

BAB II

KAJIAN LITERATUR

2.1 Konsep Sistem Informasi

Sistem informasi berdasarkan definisi sistem dan informasi di atas merupakan sebuah media yang terdiri atas beberapa komponen yang bertujuan untuk mengolah data menjadi informasi sebagai penunjang dalam pengambilan keputusan bagi penggunanya secara cepat dan tepat. Saat ini, sistem informasi berbasis komputer merupakan salah satu sistem yang banyak digunakan dikarenakan komputer sebagai alat pengolah data berperan penting [3].

Berbeda dengan teknologi informasi (*information technology*) yang merujuk pada teknologi yang digunakan oleh penggunanya untuk mengolah informasi, sedangkan sistem informasi (*information system*) merupakan sebuah alat yang mengumpulkan, menyimpan dan menganalisis informasi untuk tujuan tertentu. Ini mengapa penting bagi sebuah sistem informasi untuk dapat memberikan hasil informasi yang diolah kepada orang yang tepat, pada waktu yang tepat, dalam jumlah dan *format* yang sesuai [1].

Sistem informasi berbasis komputer dibagi menjadi beberapa jenis berdasarkan dari tujuan pengembangan sistem informasi tersebut, di antaranya [4]:

1. *Transaction Processing Systems* (TPS)

TPS merupakan sistem informasi yang merekam transaksi-transaksi bisnis seperti penjualan. Hasil dari olahan data sistem ini pada umumnya digunakan untuk mengelola jumlah stok produk berdasarkan transaksi penjualan.

2. *Management Information Systems* (MIS)

Management Information Systems (MIS) merupakan sistem informasi yang menghasilkan informasi berupa laporan pada tingkat manajemen dalam sebuah organisasi. Sistem ini memiliki fokus utama pada merencanakan, mengendalikan serta membuat keputusan.

3. *Decision Support Systems* (DSS)

Decision Support Systems (DSS) merupakan sistem informasi yang menggunakan analisa dan data dalam bentuk grafik secara mendalam sehingga dapat menelusuri setiap konsekuensi serta dapat dikustomisasi berdasarkan kebutuhan penggunanya. Sistem ini mencakupi pengambilan keputusan yang bersifat semi struktur dan tidak berstruktur.

4. *Executive Information System* (EIS)

Executive Information System (EIS) merupakan sistem yang berperan dalam menyediakan informasi atas kinerja secara keseluruhan dari sebuah organisasi. Sistem ini dikembangkan dengan tujuan mempermudah penyebaran informasi kepada para penggunanya sehingga dapat menghemat waktu. Umumnya sistem ini dapat disambungkan dengan pelayanan informasi *online* dan surat elektronik.

2.2 *Rapid Application Development*

Pengembangan sistem merupakan tahapan dimana sebuah sistem melalui proses penyusunan terlebih dahulu dengan tujuan untuk menghasilkan sistem baru ataupun menggantikan sistem yang lama. *Rapid Application Development* merupakan sebuah metode pengembangan sistem secara bertingkat. Metode ini banyak dipakai ketika ingin mengembangkan sebuah sistem informasi yang fungsional secara singkat, cepat dan dalam jangka waktu yang pendek [4], [5].

Metode ini dilakukan secara berulang dan berkelanjutan dimana model kerja sistem dikembangkan dengan menyesuaikan perubahan yang ada sesuai dengan kebutuhan pengguna. Empat tahapan dalam metode pengembangan ini oleh Tiley dibagi menjadi berikut ini [4], [5].

1. Perencanaan persyaratan

Perencanaan persyaratan menggabungkan elemen-elemen dari tahap perencanaan sistem dan analisis sistem pada metodologi *System Development Life Cycle* (SDLC). Pengguna, manajer dan staf TI berdiskusi dan menyepakati ruang lingkup, batasan, dan kebutuhan sistem. Tahapan ini selesai apabila kesepakatan telah tercapai.

