

## BAB II TINJAUAN PUSTAKA

### 2.1 Penelitian Terdahulu

Berikut penelitian-penelitian terdahulu yang dijadikan acuan untuk melakukan penelitian ini sebagai berikut:

#### 2.1.1 Penelitian Pertama

Penelitian terdahulu pertama berjudul Penerapan Sistem Informasi Manajemen Dalam Tata Kelola Perpustakaan Universitas Muhammadiyah Riau, untuk penjelasan lengkapnya dijelaskan pada tabel 2.1 dibawah ini.

**Tabel 2.1** Penerapan Sistem Informasi Manajemen Dalam Tata Kelola Perpustakaan Universitas Muhammadiyah Riau

<b>Judul Penelitian</b>	<b>Peneliti dan tahun Penelitian</b>
Penerapan Sistem Informasi Manajemen Dalam Tata Kelola Perpustakaan Universitas Muhammadiyah Riau	Aisyah Aura, Nur Azzahra Wedyati, Nikken Azzahara ZI, Nadia Fathurrahmi, SE., M. Ak (2021)
<b>Tujuan</b>	<b>Ruang lingkup</b>
Bagaimana implementasi sistem informasi manajemen yang digunakan pada perpustakaan UMRI	Meningkatkan layanan perpustakaan dan akses terhadap informasi perpustakaan
<b>Hasil Penelitian</b>	
Sistem pengelolaan layanan perpustakaan didasarkan pada sistem informasi yang memungkinkan terlaksananya seluruh kegiatan perpustakaan, mulai dari pendaftaran, peminjaman, pengembalian, penambahan, penggantian, dan pembuatan laporan	
<b>Perbedaan dengan penelitian yang akan dilakukan</b>	
Sistem yang dapat memudahkan layanan cek plagiarisme, permohonan cetak laporan TA/PKL dan pengumpulan karya	

### 2.1.2 Penelitian Kedua

Penelitian terdahulu kedua berjudul Penerapan Sistem Informasi Manajemen Dalam Pelayanan Perpustakaan di Dinas Perpustakaan dan Kearsipan Provinsi Banten, untuk penjelasan lengkapnya dijelaskan pada tabel 2.2 dibawah ini.

**Tabel 2.2** Penerapan Sistem Informasi Manajemen Dalam Pelayanan Perpustakaan di Dinas Perpustakaan Dan Kearsipan Provinsi Banten

Judul Penelitian	Peneliti dan tahun Penelitian
Penerapan Sistem Informasi Manajemen Dalam Pelayanan Perpustakaan di Dinas Perpustakaan dan Kearsipan Provinsi Banten	Elys Kusumawati, (2018)
Tujuan	Ruang lingkup
Mengetahui bagaimana penerapan Sistem Informasi Manajemen dalam layanan Perpustakaan di Dinas Perpustakaan dan Kearsipan Provinsi Banten	Penerapan Sistem Informasi Manajemen Sebagai Instrumen yang menerapkan Sistem Informasi Manajemen
Hasil Penelitian	
Layanan perpustakaan di Dinas Perpustakaan dan Kearsipan Provinsi Banten berjalan dengan baik berkat penggunaan Sistem Informasi Manajemen	
Perbedaan dengan penelitian yang akan dilakukan	
Peneliti menemukan bahwa teori yang digunakan dalam penelitian ini mempunyai kesamaan dengan penelitian sebelumnya. Namun, penelitian ini berbeda dalam hal sistem yang dikembangkan, yang dirancang khusus untuk membantu mahasiswa dalam mengajukan permintaan layanan cetak laporan serta pengumpulan karya	

### 2.1.3 Penelitian Pertama

Penelitian terdahulu ketiga berjudul Manajemen Layanan Perpustakaan *Online*, untuk penjelasan lengkapnya dijelaskan pada tabel 2.3 dibawah ini.

