

RANCANG BANGUN SISTEM INFORMASI PENELITIAN DOSEN DI FAKULTAS TEKNIK UNIVERSITAS DIPONEGORO

Irza Nur Fauzi ¹⁾, M. Arfan ²⁾

^{1,2)} Teknik Elektro Universitas Diponegoro
email : irzanurf@gmail.com ¹⁾, arfan@elektro.undip.ac.id ²⁾

Abstraksi

Kegiatan penelitian dosen di perguruan tinggi sering menghadapi kendala dalam pengelolaan data, pelaporan, dan dokumentasi, yang berdampak pada efektivitas dan efisiensi proses penelitian. Untuk mengatasi permasalahan tersebut, pemanfaatan teknologi informasi melalui pengembangan sistem informasi penelitian menjadi solusi strategis. Penelitian ini bertujuan untuk merancang dan membangun sistem informasi penelitian dosen di Fakultas Teknik Universitas Diponegoro. Data dikumpulkan melalui observasi, wawancara, dan studi pustaka. Sistem dikembangkan menggunakan metode *waterfall* melalui tahapan analisis kebutuhan, perancangan dengan *UML*, implementasi menggunakan *CodeIgniter* dan *Bootstrap*, serta pengujian. Pengujian fungsional dilakukan dengan metode *blackbox* untuk memastikan seluruh fitur berjalan sesuai spesifikasi, sedangkan pengujian performa menggunakan *JMeter* menunjukkan sistem mampu merespons dengan baik pada berbagai beban pengguna. Hasil penelitian menunjukkan bahwa sistem mendukung proses penelitian secara digital dan terstruktur, serta diharapkan dapat meningkatkan efisiensi dan transparansi manajemen penelitian.

Kata Kunci :

Metode Waterfall, Sistem Informasi Penelitian, UML, Pengujian Fungsional, Pengujian Performa

Abstract

Lecturer research activities in higher education often face challenges in data management, reporting, and documentation, which can hinder the effectiveness and efficiency of research processes. To address these issues, the utilization of information technology through the development of a research information system offers a strategic solution. This study aims to design and develop a research information system for lecturers at the Faculty of Engineering, Diponegoro University. Data were collected through observation, interviews, and literature review. The system was developed using the waterfall model, encompassing stages of requirements analysis, system design with UML, implementation using CodeIgniter and Bootstrap, and testing. Functional testing was conducted using the blackbox method to ensure all features operated according to specifications, while performance testing using JMeter demonstrated the system's ability to respond effectively under various user loads. The results indicate that the system supports digital and structured research processes and is expected to enhance the efficiency and transparency of research management.

Keywords :

Waterfall Method, Research Information System, UML, Functional Testing, Performance Testing

Pendahuluan

Perguruan tinggi memiliki tanggung jawab dalam pengembangan ilmu pengetahuan melalui kegiatan penelitian dan pengabdian kepada masyarakat, sebagaimana diatur dalam Pasal 20 Undang-Undang No. 20 Tahun 2003 tentang Sistem Pendidikan Nasional dan diperkuat oleh Permenristekdikti No. 44 Tahun 2015 [1][2]. Dalam konteks ini, dosen sebagai bagian dari sivitas akademika dituntut untuk aktif melakukan penelitian sebagai bagian dari pelaksanaan Tri Dharma Perguruan Tinggi.

Namun, pelaksanaan kegiatan penelitian dosen sering kali menghadapi kendala dalam hal pengelolaan data, pelaporan, dan dokumentasi [3]. Kebutuhan akan sistem yang mampu menyajikan informasi secara cepat, tepat, dan terintegrasi menjadi sangat penting [4], khususnya di lingkungan Fakultas Teknik Universitas Diponegoro. Oleh karena itu,

pemanfaatan teknologi informasi melalui pengembangan sistem informasi penelitian menjadi solusi strategis untuk mendukung efektivitas dan efisiensi proses penelitian dosen [5][6].

Tujuan dari penelitian ini adalah untuk merancang dan membangun Sistem Informasi Penelitian Dosen yang terintegrasi dalam Sistem Informasi Penelitian Fakultas Teknik Universitas Diponegoro berbasis web. Sistem ini diharapkan dapat memfasilitasi proses pengajuan proposal, penugasan reviewer, penilaian proposal, serta pelaporan monitoring dan evaluasi hingga laporan akhir penelitian.

Sistem informasi dikembangkan menggunakan metode *waterfall*, yang terdiri dari tahapan analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan [7]. Pemodelan sistem dilakukan menggunakan *Unified Modeling Language*

(UML), basis data MySQL, serta framework web php CodeIgniter dan Bootstrap untuk antarmuka pengguna.

Hasil dari penelitian ini adalah sistem informasi yang telah melalui pengujian fungsional (*blackbox testing*) dan pengujian performa (menggunakan *JMeter*), yang menunjukkan bahwa sistem mampu merespons dengan baik terhadap berbagai skenario penggunaan. Sistem ini memberikan kontribusi nyata dalam mendukung manajemen penelitian dosen secara digital, transparan, dan terstruktur, serta dapat dikembangkan lebih lanjut ke dalam platform mobile untuk meningkatkan aksesibilitas.

Tinjauan Pustaka

Studi Literatur Terkait

Dalam pengembangan sistem informasi untuk manajemen penelitian dosen, kajian terhadap penelitian terdahulu menjadi penting untuk memahami pendekatan, teknologi, dan permasalahan yang telah diatasi. Kajian ini memberikan gambaran mengenai state of the art serta memperkuat landasan teoritis dan praktis dari penelitian yang sedang dilakukan, khususnya dalam hal efisiensi pengelolaan data dan pengujian sistem.

Penelitian oleh Utami dkk [6] merancang sistem informasi manajemen relasi mitra perusahaan berbasis web untuk meningkatkan efisiensi pengelolaan data dan komunikasi antar pihak. Metode pengembangan yang digunakan adalah *waterfall*, dengan tahapan analisis kebutuhan, perancangan menggunakan UML, implementasi, dan pengujian. Penelitian ini relevan karena menunjukkan pendekatan teknis yang serupa dan fokus pada efisiensi manajemen data, sebagaimana yang diangkat dalam penelitian ini.

Penelitian oleh Syaputri dkk [8] mengusulkan sistematasi proses perizinan usaha menggunakan model *waterfall*, yang menunjukkan efektivitas pendekatan berurutan dalam pengembangan sistem informasi berbasis kebutuhan yang telah terdefinisi. Sementara itu, Santika dkk [9] menekankan pentingnya pendekatan sistem dalam manajemen pendidikan, dengan menyoroti keterkaitan antar komponen sistem sebagai dasar pengambilan keputusan yang terstruktur. Kedua penelitian tersebut relevan karena mendukung pemilihan metode *waterfall* dan pendekatan sistematis dalam pengembangan sistem informasi penelitian dosen yang efisien dan terintegrasi.

