

## BAB II

### KAJIAN PUSTAKA

#### A. Tinjauan Pustaka

Lukman Santoso & Juni Amanullah (2022) dalam penelitiannya mengembangkan sistem informasi akademik berbasis *web* menggunakan metode *Rapid Application Development (RAD)* untuk SMA YATPI Godong. Studi ini menyoroti transformasi sistem manual berbasis Excel ke platform digital yang terintegrasi, dengan fitur pengelolaan data siswa, guru, jadwal, dan laporan akademik. *Rapid Application Development (RAD)* diaplikasikan melalui tiga fase utama: *requirements planning, design workshop,* dan *implementation*, menggunakan PHP, MySQL, dan UML untuk pemodelan sistem. Hasil pengujian *blackbox testing* menunjukkan semua fungsionalitas sistem berjalan optimal, termasuk verifikasi dokumen dan pencetakan laporan. Penelitian ini membuktikan bahwa *Rapid Application Development (RAD)* efektif dalam mempercepat pengembangan sistem berbasis *web* sekaligus meningkatkan transparansi dan akurasi data, yang relevan dengan kebutuhan otomatisasi proses administratif di institusi pendidikan.

Nurman Hidayat & Kusuma Hati (2021) merancang sistem informasi rapor online (*SIRALINE*) untuk SD Islam Imam Syafi'i Jember dengan metode *Rapid Application Development (RAD)*. Penelitian ini mengatasi masalah ketidakefisienan pengolahan nilai manual yang rentan kesalahan perhitungan dan pemborosan waktu. *Rapid Application Development (RAD)* diimplementasikan melalui kolaborasi intensif

dengan *stakeholder* sekolah, menggunakan framework CodeIgniter, PHP, dan MySQL. Hasilnya, sistem mampu mengintegrasikan input nilai oleh guru, verifikasi oleh wali kelas, dan pencetakan rapor otomatis. Studi ini menegaskan bahwa pendekatan iteratif *Rapid Application Development (RAD)* cocok untuk proyek dengan waktu terbatas, serta mendukung kebutuhan penelitian saat ini dalam membangun sistem terotomatisasi berbasis kolaborasi pengguna.

Susilo et al. (2023) merancang sistem informasi keuangan berbasis *web* untuk Kantor Lurah Kotabaru Reteh menggunakan *Rapid Application Development (RAD)*. Penelitian ini menjawab masalah pencatatan keuangan manual yang berisiko kehilangan data dan inefisiensi waktu. *Rapid Application Development (RAD)* diaplikasikan dengan melibatkan fase *requirements planning* untuk identifikasi kebutuhan pengguna, *design workshop* menggunakan UML (*use case, Activity Diagram*), dan *implementation* dengan PHP-MySQL. Hasil pengujian *whitebox* dan *blackbox testing* membuktikan sistem mampu mengelola pemasukan/pengeluaran, menghasilkan laporan *real-time*, serta meningkatkan akuntabilitas keuangan. Penelitian ini menyediakan referensi penting tentang integrasi *Rapid Application Development (RAD)* dalam konteks pemerintahan desa, khususnya dalam menyederhanakan proses administratif yang kompleks melalui pendekatan berbasis *web*.

Penelitian oleh Gunadi (2021) mengimplementasikan metode *Rapid Application Development (RAD)* dalam membangun sistem informasi berbasis *web* untuk mengelola data anggota dan transaksi keuangan di POUK Lenteng

Agung. Penggunaan *Rapid Application Development (RAD)* memungkinkan pengembangan sistem secara iteratif dengan melibatkan umpan balik pengguna melalui prototipe berulang, sehingga kebutuhan fungsional seperti pengelolaan data anggota, transaksi iuran, dan pelaporan dapat terpenuhi secara efektif. Framework W3.CSS dan PHP digunakan untuk menghasilkan antarmuka responsif yang kompatibel dengan berbagai perangkat, sementara MySQL sebagai basis data terpusat memastikan integrasi data antar-sektor. Evaluasi *User Acceptance Test (UAT)* menunjukkan tingkat penerimaan pengguna sebesar 82,86%, membuktikan bahwa *Rapid Application Development (RAD)* efektif dalam menghasilkan sistem yang sesuai kebutuhan organisasi dengan kualitas layanan tinggi. Temuan ini relevan dengan penelitian ini, khususnya dalam konteks mengintegrasikan proses administrasi yang kompleks menjadi terpusat dan mudah diakses.

Alam et al. (2023) merancang sistem pendataan jual beli tanah di Kelurahan Binawidya menggunakan metode *Rapid Application Development (RAD)* untuk mengatasi masalah duplikasi data, kesalahan input, dan penyimpanan manual. Tahapan *Rapid Application Development (RAD)* meliputi perencanaan kebutuhan, perancangan prototipe, dan implementasi berbasis PHP-MySQL, yang menghasilkan sistem dengan fitur pengelolaan data penjual, pembeli, dan arsip surat tanah. Pengujian *Blackbox Testing* membuktikan semua fungsional sistem berjalan sesuai ekspektasi, sementara analisis PIECES menunjukkan peningkatan signifikan dalam efisiensi waktu dan akurasi data. Penelitian ini mengonfirmasi bahwa

pendekatan iteratif *Rapid Application Development (RAD)* cocok untuk proyek dengan kebutuhan dinamis, seperti sistem seleksi beasiswa, di mana adaptasi cepat terhadap masukan pengguna dan pengurangan kesalahan administratif menjadi prioritas utama.

