

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Penelitian

Berikut ini merupakan tinjauan penelitian dari lima (5) jurnal yang berbeda:

Tabel 2.1 Perbandingan terhadap penelitian terdahulu

Judul	Identitas Peneliti	Metode	Tujuan	Kekurangan penelitian	Kesimpulan
Pembangunan Sistem Informasi Perpustakaan Berbasis Web menggunakan Metode <i>Rapid Application Development</i> (RAD) (Studi pada : SMK Negeri 11 Malang)	Nur Aini, Satrio Agung Wicaksono, Issa Arwani j-ptiik, Vol. 3, No. 9, September 2019, hlm. 8647-8655	Metode <i>Rapid Application Development</i> (RAD)	Tujuan dari penelitian ini yaitu dibuatnya suatu aplikasi SI perpustakaan berbasis web agar staff perpustakaan dapat melakukan pengelolaan data-data perpustakaan, sistem tersebut bisa diakses kapanpun dan dimanapun.		Sistem informasi perpustakaan (SIPS) berbasis web pada SMKN 11 Malang terdiri dari 19 kebutuhan fungsional. Dan sistem dapat diakses oleh admin, siswa, guru, dan juga staff SMKN 11 Malang. SIPS dinyatakan baik digunakan oleh pengguna karena hasil pengujian yang akurat dan memenuhi kebutuhan pengguna
Perancangan Sistem Informasi Perpustakaan Berbasis Web Untuk Sma Islam Sunan Gunung Jati	Nur Fatimah, Yandria Elmasari. JIPI, Volume 03, Nomor 02, Desember 2018 : 130 - 137	Metode <i>research and Development</i> (R&D) dengan metode pengembangan <i>Extreme Programmin g</i> .	Tujuan dari penelitian ini yaitu dibuatlah suatuaplikasi SI perpustakaan yang dapat membantu pegawai dalam memberikan informasi tentang perpustakaan	Perancangan Sistem Informasi Perpustakaan Berbasis Web Untuk Sma Islam Sunan Gunung Jati	Berdasarkan hasil penelitian dan pembahasan yang telah diuraikan pada bab sebelumnya dapat ditarik kesimpulan yaitu sistem informasi perpustakaan berbasis web yang dikembangkan dalam penelitian ini layak digunakan dengan didukung oleh hasil pengujian <i>usability</i> dengan hasil sebesar 84,22 % (kriteria sangat layak) pada kelompok kecil dan 91,2 % (kriteria sangat layak) pada kelompok besar, serta <i>performance efficiency</i> dengan hasil sebesar 98,64 (<i>grade A</i>).

Rancang Bangun Sistem Informasi Perpustakaan Berbasis Web (Studi Kasus: Universitas Kuningan)	Aji Permana, M.Kom. Jurnal <i>Cloud Information</i> , Volume 3, Nomor 2 Maret 2018	-	Tujuan dalam penelitian ini yaitu untuk membuat suatu aplikasi yang dapat mengontrol data-data buku dari setiap fakultas.		Dalam penelitian ini telah dikembangkan model sistem informasi perpustakaan yang terintegrasi dengan sistem akademik versi baru di Universitas Kuningan, yang dapat membantu Mengelola anggota perpustakaan(mahasiswa/ karyawan).
Pengembangan Sistem Informasi Perpustakaan Berbasis Web Pada Perpustakaan Kecamatan Bungah	Nurur Rohmah, Himawat Aryadita, Adam Hendra Brata. j-ptiik, Vol. 3, No. 3, Maret 2019, hlm. 2225-2234	Metode <i>waterfal</i>	Membangun sistem informasi perpustakaan berbasis web	-	Hasil analisis kebutuhan fungsional digunakan untuk membuat diagram <i>use case</i> dan <i>use case</i> skenario yang dapat dimanfaatkan untuk membuat perancangan sistem. <i>Acceptance testing</i> dilakukan dengan metode <i>black box</i> oleh <i>stakeholder</i> dengan hasil menunjukkan bahwa semua fitur pada sistem berhasil dijalankan dan telah diterima oleh <i>stakeholder</i> .
Sistem informasi <i>inventory</i> buku perpustakaan berbasis <i>website</i> di kejaksaan tinggi jawa tengah	Khodijah Allaila, B.A Herlambang	Metode <i>waterfall</i>	Dibangunnya sebuah sistem informasi <i>inventory</i> perpustakaan berbasis <i>website</i>		Perpustakaan Kejaksaan Tinggi Jawa Tengah ini adalah gebrakan baru yang dibuat oleh perpustakaan Kejaksaan Tinggi Jawa Tengah. Diharapkan dengan adanya sistem ini dapat mempermudah layanan yang diberikan oleh perpustakaan kejaksaan tinggi jawa tengah kepada setiap pegawai/karyawan di lingkungan kejaksaan tinggi jawa tengah.

