BAB 2
TINJAUAN PUSTAKA

2.1 Permainan (*Game)*

Permainan merupakan sebuah aktivitas rekreasi dengan tujuan bersenang-senang ataupun untuk mengisi waktu luang. Permainan biasanya dilakukan secara sendiri-sendiri ataupun secara kelompok. Permainan sendiri terbagi menjadi 2 bagian yaitu [5]:

1. Permainan Anak-Anak

Banyak permainan yang sering dilakukan oleh anak-anak bersama teman-teman mereka disekitar rumah mereka. Mereka biasanya berkelompok, senang berlarian, ataupun kadang duduk melingkar untuk memainkan salah satu permainan, dan dari situlah biasanya akan tercipta suatu keakraban diantara mereka. Beberapa permainan dari masa lalu disebut dengan permainan tradisional, sedangkan untuk permainan yang muncul di jaman sekarang ini disebut dengan permainan modern dan biasanya menggunakan peralatan canggih seperti halnya *video game*. Dan berikut beberapa jenis permainan yang bisa dimainkan oleh anak-anak [5] :

a. Congklak

b. Layangan

c. Petak Umpet

d. Boneka

e. Adu Kelereng

f. Bentengan.

2. Permainan Dewasa

Untuk permainan orang dewasa biasanya memerlukan pemikiran untuk memainkannya. Dan berikut beberapa jenis permainan yang bisa dimainkan oleh orang dewasa :

a. Kartu remi

b. Domino

c. Karambol

d. Catur

2.1.1 Permainan Video *(Video Game)*

Permainan video *game* adalah permainan yang menggunakan interaksi antarmuka pengguna melalui gambar yang dihasilkan oleh piranti video. Permainan video umumnya menyediakan sistem penghargaan, misalnya *score* yang dihitung berdasarkan tingkat keberhasilan yang dicapai dalam menyelesaikan tugas-tugas yang ada di dalam permainan video tersebut. Sistem elektronik yang digunakan untuk menjalankan permainan video disebut *platform*, contohnya adalah komputer pribadi (*personal computer*) ataupun konsol permainan (*game console*) [5].

1. Permainan Komputer *(Computer Game)*

Permainan komputer adalah permainan video yang di mainkan pada komputer pribadi, dan bukan pada konsol permainan. Permainan komputer telah berevolusi dari sistem grafis sederhana menjadi sangat kompleks dan mutakhir.Permainan *onlin*e (*online game*) adalah jenis permainan video atau permainan komputer dengan menggunakan jaringan internet sebagai media interaksinya.

Permainan *online* terdiri dari dua unsur utama, yaitu *server* dan *client*. *Server* adalah penyedia layanan gaming yang merupakan basis agar *client –client*

yang terhubung dapat memainkan permainan dapat melakukan komunikasi dengan baik. Suatu *server* pada prinsipnya hanya melakukan administrasi permainan dan menghubungkan *client-client*. Sedangkan *client* adalah pengguna permainan dan memakai kemampuan *server*. Contoh permainan online adalah *Ragnarok Online, DoTA, World of Warcraft* .

2. Konsol Permainan *(Game Console)*

Konsol permainan adalah sebuah mesin elektronik yang dirancang khusus untuk memainkan permainan video. Perangkat keluarannya biasa berupa monitor komputer atau televisi dan alat masukan utamanya berupa sebuah pengendali. Konsol permainan yang pertama kali dibuat adalah *Atari*, kemudian dilanjutkan dengan *Nintendo* yang sukses pada tahun 1985-1989. Konsol permainan modern sekarang ini antara lain adalah *Playstation* buatan perusahaan *Sony*. Untuk *Console* yang berbentuk kecil dan dapat dibawah kemana-mana biasa disebut *Portable Console* [5].

