

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Penelitian ini dilakukan tidak terlepas dari hasil penelitian-penelitian terdahulu. Adapun penelitian terkait yang dijadikan sebagai referensi dalam penelitian ini adalah:

Berdasarkan penelitian yang dilakukan oleh Asep Abdurrahman dan Siti Masripah (2017) dengan judul “Metode Waterfall Untuk Sistem Informasi Penjualan” Sistem yang terjadi pada toko kue Manika yang bergerak dalam bidang penjualan makanan, masih melakukan proses transaksi penjualan secara manual, mulai dari pencatatan pelanggan, penyimpanan data-data yang berhubungan dengan proses penjualan, sehingga saat proses berlangsung terjadi kesalahan dalam pencatatan, kurang akuratnya laporan yang dibuat dan keterlambatan dalam pencarian data-data yang diperlukan. Untuk mengatasi permasalahan tersebut peneliti bermaksud menggunakan pengembangan sistem yaitu *Software Development Life Cycle* (SDLC) dengan metode *Waterfall* dalam perancangan sistem informasi, dimulai dari menganalisa kebutuhan *software*, *design*, *implementasi*, dan *testing*. Perancangan sistem informasi merupakan solusi yang terbaik untuk memecahkan permasalahan-permasalahan yang ada pada perusahaan ini, serta dengan sistem yang terkomputerisasi dapat tercapai suatu kegiatan yang efektif dan efisien dalam menunjang aktivitas pada perusahaan ini.

Berdasarkan penelitian yang dilakukan oleh Indah Kusuma Dewi, dkk (2019) dengan judul “Analisis dan Implementasi Sistem Informasi Penjualan Berbasis Web Pada UKM Tiara Cakery Batam” UKM Tiara Cakery bergerak di bidang penjualan kue oleh-oleh kota Batam dengan berbagai macam jenis kue. Objek penelitian ini membuat website untuk memudahkan customer melakukan pembelian produk tanpa mendatangi langsung lokasi penjualan. Penelitian penjualan online ini membuat *website* untuk memudahkan *customer* melakukan pembelian produk tanpa mendatangi langsung lokasi penjualan. Perancangan aplikasi yang dilakukan peneliti menggunakan metode SDLC (*Software Development Life Cycle*).

Berdasarkan penelitian yang dilakukan oleh Edmund Johnson, dkk (2019) dengan judul “Pengembangan Sistem Informasi Layanan Maintenance Dan Penjualan Sparepart, Studi Kasus: PT. Karya Lencana Mas” penelitian ini, sebuah sistem dikembangkan untuk dapat membantu pihak manajemen dalam memprediksi *sparepart* yang sebenarnya diperlukan, pemesanan untuk pemenuhan stok agar lebih efektif dan memantau pergerakan stok dan manajemen mekanik dan kegiatan *service* kendaraan yang dilakukan. Pengembangan sistem dilakukan dengan mengacu pada metode SDLC (*System Development Life Cycle*). Hasil dari penelitian ini adalah sistem berbasis desktop dengan dukungan fitur yang dapat diakses oleh kepala mekanik, kasir dan pemilik.

Berdasarkan penelitian yang dilakukan oleh Ahmad Makhdum Badawi, dkk (2019) dengan judul “Design Sistem Informasi Akuntansi Untuk Pengelolaan Transaksi Keuangan dengan Metode Software Development Life Cycle pada CV.

Surya Admaja” CV. Surya Admaja merupakan suatu perusahaan yang bergerak dibidang produksi dan penjualan beras putih yang berkualitas. CV. Surya Admaja masih menggunakan cara yang sederhana dalam menangani pengendalian keuangan yaitu dengan cara pencatatan secara manual. Di dalam pengendalian keuangan tersebut terdapat berbagai masalah yang dihadapi oleh perusahaan seperti pembuatan laporan supplier, laporan pelanggan, laporan barang, laporan karyawan laporan penjualan, laporan pembelian, laporan kas masuk, laporan kas keluar dan laporan kas perusahaan. Maka penelitian ini merancang aplikasi Surya ADM yaitu dengan siklus hidup pengembangan sistem atau *System Development Life Cycle* (SDLC).

