

BAB II

TINJAUAN PUSTAKA

2.1 Teori Dasar

Bab ini menjelaskan konsep-konsep dan definisi dari berbagai ahli terkait variabel yang diteliti, untuk memberikan pemahaman yang lebih mendalam dan relevan dengan topik yang dibahas.

2.1.1 Software Development

Pengembangan perangkat lunak, atau yang dikenal dengan *Software Development*, merupakan inti dari proyek-proyek di bidang teknologi informasi. Proses ini memiliki peran vital dalam industri teknologi informasi dan komunikasi (TIK). Seiring dengan pesatnya perkembangan teknologi, pengembangan perangkat lunak semakin kompleks dan membutuhkan pemahaman yang mendalam tentang teknologi serta prinsip-prinsip pengembangan perangkat lunak yang efektif dan berkualitas.

Penelitian ini mengangkat tema *Software Development* dalam pengembangannya. Dengan pesatnya perkembangan industri teknologi, *Software Development* turut berkembang cepat sejalan dengan kemajuan teknologi itu sendiri. Pengembangan perangkat lunak atau *software development* mencakup proses pembuatan dan pengembangan perangkat lunak melalui serangkaian tahapan yang sistematis dan terstruktur.

Dalam pengembangannya, *software development* melibatkan serangkaian tahapan yang dilakukan untuk menciptakan aplikasi. Tahapan-tahapan tersebut meliputi:

1. Analisis Kebutuhan

Tahap analisis ini dilakukan agar pengembang mengetahui dan memahami kebutuhan pengguna. Menurut (Sihombing Febrian, 2023) tahap yang paling krusial dalam pengembangan perangkat lunak adalah analisis kebutuhan. Kualitas proyek perangkat lunak sangat dipengaruhi oleh seberapa baik kebutuhan dianalisis.

2. Perencanaan

Perencanaan menjadi tahap selanjutnya dalam pengembangan aplikasi, yang meliputi pemilihan metode pengembangan yang tepat untuk mencapai tujuan proyek, serta estimasi biaya dan waktu yang dibutuhkan untuk mengembangkan aplikasi.

3. Desain

Tahapan desain mencakup perancangan aplikasi secara rinci, termasuk pengembangan antarmuka pengguna (UI), pengalaman pengguna (UX), serta arsitektur aplikasi. Desain yang matang diharapkan dapat mendukung tujuan fungsional dan estetika aplikasi, memastikan aplikasi berjalan dengan lancar dan memberikan pengalaman pengguna yang optimal.

4. Implementasi

Pada tahap implementasi, proses pengkodean dilakukan, termasuk pengembangan database dan struktur program yang mendukung fungsi aplikasi. Tahap ini mencakup penerapan desain dan logika yang telah direncanakan sebelumnya ke dalam bentuk kode yang dapat dijalankan.

5. Pengujian

Tahapan pengujian bertujuan untuk memastikan aplikasi berfungsi sesuai dengan harapan dan memenuhi kebutuhan pengguna. Pengujian dilakukan untuk mendeteksi bug, kesalahan, atau ketidaksesuaian dengan spesifikasi yang ditetapkan, guna memastikan aplikasi stabil, efisien, dan bebas dari masalah teknis.

Beragam pendekatan dalam pengembangan perangkat lunak diterapkan, salah satunya adalah metode Agile. Setiap pendekatan menawarkan keunggulan dan tantangan yang unik, yang dapat disesuaikan dengan kebutuhan spesifik serta dinamika proyek yang tengah dijalankan.

2.1.2 Metode Agile

Pada akhir 1990-an, Agile mulai dikenal luas sebagai pendekatan yang berupaya mengatasi tantangan dalam pengembangan perangkat lunak yang kaku dan sangat bergantung pada perencanaan terstruktur. Namun, cikal bakal Agile sebenarnya sudah muncul sejak 1980-an, ketika keterbatasan metode tradisional semakin terlihat, terutama dengan pesatnya perkembangan

teknologi dan meningkatnya ketidakstabilan di lingkungan bisnis (Setiawan & Effendy, 2022).

Metode agile dalam pengembangan perangkat lunak menurut (Al-Saqqa et al., 2020) dirancang sebagai solusi atas keterbatasan metode tradisional. Pendekatan ini berfokus pada efisiensi dengan menekan biaya serta mengurangi beban kerja yang tidak perlu, sambil tetap memungkinkan penyesuaian terhadap perubahan kebutuhan di setiap tahap pengembangan. Agile mengatur tugas dan koordinasi berdasarkan prinsip serta nilai-nilai yang telah ditetapkan.