## **B. Landasan Teori**

### **1. Optimalisasi**

Optimalisasi adalah proses meningkatkan efektivitas dan efisiensi suatu sistem atau kegiatan untuk mencapai hasil terbaik sesuai kriteria yang ditetapkan, dengan memanfaatkan sumber daya secara optimal. Tujuannya adalah menghasilkan solusi ideal melalui pendekatan terukur, baik dengan memaksimalkan keuntungan (*benefit*) maupun meminimalkan kerugian (*cost*) (Hidayat & Irvanda, 2022). Optimalisasi adalah proses meningkatkan efektivitas dan efisiensi suatu sistem atau kegiatan untuk mencapai hasil terbaik sesuai kriteria yang ditetapkan, dengan memanfaatkan sumber daya secara optimal. Tujuannya adalah menghasilkan solusi ideal melalui pendekatan terukur, baik dengan memaksimalkan keuntungan (*benefit*) maupun meminimalkan kerugian (*cost*) (Hidayat & Irvanda, 2022).

Dalam penelitian ini, konsep optimalisasi diaplikasikan pada pengembangan sistem berbasis *website* untuk proses seleksi penerima beasiswa di Dinas Pendidikan dan Kebudayaan Kabupaten Madiun. Sistem ini dirancang untuk mengoptimalkan tiga aspek utama: efisiensi proses melalui otomatisasi tahapan seleksi yang sebelumnya manual, akurasi penilaian melalui penerapan algoritma terstandar, serta aksesibilitas melalui

platform terpusat yang memungkinkan interaksi *real-time* antara pendaftar dan panitia. Pendekatan *Rapid Application Development* (RAD) dipilih sebagai metode pengembangan untuk mencapai optimalisasi secara cepat dan adaptif, sesuai dengan karakteristik dinamis kebutuhan instansi. Dengan demikian, optimalisasi dalam konteks ini tidak hanya bermakna perbaikan sistem secara teoritis, tetapi juga terimplementasi dalam bentuk solusi teknologi yang mampu menjawab tantangan nyata dalam proses seleksi beasiswa.

## 2. Proses

Proses adalah serangkaian langkah atau tindakan yang diambil untuk mencapai suatu tujuan tertentu. Proses digunakan dalam berbagai konteks dan dapat merujuk pada tindakan fisik, aktivitas bisnis, atau bahkan peristiwa alam. Dalam pemahaman yang lebih mendalam, proses melibatkan rangkaian kejadian yang terkoordinasi dan terstruktur (Emma Harris, 2024). Dalam penelitian ini penerapan proses ada pada pengertian dari proses seleksi beasiswa.

## 3. Seleksi

Seleksi adalah proses memilih individu atau kelompok berdasarkan kriteria tertentu dengan tujuan tertentu pula. Dalam berbagai bidang, seleksi memiliki peran penting dalam menyaring kandidat yang paling sesuai untuk tugas atau posisi tertentu. Proses ini diterapkan dalam berbagai konteks, mulai dari dunia kerja, pendidikan, hingga seleksi alam dalam biologi. Setiap bentuk seleksi memiliki tahapan dan metodenya sendiri, yang

disesuaikan dengan tujuan akhir dan kriteria yang ingin dicapai (Muallif, 2024). Dalam penelitian ini pengertian seleksi digunakan untuk menjelaskan proses seleksi beasiswa yang terjadi di Dinas Pendidikan dan Kebudayaan Kabupaten Madiun.

#### 4. *Website*

*Website* adalah kumpulan halaman informasi yang dapat diakses melalui jaringan internet, terdiri atas konten dalam bentuk teks, gambar, video, atau format lainnya. *Website* memungkinkan pengguna untuk mengakses informasi tersebut kapan saja dan di mana saja selama terhubung ke internet. Definisi ini menekankan pada aspek keterbukaan akses, keberagaman konten, serta ketergantungan pada teknologi internet sebagai sarana penyebaran informasi (Rahmi et al., 2023).



Gambar 2. 1 *Website*

Sumber: (Prasetyo, 2020)

Proses akses *website* melibatkan interaksi dinamis antara klien (pengguna) dan *server*, seperti yang terlihat pada Gambar 2.1. Ketika pengguna membuka *website* melalui browser, klien mengirimkan *request*

(permintaan) ke *server*, misalnya untuk memuat halaman beranda. *Server* kemudian memproses permintaan tersebut dengan mengambil data dari database jika konten bersifat dinamis atau mengirimkan file statis seperti HTML dan mengembalikan respons berupa kode, gambar, atau elemen lain yang dibutuhkan. Browser akhirnya merender *response* tersebut menjadi tampilan visual yang interaktif bagi pengguna.

## 5. Sistem Informasi

Sistem informasi merupakan suatu sistem formal yang terstruktur, dirancang untuk mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi. Tujuannya adalah untuk mendukung kegiatan operasional, fungsi manajerial, serta proses pengambilan keputusan dalam suatu organisasi agar berjalan lebih efektif dan efisien (Wijoyo et al., 2021).

Agar dapat berfungsi secara utuh, sebuah sistem informasi tersusun atas beberapa komponen fundamental yang bekerja secara terintegrasi. Komponen-komponen tersebut meliputi perangkat keras (*hardware*) sebagai perangkat fisik, perangkat lunak (*software*) yang berisi instruksi dan program, data sebagai bahan baku yang akan diolah menjadi informasi, prosedur yang merupakan aturan penggunaan sistem, serta sumber daya manusia (*user*) yang mengoperasikan sistem tersebut. Keterpaduan seluruh komponen ini menjadi kunci untuk menghasilkan informasi yang akurat, relevan, dan tepat waktu.