2.2 Dasar Teori

2.2.1 SMK Negeri 1 Atambua

SMK Negeri 1 Atambua merupakan satu kesatuan yang terpadu dari semua

satuan dan merupakan kegiatan pendidikan yang berkaitan antara satu dengan lainnya untuk mengusahakan tercapainya tujuan pendidikan di SMK. Sistem pendidikan yang diterapkan di SMK Negeri 1 Atambua menggunakan sistem ganda di mana siswa-siswi di SMK Negeri 1 Atambua melakukan praktek kerja di dunia usaha dan dunia industri (DU/DI) sesuai dengan program keahlian yang bersangkutan.

SMK Negeri 1 Atambua memiliki sebuah perpustakaan dengan buku yang lumayan lengkap, namun sistem pengelolaan pada perpustakaan masih dilakukan secara konvensional baik dalam hal pendaftaran anggota baru, rekapan daftar buku, peminjaman buku dan lainnya sehingga sangat memakan banyak waktu dan kurang efektif.

2.2.2 Pengertian Sistem Informasi

Berdasarkan definisi mengenai sistem dan informasi, maka dapat dinyatakan bahwa sistem informasi merupakan gabungan dari empat bagian utama. Pengertian sistem informasi menurut (Maydianto & Ridho, 2021), sistem informasi adalah sebuah hubungan dari data dan metode dengan menggunakan *hardware* serta *software* dalam menyampaikan sebuah informasi yang bermanfaat.

2.2.3 Pengertian Perpustakaan

Dalam bahasa Belanda perpustakaan disebut *bibliotheca* yang juga berasal dari bahasa Yunani *biblia* yang artinya tentang buku, kitab. Perpustakaan adalah sebuah ruangan berisi buku-buku yang tertata rapi di mana biasanya dapat dijumpai di sekolah maupun di kota-kota tertentu.

Buku dapat dibaca ditempat atau dipinjam dengan gratis oleh anggota dari perpustakaan tersebut dimana juga ada batas waktunya. Anggota dapat memperpanjang peminjaman ketika batas waktu peminjaman sudah habis. Hal tersebut akan terjadi secara kontinu. Selain buku-buku pelajaran, di sana juga ada koran, majalah, novel, dll (Fatimah & Elmasari, 2018).

Pengertian perpustakaan menurut Renatha dalam (Aini et al., 2019), menjelaskan bahwa perpustakaan adalah suatu sarana yang disediakan sekolah untuk memperoleh pengetahuan selain dari kegiatan belajar mengajar di sekolah dan universitas.

2.2.4 Sistem Informasi Perpustakaan

Sistem informasi perpustakaan menurut Siregar dalam (Aini et al., 2019) menjelaskan bahwa sistem informasi perpustakaan merupakan sistem yang dapat memenuhi segala pelayanan umum di perpustakaan seperti peminjaman buku dan lain sebagainya. Sehingga petugas perpustakaan dapat memanajemen perpustakaan dengan baik dari segi waktu, biaya, dan juga usaha.

2.3 Database

Menurut Indrajani dalam (Rani & Ahmad, 2018), *database* adalah kumpulan data yang saling berhubungan secara logis dan didesain untuk mendapatkan data yang dibutuhkan oleh suatu organisasi.

Menurut Sutanta dalam (Agung Suryadi & Andi Yulianto, 2020), basis data dapat dipahami sebagai suatu kumpulan data terhubung (*interrelated data*) yang disimpan secara bersama-sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangkapan data (kalaupun ada maka kerangkapan data tersebut harus

seminimal mungkin dan terkontrol (*controlled redundancy*), data disimpan dengan cara-cara tertentu sehingga mudah untuk digunakan atau ditampilkan kembali, data dapat digunakan satu atau lebih program-program aplikasi secara optimal, data disimpan tanpa mengalami ketergantungan dengan program yang akan menggunakannya, data disimpan dengan sedemikian rupa sehingga proses penambahan, pengembalian, dan modifikasi data dapat dilakukan dengan mudah dan terkontrol.

Di dalam suatu *database* terdapat atribut yang menunjukkan karakteristik dari entitas dan setiap atribut-atribut entitas terdapat satu atribut yang dijadikan sebagai kunci (*key*). Ada beberapa jenis kunci yaitu:

1. *Primary Key*, Atribut yang tidak hanya mengidentifikasi secara unik kemunculan pada sebuah entitas (*candidate key*).
2. *Candidate Key*, sebuah atribut yang dapat mengidentifikasi secara unik sebuah kemunculan sebuah entitas yang spesifik.
3. *Composite Key*, *Candidate Key* yang terdiri dari dua atribut atau lebih.
4. *Foreign Key*, atribut pada satu relasi yang cocok pada *candidate key* dari beberapa relasi.