Penelitian oleh Sasmito dan Mutasodirin [10] menerapkan teknik *black box testing* dengan pendekatan *equivalence partitioning* untuk menguji fungsionalitas aplikasi *Transcrop*, yang menunjukkan efektivitas metode pengujian berbasis skenario penggunaan dalam memastikan keandalan sistem. Sementara itu, Mahajan dkk [11]

mengembangkan *script* otomatisasi pengujian performa menggunakan *Apache JMeter* untuk mengukur responsivitas sistem berbasis web, yang berkontribusi pada peningkatan efisiensi proses evaluasi sistem. Secara kritis, kedua studi ini menekankan pentingnya pengujian fungsional dan performa dalam siklus pengembangan sistem informasi, yang sangat relevan dengan penelitian ini dalam memastikan sistem informasi penelitian dosen berjalan sesuai spesifikasi dan mampu menangani beban pengguna secara optimal.

Sistem

Sistem adalah kumpulan komponen yang saling terhubung dan bekerja sama untuk mencapai tujuan tertentu. Sistem terdiri dari sub sistem yang mendukung sistem utama, dan harus memiliki organisasi, hubungan timbal balik, integrasi, serta tujuan yang jelas [9]. Struktur sistem mencakup elemen-elemen dasar pembentuknya, sedangkan proses menjelaskan cara kerja elemen-elemen tersebut dalam mencapai tujuan [12].

Sistem dirancang untuk menangani aktivitas yang berulang dan rutin. Dalam konteks ini, struktur berfungsi sebagai fondasi sistem, sementara proses menggambarkan mekanisme kerja dari setiap elemen dalam sistem. Dengan demikian, sistem menjadi alat penting dalam mengelola dan menyederhanakan berbagai aktivitas yang kompleks [13].

Sistem Informasi

Sistem Informasi adalah hasil perkembangan teknologi informasi yang bertujuan mempermudah pengolahan data menjadi informasi yang berguna. Sistem ini terdiri dari prosedur dan komponen yang terorganisasi untuk mendukung pengambilan keputusan dan pemecahan masalah dalam suatu organisasi atau bisnis [14].

Unified Modelling Language (UML)

UML adalah alat bantu visual untuk menggambarkan dan menjelaskan sistem perangkat lunak berbasis objek [15]. UML membantu pemangku kepentingan memahami sistem dan memudahkan pengembangan dalam merancang alur kerja sistem. Diagram UML menyajikan berbagai sudut pandang sistem melalui elemen grafis yang disebut model [16].

UML tidak menjelaskan implementasi teknis, tetapi menggambarkan bagaimana sistem bekerja. Empat jenis diagram UML yang digunakan adalah: use case diagram, activity diagram, sequence diagram, dan class diagram. Diagram ini membantu dalam memahami interaksi pengguna, alur aktivitas, serta komunikasi antar objek dalam sistem [17].

Metode Penelitian

Metode Rancang Bangun Sistem Informasi

Penelitian ini menggunakan metode pengembangan sistem *waterfall*, yang merupakan pendekatan

sistematis dan berurutan dalam pengembangan perangkat lunak [18]. Metode ini terdiri dari beberapa tahapan utama yang dilakukan secara berurutan, yaitu [7]:

1. Analisis Kebutuhan

Tahap ini dilakukan untuk mengidentifikasi kebutuhan sistem informasi penelitian dosen di Fakultas Teknik Universitas Diponegoro. Informasi dikumpulkan melalui wawancara dengan pihak fakultas dan studi dokumen terkait proses bisnis penelitian dosen. Hasil dari tahap ini adalah spesifikasi kebutuhan fungsional dan non-fungsional sistem.

2. Perancangan Sistem

Berdasarkan hasil analisis kebutuhan, dilakukan perancangan sistem menggunakan pendekatan *Unified Modeling Language (UML)*. Diagram yang digunakan meliputi *use case diagram*, *activity diagram*, dan *class diagram*. Selain itu, dilakukan perancangan antarmuka pengguna (*UI*) dan struktur basis data menggunakan *MySQL*.

3. Implementasi

Tahap implementasi dilakukan dengan membangun sistem menggunakan *framework CodeIgniter* dan *Bootstrap*. *CodeIgniter* digunakan untuk pengembangan sisi *backend*, sedangkan *Bootstrap* digunakan untuk mendesain antarmuka yang responsif dan *user-friendly*.

4. Pengujian

Sistem diuji menggunakan dua pendekatan:

- *Blackbox testing* untuk menguji fungsionalitas sistem berdasarkan skenario penggunaan.
- *JMeter* untuk menguji performa sistem dalam berbagai kondisi beban pengguna.

5. Pemeliharaan

Setelah sistem diuji dan diimplementasikan, dilakukan pemeliharaan untuk memastikan sistem berjalan dengan baik dan dapat disesuaikan dengan kebutuhan pengguna di masa mendatang. Tahap ini juga mencakup perbaikan *bug* dan pengembangan fitur tambahan.

Metode *waterfall* dipilih karena sesuai untuk proyek dengan kebutuhan yang telah terdefinisi dengan jelas sejak awal, serta memungkinkan dokumentasi yang lengkap pada setiap tahap pengembangan [8].

Metode Deskriptif

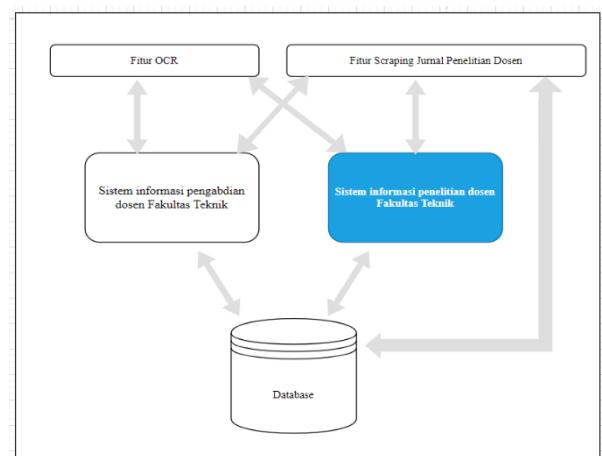
Metode deskriptif digunakan untuk memperoleh pemahaman yang mendalam mengenai kebutuhan dan proses bisnis penelitian dosen di Fakultas Teknik Universitas Diponegoro [19]. Pengumpulan data dilakukan melalui:

- *Observasi*: Pengamatan langsung terhadap proses pengajuan dan pengelolaan penelitian dosen di lingkungan fakultas.