2.1.2 Klasifikasi Permainan *(Game)*

Sejarah *game* tidak lepas dengan namanya *platform*, dari yang awalnya hanya mesin *Arcade* (Mesin ding dong) hingga sampai sekarang ini yang sudah mencapai era *Mobile Games*. Untuk lebih jelasnya, berikut ini adalah klasifikasi *game* berdasarkan jenis *Platform* yang di gunakan [5]:

1. *Arcade Games* yaitu yang sering disebut ding-dong di Indonesia, biasanya berada di daerah khusus dan memiliki box atau mesin yang memang khusus didesain untuk jenis video games tertentu dan tidak jarang bahkan memiliki fitur yang dapat membuat pemainnya lebih merasa "masuk" dan menikmati. Seperti pistol, kursi khusus, sensor gerakan, sensor injakkan dan stir mobil (beserta transmisinya).

2. *PC Games (Personal Computer)* yaitu video game yang dimainkan menggunakan *Personal Computers*.

3. *Console Games* yaitu video game yang dimainkan menggunakan *console* tertentu seperti *playstation XBOX*, *Nitendo*, dan lan-lain.

4. *Handheld Game* yaitu *game* yang dimainkan di console khusus video game yang dapat dibawa kemana-mana, contoh *nitendo DS dan Sony PSP*.

5. *Mobile Games* yaitu game yang biasa dimainkan khusus untuk *handphone atau PDA*.

Klasifikasi permainan berdasarkan pembatasan usia dari pemainnya, menurut ***ESRB*** (***Entertainment Software Rating Board***) adalah sebagai berikut [5]:

1. ***Early Childhood (3+)*** memiliki isi yang mungkin cocok untuk usia 3 keatas. Tidak mengandung materi yang tidak pantas seperti adegan kekerasan.

2. ***Everyone (6+)*** memiliki konten yang mungkin cocok untuk usia 6 keatas. Judul-judul dalam kategori ini mungkin berisi sedikit kartun, fantasi atau kekerasan ringan dan tidak jarang menggunakan bahasa yang ringan.

3. ***Everyone (10+)*** memiliki konten yang mungkin cocok untuk usia 10 keatas. Judul-judul dalam kategori ini mungkin berisi lebih banyak kartun, fantasi atau kekerasan ringan, bahasa ringan dan juga minimal tema sugestif.

4. ***Teen (13+)*** memiliki konten yang mungkin cocok untuk usia 13 keatas. Judul-judul dalam kategori ini mungkin mengandung kekerasan, tema sugestif, humor kasar, sedikit darah, simulasi perjudian, dan tidak jarang menggunakan bahasa yang kuat.

5. ***Mature (17+)*** memiliki konten yang mungkin cocok untuk orang usia 17 dan lebih tua. Judul-judul dalam kategori ini mungkin berisi intens kekerasan, darah, konten seksual dan bahasa yang kuat. *Mature* (17+) memiliki konten yang mungkin cocok untuk orang usia 17 dan lebih tua. Judul judul dalam kategori ini mungkin berisi intens kekerasan, darah, konten seksual dan bahasa yang kuat.

6. ***Adults Only (18+)*** memiliki konten yang hanya boleh dimainkan oleh orang-orang 18 tahun keatas. Judul-judul dalam kategori ini mungkin termasuk adegan yang intens, kekerasan dan juga mengandung konten seksual.
7. ***RP* (*Rating Pending*)** telah disampaikan kepada ESRB dan sedang menunggu penilaian akhir. (Simbol ini hanya muncul dalam iklan sebelum permainan dibebaskan).

2.1.3 Jenis Permaninan *(Genre Game)*

Game dari tahun-tahun semakin banyak saja jenisnya, teknologi yang dipakaipun semakin canggih. Dari ratusan atau bahkan ribuan *game* yang telah bermunculan di dunia, *game-game* tersebut dikelompokkan menjadi beberapa *genre*. Pada dasarnya, *genre-genre game* yang beredar di pasaran dapat dikelompokkan sebagai berikut :

a. Advergames

Jenis game yang secara eksplisit maupun secara menyeluruh memasukkan muatan promosi sebagai marketing *tools* ke dalam sebuah *game*. Pada *advergame*, *brand* menyatu dengan cerita, misi maupun aktifitas lain dalam game. Penerapan *advergames* ini dapat terlihat pada beberapa unsur games seperti karakter dengan *brand* tertentu,*gameplay* menampilkan ciri khas produk tertentu, banner iklan pada suatu segmen *game*, ataupun unsur lainnya.