Gambar 2. 1 Tahapan Agile Development
Sumber: (Hendra et al., 2024)

Tahapan pada metode agile mencerminkan langkah dengan metode agile yang terstruktur, tahapan dalam metode agile dapat dijabarkan menurut (Hendra et al., 2024):

1. Perencanaan (*Planning*)

Pada tahap awal ini, proses perumusan ide dan konsep sistem dilakukan dengan mendalam, sambil mempertimbangkan berbagai tantangan yang ada dan merancang solusi yang tepat untuk menghadapinya. Ini adalah fondasi penting untuk memastikan pengembangan sistem berjalan dengan jelas dan sesuai tujuan.

2. Perancangan Desain (*Design*)

Ide dan konsep yang telah terbentuk dipadukan dalam bentuk desain awal atau mockup. Tahapan ini berfokus pada visualisasi konsep agar lebih mudah dipahami dan disesuaikan dengan kebutuhan pengguna, dengan mempertimbangkan aspek fungsionalitas dan estetika yang saling mendukung.

3. Pengembangan (*Development*)

Di sini, ide yang telah dirancang diwujudkan dalam bentuk kode pemrograman yang sesungguhnya. Pengembangan sistem berlandaskan pada prinsip aturan dan kebutuhan pengguna. Selain itu, fase ini tidak hanya mengenai pembangunan teknis, tetapi juga termasuk umpan balik dari tim untuk memastikan sistem yang dibuat dapat beradaptasi dan berkembang jika diperlukan, menjaga keselarasan dengan tujuan awal.

4. Pengujian (*Testing*)

Sebelum sistem diluncurkan, tahap ini menguji kehandalan aplikasi secara menyeluruh. Pengujian berfokus pada aspek kegunaan dan performa aplikasi untuk memastikan pengalaman pengguna yang optimal, serta mengidentifikasi potensi masalah atau kekurangan yang perlu diperbaiki sebelum implementasi.

5. Dokumentasi (*Documentation*)

Di tahap ini, seluruh proses, keputusan, serta hasil yang diperoleh dari pembangunan sistem dicatat dengan teliti. Dokumentasi tidak hanya meliputi spesifikasi teknis, tetapi juga evaluasi kinerja tim dan efektivitas solusi yang diimplementasikan, memberikan gambaran menyeluruh tentang perjalanan proyek.

6. Implementasi (*Deployment*)

Setelah mendapatkan evaluasi dan persetujuan dari tim serta pihak terkait, sistem yang telah siap akan dideploy untuk digunakan oleh pengguna. Tahapan ini menandakan bahwa sistem yang telah teruji kini dapat beroperasi di lingkungan nyata, memberikan manfaat bagi penggunanya dan siap menghadapi tantangan lebih lanjut.

Prioritas utama dalam Agile adalah kecepatan dan kemampuan beradaptasi, didukung oleh komunikasi efektif dan keterlibatan aktif seluruh anggota tim, termasuk pelanggan, untuk memastikan kebutuhan mereka

terpenuhi. Metode Agile membantu organisasi menjadi lebih fleksibel dan tanggap terhadap perubahan, sehingga dapat mengurangi risiko proyek yang muncul akibat perubahan kebutuhan atau prioritas. Selain itu, pendekatan iteratif memungkinkan organisasi menerima umpan balik lebih dini dalam proses pengembangan, sehingga penyesuaian dapat dilakukan dengan lebih cepat dan efisien (Inayah, 2024). Selain itu, manajemen proyek dalam Agile mencakup perencanaan, delegasi, pemantauan, dan pengendalian proyek dengan tujuan mencapai hasil dalam batasan waktu, biaya, dan kualitas yang telah ditetapkan.

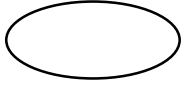
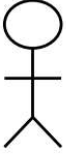
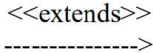
2.1.3 Unified Modeling Language (UML)

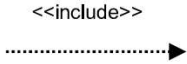
Unified Modeling Language (UML) merupakan bahasa pemodelan visual yang berfungsi untuk merancang, menggambarkan, dan mendokumentasikan sistem perangkat lunak secara terstruktur. UML berfungsi untuk membantu dalam mendeskripsikan serta merancang sistem perangkat lunak, terutama yang dikembangkan dengan pendekatan pemrograman berorientasi objek. Dengan UML, pengembang dapat memvisualisasikan struktur dan perilaku sistem secara lebih sistematis, memudahkan dalam memahami alur dan hubungan antar komponen dalam sistem yang kompleks (Nistrina & Sahidah, 2022).

1. *Use case* Diagram

Use Case Diagram digunakan untuk menggambarkan interaksi antara pengguna dan sistem yang akan dikembangkan.