Berdasarkan penelitian yang dilakukan oleh Wulan Dari (2015) dengan judul “Penerapan Metode *System Development Life Cycle* Pada Pembuatan Sistem Informasi Penjualan Produk Batik Kurowo Jakarta” Toko Batik Kurowo adalah usaha yang bergerak di bidang penjualan produk–produk yang terbuat dari kain batik, seperti pakaian, tas, hordeng, taplak meja, ikat pinggang, sajadah, mukena dan berbagai produk lainnya. Dalam proses pemasaran dan penjualan produk batik Kurowo memiliki kendala, yaitu kesulitan dalam memasarkan produknya. Berdasarkan kendala yang terjadi tersebut, maka penelitian ini merancang dan membangun Sistem Penjualan Aplikasi *E-Commerce* berbasis *online* untuk menanggulangi kendala tersebut.

II.2. Landasan Teori

II.2.1. Sistem

Definisi sistem adalah “sistem adalah kumpulan dari komponen-komponen yang saling berkaitan satu dengan yang lain untuk mencapai tujuan dalam melaksanakan suatu kegiatan pokok perusahaan”. Definisi sistem adalah "Sistem adalah rangkaian dari dua atau lebih komponen-komponen yang saling berhubungan, yang berinteraksi untuk mencapai suatu tujuan". (Arikunto, 2016).

Asal kata Sistem berasal dari bahasa latin *systema* dan bahasa Yunani *sustema*. Pengertian sistem adalah suatu kesatuan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, atau energi untuk mencapai suatu tujuan. Atau dapat juga dikatakan bahwa Pengertian Sistem adalah sekumpulan unsur /elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan. (Randi V. Palit, 2015).

II.2.2. Karakteristik Sistem

Karakteristik Sistem dibagi menjadi beberapa bagian (Agustina et al, 2013), yaitu:

1. Tujuan Sistem

Tujuan sistem merupakan target atau sasaran akhir yang ingin dicapai oleh suatu sistem. Agar target tersebut bisa tercapai, maka target atau sasaran tersebut harus diketahui terlebih dahulu ciri-ciri atau kriterianya. Upaya mencapai suatu

sasaran tanpa mengetahui ciri-ciri atau kriteria dari sasaran tersebut memungkinkan sasaran tersebut tidak akan pernah tercapai.

2. Batas Sistem

Batas sistem merupakan garis abstraksi yang memisahkan antara sistem dengan lingkungannya. Batas sistem ini sangat relatif dan tergantung kepada tingkat pengetahuan dan situasi serta kondisi yang dirasakan oleh orang yang melihat sistem tersebut.

3. Subsistem

Subsistem sebenarnya merupakan sistem di dalam suatu sistem. Ini berarti bahwa sistem berada pada lebih dari satu tingkat. Mobil adalah suatu sistem yang terdiri dari sistem-sistem di bawahnya seperti sistem mesin, sistem badan mobil, dan sistem rangka. Masing-masing sistem ini terdiri dari sistem tingkat yang lebih rendah lagi. Misal, sistem mesin adalah kombinasi dari sistem karburator, sistem generator, sistem bahan bakar, dan lain sebagainya.

4. Hubungan Sistem

Hubungan sistem disebut juga dengan *interface*, merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui *interface* ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya. Keluaran (*output*) dari satu subsistem akan menjadi masukan (*input*) untuk subsistem yang lainnya dengan melalui *interface*. Dengan *interface* satu subsistem dapat berintegrasi dengan subsistem lainnya membentuk satu kesatuan.

5. Hirarki Sistem

Hirarki sistem menggambarkan hubungan sistem dengan sistem yang lebih besar yang disebut sebagai supersistem dan hubungan dengan sistem yang lebih kecil yang disebut sebagai subsistem.

6. *Input* yang masuk ke dalam suatu sistem.