- *Wawancara*: Dilakukan dengan pihak-pihak terkait seperti dosen, admin fakultas, dan ketua departemen untuk menggali kebutuhan sistem dan permasalahan yang dihadapi dalam proses penelitian.
- *Studi Pustaka*: Penelaahan terhadap dokumen-dokumen akademik, kebijakan institusi, serta referensi ilmiah terkait sistem informasi dan manajemen penelitian.

Gambaran Umum Sistem

Sistem Informasi Penelitian Fakultas Teknik Universitas Diponegoro dirancang dan dibangun dalam satu arsitektur yang saling terintegrasi, yang selanjutnya disebut SITEDI (Sistem Informasi Penelitian dan Pengabdian) Fakultas Teknik Universitas Diponegoro. Arsitektur SITEDI dapat dilihat dalam Gambar 1



Gambar 1. Arsitektur sistem Informasi penelitian dan pengabdian dosen

Berdasarkan Gambar 1 Sistem informasi penelitian dosen merupakan salah satu komponen dari sistem informasi terpadu di lingkungan Fakultas Teknik yang mencakup berbagai fitur dan layanan, seperti sistem informasi pengabdian dosen, fitur *OCR*, dan *scraping* jurnal. Seluruh komponen ini saling terhubung dan berinteraksi melalui basis data terpusat, membentuk satu kesatuan sistem besar yang mendukung kegiatan tridharma perguruan tinggi. Namun, dalam penelitian ini, fokus akan diarahkan secara khusus pada sistem informasi penelitian dosen. Meskipun berada dalam ekosistem sistem yang saling terintegrasi, pembahasan akan dibatasi pada aspek-aspek yang berkaitan langsung dengan pengelolaan data penelitian dosen.

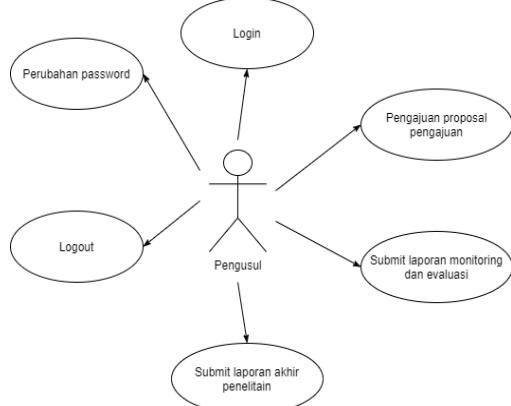
Hasil dan Pembahasan

Desain *Unified Modelling Language (UML)*

1. Use Case

Use case merupakan deskripsi fungsional yang menggambarkan interaksi antara pengguna (aktor) dan sistem untuk mencapai tujuan tertentu dalam suatu proses bisnis [20].

a. Use case pengusul

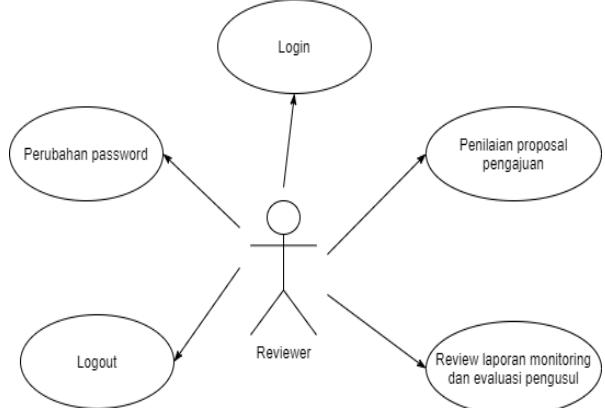


Gambar 2. Use case pengusul

Diagram *use case* pengusul pada Gambar 2 menunjukkan beberapa fitur diantaranya:

- *Login*
Fitur *login* memungkinkan pengguna membuka halaman dengan *username* dan *password* yang sesuai.
- *Logout*
Fitur *logout* memungkinkan pengguna untuk mengakhiri session dan keluar dari halaman utama.
- *Perubahan password*
Fitur perubahan *password* memungkinkan pengguna untuk melakukan perubahan *password* sesuai keinginan pengguna.
- *Pengajuan proposal pengajuan*
Fitur pengajuan proposal pengajuan memungkinkan pengguna untuk melakukan pengajuan penelitian dan pendanaan dengan mengisi *form* yang telah disediakan.
- *Submit laporan monitoring dan evaluasi (monev)*
Fitur *submit* laporan monitoring dan evaluasi (monev) memungkinkan pengguna untuk melakukan pengumpulan laporan monev yang meliputi laporan kemajuan, laporan belanja 70% dan catatan *logbook* dari penelitian yang telah dilaksanakan.
- *Submit laporan akhir*
Fitur *submit* laporan akhir memungkinkan pengguna untuk melakukan pengumpulan laporan monev yang meliputi laporan final, laporan belanja 100%, luaran dan *logbook* dari penelitian yang telah selesai dilaksanakan.

b. Use case reviewer



Gambar 3. Use case reviewer

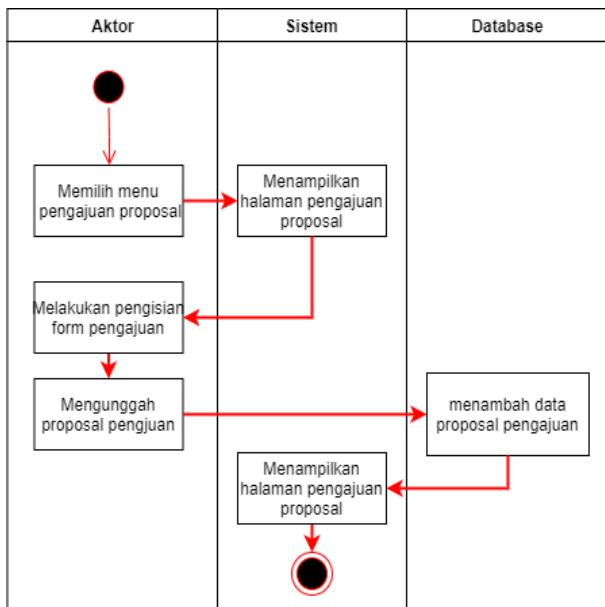
Diagram *use case* reviewer pada Gambar 3 menunjukkan beberapa fitur diantaranya:

- *Login*
Fitur *Login* memungkinkan pengguna membuka halaman dengan *username* dan *password* yang sesuai.
- *Logout*
Fitur *logout* memungkinkan pengguna untuk mengakhiri session dan keluar dari halaman utama.
- *Perubahan password*
Fitur perubahan *password* memungkinkan pengguna untuk melakukan perubahan *password* sesuai keinginan pengguna.
- *Penilaian proposal pengajuan*
Fitur penilaian proposal pengajuan memungkinkan pengguna untuk melakukan penilaian berdasarkan proposal dan komponen penilaian yang telah ditentukan.
- *Review laporan monitoring dan evaluasi*
Fitur *review* laporan monitoring dan evaluasi memungkinkan pengguna untuk melakukan *reviewing* yang meliputi *monitoring* dan mengomentari laporan monev yang masuk.

