

BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Dalam pembahasan landasan teori ini penulis mengawali dengan menelaah dan menganalisa penelitian – penelitian terdahulu yang berkaitan dengan penelitian yang dilakukan penulis. Hal ini penulis lakukan untuk mendukung dan mempertajam penelitian dan penulisan laporan penelitian ini sendiri. Berikut ringkasan penelitian terdahulu yang mempunyai kaitan dengan penelitian penulis.

Penelitian yang dilakukan oleh Ngurah Manik Mahardika dkk. dengan judul Pengembangan Sistem Informasi Layanan Sistem Kredit Poin Berbasis Website di BEMFTK Universitas Pendidikan Ganesha bertujuan untuk meningkatkan kompetensi mahasiswa secara akademik maupun non-akademik serta mengukur keikutsertaan mahasiswa pada organisasi atau perlombaan. Poin Sistem Kredit Poin ini juga akan dijadikan syarat untuk keperluan Tugas Akhir (TA/Skripsi). Dengan adanya Sistem Kredit Poin ini, mahasiswa juga dituntut untuk aktif tidak hanya di kampus, namun juga untuk aktif di lingkungan organisasi serta perlombaan.[1]

Persamaan antara penelitian ini dengan penelitian di atas adalah penelitian sama – sama memiliki tujuan untuk meningkatkan prestasi pelajar serta memiliki parameter – parameter yang sama untuk menentukan Skor Sistem Kredit Poin.

Perbedaan antara penelitian ini dengan penelitian di atas adalah perbedaan dan kebutuhan objek penelitian yang berbeda dengan cara pendekatan yang berbeda pula. Penelitian di atas berfokus hanya kepada mahasiswa dimana mahasiswa akan dituntut untuk terus aktif sehingga Poin Sistem Kredit Poin nantinya terkumpul

sebagai persyaratan Tugas Akhir. Sedangkan penelitian ini berfokus sebagai sistem yang membantu guru untuk melakukan evaluasi dari informasi yang disediakan oleh sistem nantinya.

Pada penelitian lainnya yang dilakukan oleh Ririn Jaslita dan Armida S. dengan judul Analisis Persepsi dan Harapan Mahasiswa Terhadap Implementasi SKPI dengan Sistem Kredit Poin di Fakultas Ekonomi Universitas Negeri Padang mengungkapkan bahwa saat ini mahasiswa beranggapan bahwa tujuan kuliah adalah untuk mendapatkan IPK yang tinggi agar mudah mendapatkan pekerjaan. Menurut survei NACE (2002) yang dikutip oleh Efindri (2010:156), indeks prestasi yang mencerminkan penguasaan *hard skill* menempati peringkat ke-16 dari 19 jenis kemampuan yang dibutuhkan di pasar kerja. Untuk menghadapi dunia kerja, penguasaan *soft skill* tidak kalah pentingnya dengan penguasaan *hard skill*. Dengan adanya Sistem Kredit Poin, diharapkan dapat meningkatkan dan mengembangkan kegiatan mahasiswa.[2]

Penelitian lainnya yang ditulis oleh Myrna Dwi Rahmatya dan Muhammad Faris dengan judul Pengembangan Sistem Informasi Akademik Berbasis Web Pada SMA Pasundan 2 Bandung mengungkapkan bahwa sistem informasi akademik seperti ini dapat membantu sekolah dalam mengolah data siswa, penjurusan dan penilaian.[3]

2.2 Teknologi Informasi

Teknologi Informasi adalah suatu teknologi yang akan menghasilkan suatu informasi dari proses pengolahan data mulai memperoleh data, *data preprocessing*, manipulasi data, dan penyimpanan data. Suatu informasi dapat dinilai relevan dan

akurat dapat ditentukan dari data yang diolah tersebut. Semakin baik kualitas data, maka semakin baik pula ketepatan informasi yang akan dihasilkan.[4]

2.3 Sistem Informasi

Sistem Informasi merupakan sebuah sistem yang dibangun oleh sebuah organisasi atau perusahaan yang dapat menghasilkan suatu keputusan yang tepat, cepat, dan jelas yang terdiri dari seperangkat fungsi operasional. Suatu sistem dapat memproses data sesuai dengan kebutuhannya yang merupakan kombinasi dari orang – orang, fasilitas, teknologi, dan prosedur – prosedur pengendalian.[5]

2.4 Sistem Kredit Poin

Sistem Kredit Poin merupakan sistem yang dapat membantu siswa untuk mengasah *hard skill* dan *soft skill*. Sistem Kredit Poin akan mencatat kegiatan siswa dan mengakumulasiannya dalam bentuk poin. Dengan kegiatan tersebut, Sistem Kredit Poin mendorong siswa untuk meningkatkan potensinya dengan mengembangkan aktivitas dan kreativitas siswa.[2]

2.5 Metode Pengembangan *Prototyping*

Prototyping merupakan metode pengembangan sistem dengan model fisik yang berfungsi sebagai versi awal sistem. Dengan metode *prototyping* ini akan dihasilkan suatu sistem *prototype* yang dapat diuji langsung oleh pengembang dan pengguna. *Prototype* dibangun berdasarkan kebutuhan awal yang sudah didefinisikan terlebih dahulu agar pengembangan aplikasi tetap sesuai dengan rencana awal.