Table 2. 1 *Use Case Diagram*

Simbol	Keterangan
<i>Use Case</i> 	Fungsionalitas yang ditampilkan oleh sistem saat komponen saling bertukar pesan antara unit atau aktor sering digambarkan dengan kata kerja di awal frasa nama use case.
Aktor 	Proses, orang, dan sistem lain yang berinteraksi dengan sistem informasi yang sedang dikembangkan berada di luar sistem tersebut. Aktor tidak selalu merujuk pada orang, meskipun simbolnya berupa gambar manusia; umumnya, kata benda ditempatkan di awal frasa nama aktor untuk mengidentifikasinya.
Asosiasi 	Interaksi antara aktor dan use case yang terlibat menunjukkan partisipasi aktor dalam use case, atau adanya interaksi antara use case dan aktor.
<i>Extend</i> 	Hubungan use case terlampir ke use case tambahan yang memperluas

	fungsionalitasnya, mirip dengan konsep pewarisan dalam pemrograman berorientasi objek. Biasanya, use case tambahan memiliki nama yang sama, dengan panah yang menunjuk ke use case utama.
<i>Generalization</i> 	Relasi generalisasi dan spesialisasi (Umum-Khusus) menggambarkan hubungan antara dua use case, di mana satu fungsi bersifat umum dan mencakup fungsi lainnya. Contohnya, arah panah menunjukkan ke use case yang digeneralisasi (generik).
Include 	Hubungan use case tambahan mengacu pada ketergantungan suatu use case terhadap use case lainnya untuk berfungsi atau sebagai prasyarat eksekusi. Terdapat dua cara pandang utama mengenai hal ini: pertama, include menunjukkan bahwa use case tambahan selalu dijalankan bersama use case utama; kedua, include memastikan bahwa use case tambahan hanya

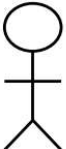
	dieksekusi jika sudah dipastikan dijalankan sebelumnya.
--	---

Sumber: (Hospita YantiS & Ani Arnomo, 2023)

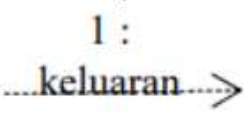
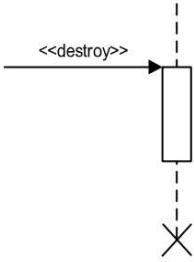
2. *Sequence Diagram*

Sequence diagram digunakan untuk memvisualisasikan urutan langkah dalam merespons suatu kejadian hingga menghasilkan output. Diagram ini menunjukkan pemicu aktivitas, proses internal sistem, serta perubahan yang terjadi sebelum output dihasilkan (Hary & Nurhaman, 2019).

Table 2. 2 Sequence Diagram

Simbol	Keterangan
Aktor 	Proses, individu, atau sistem lain yang berinteraksi dengan sistem informasi yang sedang dikembangkan berada di luar sistem itu sendiri, sehingga aktor tidak selalu merujuk pada manusia, meskipun simbol aktor digambarkan sebagai gambar orang. Umumnya, kata benda digunakan di awal frasa nama aktor untuk menggambarkannya.

<i>Lifeline</i> 	Mewakili perjalanan hidup suatu objek.
Objek 	Mendefinisikan objek yang berinteraksi melalui pesan.
Waktu Aktif 	Menyatakan bahwa objek tersebut aktif dan dapat berinteraksi, dengan setiap tindakan yang terjadi selama periode aktifnya menjadi langkah-langkah yang dilakukan di dalamnya.
Pesan Tipe <i>Create</i> 	Mendeklarasikan bahwa suatu objek menciptakan objek lain, dengan panah yang mengarah ke objek yang dibuat. Panah tersebut menunjukkan objek yang memanggil operasi atau metode, yang harus sesuai dengan diagram kelas objek yang berinteraksi.

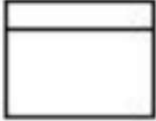
Pesan tipe <i>Send</i> 	Merupakan objek yang mengirimkan input, data, atau informasi ke objek lain, dengan panah yang mengarah ke objek pengirim.
Pesan tipe <i>return</i> 	Menunjukkan bahwa suatu objek kembali ke objek yang dituju setelah menjalankan operasi atau metode tertentu, dengan panah yang mengarah ke objek penerima pengembalian.
Pesan tipe <i>destroy</i> 	Mendeklarasikan objek untuk mengakhiri eksistensi objek lain, dengan panah yang mengarah ke objek yang dihentikan. Sebaiknya, jika ada <i>create</i>, maka diikuti dengan <i>destroy</i>.