2. Activity Diagram

Activity diagram merupakan diagram yang menggambarkan alur kerja atau urutan aktivitas dalam suatu proses sistem, termasuk interaksi antara pengguna dan sistem secara dinamis. Diagram ini membantu memvisualisasikan logika proses bisnis dan mempermudah pemahaman terhadap alur sistem secara menyeluruh [21].

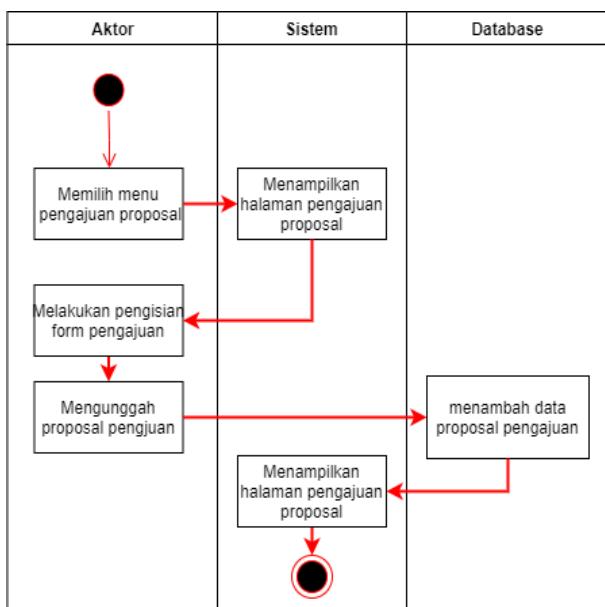
a. *Activity diagram* pengajuan proposal



Gambar 3. *Activity diagram* pengajuan proposal

Gambar 3 menggambarkan alur proses yang dilakukan oleh pengguna (pengusul) dalam mengajukan proposal penelitian melalui sistem. Proses dimulai ketika pengguna memilih menu pengajuan proposal, yang kemudian direspon oleh sistem dengan menampilkan halaman pengajuan. Selanjutnya, pengguna mengisi form pengajuan dan mengunggah dokumen proposal. Setelah data lengkap diinput, sistem akan menyimpan data proposal ke dalam database. Sebagai penanda bahwa proses telah berhasil, sistem kembali menampilkan halaman pengajuan proposal kepada pengguna.

b. *Activity diagram* review proposal



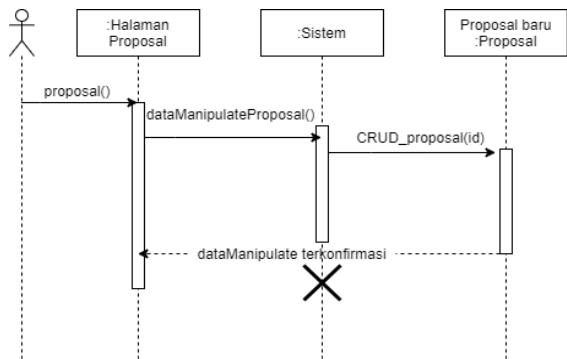
Gambar 4. *Activity diagram* review proposal

Gambar 4 menjelaskan proses penilaian proposal penelitian yang dibagi ke dalam tiga komponen utama: Aktor, Sistem, dan Database. Proses dimulai dari sisi aktor (*reviewer*) yang memilih menu penilaian proposal. Sistem kemudian merespons dengan menampilkan halaman penilaian proposal. Setelah itu, sistem membaca data proposal dari database untuk ditampilkan. Aktor kemudian melakukan manipulasi data nilai, seperti memberikan skor atau komentar terhadap proposal. Sistem kembali menampilkan halaman penilaian untuk memastikan data telah diperbarui, dan database melakukan operasi *CRUD* (*Create, Read, Update, Delete*) terhadap data proposal dan nilai yang diberikan.

3. *Sequence Diagram*

Sequence diagram merupakan diagram interaksi yang menggambarkan urutan pesan atau komunikasi antar objek dalam suatu sistem berdasarkan waktu. Diagram ini membantu memvisualisasikan bagaimana proses atau fungsi dijalankan secara berurutan dari awal hingga akhir [22].

a. *Sequence diagram* pengajuan proposal

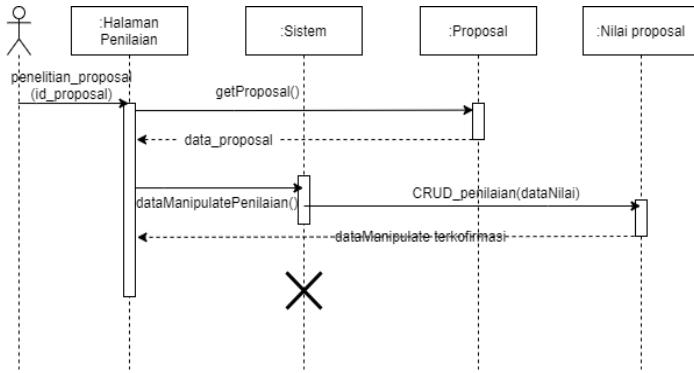


Gambar 5. *Sequence diagram* pengajuan proposal

Gambar 5 menunjukkan *sequence diagram* *CRUD* data proposal antara tiga komponen utama: Halaman Proposal, Sistem, dan Proposal Baru dalam melakukan operasi Create, Read, Update, dan Delete terhadap data proposal. Proses dimulai ketika pengguna mengakses halaman proposal dan memicu aksi *proposal()*. Halaman tersebut kemudian mengirimkan permintaan *dataManipulateProposal()* ke sistem. Sistem selanjutnya melakukan operasi *CRUD_proposal(id)* terhadap entitas data proposal baru. Setelah operasi selesai, sistem mengirimkan konfirmasi *dataManipulate terkonfirmasi* kembali ke halaman proposal sebagai tanda bahwa proses manipulasi data telah berhasil dilakukan.

layout penelitian yang berguna untuk mengarahkan user ke halaman penelitian.