Prototyping digunakan dengan tujuan untuk mengumpulkan informasi sebanyak mungkin dari setiap versi prototype yang diluncurkan sehingga akan semakin tergambar nya versi akhir dari sistem yang lebih besar nantinya.[6]

2.6 Pemrograman Berorientasi Objek

Pemrograman berorientasi objek atau *object-oriented programming (OOP)* merupakan teknik pemrograman yang menggunakan pendekatan objek. Dengan adanya OOP, pengembang dapat membangun sistem yang bersifat *reliable, reuseable, maintainable, dan extendable*. Waktu pembangunan suatu sistem dapat dipangkas dengan adanya *class*. *Class* merupakan pemodelan objek dengan karakteristik dan fungsi yang sudah dideskripsikan dan dapat diubah sesuai dengan kebutuhan sistem.[7]

2.7 Use case Diagram

Use case adalah seperangkat skenario yang diikat bersama oleh user untuk mencapai tujuan (Fowler & Scott, 1993). Pemodelan *use case* ini dibangun agar skenario interaksi sistem dan aktor (user) dapat dipahami dengan lebih mudah. Aktor pada *use case* nantinya dapat melakukan interaksi secara langsung dengan sistem. Untuk menghubungkan simbol *use case diagram* dapat menggunakan *include* (untuk menghubungkan *use case* yang bersifat wajib) dan *extend* (untuk *use case* yang bersifat opsional dan muncul setelah *use case* dasar).[8]

2.8 Activity Diagram

Activity diagram merupakan salah satu diagram pemodelan yang menggambarkan proses kerja suatu sistem. *Activity diagram* tidak mendeskripsikan aksi dari aktor atau dengan kata lain *activity diagram* hanya dapat mendeskripsikan

proses kerja suatu sistem. *Activity diagram* menggambarkan proses suatu sistem dengan vertikal yang merupakan menggambarkan rincian dari *use case* yang telah dibuat.[9]

2.9 Class Diagram

Class diagram merupakan penggambaran struktur sistem dengan metode berbasis objek. *Class diagram* suatu sistem dengan class serta atribut, metode dan hubungan antar objek. Hal ini menjadikan class diagram banyak digunakan karena kemudahannya untuk dipetakan ke bahasa pemrograman berbasis objek.[10]

2.10 Sequence Diagram

Sequence diagram merupakan diagram yang berfungsi untuk mendeskripsikan interaksi antar objek dalam suatu sistem. *Sequence diagram* juga mendeskripsikan pesan atau perintah serta waktu saat proses dieksekusi.[11]

2.11 Entity Relationship Diagram

Entity Relationship Diagram atau ERD merupakan pemodelan diagram yang digunakan untuk mendeskripsikan data dan hubungan antar entitas atau objek pada suatu sistem yang digunakan dengan tujuan membantu pembuatan konsep desain *database*. Pada setiap entitasnya, digambarkan pula *attribute* serta *key* yang ada di entitas tersebut, kemudian setiap entitas dapat dihubungkan.[12]

2.12 Black-box Testing

Black Box Testing merupakan suatu metode pengujian sistem yang didasarkan pada spesifikasi fungsional dari perangkat lunak. Penguji akan melakukan pengujian terhadap suatu sistem dengan memastikan fungsi berjalan dengan baik mulai dari pengumpulan kondisi input yang dibutuhkan agar dapat

menghasilkan output yang diharapkan. Pengujian ini juga tanpa mengetahui struktur kode dibalik sistem yang dibangun.[13]

2.13 SQL

SQL (*Structured Query Language*) adalah merupakan bahasa pemrograman yang dikembangkan oleh IBM dan digunakan untuk menjalankan sintaks yang berhubungan dengan database. Sudah banyak vendor DBMS (Database Management System) yang menggunakan SQL sebagai sintaks dasar mereka seperti Oracle, Microsoft SQL Server, MySQL, dan PostgreSQL.[14]

2.14 PostgreSQL

PostgreSQL merupakan *open-source database* dan dapat mendukung untuk digunakan dengan sintaks SQL serta memiliki fitur seperti *transactions*, *subqueries*, *triggers*, dan lain – lain. PostgreSQL juga merupakan database yang bersifat ORDBMS atau Object Relational Database Management System.[15]

2.15 Javascript

Javascript merupakan bahasa pemrograman yang dibangun oleh Netscape pada tahun 1995 dengan tujuan untuk membantu pengembangan browser mereka. Awalnya javascript merupakan bahasa pemrograman yang diperkenalkan untuk menjalankan sintaks di browser pada dokumen HTML. Untuk menjalankan javascript, hanya dibutuhkan browser tanpa membutuhkan *compiler*. [16]

2.16 Vue JS

Vue (dibaca /vju:/, seperti *view*) merupakan framework javascript yang digunakan untuk membangun tampilan *User Interface* dari suatu website. Vue merupakan suatu framework yang bersifat progresif yang dirancang secara bertahap

dengan sistem *component-based* yang membantu pengembang untuk membangun suatu sistem seefisien mungkin.[17]

2.17 Golang

Golang merupakan bahasa pemrograman *open-source* yang dikembangkan oleh Google. Golang dibangun dengan tujuan menciptakan bahasa pemrograman yang unggul dari kecepatan, skabilitas, dan kesederhanaan. Golang juga merupakan bahasa pemrograman yang dibangun dari bahasa pemrograman lainnya yaitu C.[18]

2.18 Gin

Gin merupakan framework golang yang yang digunakan untuk membangun sistem backend yang memudahkan pembangunan sistem dengan beragam fitur yang tersedia dengan performa yang baik.[19]

2.19 Application Programming Interface (API)

Application Programming Interface atau API merupakan suatu teknologi *interface* yang dapat menghubungkan satu device atau lebih untuk bertukar data antar sistem. Dengan adanya API ini memudahkan para pengembang sistem untuk saling berkolaborasi dan menghubungkan sistem yang dibangun.[20]