Input bervariasi, bisa berupa energi, manusia, data, modal, bahan baku, layanan atau lainnya. *Input* merupakan pemicu bagi sistem untuk melakukan proses yang diperlukan.

7. Proses

Proses merupakan perubahan dari *input* menjadi *output*. Proses ini mungkin dilakukan oleh mesin, orang, atau komputer. Umumnya kita mengetahui bagaimana *input* dirubah menjadi *output*, akan tetapi pada situasi tertentu proses tidak diketahui secara detail karena perubahannya terlalu kompleks. Proses dapat saja berupa perakitan yang menghasilkan satu macam *output* dari berbagai macam *input* yang disusun berdasarkan aturan tertentu.

8. *Output*

Output dapat berbentuk produk, servis, informasi dalam bentuk print out komputer atau energi seperti *output* dari dinamo. *Output* adalah hasil dari suatu proses yang merupakan tujuan dari keberadaan sistem.

9. Lingkungan Sistem

Sistem berada dalam suatu lingkungan dimana sistem tidak dapat mengontrolnya. Lingkungan sistem ada dua macam, lingkungan *eksternal* (di luar sistem) dan lingkungan *internal* (dalam sistem). Baik lingkungan *eksternal* maupun

lingkungan *internal* memiliki pengaruh yang sangat kuat terhadap sistem, sedangkan suatu sistem sedikit sekali memiliki kemampuan untuk merubah lingkungan.

II.2.3. Klasifikasi Sistem

Sistem dapat diklasifikasikan ke dalam bentuk yang lebih spesifik (Agustina et al, 2013), sebagai berikut:

1. Sistem Terbuka dan Sistem Tertutup
 - a. Suatu sistem yang dihubungkan dengan lingkungannya melalui arus sumber daya disebut sistem terbuka (*open system*). Contoh, sistem pemanas mendapatkan inputnya dari listrik dan menyediakan panasnya bagi gedung atau ruangan yang dipanasinya.
 - b. Suatu sistem yang tidak dihubungkan dengan lingkungannya adalah sistem tertutup (*closed system*). Sistem tertutup hanya terdapat dalam situasi laboratorium yang dikontrol ketat.
2. Sistem Fisik dan Sistem Konseptual

Perusahaan bisnis adalah suatu sistem fisik, karena terdiri dari sejumlah sumber daya fisik. Sistem Konseptual adalah sistem yang menggunakan 14 sumber daya konseptual, informasi dan data, untuk mewakili suatu sistem fisik. Contoh, komputer adalah suatu sistem fisik, tetapi data dan informasi yang disimpan di dalamnya dapat dipandang sebagai suatu sistem konseptual. Data dan informasi mewakili satu atau lebih sistem fisik. Sistem fisik penting karena keberadaannya; sistem konseptual penting karena penggambarannya atas sistem fisik. Contoh, jika

penyimpanan komputer menunjukkan bahwa ada 70 perkakas di dalam gudang, maka inspeksi di gudang harus mengungkapkan 70 perkakas tersebut.

II.2.4. Pengertian Informasi

Pengertian informasi merupakan suatu data yang telah diproses menjadi bentuk yang memiliki arti bagi penerima dan dapat berupa fakta maupun suatu nilai yang bermanfaat. Jadi ada suatu proses transformasi data menjadi suatu informasi dari *input* yang diproses atau dikelola yang menghasilkan suatu *output*.

Informasi adalah Data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, melakukan keputusan berdasarkan informasi tersebut dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap kembali sebagai input, diproses kembali melalui suatu model dan seterusnya yang merupakan suatu siklus. (Randi V. Palit, 2015).