b. Sequence diagram review proposal

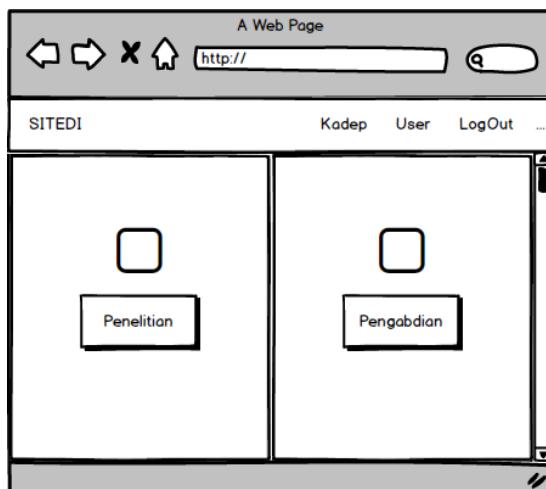


Gambar 6. Sequence diagram review proposal

Gambar 6 menunjukkan *sequence diagram review proposal* yang menggambarkan alur interaksi dalam proses penilaian proposal penelitian antara halaman penilaian, sistem, dan dua entitas data: proposal dan nilai proposal. Proses dimulai ketika halaman penilaian mengirim permintaan penelitian_proposal(id_proposal) ke sistem. Sistem kemudian mengambil data proposal melalui pesan getProposal() ke entitas Proposal, yang membalas dengan data_proposal. Setelah data proposal diterima, sistem melakukan operasi CRUD_penilaian(dataNilai) terhadap entitas Nilai Proposal. Nilai Proposal kemudian mengirimkan konfirmasi dataManipulate terkonfirmasi kembali ke sistem. Terakhir, sistem mengirimkan hasil manipulasi penilaian kembali ke halaman penilaian melalui pesan dataManipulatePenilaian.

Desain User Interface

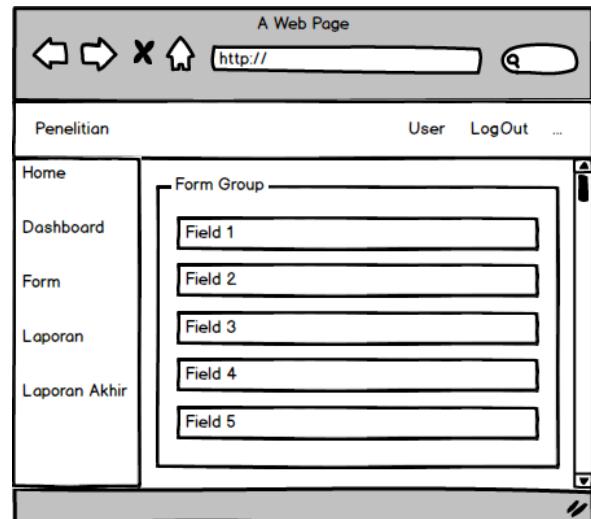
1. Homepage



Gambar 7. Usulan halaman homepage

Gambar 7 menunjukkan homepage yang merupakan halaman utama dalam SITEDI. Homepage memiliki 2 layout utama yaitu layout penelitian yang berguna untuk mengarahkan user ke halaman pengabdian dan

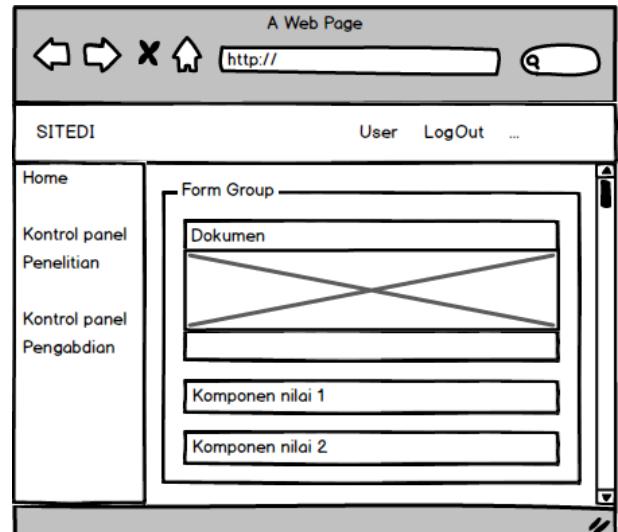
2. Pengajuan proposal penelitian



Gambar 8. Usulan pengajuan proposal penelitian

Gambar 8 menunjukkan *form pengajuan proposal* yang terdiri dari lima field input (Field 1 hingga Field 5), yang digunakan untuk mengisi informasi penting terkait proposal seperti judul, abstrak, skema pendanaan, dan dokumen pendukung.

2. Review proposal penelitian



Gambar 9. Usulan review proposal penelitian

Gambar 9 menampilkan antarmuka halaman web yang dirancang untuk memfasilitasi proses penilaian proposal oleh *reviewer*. Area utama halaman menampilkan bagian "Form Group" yang terdiri dari tampilan atau unggahan dokumen proposal serta dua komponen penilaian yang diberi label "Komponen nilai 1" dan "Komponen nilai 2".

Implementasi Sistem Informasi

1. Halaman Login

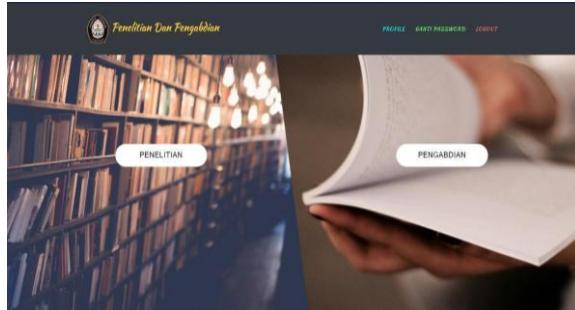


Gambar 10. Tampilan halaman *login*

Gambar 10 merupakan tampilan awal dari SITEDI. Pada tampilan awal SITEDI, *user* harus *login* terlebih dahulu sebelum masuk ke menu main menu masing-masing *role user*.

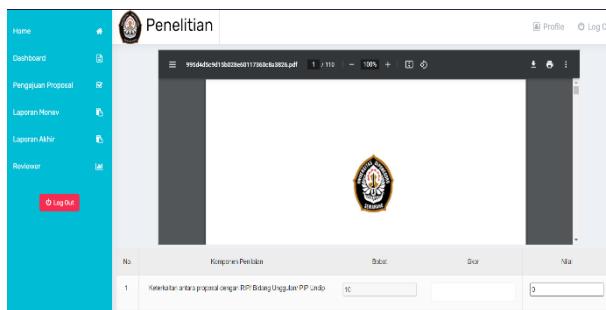
2. Halaman Utama

Setelah *login*, maka *user* akan diarahkan ke halaman utama. Halaman utama ini berisi pilihan untuk masuk ke dalam sistem penelitian atau penelitian seperti yang ditunjukkan pada Gambar 11.



