

BAB II

TINJAUAN PUSTAKA

2.1 Perancangan

Perancangan adalah suatu proses karakterisasi yang merancang penggambaran dan penggabungan komponen-komponen bebas secara individual ke dalam suatu unit fungsional yang lengkap [1]. Perancangan dapat diartikan sebagai menggambarkan, mengatur atau menguraikan beberapa komponen terpisah yang diorganisasikan dalam satu kesatuan yang utuh dan bekerja [2]. Proses ini mencakup beberapa tahap, mulai dari mengidentifikasi tujuan, melakukan analisis kebutuhan, mengumpulkan data, hingga menciptakan konsep dan desain yang diharapkan dapat memberikan hasil yang efektif.

2.2 Sistem Informasi

Sistem informasi, yang sering disingkat SI, merupakan suatu sistem yang mengintegrasikan aktivitas manusia dengan penggunaan teknologi. Sistem ini digunakan untuk mengumpulkan, menyimpan, mengelola, dan menyebarkan informasi yang relevan untuk suatu organisasi [3]. Sistem informasi adalah kumpulan dari komponen komponen komponen yang saling berhubungan melakukan pengolahan data menjadi informasi sehingga informasi tersebut dapat digunakan untuk tujuan tertentu [4].

2.3 Inventory

Inventory adalah stok barang atau material yang dimiliki oleh perusahaan untuk digunakan dalam produksi atau dijual kepada konsumen. Ini mencakup bahan baku, barang dalam proses, dan produk jadi, yang berperan penting dalam menjaga kelancaran operasi bisnis serta memenuhi permintaan pasar [5].

2.4 Agile

Metode *Agile* adalah pendekatan dalam pengembangan perangkat lunak yang menekankan pada fleksibilitas, kolaborasi, dan pengembangan bertahap. Dengan menggunakan metode ini, perubahan persyaratan dapat diterima sepanjang proses, di mana tim bekerja dalam siklus pendek yang disebut sprint untuk menciptakan perangkat lunak yang terus dapat diuji dan disesuaikan berdasarkan umpan balik pengguna [6].



Gambar 2. 1 *Agile Method*

1. Perencanaan sistem adalah salah satu tahap awal dalam proses kategorisasi yang memerlukan langkah-langkah tertentu. Kegiatan yang dilakukan meliputi wawancara, observasi, dan peninjauan arsip dokumen yang tersedia.
2. Implementasi melibatkan pembuatan menu untuk pelanggan yang merupakan hasil dari rancangan sistem baru yang telah disetujui, menggunakan bahasa pemrograman tertentu. Pada tahap ini, proses pengkodean dan desain *web* telah diselesaikan.
3. Pengujian adalah aspek penting dari suatu sistem. Pada tahap ini, pengembang melakukan pengembangan sistem melalui pengkodean dan selanjutnya menguji perangkat lunak yang telah dibuat.
4. Dokumentasi adalah proses penting yang mencakup pencatatan setiap tahap dari sistem yang telah dibangun. Dalam langkah ini, hasil dari

pengujian didokumentasikan untuk memudahkan pemeliharaan di masa yang akan datang.

5. *Deployment* atau penyebaran melibatkan pengembang dalam mendistribusikan informasi. Pada tahap ini, sistem kembali diuji untuk memastikan apakah sistem tersebut memenuhi semua persyaratan yang ditetapkan.
6. Pemeliharaan atau maintenance adalah upaya untuk menjaga sistem dalam kondisi terbaiknya. Pada tahap ini, proses pemeliharaan dilakukan secara berkala agar perangkat lunak tetap berfungsi dengan baik dan kualitasnya tetap terjaga.

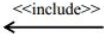
2.5 Unified Modeling Language (UML)

Unified Modeling Language (UML) yang dikenal sebagai UML digunakan dalam rekayasa, menentukan, membangun, dan bukti sistem pengembangan *software* berbasis objek.

2.5.1 Use Case Diagram

Unified Modeling Language (UML) merupakan bahasa spesifikasi standar yang digunakan untuk mendokumentasikan, mendefinisikan, dan mengembangkan perangkat lunak. Use case membantu dalam mengidentifikasi dan mendeskripsikan secara rinci interaksi antara sistem dan pengguna [7]. Beberapa simbol yang digunakan dalam use case antara lain:

Tabel 2. 1 Simbol-Simbol *Use Case Diagram*

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

2.5.2 *Activity Diagram*

Activity Diagram adalah representasi visual dari serangkaian latihan yang berlangsung dalam suatu kerangka yang memberikan makna pada respons sistem. *Activity Diagram* berfungsi untuk merepresentasikan urutan langkah atau kegiatan yang perlu diselesaikan oleh elemen dalam kerangka kerja untuk mencapai tujuan tertentu.