II.2.4.1. Kualitas Informasi

Informasi merupakan cara menambahkan pengetahuan atau mengurangi ketidakpastian pemakai informasi. Maka kualitas informasi 3 hal seperti yang dikemukakan oleh Jogiyanto, yaitu sebagai berikut:

1. Informasi harus akurat, Akurat berarti informasi harus jelas mencerminkan maksudnya. Informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan, sebab dari sumber informasi sampai ke penerima informasi kemungkinan banyak

2. Terjadi gangguan (*noise*) yang dapat mengubah dan merusak informasi tersebut.
3. Informasi harus relevan, informasi harus memiliki manfaat bagi pemakainya dan relevansi informasi bagi setiap orang akan berbeda.
4. Informasi harus tepat pada waktunya, perlu dipahami, mahalnya informasi dikarenakan harus cepatnya didapat sehingga diperlukan teknologi informasi untuk mengolah dan mengirimkannya. Informasi yang dikirim atau diterima tidak boleh terlambat diterima si penerima, sebab informasi yang usang tidak akan mempunyai nilai lagi. Apalagi jika informasi tersebut merupakan dasar untuk dijadikan dalam pengambilan keputusan. Jika pengambilan keputusan terlambat maka berakibat fatal bagi suatu organisasi.

II.2.5. Pengertian Dasar Sistem Informasi

Sistem informasi dapat didefinisikan sebagai suatu sistem di dalam suatu organisasi yang merupakan kombinasi manusia (SDM), fasilitas, teknologi, media, prosedur-prosedur, dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan SI menjadi sangat begitu penting dan tergantung kepada lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyajikan suatu dasar informasi untuk mengambil keputusan yang baik. Informasi didapatkan dari sistem informasi (*information system*) atau disebut juga dengan *processing system* atau *information processing systems*.

Komponen fisik sistem informasi dibagi menjadi 4 (empat):

1. Personal (*humanware*): pelaksana manajerial, data entry operator, computer operator, programmer, system analyst, dan database administrator.
2. Prosedur (*organiware*): kebijakan formal dan petunjuk untuk mengoperasikan sistem. Terdiri dari tatalaksana, prosedur pengolahan data, dan pedoman pemakai.
3. Perangkat pengolahan data (*technoware*): *hardware*, *software*, perangkat pendukung seperti jaringan komputer, sistem komunikasi, dan lainnya.
4. Data (*inforware*): *database*.

II.3. Sistem Informasi Penjualan

Sistem Informasi Penjualan diartikan sebagai suatu pembuatan pernyataan penjualan, kegiatan akan dijelaskan melalui prosedur, prosedur yang meliputi urutan kegiatan sejak diterimanya pesanan dari pembeli, pengecekan barang ada atau tidak ada dan diteruskan dengan pengiriman barang yang disertai dengan pembuatan faktur dan mengadakan pencatatan atas penjualan yang berlaku.

Sistem Informasi penjualan adalah suatu sistem informasi yang mengorganisasikan serangkaian prosedur dan metode yang dirancang untuk menghasilkan, menganalisa, menyebarkan dan memperoleh informasi guna mendukung pengambilan keputusan mengenai penjualan.

II.3.1. Penjualan

Definisi Penjualan adalah perkumpulan seorang pembeli dan penjual dengan melakukan tukar menukar barang dan jasa berdasarkan keputusan kedua belah pihak. (Eka Saputra, 2019).

Penjualan merupakan suatu kegiatan yang dilakukan sebagian manusia dalam menjual barang dagangan yang dimiliki baik itu barang ataupun jasa kepada pasar agar mencapai suatu tujuan yang diinginkan. (Andri Prasetyo, 2016).

Penjualan merupakan aktivitas utama perusahaan. Pendapatan perusahaan sangat ditentukan oleh besar kecilnya penjualan. Kegiatan penjualan itu sendiri berhubungan erat dengan kegiatan *marketing* atau pemasaran, dimana penjualan merupakan bagian dari marketing. Bagi perusahaan distributor, kegiatan penjualan menjadi tugas para *salesman*. (Viktor Nicolas Nore, 2013)

II.3.2. Pengertian Strategi Penjualan

Strategi Penjualan adalah perencanaan aktivitas penjualan, metode klien mencapai, perbedaan kompetitif dan sumber daya yang tersedia. Penjualan strategi sangat penting, jika tidak daerah yang paling penting yang dibutuhkan untuk menghasilkan keuntungan bagi perusahaan.

