

BAB 2

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1. Tinjauan Pustaka

Berdasarkan penelitian terdahulu (Widjijono, 2017; Yusuf dkk, 2021; Prasetio & Andika, 2022; Sitokdana & Senduk, 2022; Sutejo & Tanaamah, 2022) terdapat permasalahan yang sama terkait stok barang aktual di gudang yang tidak sesuai dengan dokumen gudang. Metode yang digunakan adalah SDLC (*System Development Life Cycle*) model *waterfall*. Menurut Prasetio & Andika (2022), dengan solusi membuat sistem informasi berbasis web menggunakan *tools use case diagram* dan *activity diagram* dapat menurunkan kesalahan pencatatan. Serupa dengan Sitokdana & Senduk (2022) meneliti studi kasus slingbag di Salatiga dan memberikan solusi pembuatan sistem informasi berbasis website dengan menggunakan *tools use case diagram* dan *class diagram*. Di samping itu Yusuf dkk (2021) memberikan solusi membuat program aplikasi dengan VBA (*Visual Basic for Application*) Excel agar memudahkan pencarian data barang dan hasil persediaan barang menjadi akurat dengan menggunakan *tools diagram konteks* dan *entity relationship diagram*. Di sisi lain Sutejo & Tanaamah (2022), Agusdino & Basry (2024), Adriant dkk (2024), Widjijono (2017) memberikan solusi perancangan sistem informasi dengan aplikasi namun terdapat perbedaan *tools* yang digunakan.

Di samping itu Novianti & Sari (2022) menggunakan Metode FAST (*Framework For The Applications Of System Thinking*) dengan *tools diagram konteks* dan DFD level satu. Masalah dan Metode yang digunakan serupa dengan penelitian Yasin dan Sari (2020) yaitu solusi perancangan sistem informasi pergudangan menggunakan VBA Excel. Hasil dari kedua penelitian menunjukkan bahwa perancangan sistem informasi memudahkan karyawan dalam pekerjaan dan dapat mengetahui data transaksi barang secara langsung dengan jelas. Di sisi lain penelitian Ikawati & Arinal (2021) menggunakan metode RAD (*Rapid Application Development*) untuk membuat sistem informasi dalam waktu yang singkat. Hasil penelitian menunjukkan bahwa sistem persediaan barang membantu dan mempercepat kerja dalam mengelola data dengan tepat dan akurat.

Rinaldi dkk (2021) dan Gairwyn & Suryadi (2025) menggunakan Metode PDCA (*Plan, Do, Check, Act*) untuk menyelesaikan masalah kesalahan pencatatan karena metode tersebut sederhana dan dapat dilakukan untuk perbaikan berkelanjutan. Annisa dkk (2021) menggunakan *tools* DMAIC (*Define, Measure, Analyze, Improve, Control*) dari Metode *Six Sigma* untuk menyelesaikan masalah yang serupa. Metode ini digunakan karena memiliki proses yang terstruktur dalam melakukan perbaikan dan membantu mengidentifikasi akar masalah dan dapat menentukan langkah *improvement*. Di sisi lain Wijaya (2023) menggunakan Metode Kaizen membantu dalam perbaikan berkelanjutan yang mendorong seluruh karyawan untuk berpartisipasi meningkatkan proses kerja menjadi lebih baik.

Perbedaan penelitian terdahulu dengan tugas akhir ini terdapat pada objek permasalahan yaitu pencatatan bahan baku yang diterima dalam satuan meter, namun setelah melalui proses produksi, terjadi konversi satuan menjadi unit produk jadi (pasang). Kondisi tersebut cenderung menimbulkan selisih data stok bahan baku aktual dengan dokumen pencatatan.

Ringkasan seluruh hasil tinjauan pustaka disajikan pada Tabel 2.1.