Dalam konteks penelitian ini, dikembangkan sebuah sistem informasi berbasis web yang secara khusus dirancang untuk mengelola

proses seleksi beasiswa di Dinas Pendidikan dan Kebudayaan Kabupaten Madiun. Sistem ini mengintegrasikan seluruh komponen tersebut untuk mengubah proses manual menjadi sebuah alur kerja digital yang terstruktur dan akuntabel.

#### 6. Beasiswa

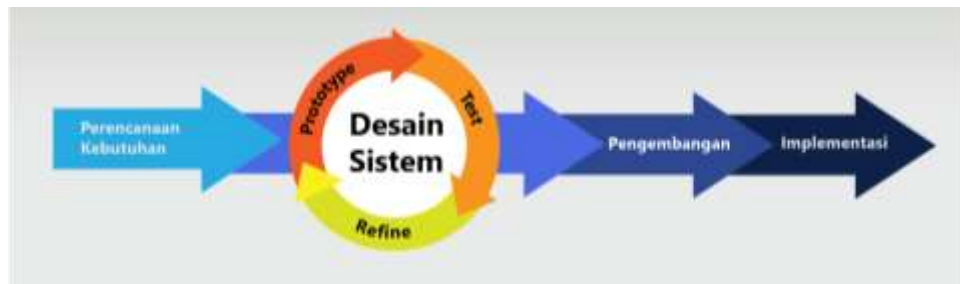
Beasiswa adalah bantuan keuangan yang diberikan kepada individu, mahasiswa, atau pelajar untuk mendukung kelangsungan pendidikan mereka. Beasiswa juga dapat diartikan sebagai bentuk apresiasi yang memungkinkan penerimanya melanjutkan pendidikan ke jenjang lebih tinggi, baik berupa akses khusus pada institusi tertentu maupun bantuan finansial (Syabaniah et al., 2022).

Keterkaitan dengan penelitian ini adalah untuk menjawab tantangan dalam proses seleksi tersebut. Penelitian ini berfokus pada optimalisasi proses seleksi beasiswa di lingkungan Dinas Pendidikan dan Kebudayaan Kabupaten Madiun, yang selama ini masih berjalan manual. Melalui pengembangan sistem informasi berbasis *web*, diharapkan proses seleksi dapat menjadi lebih akurat, cepat, dan akuntabel, sehingga mendukung efektivitas program beasiswa yang diselenggarakan oleh dinas terkait.

#### 7. *Rapid Application Development* (RAD)

Metode *Rapid Application Development* (RAD) adalah metode yang digunakan dalam pengembangan perangkat lunak tambahan dengan siklus pengembangan yang relatif singkat (Pradana et al., 2022). Pada pengembangan sistem informasi secara konvensional, biasanya dibutuhkan

waktu minimal 180 hari. Namun, dengan metode *Rapid Application Development (RAD)* sistem dapat diselesaikan lebih cepat, yaitu dalam rentang waktu 30 hingga 90 hari (Pricillia & Zulfachmi, 2021).



Gambar 2. 2 Metode *Rapid Application Development (RAD)*

Sumber: (Iqbal, 2021)

Berikut adalah langkah-langkah dari metode penelitian *Rapid Application Development*:

- a. Rencana Kebutuhan (*Requirement Planning*): Pada tahap ini, *User* dan analis sistem melakukan diskusi untuk mengidentifikasi tujuan sistem serta kebutuhan informasi yang diperlukan. Keterlibatan aktif dari kedua pihak menjadi faktor penting dalam memastikan kebutuhan sistem tercapai.
- b. Desain Sistem (*Design System*): Tahap ini melibatkan *User* secara aktif dalam proses perancangan sistem. Jika ada ketidaksesuaian desain, perbaikan segera dilakukan berdasarkan masukan *User*. Desain sistem disusun berdasarkan dokumentasi kebutuhan *User* yang sudah dibuat sebelumnya. Hasil dari tahap ini berupa spesifikasi perangkat lunak yang mencakup struktur data dan organisasi sistem secara umum.

c. Implementasi (*Implementation*): *Programmer*




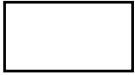


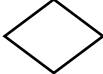


mengembangkan program berdasarkan desain yang telah disetujui oleh *User* dan analis. Sebelum diimplementasikan, program diuji untuk memastikan tidak ada kesalahan. Pada tahap ini, *User* memberikan tanggapan dan persetujuan terhadap sistem yang telah dibuat.

Metode RAD dipilih karena sangat cocok untuk kebutuhan pengembangan sistem seleksi beasiswa di Dinas Pendidikan. Pendekatan ini memungkinkan pengembangan yang cepat untuk mengejar jadwal akademik yang ketat. Dengan menyajikan prototipe secara langsung kepada staf dinas, umpan balik yang diterima menjadi lebih praktis dan relevan, memastikan sistem akhir benar-benar sesuai dengan kebutuhan operasional mereka. Proses iteratif ini juga memberikan fleksibilitas untuk melakukan penyesuaian dan secara signifikan meminimalkan risiko kegagalan, karena sistem divalidasi secara berkelanjutan oleh pengguna akhir itu sendiri.

8. Diagram Alir (*Flowchart*)

Diagram alir (*flowchart*) adalah Teknik analitis bergambar yang di gunakan untuk menjelaskan beberapa aspek dari sistem informasi secara jelas, ringkas , dan logis Bagan air mencatat cara proses bisnis dilakukan dan cara dokumen mengalir melalui organisasi. *Flowchart* adalah gambar alir akan sistem dan prosedur serta pengendalian intern yang telah dijalankan oleh perusahaan (Zainab Tuasamu et al., 2023).