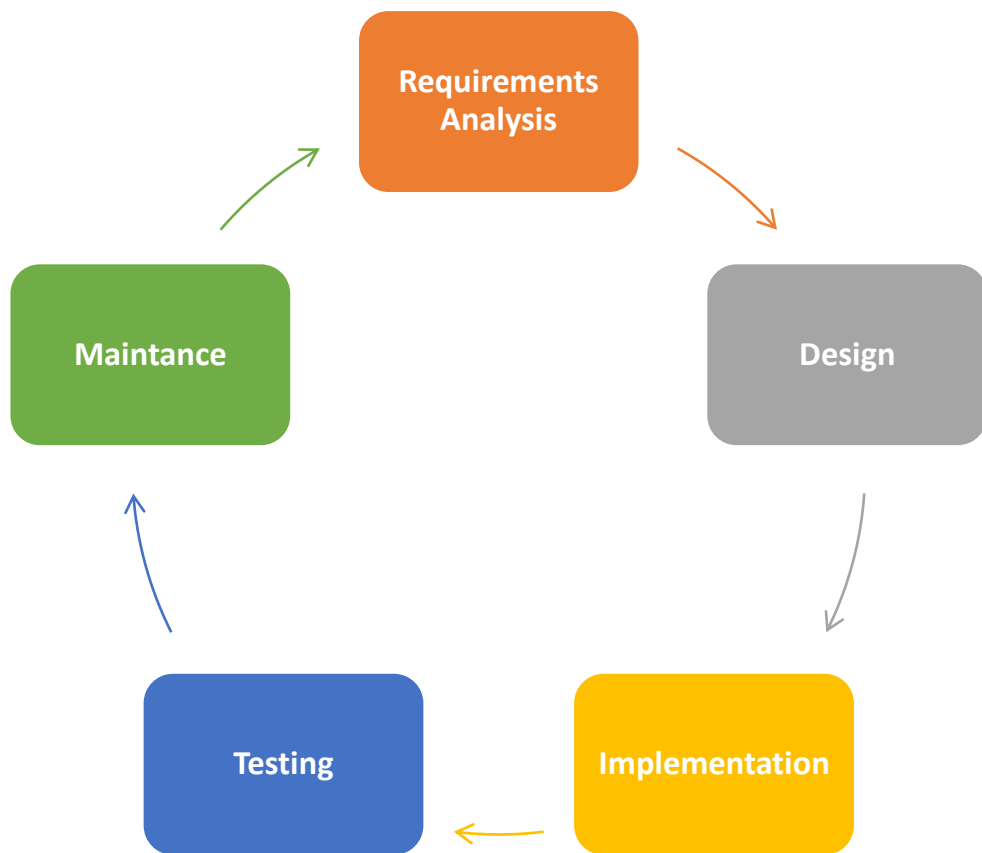
II.4. Metode Life Cycle

System Development Life Cycle (SDLC) adalah proses pembuatan dan perubahan sistem serta model dan metodologi yang digunakan untuk mengembangkan sebuah sistem. SDLC juga merupakan pola yang diambil untuk

mengembangkan sistem perangkat lunak, yang terdiri dari tahap-tahap: *requirements* (analisis kebutuhan), *analysis* (analisis sistem), *design* (perancangan), *coding / implementation* (implementasi), *testing* (pengujian), dan *maintenance* (perawatan).

System Development Life Cycle (SDLC) adalah metodologi klasik yang digunakan untuk mengembangkan, memelihara dan menggunakan sistem informasi. Siklus hidup sistem itu sendiri merupakan metodologi, tetapi polanya lebih dipengaruhi oleh kebutuhan untuk mengembangkan sistem yang lebih cepat. Pengembangan sistem yang lebih cepat dapat dicapai dengan peningkatan siklus hidup dan penggunaan peralatan pengembangan berbasis computer.