b. Action Game

Adalah tipe *game* dengan fitur utama berupa banyaknya aksi di mana pemain harus memiliki keterampilan reaksi yang cepat untuk menghindari musuh atau menghindari rintangan. Pengembang game tipe ini perlu memastikan *game* yang dibuat dioptimasi sehingga pemain memiliki pengalaman bermain yang baik, yang tidak terganggu oleh *delay* proses yang lama, contohnya : *Metal Gear Solid*.

c. Adventure Game

Adalah tipe *game* yang umumnya membuat pemain harus berjalan mengelilingi suatu tempat yang terkondisi, seperti sebuah istana, gua yang berkelok, dan planet yang jauh. Pemain melakukan navigasi suatu area mencari pesan-pesan rahasia, memperoleh obyek yang memiliki kemampuan yang bervariasi, bertempur dengan musuh, dan lain-lain. Untuk membuat *game* ini, diperlukan perencanaan yang akurat sehingga memiliki alur cerita yang menarik bagi pemain, contohnya : *Tomb Raider, Prince of Persia, Hitman*, dan lain-lain.

d. Fighting Game

Adalah tipe *game* yang intinya harus menjatuhkan lawan tandingnya, entah itu dengan pukulan, tendangan, combo, maupun dengan jurus special, contohnya : *Tekken, Mortal Kombat, Naruto*, dan lain-lain.

e. FPS (First Person Shooter)

Adalah tipe *game* yang mengunakan sudut pandang orang pertama untuk membidik atau membunuh musuh, sehingga kita hanya melihat tangannya saja dan tidak melihat tubuh karakter yang dimainkan, contohnya : *Call of Duty, Half Life, Crysis*, dan lain-lain.

f. RPG (Role Playing Game)

Adalah tipe *game* yang seringkali berupa multi karakter *game* di mana setiap pemain memiliki karakter dengan kemampuan, kekuatan, dan kelemahan yang spesifik. Para pemain saling berkompetisi, berinteraksi, dan bertempur satu sama lain. Tampilan grafis yang khas untuk setiap karakter pemain ditambah dengan *storyline* yang mendebarkan akan sangat menarik dan memberikan pengalaman yang berbeda di dalam bermain, contohnya : *Final Fantasy, Warcraft*, dan lain-lain.

g. Platform Game

Adalah tipe *game* yang mengharuskan pemain mengarahkan suatu obyek dengan melalui berbagai tahap atau tingkatan area untuk menyerang musuh dan menghindar terhadap serangan. Tipe *game* ini sedikit serupa dengan *action game*, tetapi aksinya tidak secepat *action game*. Teknik *collision detection*

sangat sering dimanfaatkan pada tipe *game* ini, contohnya : *Sonic The Hedgehog, Mario Bros, Metal Slug*.

h. Puzzle Game

Adalah tipe *game* yang umumnya membuat pemain menggunakan kemampuan berpikirnya sebagai pengganti keterampilan reaksi yang cepat karena terdapat rahasia yang perlu dipecahkan. *Game* ini lebih bersifat statis dibanding *action game*. Pembuatan *game* tipe ini seringkali ditunjang dengan AI (*Artificial Intelligence*), contohnya : *7 Wonders of The Ancient World.*

i. Racing Game

Adalah tipe *game* yang tujuannya adalah mencapai garis finish dari suatu *race*, dalam *game* ini biasanya pemain dapat memilih dan membeli kendaraan, mendandani, melakukan upgrade mesin, contohnya: *Need For Speed.*

j. Simulation Game

Adalah tipe *game* yang memberikan pengalaman atau interaksi sedekat mungkin dengan keadaan yang aslinya, meskipun terkadang keadaan tersebut masih eksperimen atau bahkan fiktif, tapi ada penekanan khusus pada detail dan pengalaman realistik menggunakan keadaan tersebut, contohnya : *Gran Turismo.*

k. Sandbox Game

Adalah tipe *game* yang umumnya ditujukan untuk menjelajahi suatu kota atau tempat dan bebas berinteraksi dengan objek disekitarnya. Mungkin *game* ini bisa di bilang mirip *RPG* tapi *Sandbox* sangat berbeda dengan *RPGgame*, perbedaannya adalah tidak adanya peningkatan level karakternya, contohnya : *GTA dan Bully*.