3. Activity Diagram

Activity Diagram memperlihatkan serangkaian langkah dan keputusan yang diambil untuk mencapai tujuan tertentu, serta hubungan antara aktivitas-aktivitas tersebut.

Table 2. 3 Activity Diagram

Simbol	Keterangan
Status awal 	Keadaan awal dari aktivitas sistem, di mana diagram aktivitas dimulai dengan keadaan awal.
Aktivitas 	Sebuah tindakan yang dilakukan oleh sistem, yang biasanya diawali dengan kata kerja.
Percabangan 	Asosiasi cabang terjadi ketika terdapat lebih dari satu opsi aktivitas.
Penggabungan 	Digunakan untuk menggabungkan asosiasi, yaitu menyatukan beberapa aktivitas menjadi satu aktivitas.
Status akhir 	Keadaan akhir dari eksekusi sistem, di mana diagram aktivitas memiliki titik akhir.

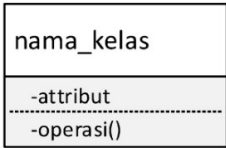
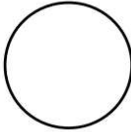
<i>Swimlane</i> 	Pemisahan unit organisasi bisnis yang bertanggung jawab atas kegiatan yang sedang berlangsung.
--	--

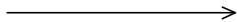
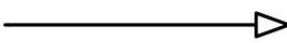
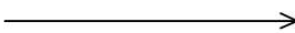
Sumber: (Hary & Nurhaman, 2019)

4. Class Diagram

Class Diagram menggambarkan struktur sistem secara statis melalui representasi kelas-kelas dan hubungan antar kelas. Diagram ini menunjukkan atribut, metode, dan keterkaitan antara berbagai entitas dalam sistem, sehingga memberikan gambaran yang jelas mengenai arsitektur dan desain sistem.

Table 2. 4 Class Diagram

Simbol	Deskripsi
Kelas 	Kelas untuk komponen sistem.
<i>Interface</i> 	Mengubah konsep antarmuka (<i>interface</i>) dalam pemrograman berorientasi objek.

<i>Association</i> 	Dalam hubungan antara kelas dan generalisasi, sering kali diikuti oleh asosiasi.
Asosiasi berarah 	Hubungan antara kelas dengan definisi umum menuju spesialisasi.
Generalisasi 	Hubungan antara kelas dengan definisi umum menuju spesialisasi.
<i>Dependency</i> 	Ketergantungan antar kelas.
<i>Aggregation</i> 	Hubungan antara kelas dan makna keseluruhan dari bagiannya.

Sumber: (Ramdany et al., 2024)

2.2 Teori Khusus

Bagian ini akan membahas teori-teori khusus yang mendasari topik penelitian ini, dengan fokus pada konsep-konsep yang lebih spesifik dan relevan dalam pengembangan berbasis web dan pengelolaan keuangan dalam konteks masjid.

2.2.1 Website

Menurut Kamus Besar Bahasa Indonesia (KBBI), arti kata web berarti sistem untuk mengakses, memanipulasi, dan mengunduh dokumen hipertaut

yang terdapat dalam komputer yang dihubungkan melalui internet; jejaring; jaringan.

Internet telah berkembang pesat dan World Wide Web (WWW) merupakan salah satu sumbernya. WWW memungkinkan informasi web disebarkan melalui pendekatan hypertext, di mana suatu teks pendek dapat menjadi acuan untuk membuka dokumen lain.

Sedangkan website memiliki definisi halaman halaman web yang memiliki hubungan dan ditempatkan pada suatu situs. Menurut (Noviana, 2022), website merupakan kumpulan halaman yang saling terhubung dan berisi informasi dalam bentuk teks, gambar, animasi, audio, dan video, yang dapat diakses melalui internet. Halaman-halaman ini dapat bersifat statis atau dinamis dan saling terhubung melalui hyperlink. Website dapat dibuat untuk keperluan pribadi, organisasi, atau perusahaan, dan dokumen-dokumen tersebut disimpan di server yang tersebar di berbagai belahan dunia, termasuk Indonesia, yang terhubung melalui jaringan internet.

2.2.2 Masjid

Dalam perspektif Islam, masjid dianggap sebagai rumah Allah di dunia, sebuah tempat suci yang menjadi pusat ibadah, pembelajaran agama, dan refleksi spiritual bagi umat Muslim (Rasyid et al., 2023). Selain berfungsi sebagai tempat ibadah, masjid juga memiliki berbagai kegiatan struktural yang memerlukan dana, mulai dari operasional hingga pengelolaan acara. Sebagai lembaga non-profit, masjid dilengkapi dengan struktur organisasi

yang mengelola berbagai aspek, termasuk pengelolaan keuangan, untuk memastikan kelangsungan dan efektivitas fungsinya dalam masyarakat.