2. Desain pengguna

Di dalam tahapan desain, pengguna berinteraksi dengan analisis sistem dan membangun model dan *prototype* yang mewakili semua masukan, proses, dan keluaran sistem. Desain pengguna merupakan proses berkelanjutan dan interaktif yang memberikan pengguna kesempatan untuk memahami, memodifikasi, dan menyetujui model sistem yang memenuhi kebutuhan mereka.

3. Konstruksi sistem

Tahapan konstruksi berfokus pada pengembangan program dan aplikasi. Akan tetapi pada tahapan ini, perubahan atau perkembangan masih bisa diusulkan dan diterapkan.

4. Penyelesaian (*cutover*)

Fase *cutover* menyerupai tahapan akhir pada fase implementasi di dalam SDLC, seperti konversi data, pengujian, dan pelatihan pengguna

2.3 Teknik Pengembangan Sistem

2.3.1 Diagram Use Case

Diagram ini banyak digunakan untuk menggambarkan interaksi antar subsistem, lingkungan sistem eksternal dan penggunanya. Diagram ini dapat menjelaskan urutan langkah-langkah dari setiap interaksi dikarenakan disertai dengan naratif penggunaannya [6].

Komponen-komponen dari *Use Case Diagram* sebagai berikut [7].







1. Sistem

Pada diagram ini, persegi digunakan sebagai bentuk atas batasan *Use Case* dengan hubungan di luar sistem.

2. Aktor

Segala entitas yang berinteraksi dengan sistem disebut dengan aktor. Komponen ini memberikan informasi atas interaksi tersebut. Ini terjadi dikarenakan

- a. Terdapat interaksi dengan sistem eksternal.
- b. Terdapat sumber lain yang digunakan pada sistem.
- c. Terdapat kebutuhan arus informasi untuk penerimanya.

Simbol	Nama	Keterangan
	Actor	Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan use case
	Use Case	Abstraksi dan interaksi antara sistem dan aktor.
	Association	Abstraksi dari penghubung antara aktor dengan use case.
	Generalisasi	Memperlihatkan spesialisasi aktor untuk bisa berperan serta dengan use case.
	Include	Memperlihatkan bahwa suatu use case seluruhnya adalah fungsionalitas dari use case lainnya.
	Extend	Memperlihatkan bahwa suatu use case adalah tambahan fungsional dari use case lainnya jika satu keadaan terpenuhi.

Gambar 2.1 Komponen *Use Case Diagram*

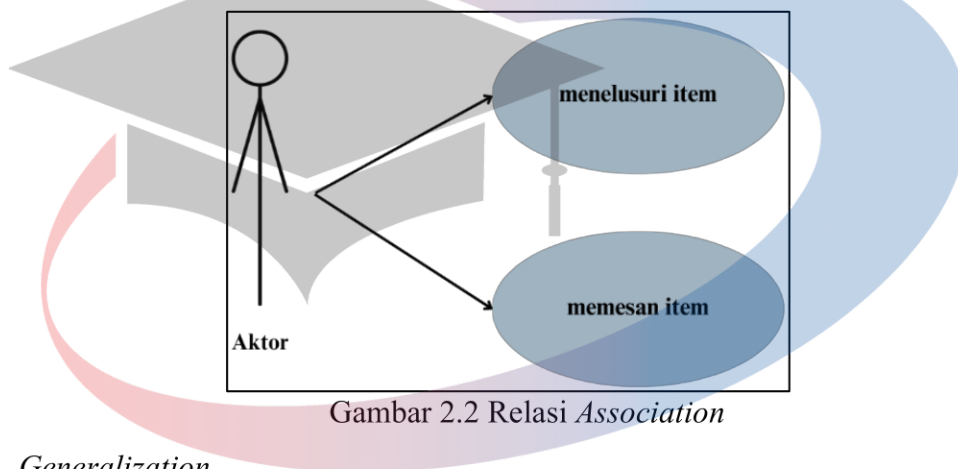
3. Use Case

Komponen ini berupa deskripsi atas fungsionalitas sistem. Ini bertujuan untuk mempermudah pihak-pihak yang terlibat dalam memahami kebutuhan fungsi-fungsi pada masa pengembangan.