Model SDLC yang dipakai dalam penelitian ini adalah model *Waterfall*. *Waterfall Model* atau *Classic Life Cycle* merupakan model yang paling banyak dipakai dalam *Software Engineering* (SE). Menurut Bassil (2012) disebut waterfall karena tahap demi tahap yang harus dilalui menunggu selesainya tahap sebelumnya dan berjalan berurutan.



Gambar II.1. SDLC(System Development Life Cycle)

Dalam pengembangan metode SDLC (*System Development Life Cycle*) memiliki beberapa tahapan yaitu:

1. Analisa Kebutuhan

Adapun tahap yang dilakukan pada analisa kebutuhan yaitu mengumpulkan data-data mengenai penjualan, transaksi, barang masuk dan barang keluar.

2. Desain Sistem

Sesuai penyelesaian yang akan dilakukan desain sistem yang harus ada pada perancangan aplikasi ini adalah merancang sistem menggunakan UML yaitu implementasi *use case diagram*, *class diagram*, *activity diagram* dan *sequence diagram* dengan menggunakan aplikasi *visio 2016*.

3. *Coding dan Testing*

Pada tahap ini akan dilakukan pembuatan program sesuai dengan apa yang direncanakan dalam perancangan yang akan digunakan dalam pembuatan aplikasi ini. Dan akan dilakukan pengecekan pada aplikasi apakah desain dan sistemnya dapat diimplementasikan ke dalam bahasa pemrograman dan dapat berjalan dengan baik. Jika tidak, maka kesalahan harus diperbaiki terlebih dahulu.

4. Pengujian Program

Selanjutnya dilakukan pengujian aplikasi secara menyeluruh dengan pengujian program melalui Blackbox testing dan melihat apakah sistem telah berjalan dengan baik sesuai dengan kebutuhan target dari aplikasi yang dirancang.

5. Hasil

Hasil dari sistem yang telah dirancang sebelumnya akan menghasilkan sebuah aplikasi pengembangan sistem informasi penjualan barang pada CV. Aditya Jaya Bersama dengan metode *Life Cycle*.

Berikut merupakan contoh model penerapan SDLC:

1. Model Prototype

Metode Prototype merupakan suatu paradigma baru dalam metode pengembangan perangkat lunak dimana metode ini tidak hanya sekedar evolusi dalam dunia pengembangan perangkat lunak, tetapi juga merevolusi metode pengembangan perangkat lunak yang lama yaitu sistem sekuensial yang biasa dikenal dengan nama SDLC atau *waterfall development model*. Prototype ini merupakan salah satu metode pengembangan perangkat lunak yang banyak

digunakan, dengan metode ini pengembang dan pelanggan dapat saling berinteraksi selama proses pembuatan system.

2. Model Rapid Application Development (RAD)

Rapid Application Development (RAD) adalah sebuah model proses perkembangan perangkat lunak sekuensial linier yang menekankan siklus perkembangan yang sangat pendek (kira-kira 60 sampai 90 hari). Model RAD ini merupakan sebuah adaptasi “kecepatan tinggi” dari model sekuensial linier dimana perkembangan cepat dicapai dengan menggunakan pendekatan konstruksi berbasis komponen. Rapid application development (RAD) atau rapid prototyping juga bisa disebutkan sebagai model proses pembangunan perangkat lunak yang tergolong dalam teknik incremental (bertingkat). RAD menekankan pada siklus pembangunan pendek, singkat, dan cepat. Waktu yang singkat adalah batasan yang penting untuk model ini. *Rapid application development* menggunakan metode iteratif (berulang) dalam mengembangkan sistem dimana *working model* (model kerja) sistem diinstruksikan di awal tahap pengembangan dengan tujuan menetapkan kebutuhan (*requirement*) pengguna. Model kerja digunakan hanya sesekali saja sebagai basis desain dan implementasi sistem akhir.