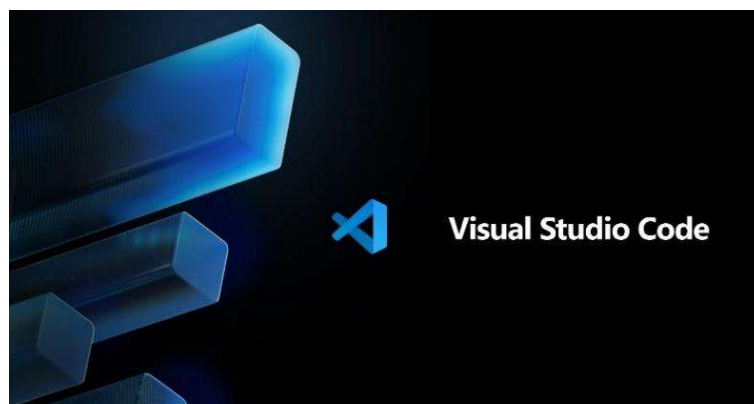
2.2.3 Manajemen Keuangan

Menurut (Fitriani, 2021) manajemen keuangan dapat didefinisikan sebagai serangkaian aktivitas yang mencakup perencanaan, penganggaran, pengawasan, dan pengelolaan dana secara efektif. Kegiatan ini melibatkan pengendalian alokasi sumber daya finansial, memastikan dana digunakan dengan tepat, serta mengoptimalkan cara penyimpanan dan pencarian dana. Fokus utamanya adalah bagaimana dana yang diperoleh dapat dikelola dan dialokasikan dengan efisien untuk mencapai tujuan yang telah ditetapkan.

2.3 Teknologi dan Perangkat Lunak

Dalam pembuatan dan pengembangan web keuangan masjid, teknologi dan perangkat lunak yang digunakan antara lain:

1. Visual Studio Code



Gambar 2. 2 Visual Studio Code

Sumber: <https://learn.microsoft.com/id-id/shows/visual-studio-code/>

Aplikasi pengembangan perangkat lunak Visual Studio Code

(VS Code) merupakan aplikasi yang populer dan gratis yang

dikembangkan oleh Microsoft. Aplikasi ini menawarkan berbagai fitur yang sangat membantu bagi para pengguna dalam mengembangkan kode dengan cepat dan mudah. Dapat diinstal di berbagai sistem operasi seperti Windows, MacOS, dan Linux, VS Code memiliki antarmuka pengguna yang intuitif dan dapat disesuaikan sesuai kebutuhan pengguna. Selain itu, aplikasi ini dilengkapi dengan banyak ekstensi dan plugin yang memungkinkan pengguna untuk menyesuaikan dan memperluas fungsionalitasnya

2. HTML

HTML (*Hypertext Markup Language*) merupakan bahasa pemrograman dasar yang menjadi fondasi pembentukan halaman web. HTML digunakan untuk membuat scripting sisi klien pada web yang memungkinkan informasi ditampilkan dalam berbagai bentuk seperti teks, grafik, dan multimedia. Selain itu, HTML juga digunakan untuk menghubungkan antar tampilan halaman web dengan hyperlink (Aminulloh et al., 2020)



Gambar 2. 3 HTML

Sumber: <https://en.wikipedia.org/wiki/HTML5>

Dalam pembuatan website, HTML sangat penting karena menentukan tata letak dan hierarki elemen-elemen dalam halaman web. Dengan HTML, pengembang web dapat menentukan bagaimana teks dan gambar ditampilkan di halaman web, termasuk jenis font, ukuran, warna, dan tata letak.

HTML terus mengalami perkembangan dan perbaruan dalam pengembangan web. HTML5, sebagai versi terbaru dari HTML, menawarkan banyak fitur baru seperti kemampuan multimedia, API browser, dan penggunaan semantik yang lebih baik untuk tag dan atribut. Hal ini menjadikan HTML5 sebagai salah satu standar dalam pengembangan web modern. Selain itu, HTML5 juga memberikan kemampuan bagi pengembang untuk membuat aplikasi web yang lebih kompleks dan interaktif

3. CSS



Gambar 2. 4 CSS

Sumber: https://en.m.wikipedia.org/wiki/File:CSS3_logo_and_wordmark.svg

CSS, atau *Cascading Style Sheets*, merupakan bahasa stylesheet yang digunakan untuk mempercantik dan mengatur tampilan visual dari halaman web. CSS memungkinkan pengembang web untuk memisahkan konten dan struktur dari tampilan halaman web, sehingga memudahkan pemeliharaan dan penggunaan ulang kode. Dalam pengembangan web, CSS digunakan untuk mengatur berbagai elemen tampilan visual seperti warna, font, layout, animasi, dan lainnya.