Gambar 11. Tampilan halaman utama

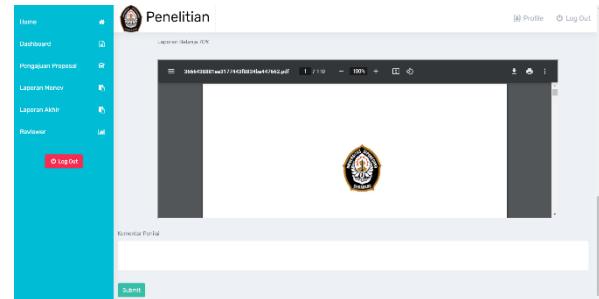
3. Halaman Penilaian Proposal



Gambar 12. Tampilan halaman penilaian proposal

Gambar 12. merupakan tampilan halaman penilaian. Pada halaman ini menampilkan Informasi mengenai proposal yang akan dinilai dan juga *form* penilaian yang telah disediakan.

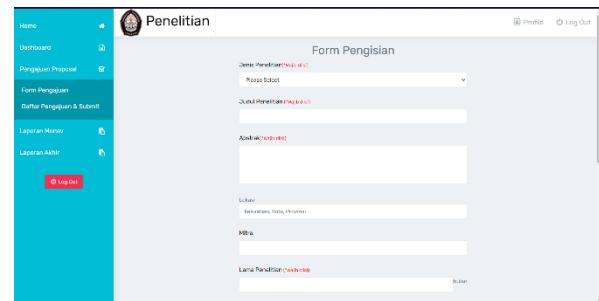
4. Halaman *Review* Laporan Monev



Gambar 13. Tampilan halaman *review* laporan monev

Gambar 13 merupakan tampilan halaman *review* laporan monev. Pada halaman ini menampilkan Informasi mengenai laporan monev yang akan direview dan juga *form* review yang telah disediakan.

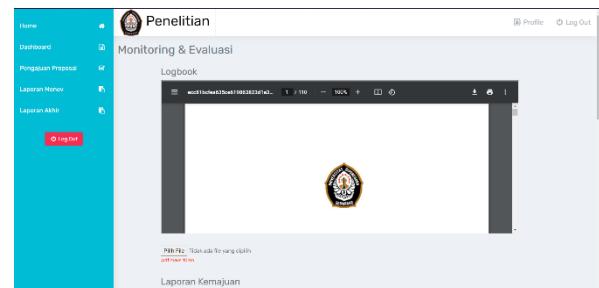
6. Halaman Pengajuan Proposal



Gambar 14. Tampilan halaman pengajuan proposal

Gambar 14 merupakan tampilan halaman daftar pengajuan proposal. Pada halaman ini menampilkan *form* pengisian yang berisi Informasi terkait penelitian yang diajukan seperti judul, abstrak, jenis penelitian, sumber pendanaan, luaran, lama pelaksanaan, jumlah pendanaan, lokasi, mitra dan anggota penelitian. di halaman ini juga terdapat *form* upload yang digunakan untuk mengunggah *file* proposal.

8. Halaman Laporan Monev



Gambar 15. Tampilan halaman laporan monev

Gambar 15 merupakan tampilan halaman laporan monev. Pada halaman ini menampilkan *form*

pengisian yang berisi informasi terkait laporan akhir dan *form upload* untuk mengunggah *file-file* pendukung laporan monev seperti laporan belanja, *logbook* dan laporan kemajuan.

Pengujian Fungsionalitas

Pengujian sistem dilakukan menggunakan metode *blackbox testing*, yaitu metode pengujian yang berfokus pada fungsionalitas sistem tanpa memperhatikan struktur internal kode program [10]. Tujuan dari pengujian ini adalah untuk memastikan bahwa setiap fitur dalam sistem informasi penelitian dosen berjalan sesuai dengan kebutuhan dan spesifikasi yang telah ditentukan [23].

Pengujian dilakukan pada beberapa fitur utama, seperti:

- Form pengajuan proposal: memastikan bahwa pengguna dapat mengisi dan mengunggah proposal dengan benar.
- Penilaian proposal oleh reviewer: memastikan bahwa reviewer dapat memberikan nilai dan komentar terhadap proposal yang ditugaskan.
- Pengisian laporan monev dan laporan akhir: memastikan bahwa pengguna dapat mengunggah dokumen dan data yang diperlukan sesuai tahapan penelitian.

Setiap skenario pengujian dilakukan dengan memberikan input yang valid dan tidak valid untuk mengamati bagaimana sistem merespons. Hasil pengujian menunjukkan bahwa sistem berhasil menampilkan halaman yang sesuai, menyimpan data ke dalam database, dan memberikan notifikasi atau umpan balik kepada pengguna sesuai dengan alur yang telah dirancang. Tidak ditemukan kesalahan fungsional pada fitur-fitur utama, yang menunjukkan bahwa sistem telah memenuhi kriteria kelayakan dari sisi fungsionalitas. Pengujian membuktikan bahwa sistem informasi yang dikembangkan telah mampu menangani proses bisnis penelitian dosen secara efektif dan dapat diandalkan dalam lingkungan operasional.

Pengujian Performa Sistem

Pengujian performa sistem menggunakan *JMeter* dengan melakukan *HTTP Request* menuju ke alamat sistem [11]. Pengujian ini akan memvariasikan 3 variabel yaitu jumlah user, waktu selang antara user satu dengan *user* lain dalam mengakses sistem inforamasi (*ramp-up*), dan jumlah perulangan tiap *user* dalam mengakses sistem Informasi (*loop-count*) [24].

1. Variasi Jumlah User

Tabel 1. Pengujian performa dengan variasi jumlah *user*

| No | Jumlah user | Average response time (ms) | Status |
|----|-------------|----------------------------|------------|
| 1 | 10 | 204 | ok (hijau) |
| 2 | 50 | 209 | ok (hijau) |
| 3 | 100 | 225 | ok (hijau) |

Berdasarkan pengujian variasi jumlah *user* pada Tabel 1 menunjukkan *hasil* dari 3 variasi yang dilakukan, yaitu untuk jumlah 10 user menghasilkan *average response time* 204 ms, untuk jumlah 50 user menghasilkan *average response time* 209 ms dan untuk jumlah 100 user menghasilkan *average response time* 225 ms. Semua pengujian menghasilkan status ok (hijau), hal ini menunjukkan sistem memiliki performa yang baik dalam menanggapi respon yang diberikan.

2. Variasi Jumlah *Ramp-up*

Tabel 2. Pengujian performa dengan variasi jumlah *ramp-up*

| No | Ramp-up (s) | Average response time (ms) | Status |
|----|-------------|----------------------------|------------|
| 1 | 1 | 206 | ok (hijau) |
| 2 | 5 | 200 | ok (hijau) |
| 3 | 10 | 195 | ok (hijau) |

Berdasarkan pengujian variasi jumlah *user* pada Tabel 2 menunjukkan *average response time* dan status yang dihasilkan dari 3 variasi jumlah *user* yang dilakukan yaitu untuk jumlah *ramp-up* 1 menghasilkan *average response time* 206 ms dengan status ok (hijau), untuk jumlah *ramp-up* 5 menghasilkan *average response time* 200 ms dengan status ok (hijau) dan untuk jumlah *ramp-up* 10 menghasilkan *average response time* 195 ms dengan status ok (hijau). Hal ini menunjukkan sistem memiliki performa yang baik dalam menanggapi respon yang diberikan.