Tabel 2. 1 Simbol dan Fungsi *Flowchart* (Zalukhu et al., 2023)

Simbol	Nama	Fungsi
	<i>Terminator</i>	Permulaan/Akhir Program
	Garis Alir ( <i>Flow Line</i> )	Arah Aliran Program
	<i>Preparation</i>	Proses inialisasi/pemberian harga awal
	Proses	Proses penghitungan/pengolahan data
	<i>Input/Output Data</i>	Proses <i>input/output</i> data, parameter, dan informasi.
	<i>Predefined Process (sub-program)</i>	Permulaan <i>sub-program</i> /proses menjalankan <i>sub-program</i> .
	<i>Decission</i>	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan terbaik untuk proses selanjutnya.
	<i>Connector</i>	Penghubung bagian – bagian <i>flowchart</i> yang berada pada satu halaman.
	<i>Off-page Connector</i>	Penghubung bagian – bagian <i>flowchart</i> yang berada pada halaman yang berbeda.

Dalam penelitian ini, *flowchart* menjadi alat fundamental yang memiliki fungsi ganda. Pertama, *flowchart* digunakan untuk memetakan

dan menganalisis alur kerja manual proses seleksi beasiswa yang ada saat ini di Dinas Pendidikan, sehingga membantu mengidentifikasi berbagai inefisiensi dan titik hambatan (*bottleneck*) secara visual. Kedua, *flowchart* dimanfaatkan untuk merancang alur kerja baru yang lebih efisien dan terotomatisasi untuk sistem berbasis *web* yang diusulkan. Dengan demikian, *flowchart* tidak hanya berfungsi sebagai alat dokumentasi, tetapi juga sebagai bukti visual yang secara efektif membandingkan kelemahan sistem lama dengan keunggulan solusi yang dikembangkan.

#### 9. *Unified Modeling Language* (UML)


*Unified Modeling Language* (UML) adalah bahasa grafis yang digunakan untuk memvisualisasikan, mendefinisikan, membangun, dan mendokumentasikan sistem pengembangan perangkat lunak berbasis objek. UML membantu dalam menggambarkan struktur dan perilaku sistem secara lebih jelas, sehingga memudahkan komunikasi antara pengembang dan pemangku kepentingan (Nabila et al., 2021).

##### 1. *Use Case Diagram*

*Use case* adalah deskripsi fungsi suatu sistem dari sudut pandang pengguna. *Use case* mendefinisikan apa saja yang akan diproses oleh sistem beserta komponen-komponennya. Dalam praktiknya, *use case* menggunakan skenario yang menggambarkan urutan langkah-langkah interaksi antara pengguna dan sistem. Selain itu, *use case* juga mengidentifikasi fungsionalitas sistem, interaksi pengguna dengan

sistem, serta hubungan antara pengguna dengan fitur sistem (Setiyani, 2021).

Tabel 2. 2 *Use Case Diagram*

Simbol	Nama	Deskripsi
	<i>Actor</i>	Digunakan untuk menjelaskan sesuatu atau seseorang yang sedang berinteraksi dengan sistem.
	<i>Use Case</i>	Menggambarkan suatu perilaku dari sistem tanpa mengungkapkan struktur internal dari sistem tersebut.
	<i>Association</i>	Jalur komunikasi antara actor dengan <i>use case</i> yang saling berpartisipasi.
	<i>Extend</i>	Penambahan perilaku ke dalam <i>use case</i> dasar yang tidak tahu tentang hal tersebut.
	<i>Use Case generalization</i>	Hubungan antara <i>use case</i> umum dengan <i>use case</i> yang lebih spesifik, yang

Simbol	Nama	Deskripsi
		mewarisi dan menambah fitur terhadapnya.
▼ <<Include>>	<i>Include</i>	Penambahan perilaku ke dalam <i>use case</i> dasar yang secara eksplisit menjelaskan penambahannya.






Dalam penelitian ini, *use case* diagram berperan sebagai cetak biru fungsional untuk merancang sistem seleksi beasiswa yang dikembangkan. Diagram ini secara spesifik digunakan untuk mengidentifikasi siapa saja pengguna sistem (aktor), seperti "Calon Pendaftar" dan "Admin Dinas", serta merinci semua fitur (*use case*) yang dapat mereka akses. Sebagai contoh, diagram ini akan memvisualisasikan bagaimana pendaftar dapat Mengisi Formulir dan admin dapat Memverifikasi Berkas. Dengan demikian, *use case diagram* memastikan bahwa seluruh kebutuhan fungsional dari proses seleksi beasiswa telah terdefinisi secara jelas dan terstruktur sebelum tahap pengembangan dimulai.

## 2. Activity Diagram

*Activity Diagram* adalah diagram yang digunakan untuk menggambarkan alur aktivitas suatu *use case* atau proses bisnis secara berurutan. Diagram ini memvisualisasikan aliran kerja atau urutan

langkah-langkah dalam menyelesaikan sebuah aktivitas secara lebih jelas.(Alfarizi et al., 2020)

Tabel 2. 3 *Activity Diagram*

Simbol	Nama	Deskripsi
	<i>Initial</i>	Menunjukkan dimana aliran kerja dimulai.
	<i>Final</i>	Menunjukkan dimana aliran kerja berakhir.
	<i>Action</i>	Langkah-langkah dalam sebuah <i>activity</i> .
	<i>Decision</i>	Menunjukkan tempat pengambilan keputusan.
	<i>Swimlane</i>	Mengelompokkan <i>activity</i> berdasarkan <i>actor</i> .


