

PERANCANGAN APLIKASI MANAJEMEN DATA PUBLIKASI DAN PENELITIAN

Eric Anthony¹⁾ Tony²⁾

^{1),2)} Program Studi Sistem Informasi, Fakultas Teknologi Informasi, Universitas Tarumanagara
Jl. Letjen S. Parman No.1, Jakarta 11440. Indonesia
email : eric.825200050@stu.untar.ac.id¹⁾ tony@fti.untar.ac.id²⁾

ABSTRAK

Aplikasi Manajemen Data Publikasi dan Penelitian pada FTI Universitas Tarumanagara (UNTAR) menggunakan teknologi HTML, CSS, dan JavaScript untuk front end, dengan jQuery untuk menyederhanakan *scripting*, serta Node.js dan Express.js untuk *backend*, didukung oleh PostgreSQL sebagai *database*. Tujuannya adalah meningkatkan efisiensi pengelolaan data akademik dengan memberikan aksesibilitas yang lebih baik. Dengan fitur-fitur seperti grafik statistik yang *interaktif*, pencarian yang mudah, dan integrasi dengan sistem identifikasi *digital*, aplikasi ini memungkinkan *staff* akademik dan dosen di FTI UNTAR untuk mengelola dan menganalisis data publikasi dan penelitian dengan lebih efisien. Evaluasi keberhasilan aplikasi ini akan memperhatikan tingkat penerimaan pengguna, efisiensi administratif, *aksesibilitas* informasi dan umpan balik dari pengguna. Aplikasi ini dapat menghemat waktu dan meningkatkan efisiensi dalam pengelolaan data publikasi dan penelitian di FTI UNTAR.

Kata Kunci

administrasi, publikasi, penelitian

1. Pendahuluan

Perkembangan teknologi informasi telah menjadi kekuatan yang mengubah cara beroperasinya berbagai sektor kehidupan, termasuk dunia akademik. Fakultas Teknologi Informasi (FTI) di Universitas Tarumanagara (UNTAR) mengakui pentingnya beradaptasi dengan perubahan ini, terutama dalam mengelola data publikasi dan penelitian yang semakin kompleks. Saat ini FTI UNTAR masih mengandalkan pihak ketiga dalam pengelolaan data publikasi, tantangan baru muncul dengan kemajuan teknologi yang menghadirkan volume data yang semakin besar dalam aktivitas akademik.

Untuk mengatasi tantangan ini, diperlukan solusi inovatif yang dapat mengintegrasikan teknologi informasi ke dalam pengelolaan data. Oleh karena itu, FTI UNTAR merencanakan untuk mengembangkan aplikasi berbasis web yang menggunakan bahasa pemrograman JavaScript dan framework Node JS, didukung oleh database PostgreSQL. Dengan

menggunakan teknologi ini diharapkan aplikasi dapat memberikan kemudahan dalam penyimpanan, pencarian, dan pengelolaan data secara efisien.

Aplikasi yang diusulkan akan memberikan akses mudah dan cepat ke informasi tentang publikasi dan penelitian yang relevan. Ini akan mengurangi waktu dan upaya yang sebelumnya dibutuhkan untuk pencarian manual yang memakan waktu. Selain itu, aplikasi akan dilengkapi dengan *fitur-fitur* tambahan seperti pengelompokan data untuk memfasilitasi pengelolaan informasi, visualisasi statistik untuk memahami tren dan pola dalam data, dan kemampuan untuk menghasilkan laporan secara otomatis untuk memfasilitasi pelaporan dan analisis lebih lanjut.

Diharapkan bahwa dengan adopsi aplikasi ini, FTI UNTAR akan dapat meningkatkan efisiensi dan *aksesibilitas* terhadap informasi akademik. *Staf* dan mahasiswa akan dapat lebih fokus pada kegiatan inti mereka tanpa terganggu oleh masalah *administratif* yang berlebihan. Ini akan membawa manfaat besar bagi kemajuan akademik dan penelitian di lingkungan FTI UNTAR, serta meningkatkan reputasi dan kontribusi institusi dalam dunia akademik.

Pada perancangan Aplikasi Manajemen Data Publikasi dan Penelitian pada FTI Untar ini menggunakan beberapa jurnal sebagai *referensi* dari penelitian terdahulu yang dapat dilihat sebagai berikut :

Jensen [1] membuat aplikasi Sistem Informasi dan Koordinasi Kegiatan Ibadah Berbasis Web untuk Komunitas *Next Gen* di Gereja *House of Prayer*. Dengan menekankan peran teknologi informasi dalam kehidupan gereja modern, penulis mencatat kendala koordinasi antar anggota pelayan dan masalah *administrasi* manual dengan *Google Form*. Aplikasi ini bertujuan untuk meningkatkan efisiensi dan efektivitas operasional pelayanan gereja, terutama di masa pandemi COVID-19. Metode pengembangan yang digunakan adalah *System Development Life Cycle (SDLC) Waterfall*, dengan tahapan yang mencakup *requirement, design, implementation, verification, dan maintenance*. Dalam hasil dan pembahasan, jurnal menyajikan peran-peran dalam sistem aplikasi, seperti *Superuser, Life Assistant, dan PIC Pengajaran*, beserta fungsi dan *aksesibilitas* masing-masing. Kesimpulan menyoroti kemampuan aplikasi dalam memudahkan koordinasi, dokumentasi data jemaat, dan meningkatkan efektivitas pelayanan

gereja. Saran untuk perbaikan mencakup peningkatan *fitur*, tampilan antarmuka yang lebih menarik, dan tambahan *fitur* monitoring data jemaat. Secara keseluruhan, jurnal ini memberikan gambaran komprehensif tentang implementasi teknologi informasi untuk meningkatkan kualitas pelayanan gereja di GBI *House of Prayer*.