l. RTS (Real Time Strategy)

Adalah tipe *game* yang mengharuskan pemainnya menggunakan taktik dan strategi untuk jeli dalam melihat setiap peluang, kelemahan musuh dan bijaksana dalam menggunakan sumber daya yang ada, contohnya : *Command and Conquer.*

m. Sport Game

Adalah tipe *game* yang berupa kompetisi antara dua pemain atau lebih, di mana pemain dapat berupa individual atau tim. Contoh *game* tipe ini antara lain sepakbola, bola basket, tenis, dan bilyard. Tergantung seberapa cepat permainan yang terjadi, aplikasi *game* perlu dioptimalkan, contohnya : *Pro Evolution Soccer.*

2.1.4 Sejarah Ramayana

Dikisahkan di sebuah negeri bernama Mantili ada seorang puteri nan cantik jelita bernama Dewi Shinta. Dia seorang puteri raja negeri Mantili yaitu Prabu Janaka. Suatu hari sang Prabu mengadakan sayembara untuk mendapatkan sang Pangeran bagi puteri tercintanya yaitu Shinta, dan akhirnya sayembara itu dimenangkan oleh Putera Mahkota Kerajaan Ayodya, yang bernama Raden Rama Wijaya. Namun dalam kisah ini ada juga seorang raja Alengkadiraja yaitu Prabu Rahwana, yang juga sedang kasmaran, namun bukan kepada Dewi Shinta tetapi dia ingin memperistri Dewi Widowati. Dari penglihatan Rahwana, Shinta dianggap sebagai titisan Dewi Widowati yang selama ini diimpikannya. Dalam sebuah perjalanan Rama dan Shinta dan disertai Lesmana adiknya, sedang melewati hutan belantara yang dinamakan hutan Dandaka, si raksasa Prabu Rahwana mengintai mereka bertiga, khususnya Shinta. Rahwana ingin menculik Shinta untuk dibawa ke istananya dan dijadikan istri, dengan siasatnya Rahwana mengubah seorang hambanya bernama Marica menjadi seekor kijang kencana. Dengan tujuan memancing Rama pergi memburu kijang ‘jadi-jadian' itu, karena Dewi Shinta menginginkannya. Dan memang benar setelah melihat keelokan kijang tersebut, Shinta meminta Rama untuk menangkapnya. Karena permintaan sang istri tercinta

maka Rama berusaha mengejar kijang seorang diri sedang Shinta dan Lesmana menunggui.

Dalam waktu sudah cukup lama ditinggal berburu, Shinta mulai mencemaskan Rama, maka meminta Lesmana untuk mencarinya. Sebelum meninggalkan Shinta seorang diri Lesmana tidak lupa membuat perlindungan guna menjaga keselamatan Shinta yaitu dengan membuat lingkaran magis. Dengan lingkaran ini Shinta tidak boleh mengeluarkan sedikitpun anggota badannya agar tetap terjamin keselamatannya, jadi Shinta hanya boleh bergerak-gerak sebatas lingkaran tersebut. Setelah kepergian Lesmana, Rahwana mulai beraksi untuk menculik, namun usahanya gagal karena ada lingkaran magis tersebut. Rahwana mulai cari siasat lagi, caranya ia menyamar yaitu dengan mengubah diri menjadi seorang brahmana tua dan bertujuan mengambil hati Shinta untuk memberi sedekah. Ternyata siasatnya berhasil membuat Shinta mengulurkan tangannya untuk memberi sedekah, secara tidak sadar Shinta telah melanggar ketentuan lingkaran magis yaitu tidak diijinkan mengeluarkan anggota tubuh sedikitpun! Saat itu juga Rahwana tanpa ingin kehilangan kesempatan ia menangkap tangan dan menarik Shinta keluar dari lingkaran. Selanjutnya oleh Rahwana, Shinta dibawa pulang ke istananya di Alengka. Saat dalam perjalanan pulang itu terjadi pertempuran dengan seekor burung Garuda yang bernama Jatayu yang hendak menolong Dewi Shinta. Jatayu dapat mengenali Shinta sebagai puteri dari Janaka yang merupakan teman baiknya, namun dalam pertempuan itu Jatayu dapat dikalahkan Rahwana.

