

OAJIS

Open Access
Journal of
Information
Systems

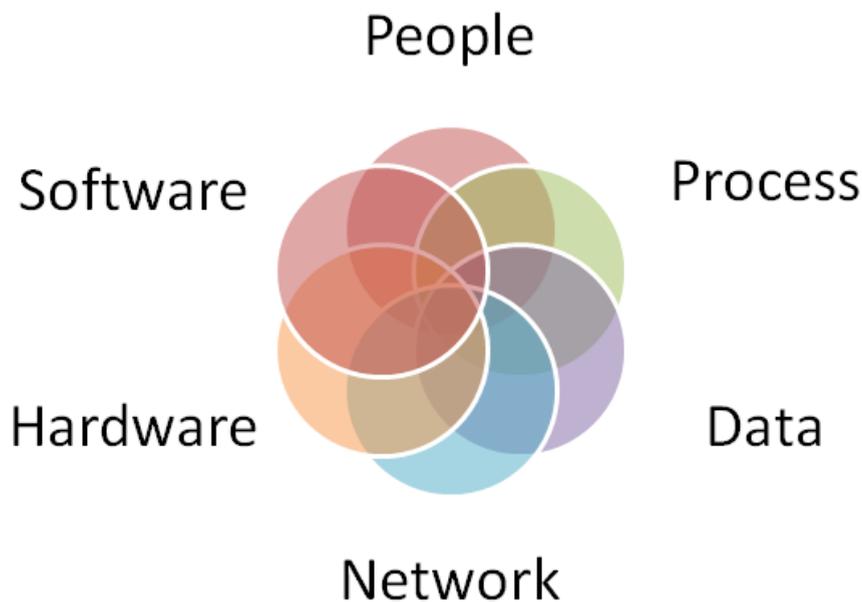
is.its.ac.id/pubs/oajis/

ISSN 1979-3979



jurnal sisfo

Inspirasi Profesional Sistem Informasi





Pimpinan Redaksi

Sholiq (Institut Teknologi Sepuluh Nopember)

Dewan Redaksi

Reny Nadlifatin (Institut Teknologi Sepuluh Nopember)

Mudjahidin (Institut Teknologi Sepuluh Nopember)

Tining Haryanti (Universitas Muhammadiyah Surabaya)

Faizal Mahananto (Institut Teknologi Sepuluh Nopember)

Rizal Risnanda Utama (Institut Teknologi Sepuluh Nopember)

Radityo Prasetyanto Wibowo (Institut Teknologi Sepuluh Nopember)

Monica Wideasri (Universitas Surabaya)

Anjik Sukmaaji (Universitas Dinamika)

Devi Septiani (Universitas Brawijaya)

Tata Pelaksana Usaha

Heppy Nuryanti (Institut Teknologi Sepuluh Nopember)

Sekretariat

Departemen Sistem Informasi – Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember (ITS) – Surabaya

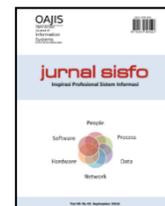
Telp. 031-5999944 Fax. 031-5964965

Email: editor@jurnalsisfo.org

Website: <http://jurnalsisfo.org>

Jurnal SISFO juga dipublikasikan di *Open Access Journal of Information Systems* (OAJIS)

Website: <http://is.its.ac.id/pubs/oajis/index.php>



Mitra Bestari

Prof. Erma Suryani, S.T., M.T., Ph.D. (Institut Teknologi Sepuluh Nopember)

Prof. Dr. Wiwik Anggraeni, S.Si., M.Kom. (Institut Teknologi Sepuluh Nopember)

Tony Dwi Susanto, S.T., M.T., Ph.D. (Institut Teknologi Sepuluh Nopember)

Yogantara Setya Dharmawan, S.Kom, M.BusProcessMgt. (Institut Teknologi Sepuluh Nopember)

Arif Wibisono, S.Kom., M.Sc., Ph.D. (Institut Teknologi Sepuluh Nopember)

Dr. Bambang Setiawan, S.Kom., M.T. (Institut Teknologi Sepuluh Nopember)

Dr. Muhammad Ainul Yaqin, S.T., M.Kom. (Universitas Islam Negeri Maulana Malik Ibrahim)

Taufik, S.T., M.Kom. (Universitas Airlangga)

Dr. Apol Pribadi Subriadi, S.T., M.T. (Institut Teknologi Sepuluh Nopember)

Muhammad Amirul Haq, S.T., M.Sc. (Universitas Muhammadiyah Surabaya)

Dhiani Tresna Absari, S.T., M.Kom. (Universitas Surabaya)

Dr. Mudjahidin, S.T., M.T. (Institut Teknologi Sepuluh Nopember)



Daftar Isi

Visualisasi Representasi Pengetahuan berbasis Ontologi untuk Memodelkan Mahasiswa Cumlaude Jenjang Sarjana <i>Nur Khofifah, Nur Laila, Sholikah Desi Purwanti, Amalinda Jayanty, Retno Aulia Vinarti</i>	1
Implementasi Metode Agile pada Pengembangan Sistem Informasi Manajemen Masjid Berbasis Website <i>Ridho Aulia Rahman, Rigen Ferdian Saputra, M. Ainul Yaqin</i>	11
Representasi Pengetahuan Berbasis Ontologi sebagai Panduan Berwisata dengan Aman <i>Muhammad Rhakan, Naufal Firjatullah Fano, Sang Intan Risqi Adi, Ziaul Haq Al Karimi, Retno Aulia Vinarti</i>	26
Impact of Digital Supply Chain in Agriculture: A Systematic Literature Review <i>Muhammad Syamil Fadlillah, Rahmatika Jagad Pramundito</i>	36
Penyusunan Dokumen SOP Sistem Manajemen Keamanan Aset Informasi Dinas Pariwisata Kebudayaan Pemuda dan Olahraga Kab. Sumenep Menggunakan Framework COBIT 5 dan ISO 27001:2013 <i>Yogantara Setya Dharmawan, Rizqi Amrullah Wildan Yani, Alif Millati Putri</i>	53
Penerapan Metodologi Agile Scrum dalam Pengembangan Situs Web AutomATEEs untuk Pembuatan Desain Kaos Berbasis AI <i>Darrell Valentino, Frans Nicklaus Gusyanto, Jhoni Ananta Sitepu, Dzaky Purnomo Rifa'i, Viera Tito Virgawan, Sholiq</i>	62
Analisis Penerapan Prinsip SOLID pada Tugas Proyek Mahasiswa UIN Malang <i>Alfred Rajendra Wijaya, Ela Ilmatul Hidayah, M. Ainul Yaqin</i>	85
Analisis Pengaruh Fitur Iklan Pada Media Sosial Terhadap Intensi Pembelian Pelanggan dengan Menggunakan Structural Equation Modelling (Studi Kasus: Instagram) <i>Andre Parvian Aristio, Mudjahidin, Made Puspa Wedanthi</i>	95

OAJIS

Open Access
Journal of
Information
Systems
is.its.ac.id/pubs/oajis/

jurnal sisfo

Jurnal Sisfo Vol. 11 No. 2 (2024)



Halaman ini sengaja dikosongkan



Implementasi Metode Agile pada Pengembangan Sistem Informasi Manajemen Masjid Berbasis Website

Ridho Aulia Rahman^{*}, Rigen Ferdian Saputra, M. Ainul Yaqin

Teknik Informatika, Sains dan Teknologi, UIN Maulana Malik Ibrahim Malang

Abstract

Mosques have an important role as places of worship and centers of religious and social activities for Muslims. However, many mosques still manage information manually, which often leads to access difficulties, lack of transparency and inefficiency. To overcome this problem, this research proposes the development of a web-based Mosque Management Information System (MISS) using the Agile method. Agile was chosen because of its flexible and structured approach, allowing for continuous feedback and adaptation to changing needs. The results of the development of the Mosque MIS include main features such as activity information, prayer schedules, Qibla direction, financial data, donation information and congregation data. This research shows that the use of a Mosque MIS can increase efficiency and transparency in mosque management. In conclusion, the Agile method has proven effective in developing Mosque MIS, producing a system that is adaptive and meets user expectations, as well as overcoming shortcomings in previous research by providing more complete features and better accessibility.