**Tabel 2.3** Penelitian Terdahulu Manajemen Layanan Perpustakaan *Online*

<b>Judul Penelitian</b>	<b>Peneliti dan tahun Penelitian</b>
Manajemen Layanan Perpustakaan <i>Online</i> di Institut Agama Islam Negeri Palangkaraya	Asma Wati, (2021)
<b>Tujuan</b>	<b>Ruang lingkup</b>
Mendeskripsikan Tentang perencanaan layanan perpustakaan <i>online</i> di Institut Agama Islam Negeri Palangkaraya, pelaksanaan layanan perpustakaan <i>online</i> dan evaluasi layanan perpustakaan <i>online</i> di IAIN Palangkaraya	Sistem Informasi merencanakan pengelolaan layanan perpustakaan <i>online</i> , melaksanakan layanan perpustakaan <i>online</i> dan mengevaluasi layanan perpustakaan <i>online</i>
<b>Hasil Penelitian</b>	
Hasil penelitian menunjukkan bahwa aplikasi <i>e-prints</i> , yang merupakan platform berbasis <i>website</i> , dapat diakses dengan mudah dan cepat untuk merencanakan layanan perpustakaan <i>online</i>	
<b>Perbedaan dengan penelitian yang akan dilakukan</b>	
Ada Kesamaan fitur yaitu <i>e-prints</i> . Namun pada fiturnya sistem cetak berbeda dengan penelitian terdahulu baik isi <i>form</i> maupun kebutuhannya dan juga <i>upload</i> karya mahasiswa	

## 2.2 Teori Terkait

### 2.2.1 PHP (*Hypertext Preprocessor*)

PHP, juga dikenal sebagai *Hypertext Preprocessor* adalah bahasa pemrograman sisi server yang digunakan untuk memproses kode program dan menerjemahkannya ke dalam kode mesin yang dapat dipahami komputer. Untuk membuat aplikasi *website* dinamis, PHP dapat diintegrasikan ke dalam HTML. PHP adalah bahasa pemrograman *opensource* yang digunakan untuk mengembangkan situs *website* sisi server. Sebagai *script* yang terintegrasi dengan HTML, PHP memungkinkan pembuatan halaman *website* yang sangat dinamis yang kontennya dibuat sesuai dengan permintaan klien (Bhakti, 2020).

*Hypertext Preprocessor* (PHP) adalah bahasa pemrograman yang biasa digunakan untuk membuat *scripting* di sisi server saat membangun aplikasi berbasis *website*. Meskipun terdapat banyak bahasa *scripting server-side* lain, PHP menjadi pondasi bagi berbagai aplikasi besar seperti *Facebook*, *Flickr*, dan *Yahoo*. PHP sering menjadi pilihan utama bagi pemula karena kemudahan penggunaannya dan dukungannya yang luas untuk pengembangan *website*.

Untuk membuat halaman *website* dinamis, PHP dapat menggunakan *database*. MySQL adalah sistem manajemen *database* yang paling umum digunakan adalah PHP, namun PHP juga mendukung sistem manajemen *database* seperti *Oracle*, *Microsoft Access*, *Interbase*, *dBase*, *PostgreSQL*, dan lainnya.

### **2.2.2 Laravel**

*Laravel* adalah sebuah *framework* PHP yang *open-source*, dirancang untuk mempermudah dan mempercepat pengembangan aplikasi *website*. *Framework* ini menawarkan berbagai fitur yang kuat dan alat-alat yang dibutuhkan untuk mengelola aspek-aspek teknis dalam pembuatan aplikasi *website* secara efisien.

*Laravel* adalah *framework* pengembangan *website* yang dibangun dengan PHP dan menggunakan pola arsitektur *Model View Controller* (MVC). Tujuan utamanya adalah untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan pemeliharaan serta meningkatkan pengalaman pengembangan aplikasi melalui sintaks yang ekspresif, jelas, dan efektif (Rahmat, 2019). Fitur-fitur modern *Laravel* meliputi *Bundles*, *Eloquent ORM* (*Object-Relational Mapping*), *Query Builder*, *Application Logic*, *Reverse Routing*, *Resource Controller*, *Class Auto Loading*, *View Composers*, *Blade*, *IoC*

*Containers, Migration, Database Seeding, Unit Testing, Automatic Pagination, Form Request, dan Middleware* adalah fitur modern *Laravel* yang sangat bermanfaat bagi pengembang dalam pembuatan aplikasi. Dengan menyediakan berbagai alat dan fitur canggih, fitur ini dimaksudkan untuk mempermudah pengembangan aplikasi (Irwan, 2013). Penggunaan *Laravel* membawa beberapa keuntungan, antara lain:

1. Struktur yang Terorganisir: *Laravel* menyediakan struktur yang terorganisir untuk pengembangan aplikasi, yang memungkinkan pengembang untuk fokus pada logika aplikasi tanpa harus memikirkan konfigurasi dasar.
2. Pemrograman yang Ekspresif: *Laravel* menyediakan sintaks yang ekspresif dan intuitif, membuatnya mudah dipahami dan digunakan oleh pengembang pemula maupun berpengalaman.
3. Pola Desain MVC: Pola ini digunakan oleh *Laravel* untuk membagi logika aplikasi dan tugasnya ke dalam komponen yang berbeda.
4. Fitur-fitur Terintegrasi: *Laravel* menyediakan berbagai fitur terintegrasi seperti sistem *routing* yang fleksibel, manajemen basis data dengan ORM (*Object-Relational Mapping*) menggunakan *Eloquent*, otentikasi pengguna, *caching*, dan masih banyak lagi.
5. Komunitas yang Aktif: *Laravel* memiliki komunitas pengembang yang luas dan dinamis, yang menyediakan dukungan, tutorial, dan sumber daya lainnya untuk membantu pengembang mempelajari dan menggunakan *framework* ini secara efektif.