Selain atribut terdapat juga kardinalitas, kardinalitas mendefinisikan jumlah kemunculan baik minimum ataupun maksimum satu entitas yang dapat dihubungkan dengan kemunculan tunggal entitas lain.

1. Satu ke satu (*one to one atau 1:1*)

Menunjukkan satu entitas hanya berhubungan dengan satu entitas lain. Misalnya hubungan antara rektor dan universitas, seorang rektor hanya dapat memimpin

pada satu universitas sedangkan dalam satu universitas dipimpin oleh seorang rektor.

2. Satu ke banyak (*one to many* atau *1:M*)

Menunjukkan satu entitas dapat berhubungan dengan banyak entitas lain. Misalnya hubungan antara telur dan induk ayam, sebutir telur hanya dihasilkan oleh seekor induk ayam sedangkan seekor induk ayam dapat menghasilkan banyak telur.

3. Banyak ke banyak (*many to many* atau *M:N*)

Menunjukkan bahwa banyak entitas dapat berhubungan dengan banyak entitas lain. Misalnya hubungan antara bisnis dan pengusaha, sebanyak M bisnis dapat dimiliki oleh N pengusaha, sedangkan M pengusaha memiliki N bisnis

2.4 Bahasa Pemrograman PHP

Menurut Adis Lena dalam (Lutfi, 2017), PHP adalah bahasa yang dirancang secara khusus untuk penggunaan pada Web. PHP adalah tool untuk pembuatan halaman web dinamis. Pada awalnya PHP merupakan kependekan dari *Personal Home Page* (Situs Personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama FI (*Form Interpreted*), yang wujudnya berupa sekumpulan script yang digunakan untuk mengolah data form dari web. Saat ini PHP adalah singkatan dari PHP:*Hypertext Preprocessor*, sebuah kepanjangan rekursif, yakni permainan kata di mana kepanjangannya terdiri dari singkatan itu sendiri: PHP: *Hypertext Preprocessor*.

2.5 Framework Laravel

Laravel adalah *framework* open source PHP berbasis web gratis yang dibuat

oleh Taylor Otwell dan ditujukan untuk pengembangan aplikasi web mengikuti *model-view controller* (MVC) atau pola arsitektur. Beberapa fitur dari Laravel adalah pengembangan sistem modul-modul yang dapat dimanajemen, mengenalkan cara yang berbeda untuk mengakses *database* relasional, utilitas yang membantu dalam penyebaran aplikasi dan pemeliharaan yang mudah. Di laravel ada banyak fungsi yang digunakan untuk berinteraksi dengan *database* seperti mengambil semua baris, mengambil baris dengan kata kunci utama, menggunakan klausul untuk menyaring baris, melakukan sisipan, melakukan *update*, dan lain-lain. Sehingga kebutuhan dalam membangun suatu sistem berbasis web dapat dilakukan dengan mudah. Awal maret 2015, laravel telah menjadi salah satu kerangka kerja PHP paling populer, bersama-sama dengan Symfony2, Nette, CodeIgniter dan Yii2.

2.6 MYSQL

Menurut Sianipar, R.H dalam (Lutfi, 2017), *MySQL* adalah salah satu aplikasi DBMS (*Database Management System*) yang sudah sangat banyak digunakan oleh para pemrogram aplikasi web. Dalam sistem *database* tak relasional, semua informasi disimpan pada satu bidang luas, yang kadangkala data di dalamnya sangat sulit dan melelahkan untuk diakses. Tetapi *MySQL* merupakan sebuah sistem *database* relasional, sehingga dapat mengelompokkan informasi ke dalam tabel-tabel atau grup-grup informasi yang berkaitan. Setiap tabel memuat bidang-bidang yang terpisah, yang mempresentasikan setiap bit informasi. *MySQL* menggunakan indeks untuk mempercepat proses pencarian terhadap baris informasi tertentu. *MySQL* memerlukan sedikitnya satu indeks pada tiap tabel. Biasanya akan

menggunakan suatu *primary key* atau pengenal unik untuk membantu penjejukan data.