3. Model Spiral

Model Spiral (spiral model) adalah salah satu bentuk dari Metode Pengembangan Perangkat Lunak atau yang disebut SDLC (*Software Development Life Cycle*), yang sangat populer digunakan dalam bidang teknologi informasi. Model ini mengadaptasi dua model perangkat lunak yang ada yaitu model prototyping dengan penulangannya dan model *waterfall* dengan pengendalian dan

sistematikanya. Model ini dikenal dengan sebutan Spiral Boehm. Pengembang dalam model ini memadupadankan beberapa model umum tersebut untuk menghasilkan produk khusus atau untuk menjawab persoalan-persoalan tertentu selama proses pengerjaan proyek.

II.5. Visual Studio 2010

Microsoft Visual Studio merupakan sebuah perangkat lunak lengkap yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasi lainnya dalam bentuk aplikasi *console*, aplikasi *Windows*, ataupun aplikasi *Web*. Kompiler yang dimasukkan ke dalam paket *Visual Studio* antara lain *Visual C++*, *Visual C#*, *Visual Basic*, *Visual Basic .NET*, *Visual InterDev*, *Visual J++*, *Visual J#*, *Visual FoxPro*, dan *Visual SourceSafe*. *Microsoft Visual Studio* dapat digunakan untuk mengembangkan aplikasi dalam *native code* (dalam bentuk bahasa mesin yang berjalan di atas *Windows*) ataupun *managed code* (dalam bentuk *Microsoft Intermediate Language* di atas *.NET Framework*).

Selain itu, *Visual Studio* juga dapat digunakan untuk mengembangkan aplikasi *Silverlight*, aplikasi *Windows Mobile* (yang berjalan di atas *.NET Compact Framework*). *Visual Studio* kini telah menginjak versi *Visual Studio 12.0* atau dikenal dengan sebutan *Microsoft Visual Studio 2013* yang diluncurkan pada 17 Oktober 2013 yang ditujukan untuk *platform Microsoft .NET Framework 4.5.1*. Versi sebelumnya *Visual Studio 2012* ditujukan untuk platform 4.5, *Visual Studio 2010* ditujukan untuk *.NET Framework 4.0*, *Visual Studio 2008* ditujukan untuk

platform .NET Framework 3.5, Visual Studio 2005 ditujukan untuk platform .NET Framework 2.0 dan 3.0. Visual Studio 2003 ditujukan untuk .NET Framework 1.1, dan Visual Studio 2002 ditujukan untuk .NET Framework 1.0. Versi-versi tersebut di atas kini dikenal dengan sebutan Visual Studio.NET, karena memang membutuhkan Microsoft .NET Framework. Sementara itu, sebelum muncul Visual Studio .NET, terdapat Microsoft Visual Studio 6.0. (Ruli; 2017)

II.6. SQL Server


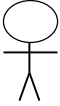
SQL Server 2008 adalah sebuah RDBMS (*Relational Database Management System*) yang sangat *powerful* dan telah terbukti kekuatannya dalam mengolah data. Dalam versi terbarunya ini, *SQL Server 2008* memiliki banyak fitur yang bisa diandalkan untuk meningkatkan *performa database*. *SQL Server 2008* memiliki suatu GUI (*Graphic User Interface*) yang kita gunakan untuk melakukan aktivitas sehari-hari berkaitan dengan database, seperti menulis T-SQL, melakukan *backup* dan *restore database*, melakukan *security database* terhadap aplikasi, dan sebagainya. Pada GUI tersebut kita bisa melakukan *settingan* terhadap *SQL Server* untuk berkerja lebih optimal. *Settingan* juga bisa dilakukan menggunakan *script* untuk memudahkan *developer* mengubah *Setting Options* pada *SQL Server 2008*. (Ruslan: 2013).




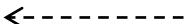
II.7. UML (*Unified Modeling Language*)

II.7.1. *Use Case Diagram*

Use case diagram adalah abstraksi dari interaksi antara *system* dan *actor*. *Use case* bekerja dengan cara mendeskripsikan tipe interaksi antara *user* sebuah *system* dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah *system* dipakai. Dengan menggunakan *use case*, proses sistem yang ada akan lebih dapat digambarkan dan dikomunikasikan kepada pemakai sistem sehingga pengguna bekerja efektif. Simbol-simbol yang digunakan dalam *Use Case Diagram* yaitu:

Tabel II.1. Tabel Simbol *Use Case Diagram*

Gambar	Keterangan
	<i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i> .
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan

	<i>use case</i> , tetapi tidak memiliki control terhadap <i>use case</i> .
	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.