Dengan demikian, *Laravel* dapat dianggap sebagai alat yang kuat dan efektif dalam pengembangan aplikasi *website*, yang memungkinkan pengembang untuk membangun aplikasi dengan cepat, aman, dan efisien (Aziz, 2020).

### 2.2.3 PostgreSQL

*Relational database management system (RDBMS) open-source PostgreSQL* sangat kuat dan canggih. Ini didukung oleh komunitas pengembang yang besar dan aktif serta telah digunakan secara luas dalam berbagai aplikasi, mulai dari aplikasi kecil hingga perusahaan skala besar. Berdasarkan lisensi BSD, *PostgreSQL* merupakan *database* yang tersedia secara gratis dan sangat populer bersama *MySQL* dan *Oracle* saat ini. *PostgreSQL* menawarkan banyak fitur bermanfaat untuk berbagai tujuan terkait *database*.

Sistem operasi seperti *Linux*, *UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64)*, dan *Windows* dapat menjalankan *PostgreSQL*. *PostgreSQL* juga memungkinkan penyimpanan biner besar-besaran termasuk *film*, musik, atau foto. Data hingga 32 TB dapat dikelola di *PostgreSQL*.

Antarmuka yang ditawarkan, salah satunya *PgAdmin* berfungsi untuk desain dan manajemen *database* menyeluruh, membantu seseorang dalam menggunakan *PostgreSQL*. *PostGIS*, *plugin backend database spasial*, dan *PgRouting*, penyedia kemampuan routing di *database PostgreSQL* dan mampu menangani masalah pencarian jalur terpendek, hanyalah dua dari banyak plugin lain untuk penggunaan *PostgreSQL* (Cookson & Stirk, 2019).

### 2.2.4 UML (Unified Modelling Language)

Bahasa standar *Unified Modeling Language* (UML) digunakan untuk mencatat, merancang, dan memodelkan sistem perangkat lunak. dan proses bisnis (Rumbaugh et al., 1999). UML adalah alat yang sangat berguna dalam pengembangan perangkat lunak karena menyediakan cara yang terstandarisasi untuk menjelaskan berbagai bagian dari sistem yang sedang dibangun (Sommerville, 2011).

Notasi grafis dalam *Unified Modeling Language* memfasilitasi para pengembang perangkat lunak dalam visualisasi struktur, perilaku, serta interaksi antar komponen dalam suatu sistem (Sutopo, 2012). Diagram aktivitas dalam UML menggambarkan aktivitas yang terlibat dalam suatu proses. Diagram *use case* memodelkan proses bisnis dan menjelaskan interaksi antara sistem dan pengguna. Diagram kelas menunjukkan struktur statis dari sistem dengan menampilkan kelas, atribut, dan hubungan antar kelas. Diagram urutan menampilkan interaksi antara objek dalam urutan peristiwa waktu tertentu. (Sommerville, 2011). Melalui penggunaan diagram ini pengembang dapat memodelkan sistem secara lengkap, memahami hubungan antar komponen, dan merancang sistem perangkat lunak dengan lebih baik.

Untuk memberikan gambaran menyeluruh tentang sistem, banyak model yang saling terintegrasi diperlukan. Ada beberapa keuntungan menggunakan UML:

- 1) UML menyediakan berbagai ide yang dapat digunakan secara rutin dan memanfaatkan pengetahuan bersama melalui mekanisme perluasan.

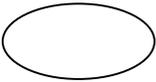
- 2) UML mudah digunakan dan dapat diterapkan untuk berbagai jenis sistem, baik perangkat lunak maupun non-perangkat lunak, serta berbagai domain, metode, dan proses.
- 3) Tidak tertutup, tidak eksklusif, dan bersifat terbuka, UML adalah standar industri untuk pemodelan.