Selain komponen, hal penting lainnya yang dibutuhkan ketika membuat sebuah *Use Case Diagram* yaitu relasi antar komponen yang terdapat pada *Use Case Diagram* tersebut [7].

a. Association

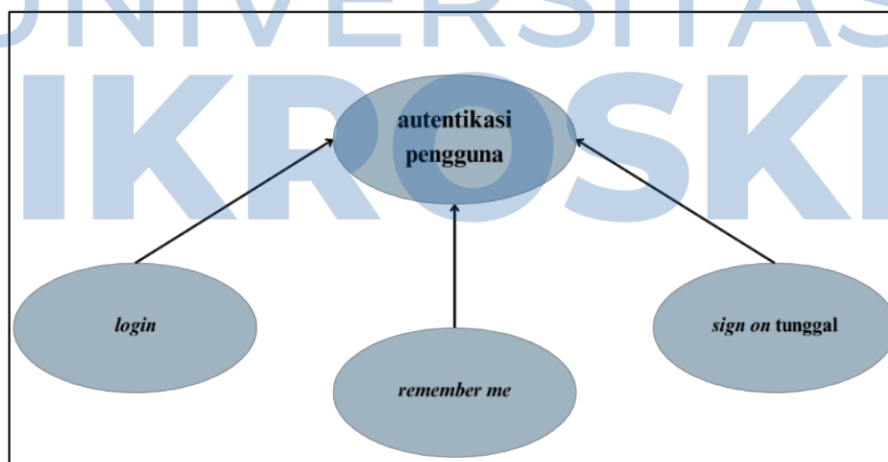
Relasi ini digunakan untuk menggambarkan identitas yang dikerjakan oleh aktor dengan sebuah *Use Case* sebagai garis penghubung.



Gambar 2.2 Relasi Association

b. Generalization

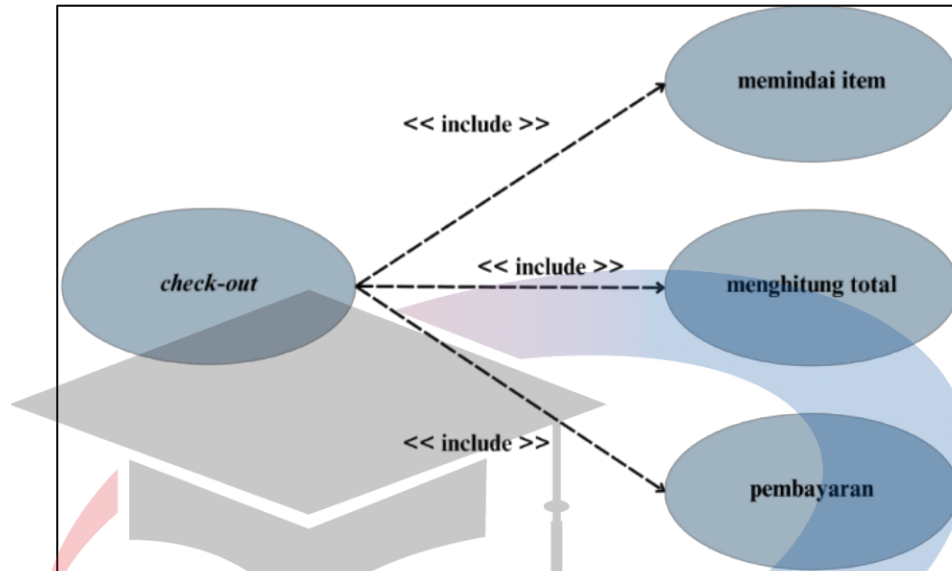
Relasi ini digunakan untuk menggambarkan 2 aktor dan juga 2 *Use Case* dimana terdapat *override* atau *inherit*.