Tabel 2.1. Ringkasan Tinjauan Pustaka

Peneliti	Objek Penelitian	Permasalahan	Metode	Hasil penelitian
Prasetio & Andika (2022)	Toko Amelia Grosir Fashion	Ketidaksesuaian stok material gudang dengan dokumen gudang	SDLC (System Development Life Cycle) model <i>waterfall</i>	Sistem informasi yang dibuat berhasil mengurangi resiko data hilang karena penyimpanan data barang tersimpan dalam <i>database</i> dan mudah dalam melakukan <i>restock</i> barang
Sitokdana & Senduk (2022)	Toko Slingbang Salatiga	Terjadi kesalahan dalam memasukkan data dan terdapat perbedaan stok antara barang dari gudang ke toko	SDLC (System Development Life Cycle) model <i>waterfall</i>	Sistem <i>website</i> yang dibuat mempermudah proses pencatatan di gudang dan lebih mudah untuk mengetahui stok barang yang tersedia di gudang
Yusuf dkk (2021)	CV Fokus Etania Zashika Karawang	Terjadi kesalahan dalam memasukkan data dan pembuatan laporan	SDLC (System Development Life Cycle) model <i>waterfall</i>	Sistem yang dirancang membantu dalam proses input data, pencarian data, dan menyajikan laporan persediaan barang
Sutejo & Tanaamah (2022)	Toko Sutejo	Pendataan barang yang tidak sesuai dengan stok aktual	SDLC (System Development Life Cycle) model <i>waterfall</i>	Sistem yang dirancang berhasil mengurangi kesalahan pencatatan barang

Tabel 2.1. Lanjutan

Peneliti	Objek Penelitian	Permasalahan	Metode	Hasil penelitian
Agusdino & Basry (2024)	CV. Sejati Steel Truss	Terdapat kesalahan pencatatan karena banyak stok barang yang keluar dan masuk	SDLC (System Development Life Cycle) model <i>waterfall</i>	Hasil perancangan sistem dapat membantu pengelolaan barang gudang
Adriant dkk (2024)	Toko Samiaji	Jumlah data stok yang dicatat tidak sesuai dengan stok aktual	SDLC (System Development Life Cycle) model <i>waterfall</i>	Sistem yang dirancang memudahkan untuk memantau dan mengolah data stok barang
Widjijono & Wehartaty (2017)	PT Lomax	Kesalahan pencatatan yang membuat jumlah stok barang berbeda	SDLC (System Development Life Cycle) model <i>waterfall</i>	Hasil sistem yang dirancang dapat menghasilkan informasi yang akurat dalam pengolahan stok barang
Novianti & Sari (2022)	PT. Meidoh Indonesia	Kesalahan pencatatan <i>input</i> data di buku	FAST (<i>Framework For The Applications Of System Thinking</i>)	Hasil sistem yang dirancang dapat memudahkan operator dalam mengontrol stok material

Tabel 2.1. Lanjutan

Peneliti	Objek Penelitian	Permasalahan	Metode	Hasil penelitian
Yasin & Sari (2020)	PT Samcon	Kesalahan input data dan tidak mengetahui stok barang yang tersedia	FAST (<i>Framework For The Applications Of System Thinking</i>)	Sistem yang dirancang membantu proses <i>input</i> dan <i>output</i> barang dapat dilihat dengan jelas
Ikawati & Arinal (2021)	PT Agree Progress International	Terjadinya kesalahan dalam melakukan <i>input</i> data karena operator kurang teliti	RAD (<i>Rapid Application Development</i>)	Hasil sistem yang dirancang dapat mengurangi kesalahan saat melakukan <i>input</i> data dan mempermudah pengelolaan data
Rinaldi dkk (2021)	PT XYZ	Terdapat selisih jumlah persediaan barang di gudang dengan data jumlah persediaan yang tercatat	PDCA (<i>Plan, Do, Check, Act</i>)	Sistem yang diterapkan berhasil mengurangi kesalahan pencatatan
Nursyanti & Partisia (2024)	PT X	Terdapat selisih stok barang antara data sistem dan jumlah aktual di gudang	<i>Root Cause Analysis</i>	Dapat berjalan dengan baik sehingga dapat mengurangi kesalahan pencatatan