### 2.2.5 Use Case Diagram

Diagram *use case* adalah diagram yang membantu Anda menggambarkan berbagai skenario penggunaan yang dikembangkan oleh sistem. Diagram ini menunjukkan apa yang harus dilakukan sistem tanpa merinci cara mencapainya. Diagram *use case* bertindak sebagai model kebutuhan sistem. yang menggambarkan fungsionalitas utama. Diagram ini terutama digunakan untuk memvisualisasikan skenario penggunaan dan interaksi antar aktor. (Steven, 2020).

Diagram *use case* digunakan untuk menggambarkan interaksi antara pengguna dan sistem. dan urutan proses yang dimiliki oleh sistem. Diagram ini menampilkan aktor, skenario penggunaan, dan koneksi di antara mereka sebagai rangkaian tindakan yang memberikan nilai yang dapat diukur kepada aktor. Dalam diagram UML, *Use Case Diagram* diwakili dengan *elips horizontal*.

**Tabel 2.4** Notasi *Use Case Diagram*

Simbol	Nama	Keterangan
	<i>Actor</i>	<i>Actor</i> adalah pelaku yang melakukan sesuatu atau mencapai tujuan di dalam sistem
	<i>Use Case</i>	Fungsi sistem yang akan dilakukan atau dikelola oleh pengguna atau sistem itu sendiri disebut sebagai <i>use case</i>

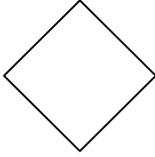
Simbol	Nama	Keterangan
	<i>Association</i>	<i>Association</i> adalah garis yang bersifat abstrak yang menjadi penghubung antar objek
<<include>>	<i>Include</i>	Sebuah garis yang menunjukkan suatu <i>use case</i> dimasukkan ke dalam <i>use case</i> lain disebut " <i>include</i> "
<<extend>>	<i>Extend</i>	hubungan antara dua kasus penggunaan di mana satu kasus penggunaan (kasus penggunaan yang diperluas) menambahkan perilaku opsional ke kasus penggunaan lainnya (kasus penggunaan yang diperluas), tergantung pada kondisi tertentu yang terjadi selama eksekusi

### 2.2.6 Activity Diagram

*Activity Diagram* atau Diagram Aktivitas, menggambarkan bagaimana aktor berinteraksi dengan sistem dengan mengilustrasikan aktivitas yang terlibat, objek yang terlibat, serta aliran proses yang mencakup keadaan (*state*), transisi antar keadaan, dan peristiwa. Diagram ini digunakan untuk menjelaskan operasi secara visual dan sebagai referensi dalam aktivitas sistem lainnya.

**Tabel 2.5** Notasi *Activity Diagram*

Simbol	Nama	Keterangan
	<i>Initial Node</i>	Simbol yang menunjukkan langkah awal kegiatan
	<i>Activity Final Node</i>	Simbol yang menunjukkan Langkah akhir kegiatan
	<i>Activity</i>	Simbol yang menunjukkan bagaimana aktor berinteraksi dengan sistem

Simbol	Nama	Keterangan
	<i>Decision</i>	Simbol yang digunakan ketika terjadi pemilihan keputusan tertentu
	<i>Connector</i>	Penghubung antara dua <i>item</i>

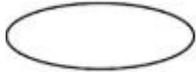
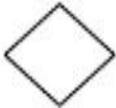
### 2.2.7 ERD (*Entity Relationship Diagram*)

Konsep dasar yang diterapkan dalam perancangan *database* adalah *Entity Relationship Diagram* (ERD). Komponen utama model ERD mencakup kumpulan entitas, himpunan relasi, dan integritas masalah. Dalam kumpulan entitas, objek di dunia nyata berbeda satu sama lain (Ridwan et al., 2022).

ERD menggambarkan model hubungan entitas-hubungan, yang mencakup konsep entitas, atribut, dan hubungan antara entitas tersebut. Dalam ERD, entitas berfungsi sebagai representasi objek atau benda dalam dunia nyata. Analis dapat menggunakan *modeling* ERD untuk menggambarkan kebutuhan dan persyaratan sistem, serta memvisualisasikan proses sistem dan komponennya. ERD memberikan gambaran tentang bagaimana entitas saling berhubungan dan atribut yang dimilikinya, tanpa langsung terkait dengan konsep proses bisnis (Adi & Maya Kristin, 2014).