Keywords: Mosque Management Information System, Web-Based, Agile

Abstrak

Masjid memiliki peran penting sebagai tempat ibadah dan pusat kegiatan keagamaan serta sosial bagi umat Islam. Namun, banyak masjid masih mengelola informasi secara manual, yang sering menyebabkan kesulitan akses, kurangnya transparansi, dan ketidakefisienan. Untuk mengatasi masalah ini, penelitian ini mengusulkan pengembangan Sistem Informasi Manajemen (SIM) Masjid berbasis web dengan menggunakan metode *Agile*. *Agile* dipilih karena pendekatannya yang fleksibel dan terstruktur, memungkinkan umpan balik berkelanjutan dan adaptasi terhadap kebutuhan yang berubah-ubah. Hasil pengembangan SIM Masjid mencakup fitur-fitur utama seperti informasi kegiatan, jadwal sholat, arah kiblat, data keuangan, informasi donasi, dan data jamaah. Penelitian ini menunjukkan bahwa penggunaan SIM Masjid dapat meningkatkan efisiensi dan transparansi dalam pengelolaan masjid. Kesimpulannya, metode *Agile* terbukti efektif dalam pengembangan SIM Masjid, menghasilkan sistem yang adaptif dan memenuhi harapan pengguna, serta mengatasi kekurangan yang ada pada penelitian sebelumnya dengan menyediakan fitur yang lebih lengkap dan aksesibilitas yang lebih baik.

Kata kunci: Sistem Informasi Manajemen Masjid, Web-Based, Agile

© 2024 Jurnal SISFO.

^{*}Corresponding Author

Email address: ridho.aulia133@gmail.com (Ridho Aulia Rahman)

<https://doi.org/10.24089/j.sisfo.2024.06.002>

1. Pendahuluan

Masjid tidak hanya berperan sebagai tempat ibadah[1], tetapi juga sebagai pusat kegiatan keagamaan dan sosial bagi umat Islam. Berbagai kegiatan rutin dan insidental diselenggarakan di masjid, seperti donasi, pembinaan, pendidikan, dan perayaan hari besar. Di era modern ini, kebutuhan akan informasi yang cepat dan akurat menjadi semakin penting, termasuk dalam pengelolaan masjid. Namun, banyak masjid masih menggunakan sistem manual dalam mengelola informasi, seperti pencatatan keuangan dan pendaftaran kegiatan. Hal ini dapat menyebabkan beberapa permasalahan nyata, seperti kesulitan dalam mengakses informasi terkini mengenai kegiatan dan pengelolaan masjid, kurangnya transparansi dalam pengelolaan data keuangan dan donasi, serta proses yang tidak efisien karena pencatatan manual memakan waktu dan rentan terhadap kesalahan manusia [2].

Fakta menunjukkan bahwa penggunaan sistem informasi yang terkomputerisasi dapat meningkatkan efisiensi dan transparansi dalam pengelolaan masjid [3]. Sebagai contoh, penelitian oleh [15] menunjukkan bahwa penggunaan Sistem Informasi Manajemen (SIM) Masjid dapat memudahkan masyarakat mengetahui informasi manajemen masjid dan membantu pengurus dalam membuat rekapitulasi data. Fokus penelitian ini adalah pada implementasi metode *Agile* dalam pengembangan SIM Masjid berbasis web. *Agile* dipilih karena pendekatan ini menawarkan pengembangan yang terstruktur dan bertahap, meminimalkan kesalahan, dan memastikan kelancaran pengembangan dengan umpan balik berkelanjutan.

Penelitian sebelumnya tentang SIM Masjid berbasis web [3] telah memberikan kontribusi dalam hal penyampaian informasi kegiatan, pengelolaan data terkomputerisasi, dan kemudahan akses informasi bagi masyarakat. Namun, berdasarkan tinjauan terhadap empat [lihat Tabel 1] penelitian terdahulu, terdapat beberapa celah yang dapat dijumpai oleh penelitian kami. Pertama, penelitian terdahulu umumnya berfokus pada fitur tertentu seperti manajemen kegiatan atau pengelolaan data keuangan masjid, sementara penelitian kami menawarkan fitur yang lebih lengkap, meliputi informasi kegiatan, jadwal shalat, arah kiblat, data keuangan detail, informasi donasi, dan data jamaah. Kedua, sistem yang dikembangkan pada penelitian terdahulu umumnya berbasis *desktop*, sehingga kurang fleksibel dalam hal akses. Penelitian kami menawarkan sistem berbasis web yang dapat diakses dari mana saja menggunakan perangkat apa saja dengan *browser* internet. Ketiga, metode pengembangan yang digunakan pada penelitian terdahulu beragam, seperti *Rapid Application Development* (RAD) yang cenderung kurang terstruktur. Penelitian kami menggunakan metode *Agile* [4] yang menawarkan pendekatan terstruktur dan bertahap untuk meminimalkan kesalahan dan memastikan kelancaran pengembangan SIM Masjid.

1. Tinjauan Pustaka

Pada tinjauan pustaka menyajikan pemahaman yang komprehensif tentang kajian terdahulu yang telah dilakukan dalam domain tertentu, menyoroti kerentanan atau celah dalam penelitian yang ada, serta memberikan landasan teoritis bagi penelitian yang sedang dilakukan. Sebagaimana yang tertera dalam Tabel 1. Meskipun penelitian sebelumnya telah berhasil mengembangkan SIM Masjid yang membantu pengelolaan agenda, manajemen data, dan operasional takmir, terdapat kekosongan dalam penerapan metode *Agile* yang lebih adaptif dan iteratif dalam proses pengembangannya. Penelitian-penelitian sebelumnya cenderung menggunakan metode pengembangan sistem yang tradisional seperti *Rapid Application Development* (RAD) dan pendekatan deskriptif, yang mungkin kurang fleksibel dalam menghadapi perubahan kebutuhan pengurus masjid dan jamaah. Oleh karena itu, penelitian ini fokus pada bagaimana metode *Agile* dapat diimplementasikan dalam pengembangan SIM Masjid untuk mengatasi keterbatasan yang ada, serta untuk menciptakan sistem yang lebih responsif dan efisien dalam mendukung operasional masjid secara berkelanjutan.