3. Variasi Jumlah *Loop-count*

Tabel 3. Pengujian performa dengan variasi jumlah *loop-count*

| No | Loop-count | Average response time (ms) | Status |
|----|------------|----------------------------|------------|
| 1 | 1 | 191 | ok (hijau) |
| 2 | 10 | 115 | ok (hijau) |
| 3 | 20 | 112 | ok (hijau) |

Berdasarkan pengujian variasi jumlah *user* pada Tabel 3 menunjukkan *average response time* dan status yang dihasilkan dari 3 variasi jumlah *user* yang dilakukan yaitu untuk jumlah *loop-count* 1 menghasilkan *average response time* 191 ms dengan status ok (hijau), untuk jumlah *loop-count* 10 menghasilkan *average response time* 115 ms dengan status ok (hijau) dan untuk jumlah *loop-count* 20 menghasilkan *average response time* 112 ms dengan status ok (hijau). Hal ini menunjukkan sistem memiliki performa yang baik dalam menanggapi respon yang diberikan.

4. Pengujian Performa *Query Database*

Pengujian performa *query database* dilakukan untuk mengetahui bagaimana performa *database* ketika menerima respon dari *query* yang diperintahkan [25]. Pengujian yang dilakukan melibatkan pengambilan data dari beberapa tabel yaitu *proposal_penelitian*, *laporan_monev_penelitian*, *laporan_akhir_penelitian*, *reviewer_penelitian*,

dosen, luaran_penelitian, sumberdana, detail_nilai, dan nilai_proposal.

Tabel 0. Pengujian performa dengan variasi jumlah *ramp-up*

| No | Tabel | Jumlah data (Byte) | Average respon se time (ms) | Status |
|----|--------------------------|--------------------|-----------------------------|------------|
| 1 | proposal_penelitian | 329502 | 19 | ok (hijau) |
| 2 | laporan_monev_penelitian | 3358 | 12 | ok (hijau) |
| 3 | laporan_akhir_penelitian | 3499 | 12 | ok (hijau) |
| 4 | reviewer_penelitian | 1049 | 9 | ok (hijau) |
| 5 | dosen | 36642 | 13 | ok (hijau) |
| 6 | luaran_penelitian | 493 | 7 | ok (hijau) |
| 7 | sumberdana | 94 | 6 | ok (hijau) |
| 8 | detail_nilai | 63178 | 18 | ok (hijau) |
| 9 | nilai_proposal | 50810 | 11 | ok (hijau) |

Berdasarkan pengujian performa *query* pada Tabel 4 menunjukkan respon yang baik dimana nilai tertinggi untuk *average response time* hanya mencapai 19 ms dengan jumlah data sebesar 329502 Byte dan semua proses pengujian memberikan status ok (hijau). Hal ini menunjukkan sistem memiliki performa yang baik dalam menanggapi respon yang diberikan.

Kesimpulan dan Saran

Kesimpulan

1. Sistem informasi penelitian dosen yang dirancang dan dibangun dalam penelitian ini berhasil mengintegrasikan proses pengajuan proposal, penilaian, serta pelaporan hasil penelitian secara digital dan terstruktur, sesuai dengan kebutuhan Fakultas Teknik Universitas Diponegoro.
2. Pengembangan sistem menggunakan metode waterfall dan pemodelan UML terbukti efektif dalam menghasilkan sistem yang fungsional dan mudah dipahami oleh pengguna, dengan antarmuka yang responsif melalui framework CodeIgniter dan Bootstrap.
3. Hasil pengujian fungsional dan performa menunjukkan bahwa sistem mampu merespons dengan baik terhadap berbagai skenario penggunaan, serta memiliki waktu respons yang cepat dan stabil, sehingga layak digunakan dalam lingkungan operasional fakultas.

Saran

1. Sistem informasi ini dapat dikembangkan lebih lanjut ke dalam bentuk aplikasi mobile untuk meningkatkan aksesibilitas dan fleksibilitas pengguna, terutama bagi dosen yang sering bekerja secara mobile.
2. Perlu dilakukan integrasi sistem ini dengan sistem informasi akademik dan kepegawaian yang sudah ada di universitas agar data penelitian dosen dapat dimanfaatkan secara lebih luas untuk keperluan pelaporan institusional dan pengambilan keputusan strategis.

Daftar Pustaka

- [1] Republik Indonesia, *Undang-Undang Republik Indonesia Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional*. Jakarta: <https://static.buku.kemdikbud.go.id/content/undang-undang/UU%20No%202020%20Tahun%202003%20tentang%20Sistem%20Pendidikan%20Nasional.pdf>, 2003.
- [2] Kementerian Ristekdikti, *Peraturan Menteri Riset, Teknologi, dan Pendidikan Tinggi Republik Indonesia Nomor 44 Tahun 2015 tentang Standar Nasional Pendidikan Tinggi*. Jakarta: <https://silemkarma.kemdiktisaintek.go.id/assets/standuan/PERMENRISTEKDIKTI-NOMOR-44-TAHUN-2015-TENTANG-SNPT-SALINAN.pdf>, 2015.
- [3] A. Ahmad *et al.*, “Bimbingan Teknis Pengisian Beban Kinerja Dosen (BKD) Melalui Aplikasi SISTER di Lingkungan Sekolah Tinggi Ilmu Administrasi Pelita Nusantara Tahun 2024,” *Jurnal Pengabdian Masyarakat Bangsa*, vol. 2, no. 8, pp. 3147–3155, Oct. 2024, doi: 10.59837/jpmba.v2i8.1414.
- [4] A. A. Zulfa, T. Ibrahim, and O. Arifudin, “Peran Sistem Informasi Akademik Berbasis Web Dalam Upaya Meningkatkan Efektivitas Dan Efisiensi Pengelolaan Akademik Di Perguruan Tinggi,” *Jurnal Tahsinia*, vol. 6, no. 1, pp. 115–134, 2025, doi: <https://doi.org/10.57171/jt.v6i1.615>.
- [5] P. A. Sunarya, M. Asri, N. Azizah, and C. P. Lim, “Evaluation of Educational Information Systems for Data and Decision Management,” *Jurnal MENTARI: Manajemen, Pendidikan dan Teknologi Informasi*, vol. 3, no. 2, pp. 118–126, Mar. 2025, doi: 10.33050/mentari.v3i2.738.
- [6] L. L. Utami, D. S. Nurfadilah, and R. Zulkifli, “RANCANG BANGUN SISTEM INFORMASI MANAJEMEN RELASI MITRA PERUSAHAAN PADA PT SARANA KREASINDO TEKNOLOGI,” *Journal of Information System Management (JOISM)*, vol. 6, no. 2, pp. 171–178, Jan. 2025, doi: 10.24076/joism.2025v6i2.1949.
- [7] D. B. J. Rao, P. Kalpana, G. S. N. Kumar, and N. M. Atcha, “A Comparative Analysis of Software Development Models: Waterfall, Agile and DevOps,” in *Intelligent Systems and Sustainable Computing*, V. S. Reddy, V. K. Prasad, J. Wang, and N. M. Rao Dasari, Eds., Singapore: Springer Nature Singapore, 2025, pp. 589–599. doi: https://doi.org/10.1007/978-981-97-8355-7_51.
- [8] L. Syaputri, E. Syahrani, M. Rivan, J. Suhanda, and S. Rahayu, “SISTEMATISASI PERIZINAN LAPAK DENGAN MODEL WATERFALL UNTUK PENDAFTARAN IZIN USAHA,” *Jurnal Riset Teknik Komputer*, vol. 1, no. 4, pp. 47–57, 2024, doi: <https://doi.org/10.69714/6srzd30>.
- [9] D. Santika, D. N. S. Haq, S. Anwar, and I. Fatoni, “PENDEKATAN SISTEM DALAM MANAJEMEN PENDIDIKAN,” *Pendas: Jurnal Ilmiah Pendidikan Dasar*, vol. 9, no. 04, pp. 782–789, 2024, doi: <https://doi.org/10.23969/jp.v9i04.19526>.