Tabel 2.1. Lanjutan

Peneliti	Objek Penelitian	Permasalahan	Metode	Hasil penelitian
Gairwyn & Suryadi (2025)	PT X	Adanya selisih stok barang dengan stok aktual	PDCA (<i>Plan, Do, Check, Act</i>) dan kaizen	Terjadi keberhasilan dalam meningkatkan keakuratan data stok
Annisa dkk (2021)	PT IMC Tekno	Ketidaksesuaian persediaan barang secara fisik dengan laporan pencatatan stok	DMAIC (<i>Define, Measure, Analyze, Improve, Control</i>)	Melakukan pengontrolan lebih lanjut dalam pencatatan barang agar mengurangi kesalahan pencatatan
Wijaya (2023)	PT Indah Prakasa Sentosa	Adanya selisih atau kehilangan data stok barang karena pekerja lupa mencatat	Kaizen, PDCA, dan 5S	Lebih menerapkan kedisiplinan pekerja untuk melakukan pencatatan

2.2. Dasar Teori

Berikut adalah penjabaran mengenai teori-teori yang digunakan berdasarkan mata kuliah pada penelitian ini.

- a. Metode SDLC (*System Development Life Cycle*) model *Waterfall* pada mata kuliah Proses Bisnis dan Sistem Informasi dengan kode STIN40502.
- b. Metode PDCA (*Plan, Do, Check, Act*) pada mata kuliah Manajemen Pergudangan dengan kode STIN76602.

2.2.1. Interelation Diagram

Interelation Diagram merupakan diagram yang berfungsi untuk mengidentifikasi akar penyebab antara satu faktor dengan faktor lainnya dan menentukan faktor dominan yang paling berpengaruh pada suatu permasalahan. Langkah-langkah pembuatan diagram interelasi menurut Andersen & Fagerhaug (2006) sebagai berikut:

- a. Menentukan faktor-faktor yang akan dianalisis yang saling berhubungan dan memberi keterangan yang jelas dan singkat.
- b. Menuliskan faktor-faktor tersebut yang saling berhubungan dalam bentuk melingkar.
- c. Menganalisis yang mempengaruhi setiap faktor dan faktor yang terpengaruh oleh faktor lainnya dan menggambarkan hubungannya dengan menggunakan anak panah.
- d. Setelah semua langkah-langkah sebelumnya dilakukan, kemudian menghitung jumlah anak panah yang masuk ke dalam faktor (indikator) dan keluar dari faktor (pendorong) dengan menuliskan jumlahnya.
- e. Selanjutnya dapat terlihat paling banyak panah yang masuk untuk suatu faktor, maka dapat disimpulkan sebagai akar penyebab utama pada suatu permasalahan.

2.2.2. Fishbone Diagram

Fishbone Diagram digunakan untuk menganalisis antar masalah utama dengan akar penyebab. Diagram ini berbentuk seperti tulang ikan dan memiliki kategori dari kemungkinan penyebab dengan metode 6M yaitu *man, machine, method, material, measurement, environment*. Langkah-langkah pembuatan *fishbone diagram* menurut Andersen & Fagerhaug (2006) sebagai berikut:

- a. Menuliskan masalah utama untuk mengetahui penyebabnya dengan jelas.
- b. Meletakkan masalah utama di ujung kanan anak panah besar.

- c. Mengidentifikasi kategori utama penyebab masalah dengan menuliskan pada cabang tulang ikan dari anak panah yang besar.
- d. Menuliskan seluruh penyebab yang terjadi di setiap kategori yang dituliskan dengan deskripsi yang singkat dan jelas.
- e. Selanjutnya dapat terlihat analisis penyebab yang teridentifikasi untuk menentukan akar penyebab yang terjadi pada suatu masalah.

2.2.3. Sistem Manajemen Gudang

Manajemen gudang dirancang untuk mengelola kegiatan yang dilakukan di gudang. Menurut Tompkins & Smith (1988), sistem manajemen gudang memiliki empat fungsi dasar yaitu menerima barang, menyimpan barang, mengambil barang, mengirimkan barang. Manajemen gudang tidak hanya meliputi empat fungsi dasar di atas, ada beberapa cara dalam membangun sistem manajemen gudang yang baik adalah sebagai berikut:

- a. Mengelompokkan barang sesuai dengan jenis dan memberi label tanggal masuk pada bahan baku, barang setengah jadi, dan barang yang sudah jadi.
- b. Menentukan metode pencatatan dengan informasi yang mudah diakses. Metode pencatatan dibagi menjadi dua yaitu pencatatan manual dan pencatatan terkomputerisasi.
- c. Memberikan pelatihan kepada pekerja dalam proses pengelolaan barang, bertujuan untuk meningkatkan kinerja saat pekerja menjalankan tugas.

