

BAB 2 TINJAUAN PUSTAKA

2.1 Studi Terdahulu

Dalam penelitian ini, terdapat penelitian terkait dari peneliti sebelumnya tentang Sistem Informasi Skripsi (Resri, 2013), dalam skripsinya yang berjudul Pengembangan Sistem Skripsi (TheSIS) FTI Universitas YARSI Berbasis Web Serta Tinjauannya Menurut Agama Islam. Dalam skripsinya tersebut peneliti melakukan pengembangan sistem informasi skripsi (TheSIS) yang bertujuan untuk mendukung kinerja suatu organisasi sehingga proses dalam aplikasi dapat berjalan dengan baik dan lancar. Proses bisnis dalam sistem TheSIS bukan merupakan proses bisnis kurikulum yang berlaku saat ini. Sehingga untuk mendukung proses kegiatan skripsi pada kurikulum saat ini masih ada beberapa modul yang kurang dan dalam penyesuaian proses bisnis kurikulum saat ini.

Adapun juga Sistem Informasi Akademik Universitas YARSI berbasis web yang berfungsi untuk menampilkan atau mencetak laporan akademik seperti nilai mahasiswa yang sering di sebut KHS (Kartu Hasil Sementara), rencana studi mahasiswa yang biasa disebut KRS (Kartu Rencana Studi), dan yang lainnya. Dalam beberapa fungsi tersebut dapat memudahkan civitas Universitas YARSI dalam kegiatan akademik mahasiswa.

2.2 Metode Pengumpulan Data

Banyak metode dalam pengumpulan data yang dapat dilakukan dalam sebuah penelitian. Metode pengumpulan data dapat dilakukan secara sendiri-sendiri ataupun dengan menggabungkan dua metode atau lebih. Beberapa metode pengumpulan data sebagai berikut:

2.2.1. Kuesioner

Kuesioner merupakan suatu metode pengumpul data yang berisikan suatu pernyataan yang harus di jawab oleh subjek secara tertulis dengan alat pengukur yang digunakan seperti skala *likert* (Anisa, 2016).

2.2.2. Wawancara

Wawancara merupakan metode pengumpul data atau alat pengumpul data yang menunjukkan peneliti sebagai pewawancara mengajukan sejumlah pertanyaan pada partisipan sebagai subjek yang diwawancarai (Galang, 2016).

2.2.3. Observasi

Observasi merupakan metode pengumpul data yang dilakukan oleh peneliti dalam *setting* alamiah dengan tujuan menggali suatu makna. Selama proses observasi, peneliti perlu membuat *field notes* selama dan sesudah proses observasi berkenaan dengan peristiwa penting yang ada dalam subjek penelitian (Galang, 2016).




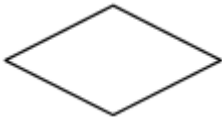
2.3 Desain Perancangan Aplikasi


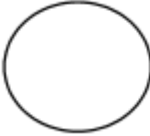

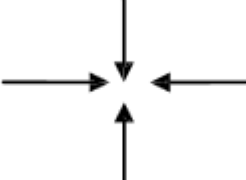




Desain perancangan aplikasi merupakan rangkaian kegiatan yang dimulai sebelum implementasi, tujuannya untuk tercapainya sebuah aplikasi yang diinginkan. Adapun perancangannya meliputi:

2.3.1. Flowchart

Flowchart adalah sekumpulan simbol-simbol yang menggambarkan urutan langkah-langkah kegiatan program dari awal hingga akhir. Simbol-simbol flowchart menurut Rachman (2018) dapat dilihat pada Tabel 1.

Tabel 1. Simbol Flowchart

Simbol	Nama	Fungsi
	<i>Terminator</i>	Untuk memulai dan mengakhiri suatu program.
	<i>Process</i>	Suatu simbol yang dapat menunjukkan setiap pengolahan yang dilakukan oleh komputer.
	Data	Untuk memasukkandata maupun menunjukkan hasil dari suatu proses.
	<i>Decision</i>	Suatu kondisi yang akan menghasilkan beberapa kemungkinan jawaban atau pilihan.