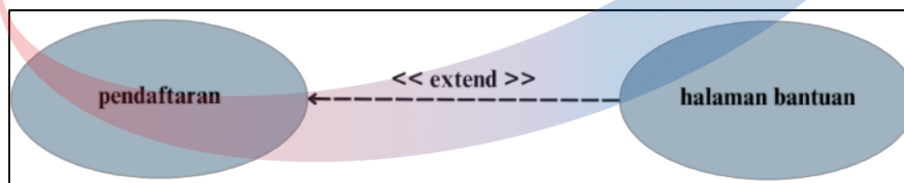
Gambar 2.3 Relasi Generalization

c. *Dependency*

Relasi ini terbagi menjadi ke dalam dua jenis berdasarkan ciri-cirinya, yaitu *intend* dimana terjadi interaksi antar *Use Case* yang saling memanggil ataupun dipanggil dan *extend* dimana relasi ini merupakan interaksi tambahan daripada relasi *intend*.



Gambar 2.4 Relasi *Intend*



Gambar 2.5 Relasi *Extend*

Perbedaan *Use Case Diagram* dengan *descriptions* berada pada visualisasi yang diberikan. *Use Case descriptions* menggunakan teks uraian dari serangkaian *Use Case* dari sebuah sistem untuk memberitahukan penggunaanya cara dalam mencapai sebuah tujuan [8].

UC1. DEPOSIT CASH	
Actors:	Customer, Bank Clerk
Triggers:	Customer asks to deposit cash into account
Contexts:	Customer arrives at counter and has asked to deposit \$500 into his business account
Pre-conditions:	Customer has account; cash legal tender
Main Flow of Events:	<ol style="list-style-type: none"> 1. The bank clerk selects access the customer account. 2. The system displays the account details. 3. The bank clerk adds \$500 to the account total. 4. The account total displayed increases by \$500. 5. The clerk prints a receipt for the customer. 6. The system acknowledges printed receipt. 7. The clerk ends the transaction.
Post-conditions:	The customer's account is increased by \$500.
Alternative Flow of Events:	<ol style="list-style-type: none"> 1. The system fails to display the account details. 2. Prompt the clerk to check customer account number. 3. Clerk repeats step 1 in main flow.
Exception Flow of Events:	<ol style="list-style-type: none"> 1. System flags problem with customer account. 2. Clerk messages manager. 3. Step 7 in the main flow.

Gambar 2.6 Contoh *Use Case Description*

2.3.2 Daftar Persyaratan Non-Fungsional

Persyaratan non-fungsional merupakan salah satu metode yang digunakan untuk menganalisis kebutuhan sistem [9]. Terdapat batasan kualitas yang harus dipenuhi oleh sebuah sistem melalui atribut-atribut non-fungsional, yaitu sebagai berikut [10, 11].

1. *Portability*

Atribut ini merupakan kemampuan sistem untuk dapat dioperasikan pada berbagai jenis *platform* perangkat lunak dan perangkat keras yang ada.

2. *Security*

Atribut ini merupakan kemampuan sistem dalam menjaga aset informasi yang dimiliki oleh sebuah organisasi dari kebocoran, modifikasi, penyalahgunaan dan penghapusan data baik secara disengaja maupun tidak disengaja.

3. *Maintainability*

Atribut ini merupakan kemampuan sistem untuk dapat dimodifikasi.

4. *Reliability*

Atribut ini merupakan kemampuan sistem untuk tetap dapat bekerja dalam suatu jangka waktu pada suatu kondisi tertentu.

5. *Scalability*

Atribut ini merupakan kemampuan sistem dalam meningkatkan kinerjanya mengikuti kebutuhan pengguna. Ini dapat berupa penambahan memori tambahan ataupun peningkatan kapasitas energi tambahan bagi komputer.

6. *Performability*

Atribut ini merupakan kemampuan sistem dalam memenuhi persyaratan waktu yang telah ditentukan.

7. *Usability*

Atribut ini merupakan kemampuan sistem untuk dapat digunakan oleh berbagai jenis pengguna dengan mudah dan efektif ketika melaksanakan sebuah tugas di sebuah kondisi tertentu.