2.2.4. Sistem Informasi

Menurut Arduin dkk (2015), konsep sistem informasi adalah suatu sistem yang dibuat oleh manusia dan komponen di dalamnya adalah aktivitas melakukan input, output, penyimpanan, mengelola data, dan mencatat informasi. Terdapat dua komponen penting dalam sistem informasi, yaitu pakar sistem informasi yang membuat dan mengembangkan sistem, klien merupakan orang yang akan menggunakan sistem informasi untuk melakukan komunikasi dan mendapatkan atau menyebarkan informasi. *Hardware* dan *software* merupakan komponen yang paling penting karena sebagai komponen utama dalam pembuatan sistem. *Hardware* adalah peralatan fisik seperti komputer, *keyboard*, dan *mouse*. *Software* merupakan sistem yang akan menghubungkan pengguna dengan *hardware*. Data memiliki beberapa bentuk di antaranya angka, huruf, gambar, kalimat, video. Data akan disimpan dalam database dan database merupakan kumpulan data yang akan menjadi sebuah informasi yang sesuai dengan kebutuhan.

2.2.5. SDLC (System Development Life Cycle) model Waterfall

Metode *waterfall* merupakan pengembangan perangkat lunak dengan menggambarkan proses pengembangan yang berjalan secara bertahap, dan setiap tahap diselesaikan secara berurutan sebelum melanjutkan ke tahap berikutnya. Metode ini sesuai dengan kebutuhan pengguna yang sudah dipahami oleh penulis mengenai permasalahan yang ada. Langkah-langkah perancangan sistem informasi menurut Hasanah & Untari (2020) adalah sebagai berikut:

a. Requirement Gathering Analysis

Tahap ini melakukan analisis dengan mengumpulkan data yang sesuai untuk kebutuhan sistem.

b. System Design

Perancangan desain yang dibagi menjadi tiga, yaitu tampilan proses, tampilan database, dan *design interface*.

c. Implementation

Implementasi sistem yang sudah dirancang dalam bentuk program menggunakan *coding*. Fitur serta fungsi sistem yang akan dimasukkan dalam bentuk program disesuaikan dengan tahap pertama dan tahap kedua.

d. Integration and Testing

Uji coba sistem atau program yang telah dibuat pada fitur-fitur yang ada dalam sistem dan program yang telah dibuat sudah sesuai dengan kebutuhan pengguna.

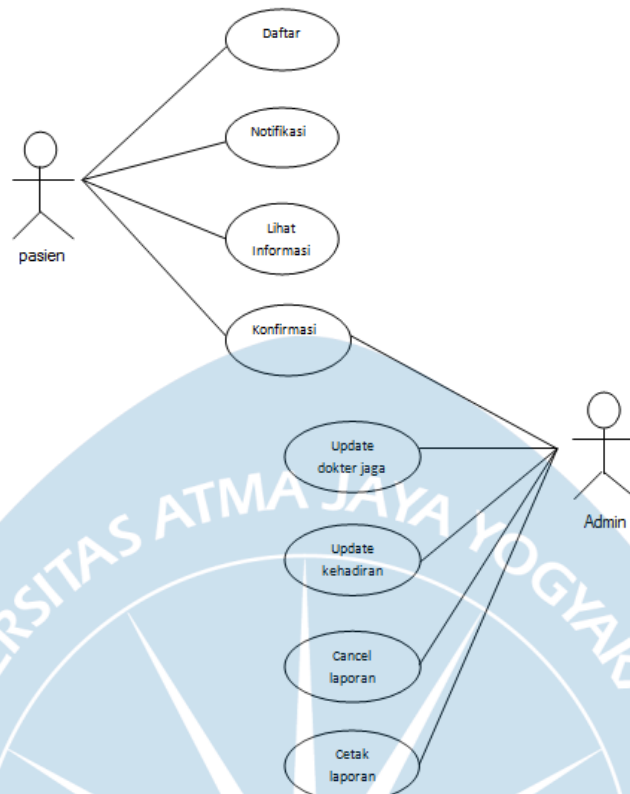
e. Operation and maintenance

Tahap ini program akan dioperasikan oleh pengguna dan akan ada perbaikan apabila program mengalami *error*.