Dalam penelitian ini, activity diagram berperan penting untuk memvisualisasikan alur kerja secara detail. Diagram ini digunakan untuk dua tujuan utama: pertama, memetakan proses seleksi manual yang ada saat ini untuk menunjukkan kerumitan dan inefisiensinya. Kedua, merancang alur kerja yang logis dan efisien untuk setiap fitur pada sistem berbasis *web* yang diusulkan, seperti proses pendaftaran hingga perankingan otomatis. Dengan demikian, diagram ini menjadi jembatan antara rancangan fungsional dan panduan implementasi



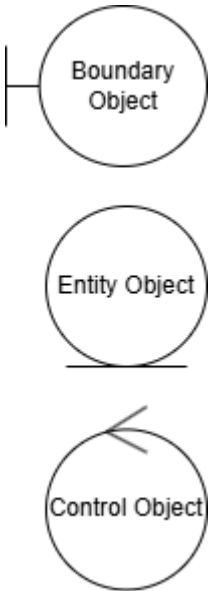
teknis, sekaligus menjadi bukti visual perbandingan antara sistem lama dan sistem baru.

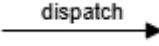
### 3. *Sequence Diagram*

*Sequence Diagram* adalah salah satu jenis interaction diagram yang menggambarkan cara dan urutan kerja sekelompok objek dalam berinteraksi. Diagram ini sering digunakan oleh pengembang perangkat lunak dan profesional bisnis untuk memahami kebutuhan sistem baru atau mendokumentasikan proses yang sudah ada (Wanigasekera & Wijayarathna, 2022).

Tabel 2. 4 Simbol Sequence Diagram (Sumirat et al., 2023)

Simbol	Keterangan
Aktor 	Komponen ini menggambarkan seorang pengguna yang berada di luar sistem dan sedang berinteraksi dengan sistem. Dalam sequence diagram, aktor biasanya digambarkan dengan simbol stick figure.
<i>Activation box</i>	Komponen <i>activation box</i> ini merepresentasikan waktu yang dibutuhkan suatu objek untuk menyelesaikan tugasnya. Semakin lama waktu yang diperlukan,

Simbol	Keterangan
	<p>maka secara otomatis <i>activation box</i>-nya juga akan menjadi lebih panjang. Komponen ini digambarkan dengan bentuk persegi panjang.</p>
<p><i>Lifeline</i></p> 	<p>Komponen ini digambarkan dengan bentuk garis putus-putus. <i>Lifeline</i> ini biasanya memiliki kotak yang berisi objek yang memiliki fungsi untuk menggambarkan aktivitas dari objek.</p>
<p>Objek</p> 	<p>Komponen objek ini digambarkan sesuai dengan symbol dan nama objek. Biasanya objek berfungsi untuk mendokumentasikan perilaku sebuah objek pada sebuah sistem. (contoh: <i>Boundary, Entity, Control</i>)</p>

Simbol	Keterangan
<p data-bbox="459 383 576 416"><i>Message</i></p> 	<p data-bbox="911 383 1356 1003">Komponen ini untuk menggambarkan komunikasi antar objek. <i>Messages</i> biasanya muncul secara berurutan pada lifeline. Komponen messages ini direpresentasikan dengan anak panah. Inti dari sebuah diagram urutan terdapat pada komponen <i>lifeline</i> dan <i>messages</i> ini.</p>

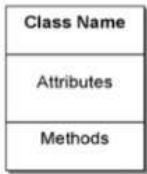




Dalam penelitian ini, *sequence diagram* digunakan untuk memodelkan interaksi teknis antar komponen sistem berdasarkan urutan waktu. Diagram ini menjawab bagaimana sebuah fitur bekerja di balik layar, misalnya dengan menggambarkan alur pesan yang terjadi saat admin menekan tombol verifikasi mulai dari antarmuka pengguna, diteruskan ke kontroler di *server*, hingga terjadi pembaruan data di *database*. Dengan demikian, *sequence diagram* berfungsi sebagai panduan detail bagi *programmer* selama proses pengkodean untuk memastikan setiap komponen sistem berinteraksi secara benar dan sesuai dengan logika bisnis yang telah dirancang.


#### d. *Class Diagram*

*Class Diagram* adalah representasi visual yang menggambarkan struktur sistem melalui definisi berbagai kelas yang dirancang untuk

membentuk suatu sistem. Diagram ini memuat detail mengenai atribut dan metode setiap kelas, serta hubungan antar kelas tersebut (Wulandari & Nurmiati, 2022).

Tabel 2. 5 *Class Diagram*

Simbol	Nama	Deskripsi
	<i>Class</i>	Sekumpulan objek dengan atribut berbeda yang melakukan operasi yang sama.
	<i>Association</i>	Hubungan antar kelas yang mempunyai arti umum dan biasanya melibatkan multiplisitas.
	<i>Directed Association</i>	Hubungan antar kelas yang artinya satu kelas digunakan oleh kelas lain.
	<i>Aggregation</i>	Mengidentifikasi seluruh bagian suatu hubungan disebut hubungan.
	<i>Composition</i>	Konfigurasi relasi dengan kelas dependen.

Simbol	Nama	Deskripsi
	<i>Dependency</i>	Menampilkan operasi pada suatu kelas yang menggunakan kelas.

Dalam penelitian ini, class diagram berperan sebagai cetak biru (*blueprint*) untuk merancang struktur database sistem seleksi beasiswa. Diagram ini secara langsung memetakan setiap kelas yang dirancang menjadi tabel, atribut pada kelas menjadi kolom, dan hubungan antar kelas menjadi kunci asing (*foreign key*) yang mengaitkan tabel-tabel tersebut. Sebagai contoh, kelas Pendaftar beserta atributnya akan diimplementasikan menjadi tabel pendaftar di dalam database. Oleh karena itu, class diagram ini berfungsi sebagai panduan teknis fundamental bagi programmer untuk membangun skema database yang logis, terstruktur, dan konsisten.