Susanto *et al.* [2] membuat Sistem Informasi Akademik Siswa Berbasis Web di SMAN 1 Ngabang, Kalimantan Barat, sebagai solusi efisien untuk pengolahan data akademik. Penulis menyoroti kendala dalam penggunaan Excel dan pencatatan manual di sekolah tersebut. Dengan menerapkan metode *Rapid Application Development (RAD)* dan *Unified Modeling Language (UML)*, penelitian ini berhasil merancang sistem dengan siklus pengembangan singkat, fokus pada perencanaan kebutuhan, desain sistem, pengembangan, dan implementasi. Pengujian sistem menggunakan metode *white box* melibatkan *admin*, kepala sekolah, guru, dan murid. Antarmuka web dan web mobile dirancang untuk memudahkan akses. Kesimpulan jurnal menekankan bahwa solusi ini berhasil meningkatkan efisiensi pengelolaan data akademik di SMAN 1 Ngabang, terutama pada akses *admin*, kepala sekolah, guru, dan siswa.

Lorenzo [3] membuat Aplikasi Penyewaan Jasa Dekorasi berbasis *web* pada Gianluca Project, sebuah bisnis dekorasi di Jakarta. Aplikasi web dirancang untuk meningkatkan efisiensi dalam pemesanan dan transaksi, menggantikan penggunaan WhatsApp. Metode pengembangan yang digunakan adalah *SDLC Waterfall* dengan tahapan analisis, desain, implementasi, pengujian, dan pemeliharaan sistem. Dengan menggunakan PHP, XAMPP, MySQL, dan konsep penyewaan. Jurnal menyoroti penerapan *SDLC Waterfall* melalui *diagram use case*, *ERD*, dan *class diagram*. Pengujian melibatkan pemilik dan *admin* Gianluca Project, dengan fokus pada pengujian logika dan fungsionalitas. Kesimpulan menekankan manfaat pengembangan sistem untuk Gianluca Project dalam meningkatkan efisiensi, layanan pelanggan, dan *aksesibilitas*. Saran yang diberikan termasuk perbaikan tampilan UI, penambahan metode pembayaran, dan peningkatan keamanan aplikasi web.

Ansar [4] merancang dan membuat aplikasi Sistem Informasi Berbasis *Website* untuk Profil Kelurahan Benteng. Fokus utama meliputi perancangan antarmuka dan sistem database, implementasi, dan pengujian. Penelitian menggunakan metode *Research and Development (R&D)* dengan lokasi di Kelurahan Benteng, Kota Palopo, dan dilaksanakan selama 4 bulan. Analisis kebutuhan dilakukan sebelum merancang sistem, mengidentifikasi masalah *administrasi* dan kebutuhan Kelurahan Benteng. Metode pengembangan *website* ini mencakup langkah-langkah seperti analisis, desain, pengkodean, dan pengujian. Hasilnya mencakup rancang bangun *website* dengan *fitur-fitur* seperti profil Kelurahan, visi misi, struktur organisasi, galeri, dan kontak. Pengujian dilakukan oleh ahli *website*, menilai

usabilitas, dan memberikan presentasi kelayakan. Ditemukan keunggulan dalam *usability* dan tampilan yang ramah pengguna, meskipun terdapat kelemahan pada sistem pesan masuk yang tidak memungkinkan *admin* untuk membalas pesan. Kesimpulannya, *website* ini dapat diimplementasikan di Kantor Kelurahan Benteng, memberikan akses informasi profil Kelurahan secara *efektif* dan memudahkan masyarakat dalam mencari informasi terkait.

Ngangi *et al.* [5] membuat Aplikasi untuk mengelola data secara online pada Bengkel Motorindo berbasis web, Bengkel Motorindo adalah sebuah bengkel penjualan suku cadang kendaraan bermotor, yang masih mengandalkan proses manual dalam penjualan, pembelian, dan pembukuan. Keterbatasan tersebut mencakup penggunaan kertas struk dan kalkulator, menyebabkan keterlambatan terutama dalam pembelian barang dalam jumlah besar. Dalam menanggulangi kendala gudang persediaan, bengkel hanya mengandalkan kwitansi dari supplier. Penelitian ini merinci implementasi sistem informasi penjualan berbasis *website* untuk mengatasi permasalahan tersebut. Metode pengembangan yang digunakan adalah *SDLC model waterfall*, dengan fokus pada analisis, desain, implementasi, pengujian, dan pemeliharaan. Hasil pengujian *black-box* menunjukkan tingkat keberhasilan sebesar 95.83%. Meskipun terdapat kekurangan seperti belum di-hosting secara *online* dan kebutuhan pengembangan pada *fitur chat online*, penelitian ini diharapkan dapat membantu Bengkel Motorindo meningkatkan efisiensi penjualan dan persediaan barang, serta mempermudah interaksi dengan pelanggan.