2.2.6. Unified Modelling Language (UML)

Unified Modelling Language (UML) merupakan salah satu *tools* yang digunakan untuk merancang pengembangan perangkat lunak menurut Hasanah & Untari (2020). UML bertujuan untuk membuat kebutuhan pemodelan visual yang spesifik, menggambarkan, membangun, dan mendokumentasikan sistem perangkat lunak. Contoh UML dapat dilihat pada Gambar 2.1 dan simbol pada Gambar 2.2.

a. Usecase diagram, berfungsi untuk memperlihatkan interaksi sistem dengan user.



Gambar 2.1. Contoh Use Case Diagram

(Sumber: www.waskhas.com)

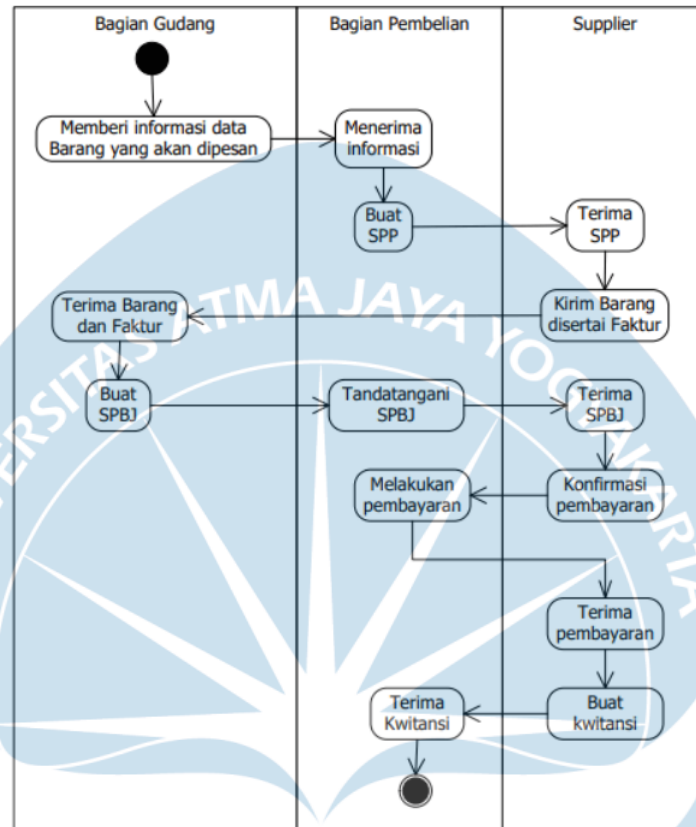
Penjelasan dari simbol yang digunakan dalam *use case* diagram dapat dilihat pada Tabel 2.2.

Tabel 2.2. Simbol Use Case Diagram

Simbol	Keterangan
 Aktor	Orang yang berkomunikasi dengan <i>usecase</i>
 Use case	Aktivitas yang ada di dalam sistem
 Association	Penghubung antara aktor dengan <i>usecase</i>

b. *Activity Diagram*

Activity diagram digunakan untuk menggambarkan proses yang terjadi dalam sistem dapat dilihat pada contoh Gambar 2.2.

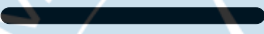


Gambar 2.2. Contoh *Activity* Diagram

(Sumber: www.dicoding.com)