4. Javascript

JavaScript adalah bahasa pemrograman yang berfungsi di sisi klien (client-side), yang biasanya digunakan bersama dengan HTML dan CSS untuk menciptakan halaman web interaktif (Chandra & Hermansyah, 2024). Dengan JavaScript, pengembang dapat menambahkan berbagai elemen dinamis, seperti animasi, validasi form, serta pengelolaan konten yang diperbarui secara real-time tanpa memerlukan muatan ulang halaman. Pengguna dapat berinteraksi

langsung dengan elemen-elemen di halaman web, sementara HTML memberikan struktur dasar dan CSS menangani desain visual



Gambar 2. 5 JavaScript

Sumber: <https://www.freelogovectors.net/javascriptlogo/>

Secara keseluruhan, JavaScript merupakan komponen esensial dalam pembuatan situs web modern dan aplikasi web, menawarkan interaktivitas dan fungsionalitas tinggi yang tidak dapat diberikan oleh HTML dan CSS saja. Dengan fleksibilitasnya yang luas, JavaScript memungkinkan pengembang untuk membangun pengalaman pengguna yang lebih dinamis, interaktif, dan responsif.

5. MySQL

Aplikasi program untuk membuat database di server web, MySQL, banyak digunakan karena kemampuannya yang andal dalam mengelola data berkapasitas besar dan memuat data dengan cepat saat diperlukan.



Gambar 2. 6 MySQL

Sumber: <https://www.logo.wine/logo/MySQL>

MySQL dipilih secara luas karena kemampuannya untuk menangani volume data yang besar dan kecepatan pemrosesan saat mengambil data. MySQL memiliki beberapa fitur kunci seperti dukungan untuk banyak pengguna dan akses terhadap database yang bersifat remote, performa yang cepat, keamanan data yang baik, dukungan untuk penggunaan transaksi dan indeks, serta dukungan untuk database berukuran besar. MySQL juga memiliki dukungan yang baik dari komunitas pengguna dan tersedia banyak referensi, tutorial, dan tools yang dapat membantu pengguna dalam pengembangan dan manajemen basis data.

6. PHP

Dalam pengembangan web, PHP digunakan untuk mengolah data dari server dan menghasilkan output dalam format HTML yang dapat ditampilkan pada browser pengguna. Dengan PHP, pengembang dapat

membuat berbagai fitur dan fungsionalitas seperti formulir, login, pembayaran, dan banyak lagi.



Gambar 2. 7 PHP

Sumber: <https://id.m.wikipedia.org/wiki/Berkas:PHP-logo.svg>

PHP juga memiliki banyak framework dan library yang tersedia untuk membantu pengembang dalam membuat aplikasi web, seperti Laravel, CodeIgniter, dan Symfony. Framework dan library ini membantu dalam mempercepat pengembangan aplikasi web, meningkatkan keamanan, dan memperbaiki kualitas kode.

7. XAMPP



Gambar 2. 8 XAMPP

Sumber: https://www.stickpng.com/id/img/agama/ikon-logo-emoji/perusahaan-teknologi/logo-xampp#google_vignette

Fungsinya adalah untuk membantu pengembangan website secara lokal pada komputer pengguna. Salah satu fitur penting dari XAMPP adalah folder Htdoc, tempat untuk menyimpan berkas-berkas seperti berkas PHP, HTML, dan script lainnya yang akan dijalankan. Selain itu, terdapat juga fitur Phpmyadmin yang digunakan untuk mengelola basis data MySQL pada komputer pengguna. Untuk membukanya, cukup membuka browser dan mengetikkan alamat <http://localhost/phpmyadmin>, sehingga akan muncul halaman phpmyadmin. Selain itu, terdapat juga Kontrol Panel yang berfungsi untuk mengelola layanan (service) XAMPP, seperti menghentikan (stop) atau memulai (start) layanan tersebut.