## 10. PHP

PHP awalnya merupakan singkatan dari *Personal Home Page*, namun kini dikenal sebagai *Hypertext Preprocessor*. PHP adalah bahasa pemrograman skrip yang banyak digunakan dalam pengembangan *web* dan dijalankan di sisi *server*, berbeda dengan *Java Virtual Machine* yang dieksekusi di sisi klien. PHP memungkinkan pengembang membuat halaman *web* dinamis yang hasilnya ditampilkan kepada pengguna sebagai HTML biasa setelah diproses oleh *server*. Karena sifatnya yang dieksekusi di *server*, PHP sering disebut sebagai bahasa pemrograman sisi *server*. Selain itu, PHP dapat diintegrasikan dengan berbagai sistem manajemen

basis data, menjadikannya alat yang fleksibel dan andal untuk pengembangan *web* (T. Maulana et al., 2024).

Dalam penelitian ini, PHP dipilih sebagai bahasa pemrograman utama pada sisi *server (server-side)* untuk membangun keseluruhan logika dan fungsionalitas sistem informasi seleksi beasiswa. Kemampuan PHP untuk mengelola data, memproses input dari formulir pendaftaran secara aman, dan menjalankan logika bisnis seleksi menjadi alasan utamanya. PHP memungkinkan pembuatan halaman *web* yang dinamis, seperti menampilkan status pendaftaran yang berbeda untuk setiap pengguna atau menyajikan daftar hasil seleksi secara *real-time*. Lebih penting lagi, kemampuannya untuk terintegrasi secara andal dengan basis data (seperti MySQL) menjadi faktor kunci untuk menyimpan dan mengelola seluruh data pendaftar secara terstruktur, menjadikan PHP fondasi teknis yang menjalankan mesin di balik layar aplikasi *web* yang dikembangkan.

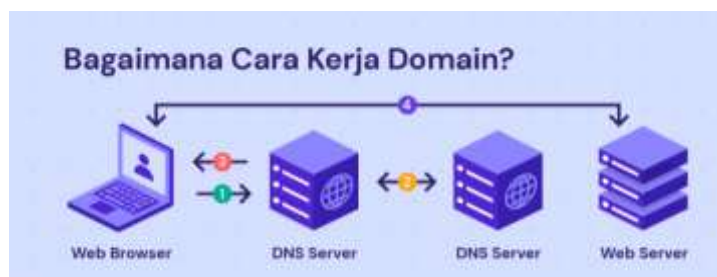
## 11. MySQL

*Structure Query Language (SQL)* adalah bahasa khusus *domain* yang digunakan untuk mengolah data dalam sistem manajemen basis data hubungan. Aplikasi RDBMS (*Relational Database Management System*) yang paling umum digunakan oleh programmer aplikasi *web* untuk mengolah basis data mereka adalah MySQL. Fungsi-fungsi dalam Bahasa pemrograman PHP biasanya digunakan untuk membuat, membaca, mengubah, atau menghapus data dalam SQL, yang kemudian dapat ditampilkan di halaman *web* (Sinlae et al., 2024).

Dalam penelitian ini, MySQL dipilih sebagai *Relational Database Management System* (RDBMS) untuk menjadi tempat penyimpanan pusat seluruh data sistem seleksi beasiswa. Pemilihan MySQL didasarkan pada beberapa keunggulan utama. Pertama, sifatnya yang *open-source* dan gratis membuatnya sangat ideal untuk proyek pengembangan di lingkungan akademik dan pemerintahan. Kedua, MySQL memiliki kompatibilitas yang sangat tinggi dengan PHP, yang merupakan bahasa utama aplikasi ini, sehingga proses integrasi antara logika aplikasi dan basis data berjalan dengan lancar dan andal. Oleh karena itu, MySQL menyediakan fondasi basis data yang kuat, aman, dan efisien untuk menampung semua informasi pendaftar, dokumen, hingga hasil akhir seleksi.

## 12. *Domain*

*Domain* adalah nama sebuah situs, biasanya nama unik yang diberikan untuk mengidentifikasi nama *server* komputer seperti *web server* atau *email server* di jaringan komputer ataupun internet. Fungsi *domain* adalah untuk mempermudah pengguna di internet pada saat melakukan akses ke *server*, selain itu juga dipakai untuk mengingat nama *server* yang dikunjungi tanpa harus mengenal deretan angka yang rumit dikenal sebagai *IP address*. Istilah lain dalam penyebutan *domain* adalah URL atau alamat *website* (Rochman et al., 2020).



Gambar 2. 3 *Domain*

Sumber: (Faradilla, 2025)

Dalam penelitian ini, penggunaan *domain* menjadi aspek teknis yang krusial agar sistem seleksi beasiswa berbasis *web* yang dikembangkan dapat diakses oleh publik. Sebuah nama *domain* yang spesifik dan profesional akan ditetapkan untuk sistem ini.

Tujuannya adalah untuk memberikan alamat akses yang mudah diingat dan kredibel bagi para calon pendaftar dan admin dinas, sehingga meningkatkan kepercayaan dan kemudahan penggunaan sistem. Tanpa *domain*, pengguna harus mengakses sistem melalui alamat IP *server*, yang tidak praktis dan tidak profesional. Oleh karena itu, *domain* menjadi gerbang utama bagi pengguna untuk dapat berinteraksi dengan aplikasi *web* yang telah dibangun.