2. Tinjauan Pustaka

2.1. Publikasi dan Penelitian

Germano dalam [6] menguraikan bahwa publikasi ilmiah adalah suatu proses yang mendukung penelitian dengan menyampaikan hasil-hasil riset melalui evaluasi ketat oleh para pakar di bidangnya melalui sistem *peer review*. Tujuannya adalah untuk menjamin tingkat objektivitas yang tinggi dan memastikan kualitas yang dapat dipercaya dari karya yang dipublikasikan.

Publikasi ilmiah menurut Germano [6] merupakan suatu proses yang mendukung penelitian dengan mengantarkan hasil-hasil riset melalui evaluasi yang ketat oleh para pakar di bidangnya melalui sistem *peer review*. Hal ini bertujuan untuk menjamin tingkat objektivitas yang tinggi dan memastikan bahwa karya yang dipublikasikan memiliki kualitas yang dapat dipercaya. Mayoritas dari karya-karya akademis diterbitkan melalui jurnal ilmiah atau dalam bentuk buku, yang merupakan wadah utama untuk menyebarkan pengetahuan dan hasil riset kepada komunitas ilmiah dan masyarakat umum.

Penelitian adalah suatu kegiatan yang dilakukan untuk mengumpulkan informasi atau data yang sistematis dan terstruktur guna menjawab pertanyaan-

pertanyaan penelitian atau menguji hipotesis tertentu [6]. Proses penelitian melibatkan langkah-langkah yang terorganisir, mulai dari perumusan masalah, pengumpulan data, analisis data, hingga pembuatan kesimpulan.

2.2. Website

Website adalah ruang informasi digital yang diakses melalui internet, diperkenalkan oleh Sir Timothy John Berners-Lee pada tahun 1991 untuk memfasilitasi pertukaran informasi. Fungsinya sebagai media komunikasi global memungkinkan pengguna untuk mengakses, mencari, dan berbagi informasi. Jenis-jenis website meliputi statis yang kontennya tidak berubah secara manual, dinamis yang diperbarui berkala seperti berita atau blog, dan interaktif yang memfasilitasi interaksi antara pengguna dan situs seperti forum online atau aplikasi e-commerce [7].

2.3. Web Browser

Web browser adalah perangkat lunak yang menampilkan halaman web menggunakan HTML dan CSS. Fungsinya adalah untuk mengambil dan menafsirkan kode-kode tersebut dari *server web*, kemudian merender halaman sesuai instruksi. Tujuan utamanya adalah memungkinkan pengguna mengakses dan menavigasi halaman *web* dengan *efisien*, serta menyediakan fitur seperti *bookmark*, riwayat penelusuran, dan alat interaksi dengan konten web. Beberapa web browser populer antara lain Firefox, Google Chrome, dan Safari. Firefox dikembangkan oleh Mozilla *Foundation* dengan dukungan luas terhadap standar web dan add-on. Google Chrome dikenal karena kecepatan dan efisiensinya, sementara Safari merupakan pilihan populer di macOS karena integrasinya dengan sistem operasi dan performa cepat [7].

2.4. Hyper Text Markup Language (HTML)

Pallavi Yadav and Paras Nath Barwal dalam [7] menjelaskan HTML adalah bahasa *standar* yang digunakan untuk membuat dan *mendesain* halaman *web* sejak awal tahun 1990-an. Dengan menggunakan *tag-tag* seperti `<head>` dan `<body>`, pengembang dapat menciptakan halaman *web* dengan berbagai *elemen* seperti teks, gambar, video, dan *formulir*. Sejarah HTML dimulai dari konsep World Wide Web pada tahun 1989 oleh Tim Berners-Lee di CERN, dan perkembangannya semakin pesat setelah dirilisnya *browser web* Mosaic pada tahun 1993. HTML terus berkembang, dengan versi terbaru HTML5 diperkenalkan pada tahun 2014, mendukung teknologi web modern. Digunakan bersama dengan CSS dan JavaScript, HTML menciptakan pengalaman *web* yang *dinamis* dan menarik bagi pengguna. Sebagai pondasi dari web, HTML tetap menjadi bahasa markah utama dalam dunia *web* hingga saat ini.

2.5. Cascading Style Sheets (CSS)

CSS (*Cascading Style Sheets*) adalah *teknologi* yang penting dalam pengembangan *web* karena mengatur tampilan halaman web yang ditulis dalam HTML. Diperkenalkan oleh W3C pada pertengahan tahun 1990-an, CSS menjadi standar industri dalam desain situs web. CSS memungkinkan pengaturan properti seperti warna, *font*, dan tata letak *elemen* HTML, menciptakan desain web yang responsif dan estetis. Dengan konsep kaskade dan warisan, CSS memungkinkan pengaturan aturan gaya secara hierarkis untuk pengelolaan desain web yang kompleks. Bersama dengan HTML, CSS menciptakan halaman web fungsional dan menarik bagi pengguna di berbagai perangkat, dari desktop hingga perangkat *mobile*. Sebagai alat yang tak tergantikan bagi pengembang *web*, CSS memainkan peran penting dalam menciptakan pengalaman pengguna yang menyenangkan dan *responsif* [7].