2.4 Penelitian Terdahulu

Menelusuri penelitian-penelitian terdahulu yang memiliki kaitan dengan penelitian ini merupakan langkah penting. Penelitian yang relevan dapat memberikan dasar yang kuat serta mengungkap inovasi dalam penerapan teknologi untuk pengelolaan keuangan. Berikut ini adalah sejumlah penelitian terdahulu yang memiliki relevansi dengan studi ini:

1. Perancangan Aplikasi Pembukuan Menggunakan Metode Agile Scrum (Putra & Tanaem, 2022)

Aplikasi pembukuan berbasis Android dikembangkan menggunakan metode Agile dengan framework Scrum, yang dirancang untuk mengelola stok barang, transaksi, dan otomatisasi laporan penjualan. Selama proses

pengembangan, peran Product Owner, Scrum Master, dan Development Team menjadi fokus utama, dengan tahapan kunci meliputi Product Backlog, Sprint Planning, hingga Sprint Review. Implementasi aplikasi menunjukkan keberhasilan dalam memenuhi kebutuhan operasional. Pengujian menggunakan metode black box memastikan seluruh fitur berfungsi dengan optimal. Pendekatan Agile terbukti efektif dalam menghasilkan solusi digital yang fleksibel dan terstruktur untuk mendukung kebutuhan bisnis.

2. **Pengujian Implementasi Sistem Pengelolaan Keuangan Masjid** (Dan et al., 2020)

Penelitian ini menghasilkan sistem pengelolaan keuangan masjid berbasis web dan Android yang dirancang untuk mempermudah pencatatan dana masuk dan dana keluar. Pengujian dilakukan dengan metode black box testing, melibatkan tujuh pengguna, termasuk bendahara dan ketua masjid. Hasil pengujian menunjukkan rata-rata skor di atas 80%, dengan nilai tertinggi mencapai 85,21%, menunjukkan bahwa sistem berfungsi dengan baik dan integrasi antara platform web dan Android berjalan lancar. Namun, penelitian ini mengidentifikasi beberapa keterbatasan, seperti belum dilakukannya pengujian performa dan beta testing, serta merekomendasikan evaluasi jangka panjang untuk menjaga keoptimalan sistem.

3. **Implementasi Metode Agile Scrum Pada Sistem Informasi Akuntansi CV Tritama Inti Persada** (Amalia Arsyad & Sumardin, 2022)

Penelitian ini menghasilkan sistem informasi akuntansi berbasis web untuk perusahaan perdagangan pengadaan barang dan jasa yang sebelumnya menghadapi kendala keterlambatan pelaporan keuangan akibat pencatatan manual. Dengan menerapkan metode Agile Scrum dan pendekatan iteratif, sistem ini dirancang untuk mencakup fitur login, dashboard staf, pengelolaan data akun, pencatatan jurnal, buku besar, neraca saldo, serta laporan keuangan seperti laba/rugi, perubahan modal, dan neraca. Implementasi sistem berhasil menggantikan proses manual dengan otomatisasi, mempercepat pelaporan pajak, dan mencapai efektivitas sebesar 86% berdasarkan hasil kuesioner. Namun, aspek teknis dan keamanan sistem masih memerlukan peningkatan lebih lanjut.

4. **Pencatatan Transaksi Keuangan Berbasis *Web* Menggunakan Model Agile Scrum Development** (Wahyuningroem, 2024)

Penelitian Wahyuningroem menghasilkan sistem pencatatan keuangan berbasis metode Agile Scrum untuk memfasilitasi pengguna dalam mencatat dan memonitor transaksi keuangan pribadi dengan lebih efisien dan akurat. Pendekatan Agile Scrum diterapkan dengan melibatkan partisipasi aktif pengguna selama proses pengembangan. Sistem ini menawarkan fitur utama seperti dashboard laporan keuangan, manajemen transaksi (pemasukan dan pengeluaran), dompet virtual, pengelompokan kategori transaksi, serta tips untuk pengelolaan keuangan.

Hasilnya menunjukkan bahwa sistem bekerja dengan baik dan memenuhi ekspektasi pengguna, membantu meningkatkan efisiensi pencatatan sekaligus memberikan visualisasi yang jelas terhadap kondisi keuangan. Penelitian ini berkontribusi dengan menghadirkan sistem yang mudah digunakan dan relevan dengan kebutuhan pengguna, dicapai melalui penerapan metode Agile Scrum dan keterlibatan langsung pengguna untuk memastikan sistem sesuai dengan kebutuhan nyata.

5. **Rancang Bangun Sistem Informasi Keuangan Tingkat RT Menggunakan Metode Agile** (Fauzan et al., 2024)

Penelitian ini merancang sistem informasi keuangan berbasis web untuk mendukung pengelolaan keuangan di tingkat RT secara lebih efisien. Sistem ini dirancang untuk mempermudah pencatatan transaksi dan pembuatan laporan keuangan. Pendekatan yang digunakan adalah metode Agile dengan tahapan utama seperti *Product Backlog* dan *Sprint Backlog*.

