design*) : membuat rancangan desain program.
4. Pembentukan *Prototype* (*Construction of prototype*) : pembuatan aplikasi berdasarkan dari pemodelan desain yang telah dibuat.
5. Penyerahan Sistem dan Umpan Balik (*Development Delivery and Feedback*) : memproduksi perangkat ssecara benar sehinga dapat digunakan oleh pengguna.



Gambar 2.1 Diagram Prototype

Pada Gambar 2.1, Tahap pertama yaitu *communication* yaitu tahap suatu perencanaan yang di lakukan, mulai dari menciptakan dan melaksanakan proses untuk memastikan bahwa perencanaan tersebut berkualitas tinggi, terpercaya, efisiensi biaya. Tahap kedua adalah *quick plan* yaitu analisis terhadap kebutuhan pengguna. Tahap ketiga adalah *modelling quick design* yaitu pembuatan desain secara umum unuk selanjutnya dikembangkan kembali. Tahap keempat adalah *construction of prototype* adalah pembuatan perangkat *prototype* termasuk pengujian dan penyempurnaan. Tahap kelima adalah *deployment, delivery, and feedback* adalah tahap penyerahan sistem ke pengguna dan umpan balik.

2.8 Pengujian *Black-box Testing*

Menurut (Raka Tamagola, Puput Budi Wintoro, 2017) Menjelaskan, *Black Box Testing* bukanlah solusi alternatif dari *White Box Testing* tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh *White Box Testing*. *Black-box Testing* cenderung untuk menemukan hal-hal sebagai berikut :

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*Interface Errors*).
3. Kesalahan struktur data dan akses basis data.
4. Kesalahan performan (*Performance Errors*).
5. Kesalahan inisialisasi dan terminasi.

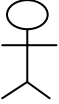

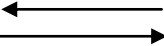
2.9 *Unified Modeling Language (UML)*

Menurut (Marini, 2019) Menjelaskan, Perencanaan berorientasi obyek biasanya menggunakan model yang dikenal dengan *Unified Modeling Language (UML)* yang merupakan sebuah bahasa pemodelan objek standar sebagai ganti dari pendekatan atau metode berorientasi objek standar. *Unified Modeling Language (UML)* adalah satu kumpulan konveksi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek.

2.10 Use Case Diagram

Menurut (Dona Yulawati, Sushanty Saleh, Indera, 2018) Menjelaskan, *Use case* menggambarkan fungsi tertentu dalam suatu sistem berupa komponen, kejadian atau kelas. *Use case* diartikan sebagai urutan langkah-langkah yang secara tindakan saling berkait (skenario), baik terotomatisasi maupun secara manual, untuk tujuan melengkapi satu tugas bisnis tunggal. Diagram *use case* bersifat statis, diagram ini memperlihatkan himpunan *use case* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna. Komponen-komponen *use case* terdapat pada tabel 2.1



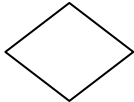


Tabel 2.1 Komponen-komponen *Use case*

Simbol	Keterangan
Aktor 	Mendefinisikan entitas diluar sistem yang memakai sistem.
Use Case 	Gambaran fungsionalitas dari suatu sistem, sehingga user mengerti kegunaan dari sistem yang akan dibangun.
Relasi 	Menceritakan suatu hubungan antara aktor dan <i>use case</i> sehingga diagram dapat mudah dipahami

2.11 Activity Diagram

Menurut (Dona Yulawati, Sushanty Saleh, Indera, 2018) Menjelaskan, Aktivitas merupakan kumpulan aksi-aksi. Aksi-aksi melakukan langkah sekali saja tidak boleh dipecah menjadi beberapa langkah lagi. Sebagai contoh fungsi matematika, pemanggilan perilaku, pemrosesan data. Aktivitas dapat mengakses atribut dan operasi dan operasi *classifier*, tiap objek terhubung dan parameter-parameter jika aktivitas memiliki hubungan dengan perilaku. Berikut komponen-komponen pada *activity diagram* pada tabel 2.2