Disaat yang sama Rama terus memburu kijang kencana dan akhirnya Rama berhasil memanahnya, namun kijang itu berubah kembali menjadi raksasa. Dalam wujud sebenarnya Marica mengadakan perlawanan pada Rama sehingga terjadilah pertempuran antar keduanya, dan pada akhirnya Rama berhasil memanah si raksasa. Pada saat yang bersamaan Lesmana berhasil menemukan Rama dan mereka berdua kembali ke tempat semula dimana Shinta ditinggal sendirian, namun sesampainya Shinta tidak ditemukan. Selanjutnya mereka berdua berusaha mencarinya dan bertemu Jatayu yang luka parah, Rama mencurigai Jatayu yang menculik dan dengan penuh emosi ia hendak membunuhnya tapi berhasil dicegah

oleh Lesmana. Dari keterangan Jatayu mereka mengetahui bahwa yang menculik Shinta adalah Rahwana! Setelah menceritakan semuanya akhirnya si burung garuda ini meninggal.

Mereka berdua memutuskan untuk melakukan perjalanan ke istana Rahwana dan ditengah jalan mereka bertemu dengan seekor kera putih bernama Hanuman yang sedang mencari para satria guna mengalahkan Subali. Subali adalah kakak dari Sugriwa paman dari Hanuman, Sang kakak merebut kekasih adiknya yaitu Dewi Tara. Singkat cerita Rama bersedia membantu mengalahkan Subali, dan akhirnya usaha itu berhasil dengan kembalinya Dewi Tara menjadi istri Sugriwa. Pada kesempatan itu pula Rama menceritakan perjalanannya akan dilanjutkan bersama Lesmana untuk mencari Dewi Shinta sang istri yang diculik Rahwana di istana Alengka. Karena merasa berutang budi pada Rama maka Sugriwa menawarkan bantuannya dalam menemukan kembali Shinta, yaitu dimulai dengan mengutus Hanuman persi ke istana Alengka mencari tahu Rahwana menyembunyikan Shinta dan mengetahui kekuatan pasukan Rahwana.

Taman Argasoka adalah taman kerajaan Alengka tempat dimana Shinta menghabiskan hari-hari penantiannya dijemput kembali oleh sang suami. Dalam Argasoka Shinta ditemani oleh Trijata kemenakan Rahwana, selain itu juga berusaha membujuk Shinta untuk bersedia menjadi istri Rahwana. Karena sudah beberapa kali Rahwana meminta dan ‘memaksa' Shinta menjadi istrinya tetapi ditolak, sampai-sampai Rahwana habis kesabarannya yaitu ingin membunuh Shinta namun dapat dicegah oleh Trijata. Di dalam kesedihan Shinta di taman Argasoka ia mendengar sebuah lantunan lagu oleh seekor kera putih yaitu Hanuman yang sedang mengintainya. Setelah kehadirannya diketahui Shinta, segera Hanuman menghadap untuk menyampaikan maksud kehadirannya sebagai utusan Rama. Setelah selesai menyampaikan maskudnya Hanuman segera ingin mengetahui kekuatan kerajaan Alengka. Caranya dengan membuat keonaran yaitu merusak keindahan taman, dan akhirnya Hanuman tertangkap oleh Indrajid putera Rahwana dan kemudian dibawa ke Rahwana. Karena marahnya Hanuman akan dibunuh tetapi dicegah oleh Kumbakarna adiknya, karena dianggap menentang, maka Kumbakarna diusir dari kerjaan Alengka. Tapi akhirnya Hanuman tetap dijatuhi

hukuman yaitu dengan dibakar hidup-hidup, tetapi bukannya mati tetapi Hanuman membakar kerajaan Alengka dan berhasil meloloskan diri. Sekembalinya dari Alengka, Hanuman menceritakan semua kejadian dan kondisi Alengka kepada Rama. Setelah adanya laporan itu, maka Rama memutuskan untuk berangkat menyerang kerajaan Alengka dan diikuti pula pasukan kera pimpinan Hanuman.