2.6. SASS (Syntactically Awesome Style Sheets)

Sass adalah bahasa *scripting preprocessor* yang meningkatkan *fleksibilitas* dalam menulis kode gaya untuk halaman *web*. Dengan dua *sintaks*, yakni *sintaks* asli dan SCSS, Sass dapat diinterpretasikan menjadi CSS, mempercantik tampilan halaman web. Sass memberikan keunggulan dengan fitur seperti *variabel*, *nesting*, *mixins*, dan warisan *selector*, meningkatkan *efisiensi* dan keterbacaan kode gaya. Dengan Sass, pengembang dapat meningkatkan produktivitas dan memudahkan pemeliharaan kode gaya, membuatnya menjadi pilihan yang efisien dalam pengembangan web [7].

2.7. Bootstrap

Bootstrap adalah kerangka kerja sumber terbuka yang digunakan untuk membangun desain web responsif. Diciptakan oleh tim pengembang di Twitter pada tahun 2010, Bootstrap menyediakan alat dan komponen siap pakai untuk mempermudah pengembangan antarmuka pengguna yang profesional. Kemudahan penggunaan dan pembaruan berkala membuat Bootstrap menjadi pilihan utama dalam menciptakan desain web yang responsif dan berkualitas tinggi [8].

2.8. Javascript

JavaScript adalah bahasa *pemrograman* penting dalam pengembangan *web*, membuat halaman *web* menjadi interaktif dan *dinamis*. Dipilih untuk pembuatan aplikasi Manajemen Data Publikasi dan Penelitian pada FTI UNTAR karena kemampuannya yang kuat dalam membangun antarmuka pengguna yang *responsif* dan *integrasi* dengan berbagai teknologi lainnya [9].

2.9. jQuery

jQuery adalah *library* JavaScript yang populer untuk membuat halaman *web* lebih *interaktif*. Fitur-fiturnya mencakup *navigasi* HTML, *manipulasi* elemen DOM, animasi, dan pengembangan aplikasi Ajax. Dipilih karena keandalannya, ketersediaan sebagai proyek open *source*, dan penerimaan luas di komunitas pengembang web. Dengan kemampuannya yang luas dan dukungan kuat, jQuery tetap menjadi pilihan utama bagi pengembang *web* [9].

2.10. NPM

NPM (Node Package Manager) adalah perangkat lunak manajemen paket yang penting dalam pengembangan JavaScript, terutama dengan Node.js. Dikembangkan oleh NPM Inc., *platform* ini memungkinkan pengelolaan *dependensi* proyek, *instalasi*, pembaruan, dan penghapusan paket dengan mudah. Fitur utamanya termasuk manajemen versi paket dan pembuatan file *package-lock.json* untuk konsistensi pengembangan lintas lingkungan. Dengan npm, pengembang dapat menggunakan *command line interface* (CLI) untuk menjalankan perintah seperti *npm install* untuk *menginstal* paket. Ini menjadi pondasi penting dalam proses pengembangan perangkat lunak modern, menyederhanakan manajemen *dependensi* dan distribusi paket-paket JavaScript [9].

2.11. Node.js

Node.js adalah lingkungan runtime JavaScript yang memungkinkan eksekusi kode di sisi server. Dirilis pada 2009 oleh Ryan Dahl, Node.js didasarkan pada mesin JavaScript V8 dari Google. Keunggulan utamanya adalah model *event-driven* dan *non-blocking I/O*, cocok untuk aplikasi *real-time* dan *web* yang membutuhkan kinerja tinggi. Node.js populer karena ekosistem paket kaya melalui NPM (Node Package Manager), yang mempercepat pengembangan dengan ribuan paket siap pakai. Untuk menggunakannya, pengembang cukup menginstal Node.js dan menulis kode JavaScript untuk *server-side*. Dengan popularitas yang terus meningkat, Node.js telah menjadi platform *server-side* yang populer untuk membangun aplikasi *scalable* dan inovatif [9].

2.12. Express.js

Express.js adalah *framework* backend untuk membangun RESTful APIs dengan Node.js. Dengan kesederhanaannya, Express menyediakan alat yang kuat untuk pengembangan aplikasi *web* dan API. Penggunaannya luas dalam mengelola *endpoint*, *routing*, *middleware*, dan penanganan permintaan HTTP. Untuk menggunakan Express, instal melalui NPM dan impor modul Express ke dalam proyek Node.js, lalu tentukan *route*-*route* API dan logika bisnis yang diperlukan. Express memungkinkan pengembang untuk membuat API dengan cepat dan efisien, menjadi pilihan utama dalam

pengembangan aplikasi *web* dan API dengan Node.js [9].