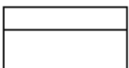


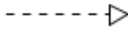

Tabel 2.2 Komponen-komponen Activity Diagram

Simbol	Keterangan
Status Awal 	Status Awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan oleh sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan 	Asosiasi percabangan dimana ada pilihan aktivitas lebih dari satu.
Penggabungan 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir

2.12 Class Diagram

Menurut (Melda Agarina, Tria Devi Miranti, Sutedi, 2019) Menjelaskan, *Class diagram* merupakan alat bantu untuk menentukan langkah-langkah kerja yang akan dilakukan oleh pemogram di mulai dari proses pengumpulan data, sampai pembentukkan tabel sesuai dengan permasalahan yang ditangani. *Class diagram* ini terlebih dahulu dirancang dalam mendukung rancangan pengolahan data elektronis supaya dapat berjalan dengan baik, dan dengan relasi yang baik akan diperoleh gambaran umum sistem yang akan di persiapkan. Penjelasan simbol *class diagram* terdapat pada tabel 2.3

Tabel 2.3 Simbol *Class Diagram*

Simbol	Keterangan
<i>Generalization</i> 	Menggambarkan relasi generalisasi
<i>Nary Association</i> 	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
<i>Class</i> 	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
<i>Collaboration</i> 	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
<i>Realization</i> 	Menggambarkan relasi.
<i>Dependency</i> 	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
<i>Association</i> 	Apa yang menghubungkan antara objek satu dengan objek lainnya

2.13 Penelitian Terkait

Dalam penyusunan skripsi ini, penulis terinspirasi dan mereferensi dari penelitian-penelitian sebelumnya yang berkaitan dengan skripsi ini. Daftar penelitian terkait sebagai berikut :

2.13.1 Penggunaan Sistem Informasi Dalam Pengolahan Nilai Siswa Berbasis E-Rapor Pada Kurikulum 2013 Kelas X SMAN 1 Sakti

Usti Ba'di Fitrillah, 2019 Menguraikan penggunaan Sistem Informasi Dalam Pengolahan Nilai Siswa Berbasis E-Rapor Pada Kurikulum 2013 Kelas X SMAN 1 Sakti menggunakan pengolahan nilai siswa, berupa input nilai siswa, mengubah nilai siswa sampai rapor siswa. Sistem ini hanya

dapat di akses oleh pengguna yang telah melakukan login ke sistem. Rancangan sistem informasi pengolahan nilai siswa ini dibuat sebagai sarana dalam menyajikan informasi laporan nilai serta data siswa yang dapat diakses dengan mudah oleh siswa dan orang tua.

2.13.2 Manajemen Program Penilaian Hasil Belajar E-rapor di SMA Negeri 3 Sidoharjo

Miftachul Jannah, Dr. Mudjito, Ak., M.Si, (2019) Menguraikan proses penginputan nilai bapak dan ibu guru harus mengentri nilai sendiri. Operator atau admin dalam e-rapor dan juga ada beberapa level yaitu administrator, wali kelas, guru mata pelajaran dan BK. Guru mata pelajaran mempunyai peran pembina ekstra misalnya menilai kehadiran siswa yang bisa mengisi BK sehingga wali kelas memverifikasi. Dalam tahap evaluasi ddari DAPODIK dan juga dari lembaga sekolah tersebut. Tujuan evaluasi tersebut adalah untuk meminimalkan kesulitan dari tidak dievaluasinya secara langsung dalam lingkungan sekolah.

2.13.3 Silent Monitoring And Cyber Law Awareness Siswa SMK Trisakti Bandar Lampung

Rionaldi Ali, Yuni Puspita Sari, R.Z Abdul Aziz, Joko Triloka, Sri Lestari (2020) Menguraikan Fenomena penggunaan perangkat komunikasi yang sudah hampir seluruhnya melekat pada masing-masing individu bahkan termasuk siswa sekolah adalah fakta yang nampak secara kasat mata. Bagi para orang tua/wali siswa untuk dapat memahami dan menggunakan aplikasi monitoring yang sudah dipasangkan adalah hal penting setelah akhlak dan awareness dalam diri pribadi para siswa dan juga dapat menghasilkan sikap dan akhlak siswa/siswi SMK Trisakti yang baik yang sejalan dengan tujuan dari pendidikan nasional.