Setibanya di istana Rahwana terjadi peperangan, dimana awalnya pihak Alengka dipimpin oleh Indrajid. Dalam pertempuran ini Indrajid dapat dikalahkan dengan gugurnya Indrajit. Alengka terdesak oleh bala tentara Rama, maka Kumbakarna raksasa yang bijaksana diminta oleh Rahwana menjadi senopati perang. Kumbakarna menyanggupi tetapi bukannya untuk membela kakaknya yang angkara murka, namun demi untuk membela bangsa dan negara Alengkadiraja.Dalam pertempuran ini pula Kumbakarna dapat dikalahkan dan gugur sebagai pahlawan bangsanya. Dengan gugurnya sang adik, akhirnya Rahwana menghadapi sendiri Rama. Pad akhir pertempuran ini Rahwana juga dapat dikalahkan seluruh pasukan pimpinan Rama. Rahmana mati kena panah pusaka Rama dan dihimpit gunung Sumawana yang dibawa Hanuman.

Setelah semua pertempuran yang dasyat itu dengan kekalahan dipihak Alengka maka Rama dengan bebas dapat memasuki istana dan mencari sang istri tercinta. Dengan diantar oleh Hanuman menuju ke taman Argasoka menemui Shinta, akan tetapi Rama menolak karena menganggap Shinta telah ternoda selama Shinta berada di kerajaan Alengka. Maka Rama meminta bukti kesuciannya, yaitu dengan melakukan bakar diri. Karena kebenaran kesucian Shinta dan pertolongan Dewa Api, Shinta selamat dari api. Dengan demikian terbuktilah bahwa Shinta masih suci dan akhirnya Rama menerima kembali Shinta dengan perasaan haru dan bahagia. Dan akhir dari kisah ini mereka kembali ke istananya masing-masing [2].

2.2 Kecerdasan Buatan / AI (Artificial Intelligence)

Kecerdasan buatan sering disebut juga dengan Artificial Intelligence (AI) merupakan salah satu bagian ilmu komputer yang mempelajari tentang bagaimana caranya agar komputer dapat melakukan pekerjaan seperti yang dapat dilakukan oleh manusia. Pada awal diciptakannya, komputer hanya difungsikan sebagai alat hitung saja. Namun seiring dengan perkembangan zaman, teknologi komputer

semakin ditingkatkan dan peran komputer semakin mendominasi kehidupan umat manusia. Komputer tidak lagi hanya digunakan sebagai alat hitung, melainkan komputer diharapkan dapat diberdayakan untuk mengerjakan hal-hal yang dapat memudahkan pekerjaan manusia.

Manusia bisa menjadi pandai dalam menyelesaikan segala permasalahan di dunia ini karena manusia mempunyai pengetahuan dan pengalaman. Pengetahuan dapat diperoleh dari proses belajar. Semakin banyak bekal pengetahuan yang dimiliki oleh seseorang tentu saja diharapkan akan lebih baik dalam menyelesaikan permasalahan. Namun bekal pengetahuan saja tidak cukup, manusia juga diberi akal untuk melakukan penalaran, mengambil kesimpulan berdasarkan pengetahuan dan pengalaman yang dimiliki. Tanpa memiliki kemampuan menalar yang baik, manusia dengan segudang pengalaman dan pengetahuan tidak akan dapat menyelesaikan masalah dengan baik, demikian pula dengan kemampuan menalar yang sangat baik, namun tanpa bekal pengetahuan dan pengalaman yang memadai, manusia juga tidak akan bisa menyelesaikan masalah dengan baik.

Agar komputer bisa bertindak seperti apa yang dapat dilakukan manusia, maka komputer juga harus diberi bekal pengetahuan dan mempunyai kemampuan untuk menalar. Untuk itu AI akan mencoba untuk memberikan beberapa metode untuk membekali komputer dengan kedua komponen tersebut agar komputer bisa menjadi mesin yang pintar [6].