2.13. Git

Git adalah sistem kontrol versi yang memungkinkan pengembang melacak perubahan dalam kode sumber secara efisien. Berasal dari pengembangan kernel Linux oleh Linus Torvalds pada 2005, Git menjadi standar de facto dalam pengelolaan kode sumber perangkat lunak. Keunggulan utamanya adalah sistem terdistribusi, memungkinkan setiap pengembang memiliki repositori lengkap di komputernya sendiri. Git memudahkan kolaborasi, pelacakan perubahan, pemulihan dari kesalahan, dan bereksperimen dengan fitur baru tanpa mengganggu kode yang sudah ada. Fitur seperti cabang dan penyatuan membantu pengembang mengelola dan mengintegrasikan perubahan dengan baik. Untuk menggunakan Git, pengembang perlu menginstalnya dan membuat repositori untuk proyek mereka. Dengan perintah-perintah seperti ``git add``, ``git commit``, dan ``git push``, serta penggunaan ``git pull`` dan ``git merge``, pengembang dapat mengelola dan menyinkronkan kode sumber proyek mereka dengan mudah [10].

2.14. PostgreSQL

PostgreSQL adalah sistem manajemen basis data relasional (RDBMS) open-source yang terkenal karena keandalannya, skalabilitasnya, dan fitur-fitur tingkat tinggi. Dengan dukungan kuat terhadap standar SQL, PostgreSQL memungkinkan pengembang untuk dengan mudah mengelola data dalam basis data relasional mereka. Kemampuan skalabilitasnya yang luar biasa, dengan dukungan untuk replikasi, partisi, dan indeks canggih, memungkinkan pengguna untuk mengelola basis data dengan ukuran besar dan menangani beban kerja yang tinggi. Cocok untuk berbagai jenis aplikasi, PostgreSQL mendukung berbagai jenis data dan fitur seperti dukungan JSON dan geospasial. Untuk menggunakan PostgreSQL, pengembang perlu menginstalnya di server mereka, membuat basis data yang diperlukan, dan menggunakan bahasa kueri SQL untuk berinteraksi dengan data. Dengan dokumentasi yang luas dan dukungan komunitas yang kuat, PostgreSQL menjadi pilihan yang solid untuk memenuhi kebutuhan pengembangan perangkat lunak [11].

3. Metode Penelitian

Aplikasi manajemen data publikasi dan penelitian di FTI Untar menggunakan metode SDLC Waterfall, yang menekankan pendekatan sistematis dari identifikasi kebutuhan hingga perawatan. Proses ini melibatkan pengumpulan data publikasi dan penelitian di FTI Untar serta wawancara dengan dosen dan staf. Tahapan Waterfall, seperti yang dijelaskan oleh Unhelkar dalam bukunya "Software Engineering with UML"[12], meliputi sebagai berikut:

1. *Requirements Analysis*

Langkah awal dalam metode *waterfall* adalah melakukan analisis kebutuhan. Pengembang perlu melakukan penelitian untuk mengenali kebutuhan pengguna terhadap sistem yang sedang dibangun. Informasi ini dapat digunakan sebagai panduan dalam menentukan layanan atau *fitur* yang perlu dikembangkan.

2. *Design*

Langkah kedua dalam metode *waterfall* adalah melakukan proses perancangan dan pengembangan berdasarkan informasi kebutuhan pengguna. Proses perancangan dilakukan untuk memudahkan pengerjaan dan mendapatkan gambaran rinci terkait tampilan sistem. Tahap desain dalam metode ini juga bertujuan untuk mengenali kebutuhan perangkat keras dan sistem yang diperlukan untuk keseluruhan proses pengembangan.

3. *Implementation*

Langkah ketiga dalam metode *waterfall* adalah implementasi, yang melibatkan proses coding. Pengembangan sistem dilakukan melalui tahap-tahap berupa modul-modul kecil yang nantinya akan digabungkan pada tahap selanjutnya dalam metode *waterfall*. Selain itu, pada fase ini, setiap modul yang telah dibuat diperiksa untuk memastikan bahwa modul tersebut memenuhi fungsi yang telah ditetapkan dan mematuhi standar yang telah ditentukan.

4. *Testing*

Langkah keempat berkaitan dengan integrasi setiap modul yang telah dibuat. Setelah menyelesaikan tahap ini, pengembang akan melakukan pengujian untuk memeriksa kinerja sistem secara menyeluruh. Selain itu, pengembang juga dapat mengidentifikasi potensi kegagalan atau kesalahan pada sistem.

5. *Deployment*

Langkah kelima SDLC adalah implementasi aplikasi ke *server* untuk digunakan. Selanjutnya *install* aplikasi dengan penyalinan file dan *konfigurasi* yang diperlukan. Setelah selesai, aplikasi siap digunakan di lingkungan produksi, memastikan *fungsionalitas* dan performa yang optimal sebelum digunakan secara luas.

6. *Maintenance*

Setelah melewati serangkaian langkah sistematis yang telah disebutkan diatas, tahap terakhir dalam metode ini adalah pemeliharaan sistem yang telah dibuat. Sistem tersebut telah didistribusikan dan aktif digunakan oleh pengguna. Tugas yang masih perlu dilakukan melibatkan pemeliharaan dan memastikan agar sistem tetap beroperasi dengan baik sesuai dengan tujuan fungsinya.