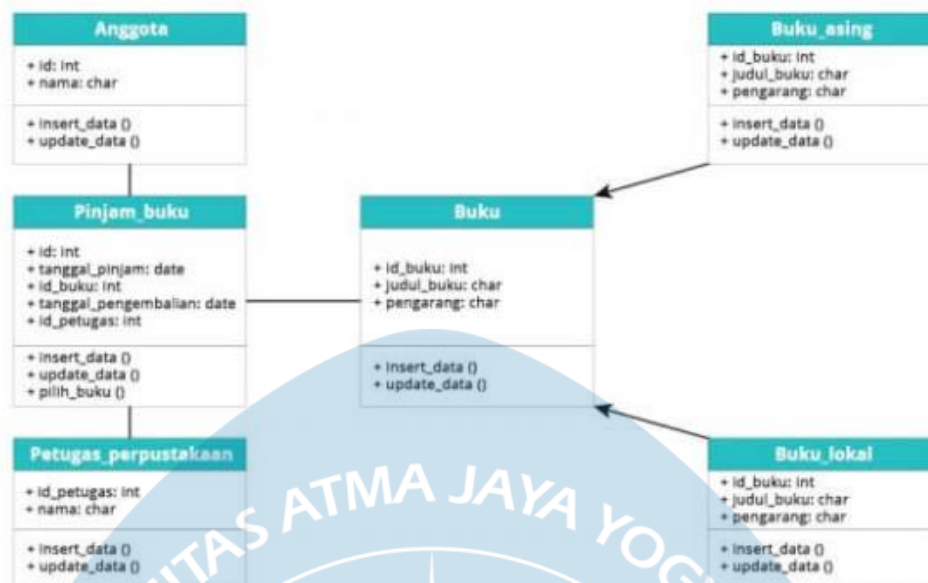
Simbol yang digunakan dalam *activity* diagram pada Tabel 2.3.

Tabel 2.3. Simbol Activity Diagram

Simbol	Keterangan
 <i>Initial State</i>	Diagram aktivitas yang menunjukkan status awal
 <i>Activity</i>	Aktivitas yang dilakukan sistem dengan menggunakan kata kerja
 <i>Decision</i>	Percabangan aktivitas yang dipilih lebih dari satu
 <i>Join</i>	Penggabungan lebih dari satu aktivitas
 <i>Final State</i>	Diagram aktivitas yang menunjukkan status akhir

c. *Class Diagram*

Class diagram adalah hubungan antar kelas dengan model desain pada sistem yang memenuhi kebutuhan satu sama lain menurut Sumirat (2023), dapat dilihat pada contoh Gambar 2.3.



Gambar 2.3. Contoh Class Diagram

(Sumber: www.dicoding.com)

Simbol yang digunakan dalam *class diagram* dapat dilihat pada Tabel 2.4.

Tabel 2.4. Simbol Class Diagram

Simbol	Keterangan
Class Atribut Operasi <i>Class</i>	Atribut: variabel yang ada pada kelas Operasi: fungsi yang ada pada kelas
_____ Asosiasi	Menggambarkan hubungan relasi antar dua kelas
_____◆ Agregasi	Menggambarkan hubungan antar dua kelas dimana salah satu kelas adalah bagian dari kelas lain
_____→ Generalisasi	Menggambarkan hubungan pewarisan antar kelas
-----→ <i>Dependency (Ketergantungan)</i>	Menggambarkan hubungan antar kelas dimana sebuah kelas bergantung pada kelas lain, namun tidak sebaliknya

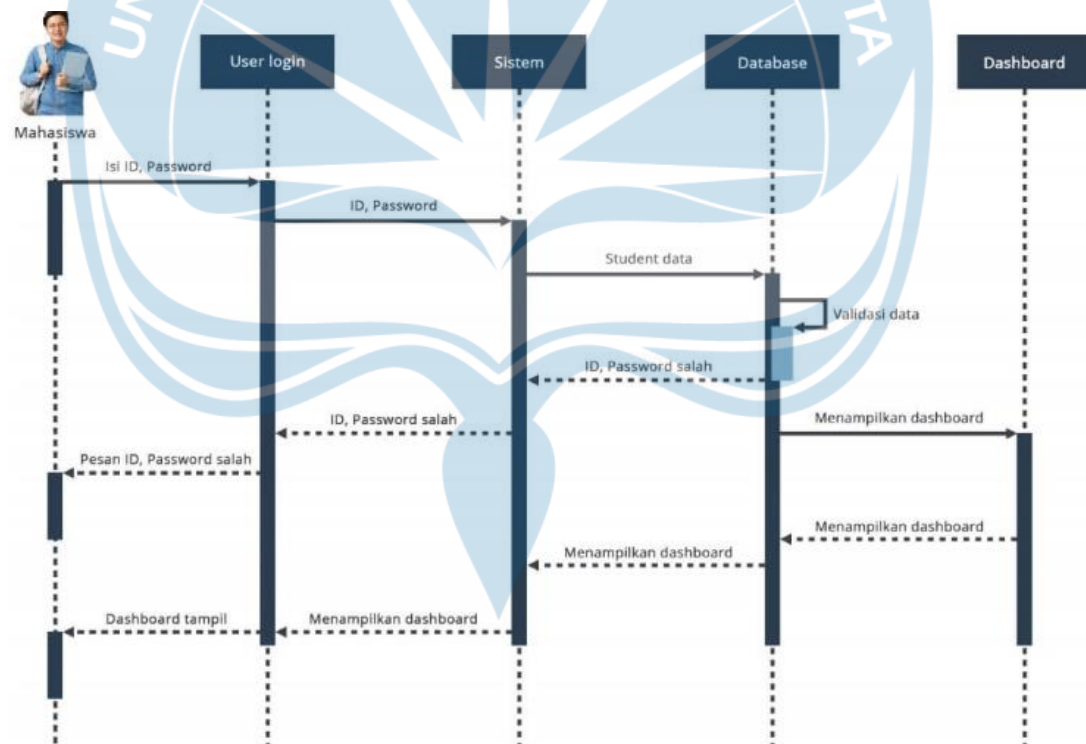
Class diagram memiliki nilai kardinalitas dapat dilihat pada Tabel 2.5.