Tabel 1. Penelitian Terdahulu

Sumber	Input	Output	Metode	Hasil
[12]	Data agenda syiar Islam	SIM Masjid dengan fitur informasi agenda	Analisis Deskriptif	Mempercepat kerja pengurus masjid dalam menyampaikan informasi agenda syiar Islam
[15]	Data manajemen masjid	SIM Masjid dengan rekapitulasi data	Pengembangan Sistem	Memudahkan masyarakat mengetahui informasi manajemen masjid dan memudahkan pengurus
[14]	Acuan pengembangan sistem	Panduan pengembangan SIM Masjid	Studi Literatur	Acuan untuk pembuatan atau pengembangan SIM Masjid
[1]	Data kegiatan masjid	SIM Masjid untuk manajer operasional takmir	Rapid Application Development (RAD)	Membantu manajer operasional takmir mengelola seluruh kegiatan masjid dan memberi informasi kepada jama'ah

2.1 Manajemen Masjid

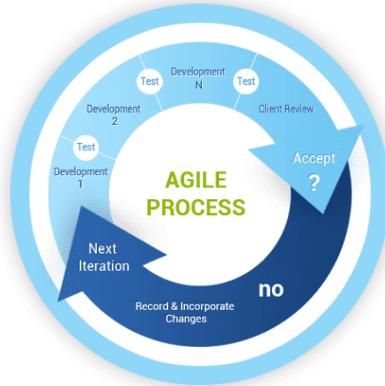
Manajemen masjid adalah aspek penting untuk memastikan fungsi dan keberlanjutan operasional masjid sebagai pusat ibadah dan kegiatan sosial. Prosesnya melibatkan perencanaan, pengorganisasian, pelaksanaan, dan pengawasan. Perencanaan mencakup penentuan visi dan misi, rencana program dan kegiatan, serta anggaran. Pengorganisasian melibatkan pembentukan struktur organisasi, rekrutmen, dan pelatihan[5]. Pelaksanaan fokus pada kepemimpinan yang efektif, komunikasi yang baik, dan implementasi program. Pengawasan bertujuan memastikan segala sesuatu berjalan sesuai rencana melalui evaluasi kinerja, audit keuangan, dan tindak lanjut. Dengan proses ini, masjid dapat berfungsi optimal sebagai pusat ibadah dan komunitas yang berkontribusi positif bagi masyarakat.

2.2 Metode Agile

Menggunakan kerangka kerja *scrum* dalam pengembangan[4], adapun tahap-tahap dari kerangka *scrum* (lihat Gambar 1) yang merupakan bagian dari metode *Agile* sebagai berikut :

- 1) *Product Backlog*: Merupakan daftar hal-hal yang ingin dikerjakan atau disebut juga *wishlist*.
- 2) *Sprint Backlog*: Tim menentukan mana yang ingin dikerjakan, membagi tugas, dan merencanakan cara mencapai tujuan sehingga menghasilkan *output* yang diinginkan.
- 3) *Sprint Planning*: Pada tahap ini, tim mulai mengerjakan *sprint backlog*. *Sprint* berlangsung selama satu bulan atau kurang dan melibatkan pertemuan harian (*daily scrum meeting*).
- 4) *Daily Scrum Meeting*: Pertemuan ini berlangsung selama 15 menit, dihadiri oleh tim pengembang untuk membahas progres, hambatan, dan hal-hal terkait dengan tujuan. *Scrum master* hadir untuk memastikan tim tetap fokus. *Output* dari *sprint* adalah produk yang siap dirilis.
- 5) *Sprint Review Meeting*: Pertemuan ini melakukan *review* terhadap produk yang siap dirilis. Pada tahap ini, tim *scrum* bertemu dengan pemangku kepentingan utama dan mendapatkan umpan balik berupa hal-hal yang disukai, tidak disukai, dan komentar lainnya.

- 6) *Sprint Retrospective*: Melihat kembali proses sebelumnya dan mengevaluasi apa yang bisa diperbaiki untuk *sprint* selanjutnya. Setelah *retrospective*, tim kembali ke *product backlog* dan memutuskan *wishlist* mana yang akan dikerjakan berikutnya.



Gambar 1. Tahap-Tahap Metode Agile

2.3 Blackbox Testing

Pada pengujian *black-box*, sistem perangkat lunak diperlakukan seperti “kotak hitam” di mana penguji tidak mengetahui detail internalnya. Alih-alih berfokus pada cara kerja sistem, penguji memvalidasi fungsionalitasnya dengan memberikan *input* dan mengamati *output* yang dihasilkan. Hal ini bertujuan untuk memastikan bahwa sistem berperilaku sesuai dengan spesifikasi yang telah ditentukan, tanpa perlu memahami detail implementasinya [8]

3. Metodologi

3.1 Data Penelitian

Penelitian ini mengumpulkan data melalui berbagai metode, termasuk wawancara dan diskusi dengan pengurus masjid dan jamaah, observasi langsung terhadap aktivitas masjid, serta analisis dokumentasi dan arsip masjid. Selain itu, prototipe sistem diuji oleh pengguna awal untuk mendapatkan umpan balik yang digunakan dalam proses penyempurnaan [6]. Tinjauan literatur dan studi kasus sebelumnya juga dilakukan untuk memberikan landasan teoritis dan panduan praktik terbaik dalam penerapan metodologi Agile.

3.2 Agile Development

Metode *Agile* [8] dipilih dalam penelitian ini karena fleksibilitas, keterlibatan pemangku kepentingan, pengembangan iteratif, dan fokus pada pengiriman nilai. Penelitian ini akan menggunakan desain PAT dan melibatkan beberapa tahapan, yaitu perencanaan, analisis, desain, pengembangan, pengujian, penerapan, dan evaluasi. Instrumen penelitian yang akan digunakan adalah wawancara, kuesioner, observasi, dan analisis log.[9] Data yang dikumpulkan akan dianalisis dengan menggunakan metode kualitatif dan kuantitatif. Penelitian ini akan dilakukan dengan mengikuti prinsip-prinsip etika penelitian

3.2.1 Analisis Kebutuhan

Melalui kolaborasi intensif dengan para ahli domain dan pemangku kepentingan bisnis pada tahap sebelumnya dengan metode PIECES [7], pemahaman menyeluruh tentang domain bisnis telah terbangun. Pemahaman ini menjadi landasan yang kokoh untuk implementasinya ke dalam perangkat lunak. Tahap ini akan menjelaskan secara rinci hasil diskusi tersebut. Tujuannya adalah untuk mengidentifikasi elemen-elemen penting yang diperoleh. Elemen-elemen inilah yang nantinya akan menjadi acuan dalam proses penulisan kode program[10]. Dari proses kolaborasi tersebut, didapatkan data-data penting, sebagai berikut:

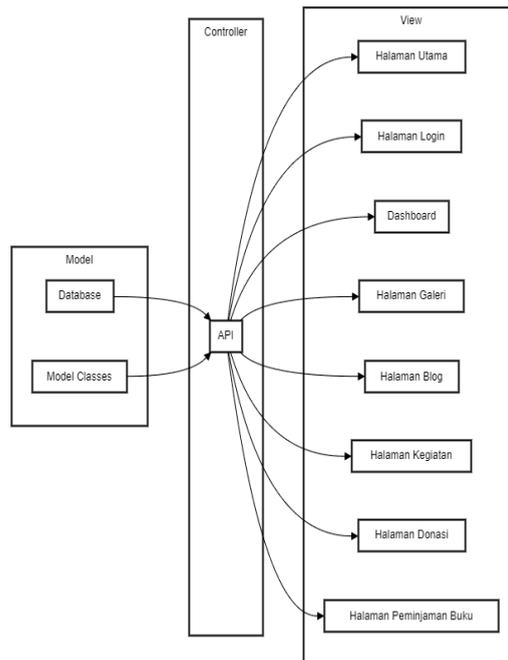
- 1) Halaman *dashboard* untuk *user/jamaah* yang dibuka pertama kali oleh *user/jamaah* ketika *login*
- 2) Halaman *AdminPanel* hanya untuk admin/pengurus yang *login*.
- 3) *jamaah* bisa berdonasi jika disetujui admin untuk mendapatkan *role* sebagai donatur
- 4) pengurus dapat membuat artikel dakwah pada fitur sistem informasi manajemen masjid
- 5) pengurus dengan *role* bendahara yang hanya bisa mengakses admin panel keuangan

3.2.2 Design

Pada bagian *design*, akan dilakukan rancangan sebuah sistem menggunakan *Architecture Diagram*, *Use Case Diagram* dan *Entity Relationship Diagram* [11].