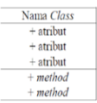
Tabel 2. 2 Simbol- simbol *Activity Diagram*

No	Gambar	Nama	Keterangan
1		Aktivitas	Mendemonstrasikan interaksi antara setiap kelas antarmuka.
2		Simpul Awal	Simpul yang memulai proses aktivitas.
3		Simpul Akhir Aktivitas	Simpul yang mengakhiri proses aktivitas.
4		Simpul Garpu	Suatu bagian yang terpecah menjadi beberapa bagian.

2.5.3 Class Diagram

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. *Class* memiliki apa yang disebut atribut dan metode atau operasi [8].

Tabel 2. 3 Simbol-simbol *Class Diagram*

Simbol	Nama	Deskripsi
	Class	Himpunan objek-objek dari berbagai atribut yang memiliki operasi yang sama.
	Association	Relasi antar kelas dengan makna umum dan biasanya disertai multiplicity.
	Directed Association	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas lain.
	Aggregation	Mengindikasikan keseluruhan bagian relationship disebut sebagai relasi.
	Composition	Relasi Composition terhadap class tempat dia bergantung.
	Dependency	Menunjukkan operasi pada suatu class yang menggunakan class yang lain.

2.6 Bahasa Pemrograman dan Perangkat Lunak Pendukung

2.6.1 Pengetian Basis Data (*Database*)

Basis data adalah kumpulan informasi logis yang terintegrasi dengan baik, saling terhubung untuk memudahkan dan mempercepat proses pengambilan serta penggunaan kembali data. Basis data memiliki peranan penting sebagai fondasi penyediaan informasi bagi pengguna, dan menjadi elemen krusial dalam sistem informasi.

2.6.2 MySQL

MySQL merupakan server yang dirancang untuk pemantauan dan pengumpulan data secara instan, berbasis informasi. MySQL mampu mengelola volume data yang besar dan dapat diakses oleh banyak pengguna secara bersamaan. Selain itu, MySQL juga berfungsi sebagai perangkat lunak yang memudahkan pengelolaan dan pembuatan basis data itu sendiri [9].

2.6.3 *Website*

Website adalah suatu *website* yang memiliki beberapa gabungan untuk memperlihatkan bagian data teks, data gambar diam atau bergerak, animasi, suara, video dan berbagai informasi yang mudah dipahami [10]. Saat ini *web* merupakan suatu sumber data yang paling banyak digunakan untuk mencari data-data yang dibutuhkan karena ada jaringan *web* terdapat *web crawler* yang dapat dimanfaatkan oleh klien untuk mencari halaman-halaman yang dibutuhkannya.

2.6.4 HTML

Bahasa standar dalam menampilkan hasil pada *website* merupakan HTML (*Hypertext Markup Language*). Ini dapat digunakan untuk menata dan merancang konten halaman situs *web*, membuat tabel pada halaman situs *web*, mempublikasikan halaman situs *web* secara online, membuat formulir yang dapat diisi *input* dan menampilkan area gambar di *browser* [11].

2.6.5 XAMPP

Cross-Platform Apache MySQL PHP dan Perl adalah singkatan dari XAMPP. XAMPP adalah perangkat lunak gratis yang mengintegrasikan berbagai aplikasi fungsional dari sistem operasi yang berbeda. Biasanya, XAMPP mengandalkan beberapa komponen perangkat lunak utama yang sangat penting dalam pengembangan *web*, termasuk server *web* Apache, database MySQL, serta *interpreter* bahasa pemrograman seperti PHP dan Perl.

2.6.6 PHP

PHP, yang merupakan singkatan dari "*Praprosesor Hiperteks*," adalah bahasa pemrograman server yang digunakan untuk menciptakan aplikasi web yang interaktif dan dinamis. Diciptakan oleh Rasmus Lerdorf pada tahun 1994, PHP telah berkembang menjadi salah satu bahasa pemrograman paling terkenal dalam dunia pengembangan *web*. Bahasa ini dapat dikombinasikan dengan HTML untuk memproses database dan konten *web*, sehingga mampu menghasilkan situs yang dinamis.[12].