2.7 Website

Menurut Dillon, Schonhaler, dan Vossen (2017 : 1), sejak awal 1990, *world wide web* atau *website* merevolusi kehidupan pribadi maupun *professional*. Web menjadi situs yang terus berkembang dan sebagai perpustakaan informasi yang ada di mana-mana yang dapat diakses melalui mesin pencari dan portal. Web menjadi tempat penyimpanan media yang memfasilitasi *hosting* dan berbagi sumber daya yang sering kali gratis dan sebagai pendukung layanan *do-it-yourself*. Web juga menjadi platform perdagangan tempat orang dan perusahaan semakin menjalankan bisnisnya.

Pengertian *website* menurut Sebok, Vermat, dan tim (2018:70) adalah kumpulan halaman yang saling terhubung yang di dalamnya terdapat beberapa *item* seperti dokumen dan gambar yang tersimpan di dalam *web server*. *Web app* adalah sebuah aplikasi yang berada dalam *web server* yang bisa *user* akses melalui *browser*. *Web app* biasanya menampilkan data *user* dan informasi dari server.

2.8 Unified modeling language (UML)



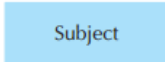

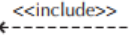


(Dennis et al., 2009) mengemukakan bahwa : *Unified Modeling Language (UML)* adalah cara untuk menampilkan representasi grafis dalam melakukan pemodelan sistem perangkat lunak dengan standar teknik pembuatan diagram.

2.8.1 Fitur Fitur Pada UML

1. Use case Diagram

(Rosa & Salahudin, 2015) *Use case* diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah apa yang diperbuat sistem, dan bukan bagaimana yang diperbuat di sistem. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dan sistem.


Tabel 2.2 Daftar simbol *use case* diagram







No.	Simbol	Nama	Keterangan
1.		<i>Actor</i>	Merepresentasikan peran seseorang atau sesuatu yang sedang berinteraksi dengan sistem.
2.		<i>Use case</i>	Serangkaian peristiwa yang terjadi ketika <i>actor</i> menggunakan sistem untuk menyelesaikan suatu proses.
3.		<i>Subject Boundary</i>	Mewakili keseluruhan sistem atau komponen utama.
4.		<i>Generalization</i>	Menunjukkan bahwa <i>use case</i> merupakan cara khusus untuk mencapai tujuan <i>use case</i> umum.
5.		<i>Include</i>	Menunjukkan bagaimana <i>use case</i> dipecah menjadi langkah-langkah yang lebih kecil.
6.		<i>Extend</i>	Menunjukkan bahwa satu <i>use case</i> dapat menambahkan fungsionalitas ke yang lain.
7.		<i>Association</i>	Menunjukkan hubungan antara <i>actor</i> dan <i>use case</i> .

2. Activity Diagram

Activity Diagram merupakan diagram yang merepresentasikan aliran kerja dari suatu proses bisnis perangkat lunak (Syarif & Nugraha, 2019). Berikut ini merupakan simbol *Activity* diagram :

Tabel 2.3 Daftar simbol *activity* diagram




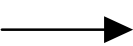

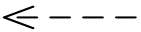
No	Simbol	Nama	Keterangan
1.		<i>Action</i>	Merepresentasikan setiap aktivitas atau tindakan.

2.		<i>Decision</i>	Untuk merepresentasikan keputusan sebelum memutuskan aliran kontrol.
3.		<i>Fork node</i>	Node kontrol yang membagi aliran menjadi beberapa aliran bersamaan.
4.		<i>Initial node</i>	Menggambarkan keadaan awal suatu sistem.
5.		<i>Final node</i>	Menggambarkan keadaan akhir suatu sistem.
6.		<i>Join node</i>	Node kontrol yang menyinkronkan banyak aliran.
7.		<i>Swimlane</i>	Mewakili banyak kategori informasi.

3. Sequence Diagram

(Pressman & Maxim, 2014) menyatakan bahwa : “*sequence* diagram merupakan diagram yang menampilkan komunikasi dinamis antara objek yang satu dengan yang lain dalam mengeksekusi sebuah tugas.”

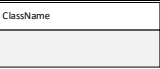

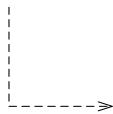

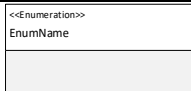
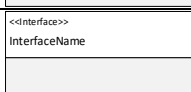
Tabel 2.4 Daftar simbol *sequence* diagram

No	Simbol	Nama	Keterangan
1		<i>actor</i>	Menggambar orang yang sedang berinteraksi dengan sistem.
2		<i>Object lifeline</i>	Menggambarkan pesan yang dikirim dan diterima untuk bisa menggambarkan sebuah aktivitas.
3		<i>Message self</i>	Menggambarkan hubungan antar objek itu sendiri untuk diurutkan sesuai dengan aktivitas yang dihasilkan.
4		<i>message</i>	Menggambarkan objek mengirim pesan ke objek lain
5		<i>activation</i>	Meggambarkan objek sedang aktif berinteraksi
6		<i>Return message</i>	Menggambarkan pengiriman balasan /keluaran pesan pada suatu objek