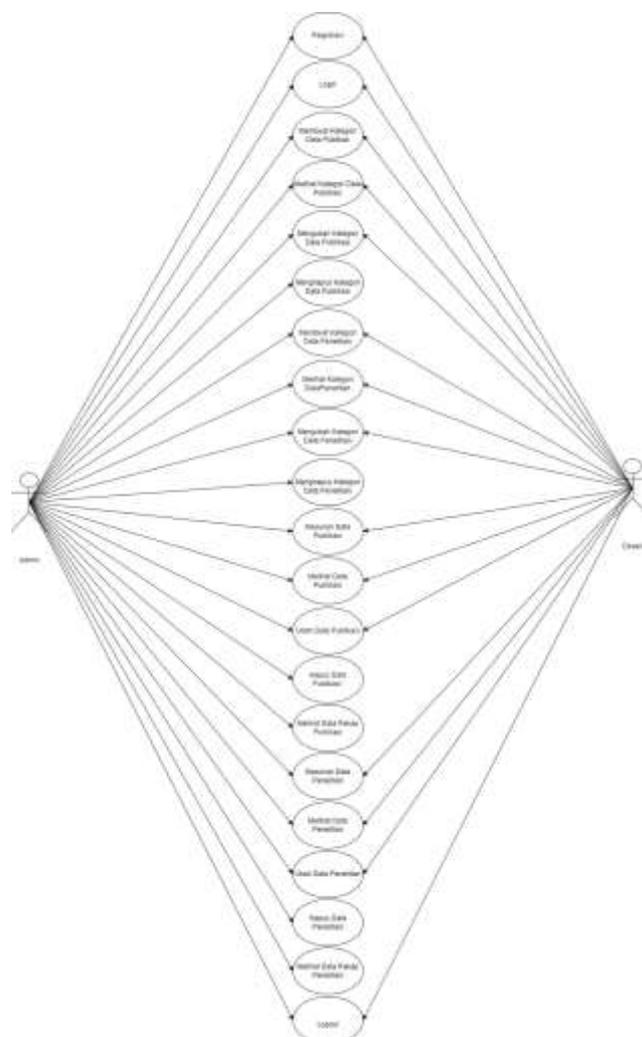
4. Hasil Perancangan

Pada Rancangan Aplikasi Manajemen Data Publikasi dan Penelitian pada FTI Untar terdapat tampilan fitur masing masing untuk *admin* dan Dosen.

Untuk menggambarkan proses rancangan Aplikasi Manajemen Data Publikasi dan Penelitian adalah dengan menggunakan *use case*, *Use Case Scenario*, *activity diagram*, *sequence diagram*, *class diagram*, dan *ERD*.

4.1. Use Case Diagram

Use case diagram adalah *representasi* visual tentang kebutuhan sistem dalam ruang masalah. Ini menggambarkan *interaksi* antara *aktor* dan sistem [13]. Dalam contoh ini, Admin dan Dosen berinteraksi dengan sistem manajemen data publikasi dan penelitian. Admin dapat login, mengelola kategori, dan data publikasi. Dosen memiliki akses serupa tanpa kemampuan menghapus data. Setelah selesai, keduanya dapat *logout*, Untuk *use case diagram* yang telah dibuat dapat dilihat pada **Gambar 1**.



Gambar 1. Use Case Diagram

4.2. Use Case Scenario

Use case scenario adalah *representasi* dari bagaimana pengguna berinteraksi dengan sistem untuk mencapai tujuan tertentu [13]. Contoh skenario ini meliputi proses penambahan, pengeditan, dan

penghapusan kategori, serta data publikasi dan penelitian oleh admin dan dosen. Pengguna memasuki informasi yang diperlukan, sistem melakukan validasi, dan menyimpan data ke dalam database. Setelah itu, pengguna dapat melihat perubahan yang telah dilakukan. Skenario juga mencakup proses melihat ringkasan data publikasi dan penelitian. Semua proses ini membantu pengembang memahami bagaimana pengguna berinteraksi dengan sistem dalam manajemen data publikasi dan penelitian.

4.3. Activity Diagram

Activity diagram merupakan pemodelan aliran, atau proses, dalam suatu sistem yang dapat digunakan untuk mendokumentasikan perilaku internal dalam use case, antar use case, atau bisnis secara keseluruhan. Diagram ini dapat mencakup berbagai tingkat detail, baik pada level teknis terperinci maupun pada level bisnis. Misalnya, untuk proses registrasi admin, diagram menunjukkan langkah-langkah mulai dari inisiasi halaman registrasi hingga pembuatan akun baru dan penyimpanan kredensial yang valid di database [13]. Proses-proses lain seperti penambahan, pengeditan, dan penghapusan kategori atau data publikasi dan penelitian juga terdokumentasi dengan detail yang serupa. Keseluruhan diagram memberikan gambaran menyeluruh tentang alur kerja sistem dalam berbagai situasi dan peran pengguna yang berbeda, baik sebagai admin maupun dosen.

4.4. Sequence Diagram

Sequence diagram merupakan representasi visual dari skenario dalam sebuah use case. Misalnya, dalam skenario pendaftaran, pengguna mengisi formulir pendaftaran dengan informasi yang diperlukan, kemudian aplikasi memvalidasi dan menyimpan informasi tersebut. Untuk skenario login, pengguna memasukkan kredensial mereka, aplikasi memvalidasi, dan memberikan akses ke dashboard jika valid [13]. Diagram juga mencakup skenario penelusuran kategori dan data publikasi serta penelitian, baik untuk pengguna umum maupun dosen, dengan langkah-langkah yang serupa. Semua skenario tersebut direpresentasikan secara visual dalam diagram urutan.