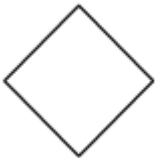

	<i>Preparation</i>	Untuk menyediakan tempat-tempat pengolahan data dalam <i>storage</i> .
	<i>Connector</i>	Untuk masuk dan keluarnya suatu prosedur pada lembar kertas yang sama.
	<i>Off-page Connector</i>	Untuk masuk dan keluarnya suatu prosedur pada lembar kertas yang lain.
	<i>Flow</i>	Prosedur yang dapat dilakukan dari atas ke bawah, kiri ke kanan atau sebaliknya.
	<i>Document</i>	Untuk data yang berbentuk informasi.
	<i>Predefined Process</i>	Untuk sekumpulan langkah proses yang ditulis sebagai prosedur.
	<i>Display</i>	Untuk output yang ditunjukkan suatu <i>device</i> , seperti monitor.
	<i>Magnetic</i>	Untuk menyimpan data

2.3.2. Entity Relationship Diagram

Menurut Rachman (2018) entity relationship diagram adalah diagram yang berisi komponen-komponen himpunan entitas dan himpunan relasi yang

masing-masing dilengkapi dengan atribut yang merepresentasikan seluruh fakta yang ditinjau. Simbol-simbol ERD dapat dilihat pada Tabel 2.

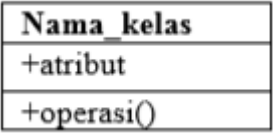






Tabel 2. Simbol ERD

Simbol	Nama	Keterangan
	Entitas	Menunjukkan himpunan entitas yang merupakan suatu obyek yang dapat diidentifikasi dalam lingkungan pengguna.
	Aliran Data	Menunjukkan aliran atau arus data dimana informasi sedang melintas atau menuju ke suatu proses.
	Proses	Menunjukkan suatu proses yang dilakukan dimana beberapa tindakan dijalankan.
	Atribut	Menunjukkan atribut yang dapat berfungsi untuk memberikan deskripsi karakter entitas.

2.3.3. Class Diagram

Class diagram menggambarkan struktur sistem dari segi pengertian kelas-kelas yang akan dibuat untuk membangun sebuah sistem informasi. Kelas memiliki atribut dan metode atau operasi (Winda dan Umi, 2016). Simbol class diagram dapat dilihat pada table 3.






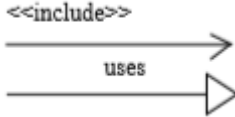
Tabel 3. Simbol class diagram

Simbol	Nama	Deskripsi
	Kelas	Kelas pada struktur sistem
	Antarmuka	Sama dengan konsep antarmuka dalam pemograman berorientasi objek.
	Asosiasi	Relasi antarkelas dengan makna umum, asosiasi biasanya disertai dengan <i>multiplicity</i> .
	Asosiasi berarah	Relasi antarkelas dengan makna kelas yang satu digunakan kelas lain.
	Generalisasi	Relasi antarkelas dengan makna generalisasi.
	<i>dependency</i>	Relasi antarkelas dengan makna kebergantungan antarkelas.
	Agregasi	Relasi antarkelas dengan makna semua bagian.

2.3.4. Use Case Diagram

Use case diagram merupakan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat (Winda dan Umi, 2016). Terdapat beberapa simbol dalam menggambarkan use case diagram. Simbol use case diagram dapat dilihat pada table 4.

Tabel 4. Simbol Use Case Diagram

Simbol	Nama	Deskripsi
	<i>Use Case</i>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit dan aktor.
	Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat.
	Asosiasi	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi.
	<i>Extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambah dapat berdiri sendiri.
	Generalisasi	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> untuk menjalankan fungsinya.