3.2.2.1 Architecture Diagram

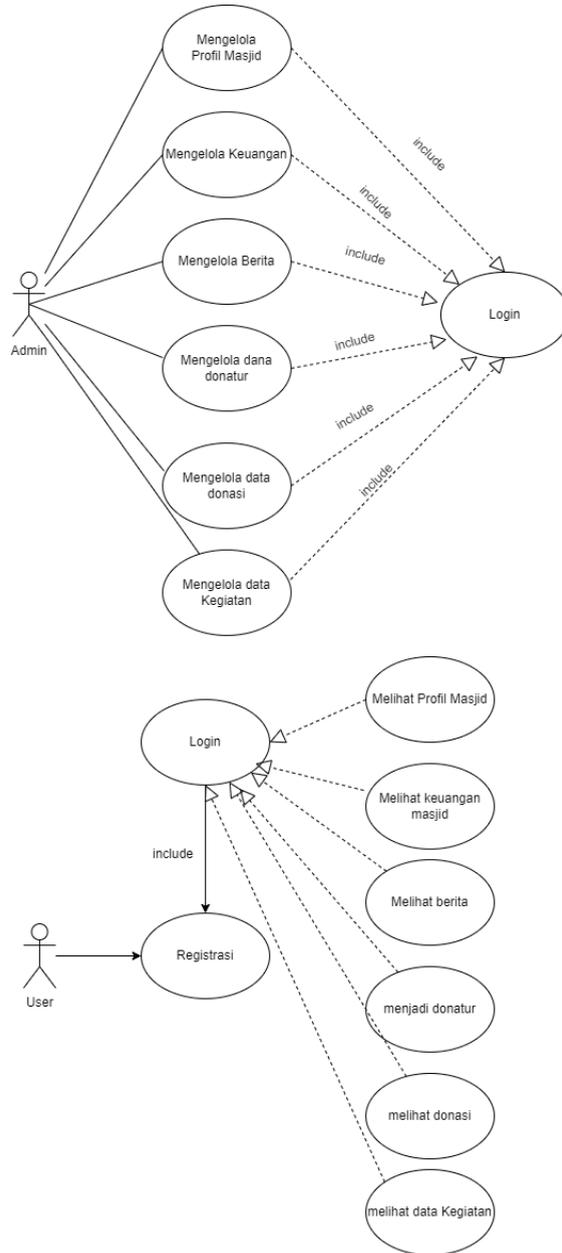
Diagram arsitektur adalah gambaran visual dari struktur dan keterhubungan komponen-komponen dalam sebuah sistem atau proyek perangkat lunak (lihat Gambar 2). Diagram arsitektur menyajikan pandangan grafis atas organisasi dan hubungan antara bagian-bagian dalam sistem tersebut.



Gambar 2. *Architecture Diagram* Sistem Informasi Masjid (SIM)

3.2.2.2 Use Case Diagram

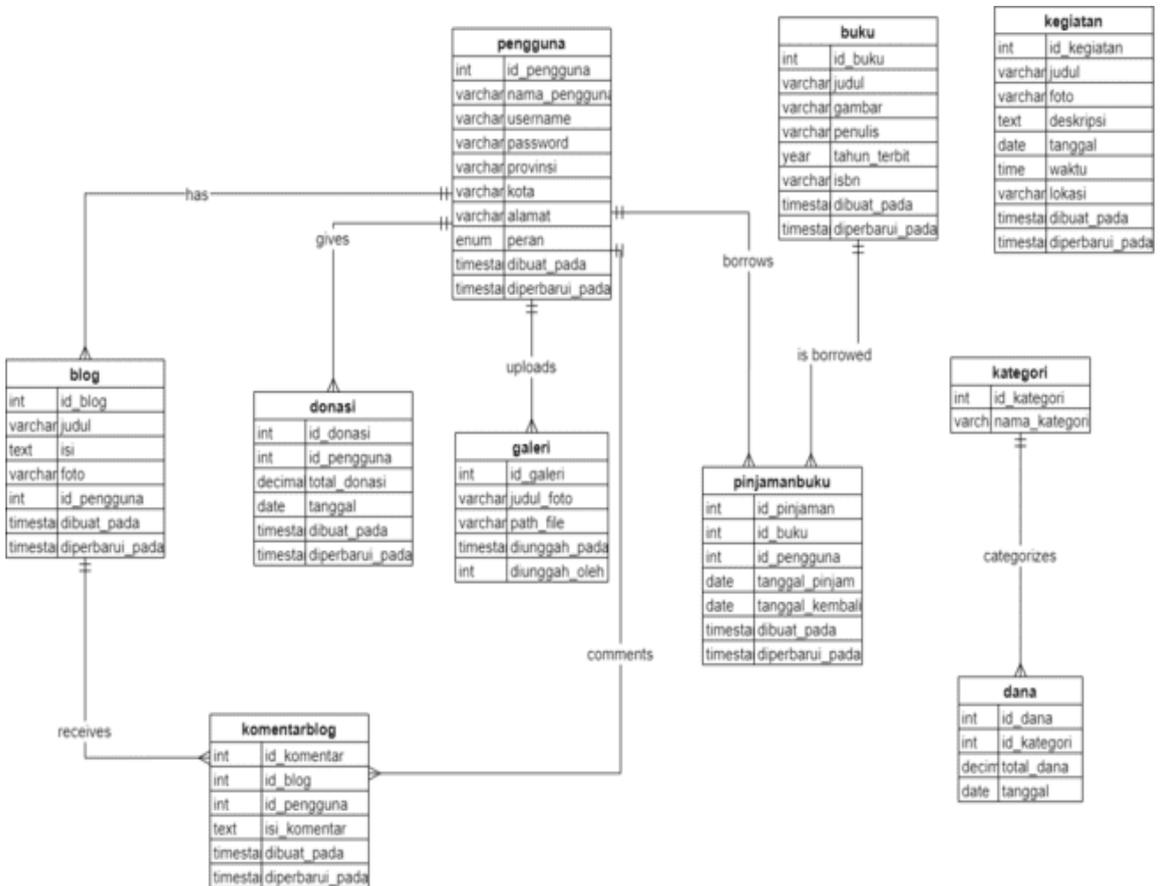
Gambar 3 adalah deskripsi visual dalam *Unified Modeling Language* (UML) [14] yang memperlihatkan kebutuhan fungsional dari suatu sistem dengan menggambarkan interaksi antara aktor (entitas eksternal) dan *use case* (fungsi atau proses tertentu dalam sistem).



Gambar 3. Use Case Diagram Sistem Informasi Masjid (SIM)

3.2.2.3 Entity Relationship Diagram

Entity Relationship Diagram (ERD) adalah gambaran visual yang digunakan untuk mengilustrasikan struktur dan keterhubungan antara entitas dalam sebuah basis data (lihat Gambar 4). Dengan menggunakan simbol-simbol grafis, ERD mengenali entitas-entitas utama yang menyimpan data, atribut-atribut yang mencerminkan karakteristik entitas, dan juga menampilkan hubungan antara entitas-entitas tersebut sesuai dengan peran yang dimiliki [15].