4.5. Class Diagram

Class diagram adalah diagram struktur dalam UML yang menggambarkan struktur dan hubungan antar class, atribut, dan metode dalam sebuah sistem. Diagram ini statis dan berguna untuk memvisualisasikan struktur sistem dan memahami skema program secara umum. Desain modelnya terbagi menjadi penjabaran database dan modul MVC, termasuk class *interface*, *control*, dan *entity*. Class diagram membantu dalam analisis bisnis dan menyajikan gambaran tentang sistem atau perangkat lunak beserta relasinya [13]. Keunggulannya meliputi kemampuan dalam menjelaskan model data,

memberikan gambaran yang jelas tentang aplikasi, dan mendukung penyampaian kebutuhan sistem. Untuk *Class Diagram* untuk aplikasi manajemen data publikasi dan penelitian dapat dilihat pada **Gambar 2**.



Gambar 2. Class Diagram

4.6. ERD

Entity Relationship Diagram (ERD) merupakan alat penting dalam perancangan aplikasi manajemen data publikasi dan penelitian di FTI UNTAR. ERD memiliki tiga fungsi utama: memodelkan hasil analisis data, memodelkan data secara konseptual, dan memodelkan objek-objek dalam sistem. Ini membantu pengembang memahami kebutuhan sistem, memetakan konsep data secara abstrak, dan memahami interaksi antar objek [13]. ERD yang telah dibuat dapat dilihat pada **Gambar 3** telah menjadi dasar dalam perancangan dan pengembangan aplikasi yang efisien dan sesuai dengan kebutuhan FTI UNTAR.

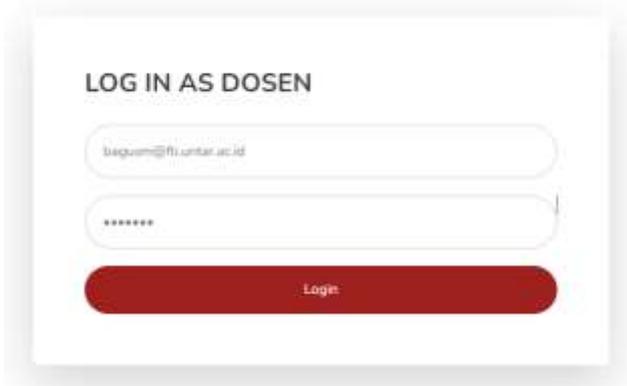


Gambar 3. ERD

4.7. Aplikasi

Aplikasi ini memiliki fitur login yang memungkinkan dosen dan admin untuk mengaksesnya dengan menggunakan email dan password masing-masing, untuk menjaga keamanan data dan informasi penting. Juga terdapat fitur untuk menolak akses jika password yang dimasukkan salah, baik untuk dosen maupun admin. Selain itu, tersedia dashboard yang menampilkan statistik penelitian, kemampuan untuk mengubah password sendiri, filter data penelitian

berdasarkan tahun, dan grafik rekapitulasi penelitian dan biaya penelitian. Dosen dan admin dapat melihat detail penelitian, menambahkan data penelitian melalui Excel, serta melakukan pencarian data penelitian yang lebih canggih berdasarkan kriteria tertentu. Pengelolaan kategori penelitian dan subkategori juga dapat dilakukan, bersama dengan manajemen data mahasiswa yang terlibat dalam penelitian. Semua fitur ini bertujuan untuk meningkatkan efisiensi dan keamanan dalam manajemen dan pelaksanaan penelitian di lingkungan akademik.



Gambar 4. Tampilan Login



Gambar 5. Tampilan Menu Dashboard



Gambar 6. Tampilan Menu Kategori

4.8. Hasil User Acceptance Test (UAT)

Hasil pengujian oleh Dosen dan Admin Fakultas Teknologi Informasi Universitas Tarumanagara Jakarta menunjukkan kesuksesan aplikasi dengan seluruh skenario uji berhasil dilalui, membuktikan bahwa

kualitas dan konsistensi fungsionalitasnya. Antarmuka pengguna yang intuitif dan mudah dipahami menjadi fokus utama dalam proses pengujian, mendapat *feedback* positif yang memfasilitasi interaksi yang lancar. Keunggulan utama aplikasi ini adalah kemudahan pengguna, memungkinkan pengguna dari berbagai latar belakang dengan cepat dan mudah untuk mempelajari cara penggunaan. Fitur-fitur aplikasi ini, tidak hanya itu saja aplikasi ini juga berjalan sesuai kebutuhan pengguna, menggambarkan komitmen pengembang untuk menyediakan solusi yang sesuai dengan kebutuhan lingkungan akademis di Fakultas Teknologi Informasi Universitas Tarumanagara Jakarta. Bapak Novario Jaya Perdana, S.Kom., M.T. Sekretaris Program Studi Sarjana Sistem Informasi di FTI Untar, memberikan apresiasi terhadap aplikasi yang telah menyesuaikan dengan kebutuhan di FTI Untar, serta menekankan potensi aplikasi ini sebagai aplikasi yang sangat berguna dalam mencari dan merakap data terkait publikasi dan penelitian dengan tingkat efisiensi dan akurasi yang tinggi. Dukungan aplikasi ini diharapkan dapat membawa dampak positif bagi pengembangan akademik di FTI Untar dengan mempermudah proses pengumpulan dan pengelolaan data. Bapak Ruwanto selaku Kasubag Umum dan Perlengkapan di FTI Untar, setelah melakukan UAT (*User Acceptance Testing*), menyatakan bahwa fitur-fitur utama berfungsi dengan baik namun menyarankan penambahan pencarian dosen selain NIDN dan kemampuan dashboard untuk memfilter kategori. Penilaian pengguna menunjukkan tingkat kepuasan yang tinggi dengan skor rata-rata 8 dari 10, menilai aspek seperti kemudahan penggunaan antarmuka, kelengkapan fitur, responsivitas sistem, dan keamanan, sehingga secara keseluruhan aplikasi ini memenuhi kebutuhan dan diharapkan dapat meningkatkan efisiensi pengelolaan data penelitian dan publikasi di FTI Untar.