Sistem ini dilengkapi dengan berbagai fitur, termasuk halaman login admin, dashboard, manajemen pendapatan dan pengeluaran, laporan keuangan, serta pengelolaan pengguna. Teknologi yang digunakan meliputi PHP dengan framework Laravel dan MySQL sebagai basis data. Tahapan Agile yang diimplementasikan mencakup proses pengembangan, pengujian, hingga *User Acceptance Test* (UAT).

Hasilnya, sistem berhasil memenuhi kebutuhan pengguna dan mendapatkan tanggapan positif. Penelitian ini berkontribusi secara

signifikan dalam meningkatkan efisiensi dan akurasi pengelolaan keuangan di tingkat RT.

6. Perancangan Aplikasi Pencatatan Laporan Keuangan Dengan Menggunakan Metode Agile Development Scrum (Sabila et al., 2021)

Penelitian ini berfokus pada permasalahan pencatatan keuangan di UMKM, khususnya di Kedai Intikopi, yang masih menggunakan metode manual. Pencatatan manual ini sering menimbulkan ketidakkonsistenan dan kesalahan dalam transaksi keuangan. Untuk mengatasi hal ini, penelitian ini mengadopsi metode Agile Development Scrum yang melibatkan tahapan analisis proses bisnis, identifikasi kebutuhan pengguna, desain sistem, perancangan program, dan pengujian aplikasi.

Aplikasi yang dikembangkan berhasil menggantikan pencatatan manual dengan sistem berbasis web yang lebih efisien. Data keuangan disimpan dalam database MySQL, memungkinkan pengelolaan keuangan yang lebih terstruktur dan terkendali. Hasil pengujian menunjukkan bahwa aplikasi ini berfungsi dengan baik, memenuhi standar fungsionalitas, keandalan, dan kegunaan, serta siap diterapkan secara lebih luas di UMKM lainnya.

7. Systematic Literature Review Analisis Metode Agile dalam Pengembangan Aplikasi Mobile (Yusril et al., 2021)

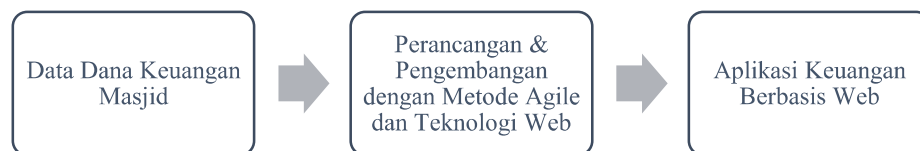
Penelitian ini melakukan Systematic Literature Review (SLR) untuk menganalisis penerapan metode agile dalam pengembangan aplikasi

mobile. Tujuan penelitian ini adalah untuk mengidentifikasi tren model agile yang digunakan serta bidang aplikasi yang mengimplementasikan metode tersebut. Dengan menganalisis 44 literatur yang diterbitkan antara tahun 2015 hingga 2020, penelitian ini menemukan bahwa model Extreme Programming (XP) adalah yang paling banyak digunakan, mencapai 41%. Sementara itu, bidang yang paling banyak mengadopsi metode agile adalah bidang produktivitas, dengan persentase 23%.

Penelitian ini memberikan wawasan yang komprehensif mengenai penggunaan metode agile dalam pengembangan aplikasi mobile, yang dapat membantu pengembang dan peneliti dalam memilih model agile yang tepat sesuai dengan kebutuhan proyek mereka..

2.5 Kerangka Pemikiran

Pemanfaatan teknologi berbasis web dalam bentuk aplikasi keuangan dapat meningkatkan efisiensi dan mempermudah pengguna dalam mengelola keuangan masjid. Dengan adanya aplikasi berbasis web ini, pengelolaan dana masjid dapat dilakukan dengan lebih terstruktur dan tepat sasaran.



Gambar 2. 9 Kerangka Pemikiran

Sumber: Data Penelitian, 2024

Penelitian ini menjadikan data keuangan masjid sebagai input, mengadopsi metode Agile serta proses perancangan menggunakan bahasa

pemrograman sebagai langkah pengembangan, dan menghasilkan aplikasi keuangan berbasis web sebagai output.

Penelitian ini berfokus pada pengembangan aplikasi keuangan berbasis web yang dirancang untuk mendukung pengelolaan dana masjid secara lebih efisien. Aplikasi berbasis web ini akan dilengkapi dengan fitur-fitur yang memungkinkan pengguna memantau dan mengelola keuangan masjid secara real-time. Melalui aplikasi ini, pengelolaan dana diharapkan menjadi lebih terstruktur, meminimalkan risiko kesalahan, serta meningkatkan akuntabilitas. Kerangka pemikiran penelitian ini menitikberatkan pada pengembangan solusi yang mampu memenuhi kebutuhan pengelolaan keuangan masjid.