Tabel 2.5. Nilai Kardinalitas

Nilai Kardinalitas	Keterangan
0..1	Nol atau satu
1	Hanya satu
0..*	Nol atau lebih
1..*	Satu atau lebih
0..n	Nol sampai n (dengan n > 1)
1..n	Satu sampai n (dengan n > 1)

d. *Sequence Diagram*

Sequence diagram digunakan untuk menjelaskan proses interaksi objek berdasarkan urutan waktu, dapat dilihat pada contoh Gambar 2.4.



Gambar 2.4. Contoh Sequence Diagram

(Sumber: www.dicoding.com)

Simbol yang digunakan dalam *sequence diagram* dapat dilihat pada Tabel 2.6.

Tabel 2.6. Simbol Sequence Diagram

Simbol	Keterangan
 Aktor	Pengguna atau <i>user</i> yang di luar sistem sedang melakukan interaksi dengan sistem
 Objek	Menggambarkan sebuah kelas
 <i>Lifeline</i>	Menggambarkan garis hidup objek
 <i>Activation box</i>	Menggambarkan waktu yang dibutuhkan oleh objek dalam menyelesaikan tugasnya
 <i>Message</i>	Menggambarkan komunikasi antar objek
 <i>Message to self</i>	Menggambarkan pesan balik dari objek sebelumnya

2.2.7. Black Box Testing

Menurut Wicaksono (2020) dari buku *blackbox testing* teori dan studi kasus, penulis Kurniawati (2018) menjelaskan bahwa *black box testing* adalah salah satu metode pengujian perangkat lunak tanpa melihat struktur program tersebut. Metode ini bertujuan untuk melakukan pengujian pada *user* saat menggunakan perangkat lunak tersebut. Parameter uji coba perangkat lunak dapat dilihat pada Tabel 2.7.

Tabel 2.7. Parameter Uji Perangkat Lunak

Parameter	Keterangan
<i>Functional testing</i>	Pengujian seluruh sistem berfungsi dengan baik dan tidak mengalami <i>error</i>
<i>Security testing</i>	Pengujian sistem terlindungi dari akses yang tidak sah
<i>Interface testing</i>	Pengujian untuk interaksi antar sistem berjalan dengan baik
<i>Usability testing</i>	Pengujian untuk melihat kepuasan pengguna dalam menggunakan sistem

2.2.8. PDCA (*Plan, Do, Check, Act*)

Metode ini digunakan untuk memperbaiki dan mengurangi kesalahan yang terjadi pada prosedur kerja. Langkah dan penjelasan metode PDCA menurut Syukron & Kholil (2012) sebagai berikut:

- Plan, menentukan tujuan perbaikan dan membuat rencana perbaikan.
- Do, menyusun dan melaksanakan rencana yang telah dibuat.
- Check, melakukan evaluasi dari rencana yang sudah dibuat sebelumnya.
- Act, melakukan perbaikan sesuai dengan evaluasi jika ada.