### 13. *Hosting*

*Hosting* (atau *web hosting*) adalah layanan penyewaan sumber daya pada sebuah *server* yang terhubung ke internet, yang memungkinkan individu atau organisasi untuk menempatkan situs *web* atau aplikasi *web* mereka. Layanan ini menyediakan ruang penyimpanan (*space*) untuk semua *file* yang membentuk aplikasi, termasuk kode program (seperti PHP),

*database* (seperti MySQL), gambar, dan dokumen lainnya. Penting untuk dipahami bahwa *hosting* adalah layanannya, sedangkan *server* adalah komputernya, dan *web server* (seperti Apache atau Nginx) adalah perangkat lunak di dalam *server* yang bertugas menyajikan halaman *web* kepada pengguna (Rochman et al., 2020).



Gambar 2. 4 Hosting

Sumber: (Ariata, 2024)

Dalam penelitian ini, *hosting* merupakan infrastruktur fundamental yang diperlukan untuk menempatkan (*deploy*) sistem seleksi beasiswa berbasis web agar dapat diakses secara *online* oleh seluruh pengguna. *Hosting* akan menjadi tempat untuk menyimpan seluruh komponen sistem, mulai dari *file-file* kode PHP, *database* MySQL yang berisi data pendaftar, hingga dokumen-dokumen yang diunggah oleh peserta. Dengan menggunakan layanan *hosting*, sistem dipastikan dapat beroperasi selama 24 jam sehari dan dapat diakses dari mana saja melalui internet menggunakan nama domain yang telah ditentukan. Oleh karena itu, pemilihan dan konfigurasi *hosting* adalah langkah teknis terakhir yang

krusial untuk mengubah sistem dari lingkungan pengembangan lokal menjadi aplikasi yang berfungsi penuh di dunia nyata.

#### 14. VSCode

Microsoft *Visual Studio Code* (VS Code) adalah editor kode dan teks yang gratis serta *open-source*. Meskipun ukurannya relatif kecil, VS Code memiliki banyak fitur penting yang membuatnya menjadi salah satu editor kode paling populer dalam beberapa tahun terakhir. VS Code mendukung berbagai bahasa pemrograman, seperti Java, C++, Python, CSS, Go, dan Docker. Selain itu, pengguna dapat menambahkan dan membuat ekstensi baru, seperti *code linker*, *debugger*, serta dukungan untuk pengembangan *cloud* dan *web*. Dibandingkan dengan editor teks lainnya, antarmuka pengguna VSCode menawarkan interaktivitas yang tinggi (bin Uzayr, 2022).

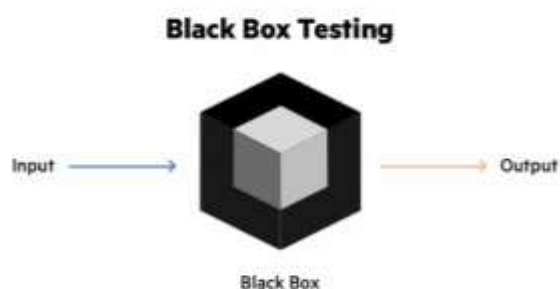
Dalam penelitian ini, Visual Studio Code berperan sebagai alat bantu utama (*tool*) atau *code editor* yang digunakan oleh peneliti/pengembang selama tahap konstruksi dan implementasi sistem.

Penting untuk dicatat bahwa VS Code adalah lingkungan pengembangan dan bukan merupakan bagian dari sistem akhir yang nantinya diakses oleh pendaftar atau admin dinas. VS Code dipilih karena dukungannya yang sangat baik untuk teknologi yang digunakan dalam proyek ini, yaitu PHP, HTML, CSS, JavaScript, dan SQL. Ketersediaan berbagai ekstensi seperti *syntax highlighting* untuk PHP dan *database client* untuk mengelola MySQL langsung dari editor, secara signifikan

meningkatkan produktivitas dan mempercepat proses penulisan kode (*coding*) dan pencarian kesalahan (*debugging*). Hal ini menjadikan VS Code sebagai pilihan yang efisien dan efektif untuk menulis dan mengelola seluruh kode sumber aplikasi seleksi beasiswa ini.

#### 15. *Black Box Testing*

*Black-box testing* adalah metode pengujian perangkat lunak yang tidak memerlukan akses ke struktur internal atau kode sumber dari sistem yang diuji. Dalam konteks pengujian *Deep Neural Networks* (DNN), metode ini diterapkan dengan cara mengevaluasi output sistem berdasarkan input yang diberikan tanpa memperhatikan proses internal yang terjadi di dalam model. *Black-box* testing lebih berfokus pada memeriksa ketepatan output terhadap ekspektasi yang telah ditentukan sebelumnya, dan seringkali digunakan dalam situasi di mana akses ke data pelatihan atau model internal terbatas atau tidak memungkinkan (Aghababaeyan et al., 2023).



Gambar 2. 5 *Black Box Testing*

Sumber: (M. Maulana, 2023)

Dalam penelitian ini, metode *black-box testing* diterapkan untuk memvalidasi seluruh fungsionalitas sistem seleksi beasiswa dari sudut pandang pengguna akhir yaitu pendaftar dan admin tanpa perlu mengetahui struktur kode internalnya. Pengujian ini mencakup serangkaian skenario, mulai dari proses pendaftaran dan pengisian formulir oleh pendaftar, hingga proses verifikasi data dan pembuatan laporan oleh admin. Tujuannya adalah untuk memastikan setiap fitur memberikan *output* yang benar sesuai dengan *input* yang diberikan dan sistem secara keseluruhan berjalan sesuai dengan kebutuhan fungsional yang telah dirancang.