- [10] G. W. Sasmito and M. A. Mutasodirin, "Black Box Testing with Equivalence Partitions Techniques in Transcrop Applications," in *2023 6th International Conference of Computer and Informatics Engineering (IC2IE)*, IEEE, Sep. 2023, pp. 53–58. doi: 10.1109/IC2IE60547.2023.10331562.
- [11] G. Mahajan, Dr. V. Attar, and S. Kalamkar, "Generation of JMeter scripts for performance testing of Moodle server," in *2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, IEEE, Dec. 2022, pp. 2277–2281. doi: 10.1109/ICAC3N56670.2022.10074284.
- [12] T. Haruna, "Global structure of directed networks emerging from a category theoretical formulation of the idea 'Objects as processes, interactions as interfaces,'" in *ECAL 2011: The 11th European Conference on Artificial Life*, MIT Press, 2011. doi: 10.7551/978-0-262-29714-1-ch049.
- [13] J. A. O'Brien and G. M. Marakas, *Introduction to Information Systems*, 16th ed. New York: McGraw-Hill Irwin, 2013. [Online]. Available: <https://books.google.co.id/books?id=R6o7PgAAQAAJ>
- [14] R. Stair and G. Reynolds, *Principles of Information Systems*, 14th ed. Boston: Cengage Learning, 2020. [Online]. Available: <https://books.google.co.id/books?id=j0VjwAEACAAJ>
- [15] M. Guo, C. Zhang, and F. Wang, "What is the Further Evidence about UML? - A Systematic Literature Review," in *2017 24th Asia-Pacific Software Engineering Conference Workshops (APSECW)*, IEEE, Dec. 2017, pp. 106–113. doi: 10.1109/APSECW.2017.28.
- [16] C. Lange, M. Wijns, and M. Chaudron, "A Visualization Framework for Task-Oriented Modeling Using UML," in *2007 40th Annual Hawaii International Conference on System Sciences (HICSS'07)*, IEEE, 2007, pp. 289a–289a. doi: 10.1109/HICSS.2007.44.
- [17] A. Belghiat, "Interleaving semantics and verification of UML 2 dynamic interactions using process algebra," *Sci Comput Program*, vol. 246, p. 103334, Dec. 2025, doi: 10.1016/j.scico.2025.103334.
- [18] A. A. Aguilera and O. S. Gómez, "A study on quality and efficiency of a waterfall-like software development process applied by pair and solo programmers [Estudio de calidad y eficiencia de un enfoque de desarrollo software secuencial con programadores solos y en pareja]," *Ingeniare. Revista chilena de ingeniería*, vol. 27, no. 2, pp. 304–318, Apr. 2019, doi: 10.4067/S0718-33052019000200304.
- [19] W. T. Saputro, "Metode deskripsi untuk mengetahui pola belanja konsumen pada data penjualan," *INTEK: Jurnal Informatika dan Teknologi Informasi*, vol. 3, no. 1, pp. 25–33, 2020, doi: <https://doi.org/10.37729/intek.v3i1.480>.
- [20] Z. Wang, "A modeling approach for use-cases model in UML," in *2012 IEEE Fifth International Conference on Advanced Computational Intelligence (ICACI)*, IEEE, Oct. 2012, pp. 176–179. doi: 10.1109/ICACI.2012.6463145.
- [21] P. E. Patel and N. N. Patil, "Testcases Formation Using UML Activity Diagram," in *2013 International Conference on Communication Systems and Network Technologies*, IEEE, Apr. 2013, pp. 884–889. doi: 10.1109/CSNT.2013.191.
- [22] H. Cheers and Y. Lin, "Reverse Engineering UML Sequence Diagrams for Program Comprehension Activities," in *2020 5th International Conference on Innovative Technologies in Intelligent Systems and Industrial Applications (CITISIA)*, IEEE, Nov. 2020, pp. 1–10. doi: 10.1109/CITISIA50690.2020.9371851.
- [23] I. Darma, G. Dana, I. Bhaskara, N. Yuniari, I. Kumara, and I. Raharja, "Pengembangan Website Dinamis Fakultas dan Program Studi dengan Metode Agile," *Jurnal Komputer, Informasi dan Teknologi*, vol. 5, no. 1, p. 12, Feb. 2025, doi: 10.53697/jkomitek.v5i1.2282.
- [24] H. Alnuhait, W. Alzyadat, A. Althunibat, H. Kahtan, B. Zaqaibeh, and H. A. Al-Khawaja, "Web application performance assessment: A study of responsiveness, throughput, and scalability," *International Journal of ADVANCED AND APPLIED SCIENCES*, vol. 11, no. 9, pp. 214–226, Sep. 2024, doi: 10.21833/ijaas.2024.09.023.
- [25] J. Agnihotri and R. Phalnikar, "Development of Performance Testing Suite Using Apache JMeter," in *Intelligent Computing and Information and Communication*, S. Bhalla, V. Bhateja, A. A. Chandavale, A. S. Hiwale, and S. C. Satapathy, Eds., Singapore: Springer Singapore, 2018, pp. 317–326. doi: https://doi.org/10.1007/978-981-10-7245-1_32.