2.6.7 CSS

Cascading Style Sheet atau yang lebih dikenal sebagai CSS, adalah teknologi *web* yang digunakan untuk mengatur elemen-elemen HTML melalui berbagai properti agar tampil dalam gaya yang diinginkan [13]. CSS mencakup atribut seperti warna, ukuran, font, dan tata letak dari berbagai elemen di halaman *web*, memberikan kebebasan bagi pengembang untuk menyusun tampilan yang menarik sesuai kebutuhan.

2.6.8 Visual Studio Code

Visual Studio Code merupakan editor kode sumber berbasis *desktop*, ringan, dan kuat untuk *Windows*, *MacOS*, dan *Linux* yang mendukung JavaScript, TypeScript, dan Node.JS serta berbagai bahasa dan runtime lainnya melalui ekosistem ekstensinya yang luas. (*Visual Studio Code*, 2022).

2.7 Figma

Figma adalah aplikasi desain berbasis *cloud* dan alat *prototyping* untuk proyek digital. Figma dibuat untuk dapat membantu para penggunanya agar bisa berkolaborasi dalam proyek dan bekerja dalam bentuk tim [14]. Figma merupakan salah satu *design tool* yang biasanya digunakan untuk membuat tampilan aplikasi *mobile*, dekstop, maupun *website* seperti membuat presentasi dan *prototype*.

2.8 Draw.io

Draw.io adalah sebuah *website* yang didesain khusus untuk menggambarkan diagram secara *online*, dan draw.io tidak memerlukan instalasi apapun. Draw.io merupakan tumpukan teknologi sumber terbuka untuk membangun aplikasi diagram. Draw.io digunakan untuk membuat *user flow* dan diagram lainnya.

2.9 Scrum

2.9.1 Pengertian Scrum

Scrum merupakan metode perencanaan berulang untuk mengidentifikasi sebab yang dapat diantisipasi kemudian mengendalikan kerugian yang diakibatkan adanya kesalahan rencana awal [15]. *Scrum* yang pertama kali dikembangkan oleh Jeff Sutherland dan Ken Schwaber pada awal tahun 1990an yang kini menjadi salah satu metode yang umum dimanfaatkan dalam mengelola proyek pengembangan *software* yang kompleks.

2.9.2 Fungsi Scrum

Scrum berfungsi sebagai penyedia sistem yang efektif untuk mengelola dan mengembangkan item pemrograman dengan cara yang fleksibel dan kooperatif. Dalam struktur ini terdapat 3 kemampuan mendasar, sebagai berikut:

- a. Pemilik *Product* (*Product Owner*)

Product owner adalah orang yang akan bertanggung jawab atas keberhasilan produk dan menentukan kebutuhan serta pengelolaan fitur.

b. *Scrum Master*

Scrum master adalah orang yang bertugas dalam memastikan jika scrum telah diterapkan dengan benar tanpa adanya hambatan yang menghalangi kemajuan tim.

c. Tim Pengembang (*Development*)

Development merupakan orang yang bertanggung jawab atas desain, pengembangan dan pengujian produk.

2.9.3 Tahapan *Scrum*

Sebelum mengetahui fase-fase teknik *scrum*, rekan tim sebaiknya mengetahui dulu isi strategi scrum. Berikut adalah artefak teknik *scrum*, yaitu:

a. *Product Backlog*

Product Backlog adalah proses pengumpulan kebutuhan melalui daftar prioritas yang mencerminkan kebutuhan sistem. Pada fase ini, penulis melakukan analisis terhadap sistem yang sedang berjalan di dalam program tersebut [16]. Daftar lengkap kebutuhan yang dihasilkan dari pemangku kepentingan menjadi acuan dalam *Product Backlog* untuk item-item yang sedang dikembangkan.

Tabel 2. 4 Tahapan *Product Backlog*

Sumber : [17]

No	Tahapan <i>Product Backlog</i>
1	<i>Product owner</i> membuat daftar prioritas dari semua fitur, perbaikan dan perubahan yang diperlukan dalam pengembangan produk.
2	<i>Product backlog</i> diperbarui secara teratur berdasarkan umpan balik dan perubahan kebutuhan.