2.4 Perangkat Lunak Sistem

Dalam mengembangkan suatu aplikasi atau sistem seperti TheSIS, terdapat beberapa perangkat lunak yang digunakan meliputi:

2.4.1. Hypertext Markup Language (HTML)

HTML adalah sebuah bahasa markah yang digunakan untuk membuat sebuah halaman *web*, menampilkan berbagai informasi di dalam sebuah penjelajah *web*.

HTML bukanlah bahasa pemrograman, tetapi bahasa *markup*, dalam HTML tidak akan ditemukan struktur yang biasa di temukan dalam bahasa pemrograman seperti *IF*, *LOOP*, maupun *variable*.

Fungsi dari HTML di antaranya untuk membuat sebuah halaman *website* dan juga menampilkan berbagai informasi yang di masukkan ke dalam HTML tersebut seperti gambar, teks, video, dan lainnya sehingga bisa ditampilkan melalui *web browser* (Syaripuddin, 2018).

2.4.2. Hyper Preprocessor (PHP)

PHP merupakan sebuah bahasa pemrograman yang berupa kode atau *script* yang bisa ditambahkan ke dalam bahasa pemrograman HTML, PHP itu sering digunakan untuk merancang, membuat, dan juga memprogram sebuah *website*.

PHP digunakan pada halaman *website* untuk mengolah isi data dari *website* tersebut yang akan dilihat pengguna dari *website* tersebut. PHP memiliki kegunaan juga untuk membuat sebuah halaman utama atau *homepage* dari sebuah *website* menjadi lebih bagus dan bersifat dinamis serta elegan (Syaripuddin, 2018).

2.4.3. Cascading Style Sheet (CSS)

CSS merupakan kumpulan perintah yang dibentuk dari berbagai sumber yang disusun menurut urutan tertentu sehingga dapat mengatasi masalah *style* atau bahasa yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa *markup*. Fungsi CSS itu sendiri untuk merancang, mengubah, mendesain, dan membentuk halaman pada *website* (Purwadi dan Alimudin, 2016).

Adapun kelebihan menggunakan CSS :

- a. Jika kita ingin mengubah tema halaman *web*, cukup modifikasi pada *css* saja.
- b. Satu *CSS* dapat digunakan banyak halaman *web*.

Kekurangan menggunakan *CSS* :

- a. Terkadang juga ada browser yang tidak support *CCS*.
- b. Dibutuhkan waktu lebih lama dalam membuatnya.

2.4.4. My Structure Query Language (MySQL)

MySQL merupakan sebuah sistem manajemen berbasis data *SQL* yang sangat terkenal dan bersifat *open source*. MySQL bekerja dalam berbagai *platform*.

MySQL memiliki *library* yang dapat ditempelkan pada aplikasi yang berjalan sendiri, sehingga dapat digunakan pada komputer yang tidak mempunyai jaringan. MySQL juga dapat menangani basis data dalam skala besar, basis data dalam *server* MySQL dapat berisi 50 juta *record* (Purwadi dan Alimudin, 2016).

2.5 Usability Testing

Usability (kegunaan) merupakan bagian dari bidang ilmu *Human Computer Interaction (HCI)*. *Human Computer Interaction* merupakan bidang ilmu yang mempelajari tentang mendesain tampilan layar komputer dalam suatu aplikasi sistem informasi agar nyaman dipergunakan oleh pengguna (Dedi, 2014).

Usability adalah sejauh mana suatu sistem dapat digunakan oleh pengguna tertentu untuk mencapai target yang ditetapkan dengan efisiensi dan mencapai kepuasan penggunaan dalam konteks tertentu. Konteks penggunaan terdiri dari pengguna, tugas, peralatan (*hardware, software, dan material*). Berdasarkan definisi tersebut *usability* diukur berdasarkan komponen:

- a. Kemudahan (*learnability*) didefinisikan seberapa cepat pengguna mahir dalam menggunakan sistem serta kemudahan dalam penggunaan menjalankan suatu fungsi dan apa yang pengguna inginkan dapat mereka dapatkan.