8. *Flexibility*

Atribut ini merupakan kemampuan sistem dalam beradaptasi mengikuti faktor eksternal dan internal yang beragam ataupun berbeda.

2.4 Koperasi Sekolah

Koperasi sekolah merupakan organisasi yang pada umumnya beranggotakan pihak-pihak yang bersangkutan dengan lingkungan sekolah yang bertujuan untuk meningkatkan kesejahteraan para warga sekolah. Buku, pensil, dan pakaian seragam merupakan contoh dari persediaan yang terdapat pada koperasi sekolah [12]. Fungsi bisnis utama yang terdapat pada umumnya terdapat pada sebuah koperasi sekolah adalah sebagai berikut.

1. Pembelian merupakan salah satu fungsi penting sebuah proses bisnis yang melibatkan berbagai pihak dengan dasar rasional yang didapatkan melalui informasi yang memadai. Umumnya pembelian pada sebuah organisasi dilakukan secara berulang, baik dari periode pembelian hingga jumlah kuantitas barang yang dibeli [13].
2. Penjualan merupakan sebuah proses bisnis dimana sebuah pihak selaku penjual memberikan atau memenuhi kebutuhan dan keinginan kepada pembeli agar mencapai kesepakatan yang dapat menguntungkan kedua belah pihak sebagai imbalan atas jasa ataupun barang yang telah diberikan [14].
3. Persediaan merupakan barang-barang yang digunakan oleh perusahaan dalam proses produksi maupun penjualan. Oleh karena itu, penting untuk tetap mengelola persediaan agar proses produksi dan penjualan tidak terganggu. Persediaan seringkali dianggap sebagai aktivitas dari sebuah perusahaan. Ini mencakup barang-barang yang telah siap dijual maupun barang-barang yang masih dalam proses produksi [14].

2.5 Pengembangan Aplikasi Desktop

Aplikasi *desktop* adalah aplikasi yang dapat di-*instal* di dalam *operating system* terkhususnya Windows dan mampu beroperasi secara *offline*. Pengembangan Aplikasi *desktop* dapat menjadi solusi untuk membangun antarmuka pengguna grafis (GUI) untuk menghasilkan aplikasi dengan segudang fitur yang dapat membantu menyelesaikan masalah *user*[15].

2.6 Pengembangan Aplikasi Web

Pemrograman berbasis web memadukan pemrograman, *jaringan komputer*, interaksi manusia dan komputer, dan basis data untuk mengembangkan sistem *web*. Cara kerja dari sistem *web* ada pada teknologi *client/server* di mana pihak *client* mengakses aplikasi tersimpan pada *web server* melalui *internet browser*. Lima komponen utama pada teknologi *web* yaitu[16]:

1. *Browser*

Komponen merupakan sebuah perantara antarmuka bagi pengguna untuk mengakses *web server* untuk kemudian menampilkan kembali informasi *web server* kembali ke perangkat pengguna.

2. *Web server*

Komponen memiliki peran sebagai penerima (*response*) permintaan (*request*) dari *browser* pengguna. *Request* dikirim menggunakan *Hypertext Transfer Protocol* (HTTP) melalui *browser* berupa HTML. Ini merupakan protokol sederhana dari proses transporter *request* dan *response* antara *web browser* dan *web server*.

3. *Application server*

Permintaan dinamis yang mengandung logika akan diproses oleh *application server* seperti *script* dan *syntax*.

4. *Relational Database Management System* (RDBMS)

Relational Database Management System (RDBMS) berperan untuk mengelola semua basis data pada aplikasi sistem komputer yang tersimpan ke dalam bentuk beberapa tabel yang saling memiliki relasi. Ini berguna untuk mengurangi terjadinya redudansi data.

5. *Web development environment*

Pengembangan aplikasi berbasis web memerlukan lingkungan yang mendukung WWW dimana perangkat lunak yang digunakan untuk pengembangan aplikasi berbasis *web*.