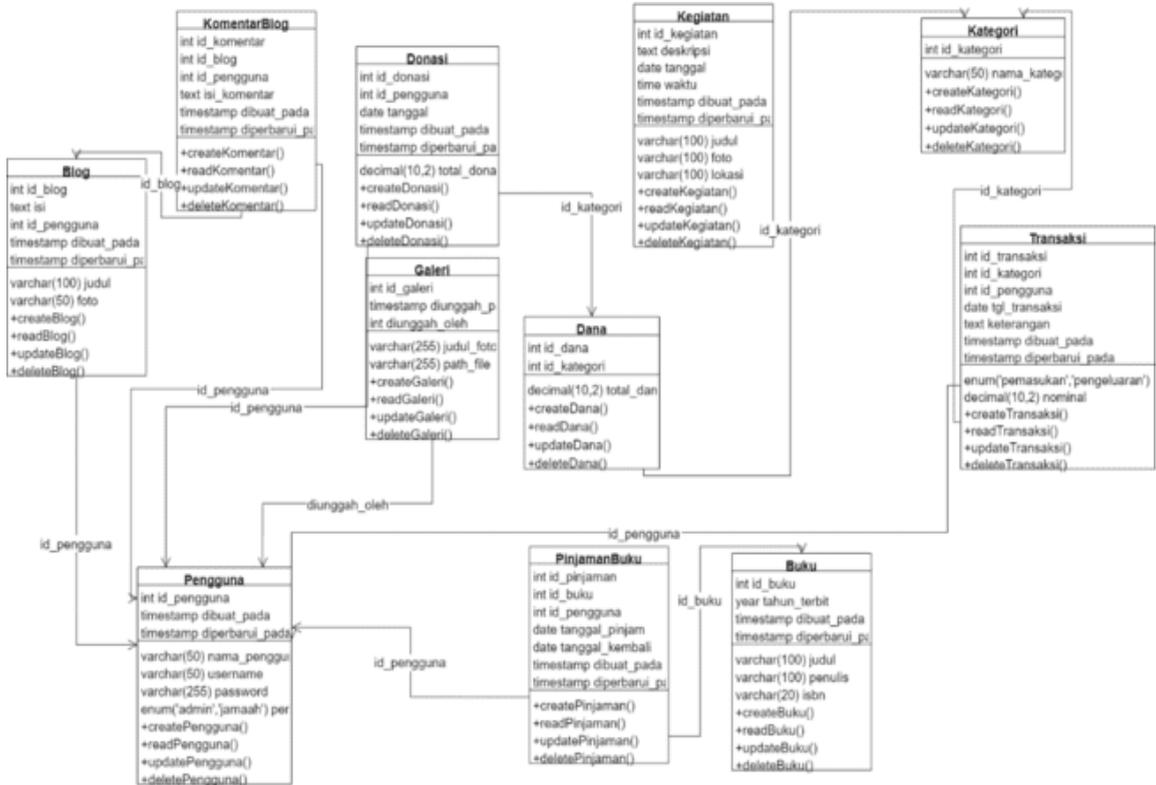
Gambar 4. *Entity Relationship Diagram* Sistem Informasi Masjid (SIM)

3.2.2.4 Class Diagram

Diagram kelas adalah representasi visual dari struktur kelas dalam sistem perangkat lunak yang divisualisasikan pada Gambar 5. Ini menggambarkan kelas-kelas dalam sistem, hubungan antara kelas-kelas tersebut, serta atribut dan metode yang dimiliki oleh setiap kelas. Diagram kelas membantu dalam merancang dan memodelkan struktur sistem secara keseluruhan, memfasilitasi pemahaman tentang bagaimana komponen-komponen sistem saling berinteraksi. Ini juga digunakan sebagai panduan dalam proses implementasi perangkat lunak, karena menyediakan pandangan yang jelas tentang hierarki dan hubungan antara kelas-kelas yang ada. Dengan demikian, diagram kelas menjadi alat yang penting dalam rekayasa perangkat lunak untuk memahami, merancang, dan mengelola sistem perangkat lunak yang kompleks.

3.2.3 Product Backlog

Product backlog isinya adalah daftar hal yang mau dikerjakan atau disebut, atau disebut juga *wishlist*. Penentuan dari pemilihan fitur *backlog* nya akan dibuatkan berdasarkan prioritas dari *product owner*. Dalam tahapan *product backlog* ini memerlukan waktu 15 hari yang akan ditampilkan di dalam Tabel 2 sebagai berikut.



Gambar 5. Class Diagram Sistem Informasi Masjid (SIM)

Tabel 2. Product Backlog

No	Nama Backlog	Kepentingan (1-100)	Perkiraan Waktu	Demo
1	Kebutuhan Dokumen dan Data	100	2 Hari	Mengumpulkan semua data menggunakan observasi dan wawancara untuk selanjutnya ke tahap suatu rancangan.
2	Pembuatan Suatu Rancangan UML	100	2 Hari	Memeriksa UML yang dibuatkan kesesuaiannya dengan Website yang dibuat.
3	Halaman Utama User	100	1 Hari	Di dalam halaman utama, yang akan menampilkan tampilan awal website masjid, termasuk informasi umum dan navigasi utama.
4	Halaman Login User	100	1 Hari	User dapat login dengan <i>username/email</i> dan <i>password</i> yang telah dibuat.

5	Dashboard dan Profil User	100	1 Hari	Setelah <i>login</i> berhasil, <i>user</i> akan diarahkan ke <i>dashboard</i> yang berisi informasi profil dan akses ke fitur lainnya.
6	Melihat Blog	90	1 Hari	<i>User</i> dapat melihat artikel blog terkait kegiatan dan informasi masjid.
7	Melihat Laporan Keuangan	95	1 Hari	<i>User</i> dapat melihat laporan keuangan masjid, termasuk pemasukan dan pengeluaran.
8	Melihat Galeri	85	1 Hari	<i>User</i> dapat melihat foto-foto kegiatan masjid yang diunggah di galeri.
9	Melihat Kegiatan	95	1 Hari	<i>User</i> dapat melihat jadwal kegiatan masjid, termasuk deskripsi dan foto kegiatan.
10	Melakukan Donasi	95	2 Hari	<i>User</i> dapat melakukan donasi secara online dan melihat riwayat donasi mereka.
11	Meminjam Buku	90	2 Hari	<i>User</i> dapat meminjam buku dari perpustakaan masjid, termasuk mengatur tanggal pinjam dan kembali.
12	Berkomentar pada Blog	85	1 Hari	<i>User</i> dapat menambahkan komentar pada artikel blog dan melihat komentar yang telah ditambahkan oleh user lain.
13	Manajemen Blog (Admin)	90	2 Hari	Admin dapat membuat, mengedit, menghapus, dan melihat artikel blog terkait kegiatan dan informasi masjid.
14	Manajemen Kegiatan (Admin)	95	2 Hari	Admin dapat membuat, mengedit, menghapus, dan melihat jadwal kegiatan masjid, termasuk deskripsi dan foto kegiatan.
15	Manajemen Donasi (Admin)	95	2 Hari	Admin dapat melihat informasi donasi, mengelola data donasi, dan melihat total donasi yang terkumpul.
16	Manajemen Dana (Admin)	95	2 Hari	Admin dapat mengelola informasi dana, termasuk pemasukan dan pengeluaran serta melihat total dana yang tersedia.
17	Manajemen Buku (Admin)	90	2 Hari	Admin dapat menambahkan, mengedit, menghapus buku dari perpustakaan masjid.
18	Manajemen Pinjaman Buku (Admin)	90	2 Hari	Admin dapat mengelola peminjaman buku, termasuk tanggal pinjam dan kembali.
19	Manajemen Galeri (Admin)	85	2 Hari	Admin dapat mengunggah, mengedit, menghapus, dan melihat foto-foto kegiatan masjid.
20	Manajemen Transaksi (Admin)	95	2 Hari	Admin dapat mencatat dan melihat transaksi keuangan masjid, baik pemasukan maupun pengeluaran.