Gambar 7. Bukti UAT dengan Bapak Novario



Gambar 8. Hasil UAT dengan Bapak Novario



Gambar 9. Bukti UAT dengan Bapak Ruwanto



Gambar 10. Hasil UAT dengan Bapak Ruwanto

4. Kesimpulan dan Saran

Hasil evaluasi dan analisis pengujian program menunjukkan keberhasilannya dalam memenuhi sebagian besar spesifikasi dan fitur yang diinginkan, serta responsif dalam kinerja dan keamanan data yang terjamin. Program ini menonjol dalam antarmuka pengguna yang ramah, fitur komprehensif, kinerja handal, dan keamanan data yang kuat. Dengan demikian, program ini efektif dalam mendukung manajemen data akademik, meningkatkan efisiensi dan kualitas proses akademik dan penelitian.

Program dapat ditingkatkan melalui pengembangan fitur analisis data yang lebih canggih, optimalisasi

kinerja, dan penambahan fitur kolaborasi untuk meningkatkan produktivitas akademik. Selain itu, peningkatan aksesibilitas dan integrasi dengan sistem eksternal akan meningkatkan efisiensi pertukaran data, sementara peningkatan keamanan dan penyediaan dukungan pengguna yang lebih baik akan meningkatkan kepercayaan dan kenyamanan pengguna. Dengan pengembangan dalam arah ini, program akan lebih mampu mengatasi tantangan yang terus berkembang dalam lingkungan akademik.

REFERENSI

- [1] J. Wang, Tony, and Wasino, "Sistem Informasi Pelayanan Dan Koordinasi Kegiatan Ibadah Berbasis Web," *J. Ilmu Komputer. dan Sist. Inf.*, vol. 11, no. 1, 2023.
- [2] A. Setiawan Susanto, E. Dewayani, and T. Sutrisno, "Sistem Informasi Akademik Siswa Berbasis Website Pada Sman 1 Ngabang," *J. Ilmu Komputer. dan Sist. Inf.*, vol. 10, no. 2, 2022.
- [3] I. Lorenzo and Tony, "Perancangan Sistem Informasi Penyewaan Jasa Dekorasi Berbasis Web Pada Gianluca Project, 2023.
- [4] Nirsal and Riska Padillah Ansar, "Rancang Bangun Sistem Informasi Berbasis Website Profil Kelurahan Benteng," *J. Artif. J. Inform. dan Sist. Inf.*, vol. 1, no. 1, pp. 1–7, 2023.
- [5] E. Alfonsius *et al.*, "Sistem Informasi Penjualan Sparepart Motor Berbasis Website (Studi Kasus Pada Bengkel Motorindo)," *J. Inf. Technol. Softw. Eng. Comput. Sci.*, vol. 1, no. 2, pp. 75–83, 2023.
- [6] W. Germano, *Getting It Published*, no. 5. 2021.
- [7] Pallavi Yadav and Paras Nath Barwal, "Designing Responsive Websites Using HTML And CSS | Pallavi Yadav | 11 Citations, 2024.
- [8] bin S. Uzayr, *Mastering Bootstrap: A Beginner's Guide*. CRC Press, 2022. [Online]. Available: <https://zeba.academy>
- [9] S. bin Uzayr, *Mastering jQuery*. 2023.
- [10] N. Biswas, *MERN Projects for Beginners*. 2021.
- [11] Baji Shaik and D. K. Chemuduru, *Procedural Programming with PostgreSQL PL*.
- [12] C. S. de Almeida *et al.*, "Software Engineering With UML," *Rev. Bras. Linguística Apl.*, vol. 5, no. 1.
- [13] E. Sugawara and H. Nikaido, "Properties of AdeABC and AdeJJK efflux systems of *Acinetobacter baumannii* compared with those of the AcrAB-TolC system of *Escherichia coli*.

Eric Anthony, Mahasiswa Program Studi Sistem Informasi Fakultas Teknologi Informasi Universitas Tarumanagara sejak 2020.

Tony, memperoleh gelar S.Kom. pada tahun 2005 dari Universitas Tarumanagara, M.Kom. pada tahun 2010 dari Universitas Indonesia, dan Ph.D. pada tahun 2021 dari Curtin University. Saat ini sebagai staf pengajar di Fakultas Teknologi Informasi Universitas Tarumanagara.