4. Class Diagram

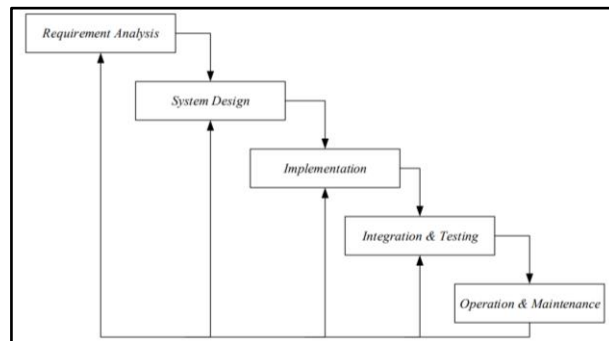
Menurut (Pressman & Maxim, 2014), *class* diagram yaitu diagram yang menampilkan sifat diam dari sistem dalam pemodelan kelas. Sedangkan menurut (Dennis et al., 2009) *class* diagram adalah diagram struktur yang merepresentasikan hubungan antar kelas yang satu dengan kelas yang lain pada pemodelan sistem.

Tabel 2.5 Daftar simbol *class* diagram

No	Simbol	Nama	Keterangan
1		<i>Class</i>	Menggambar himpunan dari objek-objek Yang berbagi atribut serta operasi yang sama
2		<i>Association</i>	Menggambarkan hubungan antara objek satu dan objek yang lainnya
3		<i>Dependency</i>	Menggambarkan hubungan di mana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
4		<i>Realization</i>	Menggambarkan operasi yang benar-benar dilakukan oleh suatu objek
5		<i>Enumeration</i>	Meggambarkan tipe data yang ditentukan pengguna
6		<i>Interface</i>	Menggambarkan sekumpulan atribut yang mendefinisikan sekumpulan perilaku yang berkaitan.

2.9 Metode *Waterfall*

Pada penelitian ini metode perancangan sistem yang digunakan adalah *System Development Life Cycle (SDLC)* dengan model *waterfall*. Model *waterfall* menyarankan pengembangan perangkat lunak secara sistematis dan berurutan yang dimulai dari tingkatan sistem tertinggi dan berlanjut ke tahap analisis, desain, pengkodean, pengujian dan pemeliharaan. Kelebihan dari metode ini adalah terstruktur, dinamis, dan *sequential*. Alur metode *waterfall* dapat dilihat pada gambar 2.1.



Gambar 2.1 Metode *waterfall*

Secara garis besar metode *waterfall* mempunyai tahapan-tahapan sebagai berikut:

1. *Requirement Analysis*

Seluruh kebutuhan *software* harus bisa didapatkan dalam fase ini, termasuk di dalamnya pengumpulan data melalui metode wawancara dan observasi terhadap keinginan pemakai nantinya.

2. *System Design*

Tahap ini dilakukan sebelum melakukan *coding*. Tahap ini melibatkan pegawai perpustakaan SMK Negeri 1 Atambua. Bertujuan untuk memberikan gambaran apa yang seharusnya dikerjakan, komponen apa saja yang diperlukan dan bagaimana tampilannya.

3. *Implementation*

Dalam tahap ini dilakukan pemrograman. Pembuatan *software* dipecah menjadi modul-modul kecil yang nantinya akan digabungkan dalam tahap berikutnya. Pada tahap ini perancangan sistem menggunakan Bahasa pemrograman PHP dengan *framework* laravel dan *MySQL* sebagai *database*.

4. *Integration & Testing*

Pada tahap ini dilakukan penggabungan modul-modul yang sudah dibuat dan dilakukan pengujian. Ini dilakukan untuk mengetahui apakah *software* yang dibuat telah sesuai dengan desainnya dan masih terdapat kesalahan atau tidak. Untuk metode pengujian yang dilakukan pada penelitian ini adalah metode pengujian langsung yaitu dengan menggunakan pengujian *black box*. Digunakan untuk menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang. Kebenaran perangkat lunak yang diuji hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan.

5. *Operation & Maintenance*

Ini merupakan tahap terakhir dalam model *waterfall*. *Software* yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Dalam tahap ini *updating* yang memungkinkan program untuk menyesuaikan diri dengan perubahan kondisi, *new functionality* dengan menambahkan fitur baru ke dalam sistem tanpa mengganggu proses yang sedang berjalan.