3	<i>Product backlog</i> bekerja secara terus menerus dengan tim pengembangan untuk memastikan pemahaman yang sama tentang kebutuhan produk dan untuk membahas item dalam <i>product backlog</i> .
---	--

b. *Sprint*

Sprint backlog adalah proses pemenuhan kebutuhan sesuai yang direncanakan pada *product backlog* dan *sprint planning* yang telah ditentukan. Dari hasil analisis penulis yang telah didapat maka kebutuhan sistem informasi dapat diketahui [18].

Tabel 2. 5 Tahapan *Sprint Backlog*

Sumber : [17]

No	Tahapan <i>Sprint Backlog</i>
1	Produk owner dan tim pengembangan bekerja sama untuk memilih item dari <i>product backlog</i> untuk <i>sprint backlog</i> .
2	Tim pengembangan memecah setiap item menjadi tugas-tugas yang lebih kecil setelah memilihnya dan dapat dikerjakan dalam waktu yang singkat.
3	Setelah estimasi selesai, tim pengembangan bekerja sama untuk menetapkan prioritas pada tugas-tugas dalam <i>sprint backlog</i> .

c. *Increment*

Increment adalah hasil kerja yang sedang berlangsung, di mana berbagai *user story* yang selaras dikumpulkan untuk menghasilkan produk yang berfungsi, meskipun belum sepenuhnya selesai. Bagi mitra, *increment* berfungsi sebagai indikator kemajuan tim dalam proyek.

Tabel 2. 6 Tahapan Increment

Sumber : [17]

No	Tahapan <i>Increment</i>
1	Tim pengembangan bekerja selama sprint untuk menghasilkan incremen, yang merupakan hasil <i>tangible</i> dari pekerjaan mereka. Mereka merancang, mengembangkan, dan menguji fitur fitur yang telah dipilih dari produk <i>backlog</i> .
2	Tim pengembangan melakukan pengujian kualitas terhadap increment, untuk memastikan bahwa fitur fitur baru berfungsi sesuai dengan harapan dan tidak menimbulkan masalah lain dalam produk.
3	Setelah instrument selesai, tim pengembangan melakukan demo kepada pemangku kepentingan untuk menunjukkan versi terbaru yang telah ditambahkan.

Berdasarkan artefak yang sudah diuraikan di atas, maka teknik *scrum* dibedakan dalam beberapa fase sebagai berikut:

a. *Sprint Planning*

Metode *Scrum* berisi proyek penelitian dikembangkan melalui beberapa *sprint*. Pada fase ini, semua pemangku kepentingan bekerja sama untuk menetapkan tujuan *sprint*, menyusun daftar tugas yang perlu diselesaikan, serta memprioritaskan pekerjaan dalam *sprint* tersebut sesuai dengan kapasitas pengembangan yang ada [19].

b. *Daily Scrum*

Daily scrum yang juga dikenal sebagai *Standups* atau *Daily Standups*, adalah pertemuan singkat yang diadakan setiap pagi selama sprint untuk membahas rencana kerja harian tim, kendala yang dihadapi, serta solusinya. Pada fase ini,

pengembang harus menyajikan hasil akhir dari aktivitas yang dilakukan selama sprint [19].

c. *Sprint Review*

Pada tahap ini, tim proyek mengadakan rapat untuk mempresentasikan hasil akhir sprint kepada para pemangku kepentingan. Hasil yang diperoleh selama fase *scrum* kemudian disampaikan kepada klien.

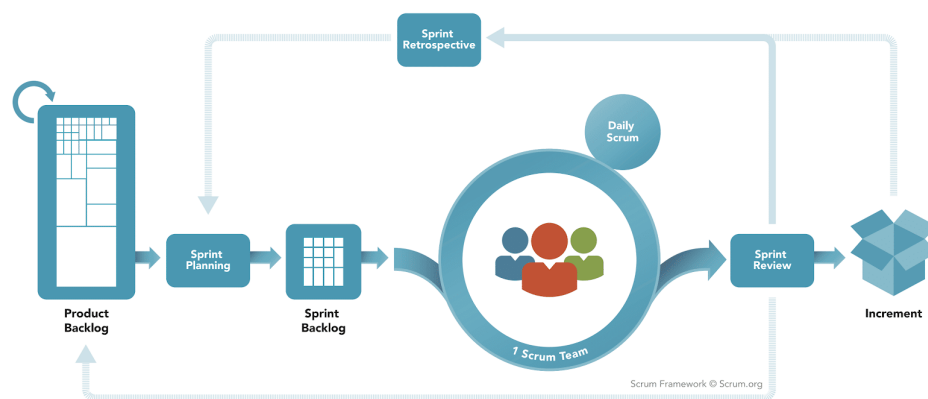
d. *Retrospective*

Dalam tahap ini tim merenungkan kembali pekerjaan yang telah dilakukan dengan menyajikan upaya kolaboratif yang dihasilkan. Selama rapat ini, kelompok memutuskan untuk melakukan perbaikan untuk iterasi selanjutnya