### C. Keaslian Penelitian

Tabel 2. 6 Matriks Literatur Review dan Posisi Penelitian

No.	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
1.	Implementasi Metode <i>Rapid Application Development</i> (RAD) pada Sistem Informasi Administrasi Keanggotaan	Gunadi, <i>Jurnal Informatik</i> , 2021	Membangun sistem informasi berbasis <i>web</i> untuk mengelola data anggota dan transaksi keuangan di POUK Lenteng Agung.	<i>Rapid Application Development</i> (RAD) efektif menghasilkan sistem terintegrasi dengan tingkat penerimaan pengguna 82,86%. W3.CSS dan PHP meningkatkan responsivitas antarmuka.	Kelemahan: Pengujian terbatas pada <i>Blackbox Testing</i> ; belum ada analisis keamanan data mendalam.	Meskipun sama-sama menggunakan RAD, penelitian tersebut berfokus pada manajemen data administratif (data anggota). Penelitian ini memiliki kompleksitas yang lebih tinggi karena berfokus pada proses pengambilan keputusan (seleksi) yang melibatkan kriteria, penilaian, dan perankingan, di mana aspek objektivitas dan transparansi menjadi

No.	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
						tantangan utama yang tidak dibahas pada penelitian sebelumnya.
2.	Penerapan <i>Rapid Application Development</i> (RAD) untuk Sistem Pendataan Jual Beli Tanah	Alam et al., <i>Jurnal Testing dan Implementasi Sistem Informasi</i> , 2023	Merancang sistem pendataan jual beli tanah berbasis web untuk meningkatkan efisiensi administratif di Kelurahan Binawidya.	<i>Rapid Application Development</i> (RAD) mempercepat pengembangan sistem, mengurangi kesalahan input, dan meningkatkan akurasi data.	Saran: Perlu pengembangan UI/UX lebih intuitif. Kelemahan: Belum ada integrasi dengan sistem pemerintah pusat.	Penelitian sebelumnya menerapkan RAD untuk pencatatan transaksional yang alurnya cenderung linear. Celah penelitian ini terletak pada penerapan RAD untuk proses seleksi yang kompetitif dan multi-kriteria. Sistem ini tidak hanya mencatat, tetapi juga harus mampu memproses, membandingkan, dan merangking data pendaftar secara adil dan transparan.

No.	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
3.	Implementasi <i>Rapid Application Development</i> (RAD) dalam Sistem Keuangan Pemerintahan Desa	Susilo et al., <i>Jurnal Testing dan Implementasi Sistem Informasi</i> , 2023	Membangun sistem keuangan terkomputerisasi untuk Kantor Lurah Kotabaru Reth.	Sistem meningkatkan efisiensi pencatatan pemasukan/pengeluaran dan meminimalkan risiko kehilangan data.	Kelemahan: Ketergantungan pada koneksi internet; belum ada fitur <i>backup</i> otomatis.	Fokus penelitian sebelumnya adalah akuntansi dan pelaporan keuangan. Penelitian ini mengisi celah dengan menerapkan RAD pada domain manajemen sumber daya manusia (seleksi talenta) di sektor publik. Tantangannya bukan hanya pada akurasi pencatatan, tetapi pada akuntabilitas dan objektivitas proses pengambilan keputusan terhadap calon penerima beasiswa.
4.	Pengembangan Sistem E-Cashier	Sholihin et al., <i>INFOTECH: Jurnal</i>	Membuat sistem kasir elektronik berbasis <i>web</i>	<i>Rapid Application Development</i> (RAD) memungkinkan	Saran: Penambahan fitur analitik penjualan. Kelemahan: Belum	Sistem <i>e-cashier</i> adalah sistem transaksional ( <i>Point of Sales</i> ) dengan alur kerja yang sederhana. Penelitian

No.	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
	Berbasis <i>Rapid Application Development</i> (RAD) untuk UMKM	<i>Informatika &amp; Teknologi</i> , 2024	untuk Marapi Kopi guna mengoptimalkan transaksi dan manajemen stok.	pengembangan cepat dengan iterasi prototipe, menghasilkan sistem yang sesuai kebutuhan UMKM.	kompatibel dengan perangkat <i>mobile</i> tertentu.	ini memiliki perbedaan signifikan karena membangun sistem pendukung keputusan ( <i>Decision Support System</i> ) yang alur kerjanya lebih kompleks, melibatkan tahapan verifikasi, validasi, dan penilaian berdasarkan aturan yang telah ditetapkan, sebuah domain yang tidak disentuh oleh penelitian <i>e-cashier</i> .
5.	Penerapan <i>Rapid Application Development</i>	Hidayat & Hati, <i>Jurnal Sistem Informasi</i>	Mengembangkan sistem rapor online untuk SD Islam Imam	Sistem mengurangi waktu pengolahan nilai hingga 40% dan	Saran: Pelatihan pengguna intensif. Kelemahan: Tidak ada	Meskipun sama-sama berada di <i>domain</i> pendidikan, sistem rapor <i>online</i> berfokus pada pengolahan dan agregasi nilai untuk setiap individu.

No.	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
	(RAD) pada Sistem Informasi Rapor Online	<i>STMIK Antar Bangsa</i> , 2021	Syafi'i guna menggantikan proses manual berbasis Excel.	meminimalkan kesalahan perhitungan.	integrasi dengan platform lain (e.g., e-learning).	Celah penelitian ini adalah pada aspek seleksi yang kompetitif, di mana sistem tidak hanya mengolah data, tetapi harus mampu membandingkan dan merangking satu pendaftar dengan pendaftar lainnya secara objektif berdasarkan serangkaian kriteria yang kompleks.