3.2.4 Sprint Backlog

Sprint backlog adalah daftar pekerjaan yang harus diselesaikan oleh tim pengembang selama satu *sprint* dalam metode *Agile Scrum*, terdiri dari *user stories* atau *tasks* yang dipilih dari *product backlog*, lengkap dengan detail, deskripsi, dan kriteria penerimaan. Setiap tugas memiliki estimasi waktu atau *effort*, seperti *story points*, dan semuanya berkontribusi pada pencapaian *sprint goal* yang telah ditetapkan[13]. *Sprint*

backlog bersifat dinamis dan dapat diperbarui sepanjang *sprint* untuk memastikan tim tetap pada jalur yang benar dalam menyelesaikan pekerjaan yang telah direncanakan. Dan pada Tabel 3 dibawah menampilkan *Sprint Backlog*.

Tabel 3. *Sprint Backlog*

Sprint	Nama Backlog	Kepentingan (1-100)	Perkiraan Waktu	Tugas	Status
1	Kebutuhan Dokumen dan Data	100	2 Hari	Mengumpulkan data	<i>In Progress</i>
	Pembuatan Suatu Rancangan UML	100	2 Hari	Membuat rancangan UML	<i>To Do</i>
2	Halaman Utama <i>User</i>	100	1 Hari	Membuat halaman utama	<i>To Do</i>
	Halaman <i>Login User</i>	100	1 Hari	Membuat halaman login	<i>To Do</i>
3	<i>Dashboard</i> dan Profil <i>User</i>	100	1 Hari	Membuat <i>dashboard user</i>	<i>To Do</i>
4	Melihat Blog	90	1 Hari	Membuat halaman blog	<i>To Do</i>
	Berkomentar pada Blog	85	1 Hari	Membuat fitur komentar pada blog	<i>To Do</i>
5	Melihat Laporan Keuangan	95	1 Hari	Membuat halaman laporan keuangan	<i>To Do</i>
6	Melihat Galeri	85	1 Hari	Membuat halaman galeri	<i>To Do</i>
	Melihat Kegiatan	95	1 Hari	Membuat halaman kegiatan	<i>To Do</i>
7	Melakukan Donasi	95	2 Hari	Membuat fitur donasi	<i>To Do</i>
	Meminjam Buku	90	2 Hari	Membuat fitur peminjaman buku	<i>To Do</i>
8	Manajemen Blog (Admin)	90	2 Hari	Membuat halaman manajemen blog	<i>To Do</i>
	Manajemen Kegiatan (Admin)	95	2 Hari	Membuat halaman manajemen kegiatan	<i>To Do</i>
	Manajemen Donasi (Admin)	95	2 Hari	Membuat halaman manajemen donasi	<i>To Do</i>
	Manajemen Dana (Admin)	95	2 Hari	Membuat halaman manajemen dana	<i>To Do</i>
	Manajemen Buku (Admin)	90	2 Hari	Membuat halaman manajemen buku	<i>To Do</i>
	Manajemen Pinjaman Buku (Admin)	90	2 Hari	Membuat halaman manajemen pinjaman	<i>To Do</i>
	Manajemen Galeri (Admin)	85	2 Hari	Membuat halaman manajemen galeri	<i>To Do</i>
	Manajemen Transaksi (Admin)	95	2 Hari	Membuat halaman manajemen transaksi	<i>To Do</i>

3.2.5 Sprint Planning

Pada Tabel 4 *sprint planning* ini adalah untuk mencapai peningkatan iteratif pada produk dengan menambahkan fitur-fitur baru yang disepakati oleh tim

3.2.6 Testing

Dalam tahap ini, sistem akan diuji menggunakan metode *blackbox testing* [12]. Metode ini adalah jenis pengujian perangkat lunak di mana proses pengujian dilakukan tanpa memerlukan pemahaman mendalam tentang cara kerja internal sistem atau komponennya. Fokus pengujian adalah pada evaluasi *input* dan hasil *output* tanpa memperhatikan secara terperinci logika internal atau struktur kode. Skenario pengujian yang akan diterapkan pada sistem ini adalah pada Tabel 5.

Tabel 4. *Sprint Planning*

Sprint	No	Nama Backlog	Kepentingan (1-100)	Perkiraan Waktu	Tugas	Assignee
1	1	Kebutuhan Dokumen dan Data	100	2 Hari	Mengumpulkan data menggunakan observasi dan wawancara	Ridho
	2	Pembuatan Suatu Rancangan UML	100	2 Hari	Memeriksa UML yang dibuatkan kesesuaiannya dengan website yang dibuat	Ridho
2	3	Halaman Utama <i>User</i>	100	1 Hari	Menyusun desain halaman utama	Ridho
	4	Halaman <i>Login User</i>	100	1 Hari	Membuat halaman <i>login</i>	Ridho
3	5	<i>Dashboard</i> dan Profil <i>User</i>	100	1 Hari	Membuat desain <i>dashboard</i> dan profil user	Ridho
	6	Melihat Blog	90	1 Hari	Membuat desain halaman untuk melihat blog	Ridho
4	12	Berkomentar pada Blog	85	1 Hari	Membuat fitur untuk berkomentar pada blog	Ridho
	7	Melihat Laporan Keuangan	95	1 Hari	Menyiapkan struktur halaman laporan keuangan	Ridho
5	8	Melihat Galeri	85	1 Hari	Menyiapkan desain halaman untuk melihat galeri	Ridho
	9	Melihat Kegiatan	95	1 Hari	Merencanakan desain halaman untuk melihat kegiatan	Ridho
6	10	Melakukan Donasi	95	2 Hari	Menyiapkan desain fitur untuk melakukan donasi	Ridho
	11	Meminjam Buku	90	2 Hari	Menyiapkan desain halaman untuk meminjam buku	Ridho
7	13	Manajemen Blog (Admin)	90	2 Hari	Menyiapkan desain halaman untuk manajemen blog (admin)	Ridho
	14	Manajemen Kegiatan (Admin)	95	2 Hari	Menyiapkan desain halaman untuk manajemen kegiatan (admin)	Ridho
8	15	Manajemen Donasi (Admin)	95	2 Hari	Menyiapkan desain halaman untuk manajemen donasi (admin)	Ridho
	16	Manajemen Dana (Admin)	95	2 Hari	Menyiapkan desain halaman untuk manajemen dana (admin)	Ridho
8	17	Manajemen Buku (Admin)	90	2 Hari	Menyiapkan desain halaman untuk manajemen buku (admin)	Ridho
	18	Pinjaman Buku (Admin)	90	2 Hari	Menyiapkan desain halaman untuk manajemen pinjaman buku (admin)	Ridho
19	19	Manajemen Galeri (Admin)	85	2 Hari	Menyiapkan desain halaman untuk manajemen galeri (admin)	Ridho
	20	Manajemen Transaksi (Admin)	95	2 Hari	Menyiapkan desain halaman untuk manajemen transaksi (admin)	Ridho

4. Hasil dan Pembahasan

4.1 Hasil

4.1.1 Implementasi Product Backlog

Selama fase implementasi, *Product Backlog* yang telah disusun menunjukkan daftar fitur yang direncanakan untuk dikembangkan berdasarkan prioritas dan kebutuhan pengguna. Hasil dari fase ini meliputi:

- 1) Kebutuhan Dokumen dan Data: Data dikumpulkan melalui observasi dan wawancara dengan para ahli domain dan pemangku kepentingan. Semua data yang dikumpulkan mendukung dalam tahap pembuatan rancangan.