e. *Refinement*

Pada tahap *refinement* tim berjumpa untuk membicarakan dan fokus pada kebutuhan baru. Kebutuhan-kebutuhan ini selanjutnya disatukan agar dapat membuat produk dan *sprint backlog*.

Saat sudah mengetahui tahapan-tahapan dan keunikan dalam teknik *scrum*, selanjutnya melakukan susunan langkah-langkah pengembangan strategi scan dan dapat dilihat pada gambar di bawah ini.



Gambar 2. 2 Tahapan *Scrum*

2.10 BlackBox Testing

Black-box testing adalah metode pengujian perangkat lunak yang menitikberatkan pada fungsionalitas sistem, tanpa perlu menganalisis kode internalnya. Dalam pendekatan ini, penguji hanya fokus pada pengujian input dan output, untuk memastikan bahwa perangkat lunak beroperasi sesuai dengan spesifikasi yang telah ditentukan, tanpa terlibat dalam struktur atau logika kode. [20] .

2.11 Penelitian Terkait

Penelitian ini merupakan rangkuman dari berbagai penelitian sebelumnya yang memiliki relevansi dengan topik yang sedang dibahas. Penelitian-penelitian terdahulu yang telah dikumpulkan ini berperan sebagai acuan dalam pengembangan metode untuk memecahkan masalah yang dihadapi. Untuk melakukan pengembangan metode penelitian, dilakukan studi pustaka yang mendalam. Berikut ini adalah kumpulan hasil penelitian sebelumnya:

Tabel 2. 7 Penelitian Terdahulu

No	Penulis	Judul	Metode Penelitian	Hasil
1.	Ruhamah, Andinislam (2024)	Perancangan Sistem Informasi Inventorybarang Pada Puskesmas Wara Berbasis Website	<i>Waterfall</i>	Hasil yang didapat dalam penelitian ini adalah sistem inventori barang yang menyediakan form cetak laporan data barang, dan juga cetak laporan data barang masuk dan keluar. Penelitian ini menggunakan pendekatan kuantitatif

				dengan data primer dan menerapkan model pengembangan metode <i>Waterfall</i> .
2.	Andi, Nurul (2023)	Sistem Informasi Persediaan Barang Berbasis <i>Web</i> Pada Toko Throve Store Soppeng	SDLC	Penelitian ini menghasilkan sistem informasi persediaan barang berbasis web pada toko Throve Store Soppeng. Dalam merancang sistem tersebut peneliti menggunakan metode SDLC.
3.	Dzikri Febriansyah (2023)	Sistem Inventori Berbasis <i>Web</i> Dengan Metode <i>Agile Scrum</i> Pada Arta Jaya Teknik	<i>Agile Scrum</i>	Penelitian ini menghasilkan sistem inventori barang yang saling integrasi antara data barang masuk dan data barang keluar menggunakan metode <i>Agile Scrum</i> . Berdasarkan hasil uji BlackBox yang telah dilakukan didapatkan bahwa sistem inventori yang dibangun dapat mengakomodasi proses bisnis dalam

				bidang inventori perusahaan.
4.	Moch, Ratih, Lilia (2022)	Perancangan Sistem Stok Barang Dengan Metode FIFO	FIFO	Penelitian ini menghasilkan sistem informasi pencatatan persediaan barang menggunakan metode FIFO untuk mengelola barang dagangannya. Sehingga memudahkan toko Bu Kurun dalam mengatur pencatatan stok barangnya.
5.	Sandi, Sulton, Muhammad Alfaqih (2022)	Penggunaan Metode Scrum Dalam Membentuk Sistem Informasi Penyimpanan Gudang Berbasis Web	<i>Scrum</i>	Penelitian ini menghasilkan sistem informasi penyimpanan Gudang berbasis <i>web</i> dengan metode <i>scrum</i> , yang dapat menyelesaikan suatu permasalahan dalam perubahan <i>requirement</i> . .