- b. Efisiensi (*efficiency*) didefinisikan sebagai sumber daya yang dikeluarkan guna mencapai ketepatan dan kelengkapan tujuan.
- c. Mudah diingat (*memorability*) didefinisikan bagaimana kemampuan pengguna mempertahankan pengetahuannya setelah jangka waktu tertentu.
- d. Kesalahan dan kemandirian (*errors*) didefinisikan berapa banyak kesalahan-kesalahan apa saja yang dibuat pengguna.
- e. Kepuasan (*satisfaction*) didefinisikan kebebasan dari ketidaknyamanan, dan sikap *positif* terhadap penggunaan sistem.

2.5.1. Pengukuran *Usability* dengan *Use Questionnaire*

Kuesioner yang dapat digunakan untuk mengukur *usability* adalah *Computer System Usability Questionnaire* (CSUQ) dan *Usability Metric for User Experience* (UMUX). Beberapa aspek pengukuran *usability* yaitu efisiensi dan kepuasan. Kuesioner dibuat dalam bentuk skor tujuh point dengan model skala *likert*, untuk pengukuran tingkat persetujuan pengguna terhadap pernyataan hasil pengukuran, kemudian dilakukan analisis terhadap masing-masing parameter (Lewis, 2018).

Didalam CSUQ ada 16 item pertanyaan yang menghasilkan empat skor, yang ke-1 (satu) secara keseluruhan pertanyaan, dan yang ke-2 (dua) sampai yang ke-4 (empat) merupakan subskala (Lewis, 2018). Aturan untuk komputasi CSUQ adalah :

1. Secara keseluruhan : rata-rata tanggapan untuk item 1 – 16 (semua item)
2. Sistem Kegunaan (SysUse) : rata-rata item 1-6
3. Kualitas Informasi (InfoQual) : rata-rata item 7-12
4. Antarmuka Kualitas (IntQual) : rata-rata item 13 – 16

Sedangkan UMUX merupakan kuesioner yang relatif singkat dan dirancang untuk mendapatkan kegunaan yang dirasakan oleh pengguna, tetapi hanya menggunakan 4 (empat) pertanyaan. UMUX item bervariasi dalam nada, memiliki tujuh langkah skala 1 (sangat tidak setuju) sampai 7 (sangat setuju) dengan skor item dimanipulasi untuk mendapatkan skor keseluruhan yang berkisar dari 0 - 100. Ada beberapa tingkatan yang diinginkan untuk UMUX, yaitu :

1. Keandalan (Koefisien alpha lebih besar dari 0,80).
2. Validitas konkuren (korelasi dengan SUS lebih besar dari 0,55).
3. Kepekaan terhadap berbagai tingkat variabel independen.

2.6 Grounded Theory

Grounded Theory adalah sebuah teknik analisa dalam mempelajari pengalaman masyarakat dengan sebuah proses dan membangun sebuah teori untuk menjelaskan proses itu bekerja (Simon, 2013). Ada beberapa tahapan dalam menggunakan metode ini, sebagai berikut:

- a. *Open Coding*, sebuah proses dalam membentuk kategori untuk setiap *code*.

Ada beberapa tahapan dalam membuat *open coding* diantaranya:

1. Membuat *code*.
2. Membuat konsep dan mengelompokkan *code* ke konsep yang relevan.
3. Membuat kategori dan mengelompokkan konsep ke kategori yang relevan.

- b. *Axial Coding*, proses untuk menemukan relasi antara kategori.

Ada beberapa tahapan dalam membuat *axial coding* diantaranya:

1. Membuat Fenomena.
2. Membuat *causal condition*, kondisi yang terjadi dilingkungan nyata.
3. Membuat *intervene condition*, strategi aksi yang dapat dilakukan berdasarkan kondisi yang ada.
4. Membuat konsekuensi.

- c. *Selective Code*, Melakukan proses iterasi *open coding* dan *axial coding* hingga semua *code* sudah terdefinisikan.
- d. *Output Grounded Theory*, sebuah keluaran dari hasil analisa *Grounded Theory*.