- 2) Rancangan UML: Diagram UML yang dibuat diverifikasi untuk memastikan kesesuaiannya dengan spesifikasi yang diharapkan oleh sistem .
- 3) Pengembangan Halaman Utama, *Login*, dan *Dashboard User*: Halaman utama, halaman *login*, dan *dashboard user* dirancang dan diimplementasikan sesuai dengan kebutuhan yang diidentifikasi.

Tabel 5. *Testing*

Skenario	Input	Output
Membuat akun <i>user</i>	Data <i>User</i>	Status Berhasil
<i>Login User</i>	<i>Email</i> , <i>Password</i>	<i>Dashboard User</i>
<i>Login admin</i>	<i>Email</i> , <i>Password</i>	<i>Dashboard</i> Admin
Membuat artikel	judul, penulis,isi, tanggal nama kegiatan, deskripsi, tanggal, waktu mulai dan selesai, tempat, dan foto kegiatan	Artikel Berhasil Ditambahkan
Membuat data kegiatan		kegiatan berhasil ditambahkan
Melihat semua informasi pada <i>dashboard user</i>	<i>Authorization user</i>	Informasi berjalan dengan baik

4.1.2 Implementasi Sprint Backlog

Setiap sprint dijalankan dengan fokus pada penyelesaian item dari *Sprint Backlog*. Berikut adalah hasil dari beberapa *sprint*:

- 1) *Sprint 1*:
 - a. Kebutuhan Dokumen dan Data: Tugas pengumpulan data selesai sesuai jadwal .
 - b. Pembuatan Rancangan UML: UML dirancang dengan lengkap dan diverifikasi oleh tim .
- 2) *Sprint 2*:
 - a. Halaman Utama *User* dan *Login User*: Kedua halaman ini dirancang dan diuji untuk memastikan fungsionalitas dasar dapat berjalan dengan baik .
- 3) *Sprint 3*:
 - a. *Dashboard* dan *Profil User*: *Dashboard* yang berisi informasi profil *user* selesai diimplementasikan dan berfungsi dengan baik sesuai dengan skenario pengujian .

4.1.3 Hasil Sprint Planning

Sprint Planning yang dilakukan setiap periode *sprint* berhasil mengidentifikasi dan mendistribusikan tugas secara efisien kepada tim. *Assignee* yang bertanggung jawab berhasil menyelesaikan tugas-tugasnya sesuai dengan jadwal yang telah ditentukan.

4.1.4 Hasil Pengujian

Pengujian sistem dilakukan dengan metode *blackbox testing*. Hasil pengujian menunjukkan bahwa semua skenario yang diuji telah berhasil memenuhi kriteria yang diharapkan:

- 1) Pembuatan Akun *User*: Akun berhasil dibuat dengan *input* data *user* yang valid.
- 2) *Login User*: *User* berhasil *login* dan diarahkan ke *dashboard user*.
- 3) *Login Admin*: Admin berhasil *login* dan diarahkan ke *dashboard* admin.
- 4) Membuat Artikel: Artikel berhasil ditambahkan dengan *input* yang lengkap dan valid .

- 5) Membuat Data Kegiatan: Kegiatan berhasil ditambahkan dengan *input* yang lengkap dan valid.
- 6) Melihat Informasi pada *Dashboard User*: Informasi pada *dashboard user* berjalan dengan baik dan menampilkan data yang diharapkan .

4.2 Pembahasan

4.2.1 Evaluasi Hasil Sprint

Implementasi dari *product backlog* dan *sprint backlog* menunjukkan bahwa metode *Agile* yang digunakan efektif dalam menyelesaikan tugas-tugas secara iteratif dan inkremental. Namun, terdapat beberapa kendala yang dihadapi selama *sprint*, seperti:

- 1) Komunikasi Tim: Terjadi beberapa kali miskomunikasi dalam tim yang mengakibatkan beberapa fitur harus diulang pengerjaannya .
- 2) Estimasi Waktu: Beberapa tugas membutuhkan waktu lebih lama dari yang diestimasi karena kompleksitas yang tidak terduga .

Langkah-langkah perbaikan yang diambil termasuk pertemuan rutin tim untuk memastikan semua anggota tim memiliki pemahaman yang sama tentang tugas yang dikerjakan.

4.2.2 Kualitas Produk

Secara keseluruhan, sistem informasi manajemen masjid yang dikembangkan memenuhi kebutuhan fungsional yang diharapkan. Berikut adalah beberapa aspek yang dievaluasi:

- 1) Fungsionalitas: Semua fitur utama yang diidentifikasi dalam *product backlog* berhasil diimplementasikan dan berfungsi dengan baik.
- 2) Kinerja Sistem: Sistem berjalan dengan lancar dan responsif, memberikan pengalaman pengguna yang baik.
- 3) *Feedback* Pengguna: Pengguna memberikan *feedback* positif terhadap sistem, terutama dalam kemudahan penggunaan dan efisiensi pengelolaan data masjid.

4.2.3 Manfaat dan Dampak

Implementasi sistem informasi manajemen masjid memberikan beberapa manfaat signifikan:

- 1) Efisiensi Operasional: Sistem membantu dalam meningkatkan efisiensi operasional masjid melalui otomatisasi tugas-tugas administratif .
- 2) Transparansi dan Akuntabilitas: Dengan adanya fitur laporan keuangan dan manajemen donasi, transparansi dan akuntabilitas pengelolaan keuangan masjid meningkat .
- 3) Pengelolaan Data: Sistem ini mempermudah pengelolaan data jamaah, keuangan, dan kegiatan masjid, serta menyediakan akses mudah bagi jamaah untuk berinteraksi dengan masjid .

4.2.4 Perbandingan dengan Sistem Sebelumnya

Sistem baru ini menunjukkan beberapa keunggulan dibandingkan dengan metode pengelolaan sebelumnya:

- 1) Otomatisasi: Tugas-tugas yang sebelumnya dilakukan secara manual sekarang dapat diotomatisasi, mengurangi beban kerja pengurus masjid .

- 2) Aksesibilitas: Informasi dan layanan masjid kini lebih mudah diakses oleh jamaah melalui *platform online*.
- 3) Efisiensi: Waktu yang dibutuhkan untuk mengelola data dan kegiatan masjid berkurang secara signifikan.

4.2.5 Efisiensi Penggunaan Metode *Agile*

Penggunaan metode *Agile* dalam pengembangan sistem informasi manajemen masjid dinilai melalui beberapa aspek yang menunjukkan peningkatan efisiensi:

- 1) Respons Terhadap Perubahan: Dalam metode *Agile*, tim pengembang mampu merespons perubahan kebutuhan pengguna dengan lebih cepat dibandingkan dengan pendekatan tradisional yang kaku. Fleksibilitas ini memudahkan tim untuk melakukan penyesuaian yang diperlukan tanpa mengorbankan *timeline* proyek secara signifikan.
- 2) Iterasi dan *Feedback* Berkelanjutan: Pendekatan *Agile* memungkinkan pengumpulan *feedback* secara terus-menerus dari pemangku kepentingan selama siklus pengembangan. Hal ini memberikan peluang untuk segera mengatasi masalah yang muncul, mengurangi risiko kesalahan yang lebih besar di akhir proyek.
- 3) Peningkatan Kolaborasi Tim: Melalui pertemuan rutin (*stand-up meetings*) dan *sprint review*, metode *Agile* meningkatkan komunikasi dan kolaborasi antar anggota tim. Meskipun terdapat kendala komunikasi, metode ini membantu meminimalkan miskomunikasi dan memastikan bahwa setiap anggota tim berada pada jalur yang sama.
- 4) Kecepatan Penyelesaian Tugas: Walaupun tidak ada perbandingan langsung dengan sistem sebelumnya, metode *Agile* yang berbasis *sprint* memastikan bahwa setiap tugas atau fitur diselesaikan dalam waktu yang relatif singkat, sesuai dengan prioritas yang telah ditentukan.
- 5) Kualitas Produk Akhir: Pengujian yang dilakukan secara iteratif memastikan bahwa setiap fitur yang dikembangkan diuji secara menyeluruh sebelum diintegrasikan ke dalam sistem. Ini meningkatkan kualitas produk akhir dan mengurangi jumlah kesalahan atau *bug* yang terdeteksi setelah implementasi.