**Tabel 2.6** Notasi *Entity Relationship Diagram*

Simbol	Nama	Keterangan
	Entitas	Mewakili suatu objek dan dapat dibedakan dengan objek lain

	Atribut	Dimiliki oleh suatu entitas tertentu dan dapat menggambarkan karakteristik entitas tersebut
Simbol	Nama	Keterangan
	<i>Relationship</i>	Menunjukkan keterkaitan antara berbagai entitas
	<i>Arrow</i>	Garis yang menghubungkan himpunan entitas, sifat, dan hubungan

### 2.2.8 CDM (*Conceptual Data Model*)

CDM (*Conceptual Data Model*) merupakan suatu pendekatan yang mewakili pandangan pengguna terhadap data yang disimpan dalam *database*. CDM disusun sebagai tabel tanpa menentukan tipe data dan menjelaskan hubungan antar tabel untuk implementasi basis data. CDM merupakan evolusi dari *Entity Relationship Diagram* (ERD). CDM menitikberatkan pada entitas, atribut, dan hubungan antar entitas tanpa memperhatikan detail teknis seperti tipe data atau *indeks*. CDM membantu memahami kebutuhan data organisasi dengan lebih baik dan menjadi dasar untuk merancang basis data yang efisien (Hutahaean Jeperson, 2015).

Menurut Imania (2019) menyatakan bahwa CDM (*Conceptual Data Model*) memiliki beberapa komponen penting yang harus diperhatikan. Entitas, hubungan, dan keturunan termasuk dalam kategori ini. Untuk membuat entitas, pilih opsi "Entitas" pada palet dan tempatkannya di area kerja. Opsi "Relasi" menghubungkan entitas satu sama lain, dan pengguna perlu mengakses *tab "General"* untuk memberi

nama relasi. Saat itu, *inheritance* dibuat dengan menghubungkan entitas induk dengan simbol *inheritance* dan memberikan nama pada *inheritance* yang baru dibuat.

### 2.2.9 PDM (*Physical Data Model*)

*Physical Data Model* (PDM) adalah kerangka kerja struktural berdasarkan tabel yang menggambarkan data dan hubungan antar data. dalam model ini didefinisikan oleh kolom-kolom yang masing-masing memiliki nama dan tipe data yang unik. PDM menyediakan pengetahuan mendalam tentang penyimpanan dan pengaturan data secara fisik dalam basis data, termasuk *indeks*, pembatasan, dan teknik akses. Model ini relevan untuk diterapkan karena memastikan bahwa nama tabel dan struktur yang digunakan sesuai dengan yang diterapkan dalam Sistem *Database Management System* (DBMS). PDM membantu dalam pengambilan dan pengelolaan data yang efektif di lingkungan basis data dengan memberikan gambaran teknis mengenai struktur data (Hutahaean Jeperson, 2015).

## 2.3 Gambaran Perusahaan

Perpustakaan STIKI Malang adalah unit penting yang mendukung kegiatan akademik dan penelitian di lingkungan kampus. Dengan koleksi yang sangat beragam, Perpustakaan ini berperan sebagai sumber daya yang sangat penting bagi seluruh anggota komunitas akademik. Koleksi yang lengkap dan terbaru membantu mereka mengakses informasi yang diperlukan untuk mendukung proses belajar mengajar serta penelitian.

Perpustakaan STIKI Malang menawarkan beberapa layanan untuk mempermudah akses informasi dan mendukung kebutuhan akademik. Salah satu layanan unggulan adalah cek plagiarisme, yang memungkinkan mahasiswa memeriksa keaslian karya tulis mereka menggunakan *software* khusus. Dengan demikian, kualitas akademik dan integritas intelektual tetap terjaga.

Selain itu, perpustakaan juga menyediakan layanan penjilidan laporan tugas akhir (TA) / praktek kerja lapangan (PKL). Layanan ini memudahkan mahasiswa dalam menyiapkan dokumen resmi yang rapi dan sesuai standar yang ditetapkan oleh institusi. Dengan demikian, mahasiswa tidak perlu mencari tempat penjilidan di luar kampus.

Perpustakaan juga menyediakan layanan pengumpulan berkas tugas akhir, yang memastikan bahwa semua dokumen yang diperlukan tersimpan dengan baik dan dapat diakses oleh pihak-pihak terkait. Layanan ini membantu memastikan bahwa semua persyaratan administrasi terpenuhi dengan baik dan tepat waktu.

Sebelum menyelesaikan studi, mahasiswa diharuskan mengurus layanan bebas tanggungan perpustakaan. Layanan ini memastikan bahwa tidak ada kewajiban atau pinjaman buku yang belum dikembalikan, sehingga mahasiswa dapat menyelesaikan proses administrasi dengan lancar. Dengan berbagai layanan yang diberikan, perpustakaan STIKI Malang berkomitmen untuk menjadi mitra yang handal dalam mendukung aktivitas akademik dan penelitian, serta memastikan bahwa kebutuhan informasi para pengguna dapat terpenuhi dengan baik.