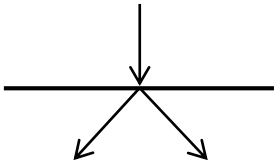
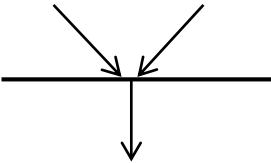
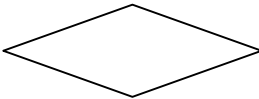
(Sumber: Ade Hendini; 2016)

II.7.2. Activity Diagram

Activity diagram adalah diagram yang menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis yang perlu diperhatikan adalah bahwa *activity diagram* menggambarkan aktifitas sistem bukan apa yang dilakukan aktor, jadi aktifitas yang dapat dilakukan oleh sistem. *Activity diagram* juga menyediakan pendekatan untuk proses pemodelan paralel. Pada dasarnya

diagram *Activity* sering digunakan oleh *flowchart*, diagram *Activity* berfokus pada aktifitas-aktifitas yang terjadi yang terkait dalam suatu proses tunggal. Jadi dengan kata lain, diagram ini menunjukkan bagaimana aktifitas-aktifitas tersebut bergantung satu sama lain. Simbol-simbol yang digunakan dalam *activity diagram*, yaitu:

Tabel II.2. Tabel Simbol *Activity Diagram*

Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activates</i> , menggambarkan suatu proses/ kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan pararel menjadi satu.
	<i>Join</i> (penggabungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true, false</i> .

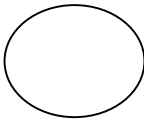
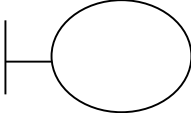
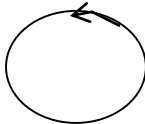
New Swimline	<i>Swimlane</i> , pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa.
--------------	--------------------------------------------------------------------------------------------


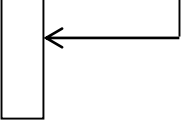


(Sumber: Ade Hendini; 2016)

II.7.3. Sequence Diagram

Sequence Diagram merupakan salah satu *diagram Interaction* yang menjelaskan bagaimana suatu operasi itu dilakukan; *message* (pesan) apa yang dikirim dan kapan pelaksanaannya. Diagram ini diatur berdasarkan waktu. Obyek-obyek yang berkaitan dengan proses berjalannya operasi diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang terurut. Simbol-simbol yang digunakan dalam *sequence diagram*, yaitu:

Tabel II.3. Tabel Simbol Sequence Diagram

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>formentry</i> dan <i>form</i> cetak
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada

	entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> , <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

(Sumber: Ade Hendini; 2016)

II.7. Sejarah Singkat CV. Aditya Jaya Bersama

CV. Aditya Jaya Bersama didirikan oleh bapak Budi Pranoto pada tanggal 01 September 2017, Lokasi Jl.Musyawarah D Desa Saentis yang bermula melayani jasa pemasangan atap baja ringan dan plafon. Berbekal kemampuan dan pengetahuan serta wawasan yang luas dalam bidang pemasaran jasa layanan perusahaan mampu berkembang dari tahun ke tahun. CV. Aditya Jaya Bersama mengalami jatuh bangun dalam membangun usahanya. Hingga pada akhirnya perusahaan mampu mengembangkan usahanya menjual produk diantaranya gypsum, panel, furing, PVC, bermacam jenis cat dan ukuran, dll. Saat ini, CV. Aditya Jaya Bersama telah mengembangkan proses bisnisnya sebagai kontraktor. Dimana proyek-proyek pemasangan rangka atap baja ringan dan plafon yang ditanganinya mencakup sebagian wilayah Sumatera Utara. Hal tersebut tidak lepas dari relasi yang dimiliki oleh bapak Budi Pranoto.