4.2.6 Peningkatan Efisiensi Operasional Masjid

Implementasi sistem informasi manajemen masjid menggunakan metode *Agile* menunjukkan beberapa indikasi peningkatan efisiensi operasional, meskipun pengukuran kuantitatif langsung tidak dilakukan. Berikut adalah beberapa aspek yang menunjukkan peningkatan tersebut:

- 1) Otomatisasi Proses Administratif: Sebelum pengembangan sistem ini, banyak proses administratif seperti pengelolaan data jamaah, pencatatan keuangan, dan pelaporan kegiatan dilakukan secara manual. Dengan sistem baru, banyak dari proses ini telah diotomatisasi, mengurangi waktu dan tenaga yang diperlukan untuk tugas-tugas tersebut.
- 2) Penyederhanaan Akses Informasi: Dengan *dashboard* yang dirancang untuk *user* dan admin, informasi penting seperti laporan keuangan, jadwal kegiatan, dan status donasi menjadi lebih mudah diakses. Ini mempermudah pengurus masjid dalam mengambil keputusan cepat, dan meningkatkan transparansi informasi bagi jamaah.
- 3) Pengurangan Kesalahan Manual: Dengan otomatisasi dan pengujian berkelanjutan, jumlah kesalahan yang biasanya terjadi akibat proses manual dapat dikurangi. Hal ini tidak hanya meningkatkan akurasi data tetapi juga mengurangi waktu yang dihabiskan untuk perbaikan atau revisi data.
- 4) Kemudahan Pengelolaan Kegiatan dan Donasi: Sistem baru memungkinkan jamaah untuk dengan mudah mendaftarkan diri dalam kegiatan masjid dan melakukan donasi secara online. Ini tidak hanya meningkatkan partisipasi jamaah tetapi juga mempermudah pengelola masjid dalam melacak dan mengelola donasi serta kehadiran jamaah dalam kegiatan.

5. Kesimpulan

5.1 Simpulan

Penggunaan metode Agile dalam pengembangan sistem informasi manajemen masjid terbukti efektif dalam memberikan hasil yang sesuai dengan kebutuhan pengguna. Meskipun efisiensi operasional masjid mengalami peningkatan, klaim ini terutama didasarkan pada kualitas implementasi dan kepuasan pengguna, tanpa data kuantitatif langsung yang mengukur peningkatan efisiensi secara eksplisit.

5.2 Saran

Untuk pengembangan lebih lanjut, disarankan untuk:

- 1) Melakukan Pelatihan Pengguna: Melatih pengurus masjid dan jamaah untuk menggunakan sistem secara efektif .
- 2) Pengembangan Fitur Tambahan: Mengembangkan fitur-fitur tambahan berdasarkan *feedback* dari pengguna untuk terus meningkatkan sistem .
- 3) Pemeliharaan dan Pembaruan Sistem: Melakukan pemeliharaan rutin dan pembaruan sistem untuk memastikan sistem tetap berjalan optimal dan aman .

6. Daftar Rujukan

- [1] Ridwanullah, A.I. and Herdiana, D., 2018. Optimalisasi pemberdayaan masyarakat berbasis masjid. *Ilmu Dakwah: Academic Journal for Homiletic Studies*, 12(1), pp.82-98.
- [2] Untung, S., 2019. Masjid dan Aktivitas Dakwah di Era Revolusi Industri 4.0. *Religia*, pp.227-247.
- [3] Awaluddin, M.I., Arifin, R.W. and Setiyadi, D., 2020. Implementasi Framework Laravel Pada Sistem Informasi Pengelolaan Aset Laboratorium Komputer. *Bina Insani Ict Journal*, 7(2), pp.187-197.
- [4] Sohail, F., Zia, S.S., Qureshi, R., Naseem, M. and Haider, H., 2021. Impact of agile methodology on software development life cycle. *Pakistan Journal of Engineering and Technology*, 4(2), pp.153-158.
- [5] Anggraini, R., 2021. Rancang Bangun Sistem Informasi Administrasi Pengelolaan Dana Masjid Berbasis Web (Studi Kasus: Masjid Al-Muttaqin). *Jurnal Teknologi dan Sistem Informasi*, 2(3), pp.109-118.
- [6] Muhammad, R.P. and El Ibrahim, G., 2024. Rancang Bangun Sistem PPDB Online Studi Kasus SMK Muhammadiyah Gamping Menggunakan Metode Extreme Programming. *Jurnal Informatika dan Teknik Elektro Terapan*, 12(2).
- [7] Susanti, S., Junianto, E. and Rachman, R., 2017. Implementasi Framework Laravel Pada Aplikasi Pengolah Nilai Akademik Berbasis Web. *Jurnal Informatika*, 4(1).
- [8] AGILE, K.M.M.M. and ARIFIN, I.M., Sistem Informasi Manajemen dan pengelolaan.
- [9] Allard, M.F. and Voutama, A., 2024. Rancang Bangun Sistem Informasi Reservasi Hotel "Hotel Hebat" Berbasis Website. *Jurnal Informatika dan Teknik Elektro Terapan*, 12(2).
- [10] Mike, A., Zen, B.P. and Utami, A., 2023. Penerapan Metode Agile Pada Website Indekost Sruntul Menggunakan Framework Laravel. *Jurnal Ilmiah Media Sisfo*, 17(2), pp.266-279.
- [11] Parameswari, S.D., Prasetyo, N.A. and Junaidi, A., 2022. Aplikasi Berbasis Web Deteksi Undertone Menggunakan Metode Agile Untuk Rekomendasi Makeup. *Jurnal Ilmiah Media Sisfo*, 16(1), pp.62-70.
- [12] Putra, M.R.S., Santoso, N. and Kurniawan, T.A., 2021. Pengembangan Sistem Informasi Manajemen Masjid Al Ghifari berbasis Web. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 5(11), pp.4711-4718.
- [13] Peta, G.E., Zidanie, I. and Setiawan, M.A., 2023. Perancangan Sistem Informasi Masjid Nurut Taqwa Berbasis Web dengan Metode Incremental. *Scientia Sacra: Jurnal Sains, Teknologi dan Masyarakat*, 3(2), pp.176-189.
- [14] Herfandi, H., Yuliadi, Y., Zaen, M.T.A., Hamdani, F. and Safira, A.M., 2022. Penerapan Metode Design Thinking Dalam Pengembangan UI dan UX. *Building of Informatics, Technology and Science (BITS)*, 4(1), pp.337-344.
- [15] Peta, G.E., Zidanie, I. and Setiawan, M.A., 2023. Perancangan Sistem Informasi Masjid Nurut Taqwa Berbasis Web dengan Metode Incremental. *Scientia Sacra: Jurnal Sains, Teknologi dan Masyarakat*, 3(2), pp.176-189.