2.2 OOP (Object Oriented Programming)

Metodologi berorientasi objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya. Metodologi berorientasi objek merupakan suatu cara bagaimana sistem perangkat lunak dibangun melalui pendekatan objek secara sistematis. Metode berorientasi objek didasarkan pada penerapan prinsip-prinsip pengelolaan kompleksitas. Metode berorientasi objek meliputi rangkaian aktivitas analisis berorientasi objek, perancangan berorientasi objek, pemrograman berorientasi objek dan pengujian berorientasi objek [7].

Dengan menggunakan OOP maka dalam melakukan pemecahan suatu masalah tidak harus melihat bagaimana cara menyelesaikan suatu masalah tersebut tetapi objek-objek apa yang dapat melakukan pemecahan masalah tersebut. Sebagai contoh seseorang yang memiliki sebuah departemen yang memiliki manager, sekretaris, petugas administrasi data dan lainnya. Misal manager tersebut ingin memperoleh data dari bag administrasi maka manager tersebut tidak harus mengambilnya langsung tetapi dapat menyuruh petugas bagian administrasi untuk mengambilnya. Pada kasus tersebut seorang manager tidak harus mengetahui bagaimana cara mengambil data tersebut tetapi manager bisa mendapatkan data tersebut melalui objek petugas adminiistrasi. Jadi untuk menyelesaikan suatu masalah dengan kolaborasi antar objek-objek yang ada karena setiap objek memiliki deskripsi tugasnya sendiri.

Pemrograman orientasi-objek menekankan konsep berikut:

1. **Kelas (*Class*)** adalah penggambaran satu set objek yang memiliki atribut dan behavior yang sama. Kelas mirip tipe data pada pemograman non objek,tapi lebih komprehensif karena terdapat struktur sekaligus karakteristiknya. Kita dapat membentuk kelas baru yang lebih spesifik dari kelas *general*-nya. Kelas dan objek merupakan jantung dari pemograman berorientasi objek. Untuk menghasilkan program jenis ini sangat penting selalu berfikir dalam konsep objek.

2. **Abstraksi (*Abstract*) -** Kemampuan sebuah program untuk melewati aspek informasi yang diproses olehnya, yaitu kemampuan untuk memfokus pada inti. Setiap objek dalam sistem melayani sebagai model dari "pelaku" abstrak yang dapat melakukan kerja, laporan dan perubahan keadaannya, dan berkomunikasi dengan objek lainnya dalam sistem, tanpa mengungkapkan bagaimana kelebihan ini diterapkan. Proses, fungsi atau metode dapat juga dibuat abstrak, dan beberapa teknik digunakan untuk mengembangkan sebuah pengabstrakan.

3. **Objek (*Object*)** adalah teknik dalam menyelesaikan masalah yang kerap muncul dalam pengembangan perangkat lunak. Teknik ini merupakan titik

kulminasi dalam menemukan cara yang efektif dalam membangun sistem dan menjadi metode yang paling banyak dipakai oleh para pengembang perangkat lunak saat ini. Orientasi objek merupakan teknik pemodelan sistem riil yang berbasis objek. Inti dari konsep ini adalah objek yang merupakan model dari sistem nyata.

4. **Enkapsulasi (*Encapsulation*)** adalah penggabungan potongan informasi dan perilaku spesifik yang bekerja pada informasi tersebut, kemudian mengemasnya menjadi apa yang disebut objek. Dalam perbankan kita mengenal objek rekening memiliki perilaku misalnya buka, tutup, penarikan, penyimpanan, ubah nama, ubah alamat dan sebagainya.

5. **Polimorfisme (*Polimorfism*)** adalah suatu fungsionalitas yang diimplementasikan dengan berbagai cara yang berbeda. Pada terminologi berorientasi objek, ini berarti kita dapat memiliki berbagai implemntasi untuk sebagian fungsionalitas tertentu. Sebagai contoh, misalkan kita akan mengembangkan sistem berbasis grafis. Saat pengguna mau menggambar sesuatu, misalnya garis atau lingkaran, sistem akan memunculkan perintah gambar. Manfaat dari polimorfisme adalah kemudahan pemeliharaanya. Jika perlu menambahkan gambar baru sedangkan fungsi umumnya tidak mengalami perubahan.

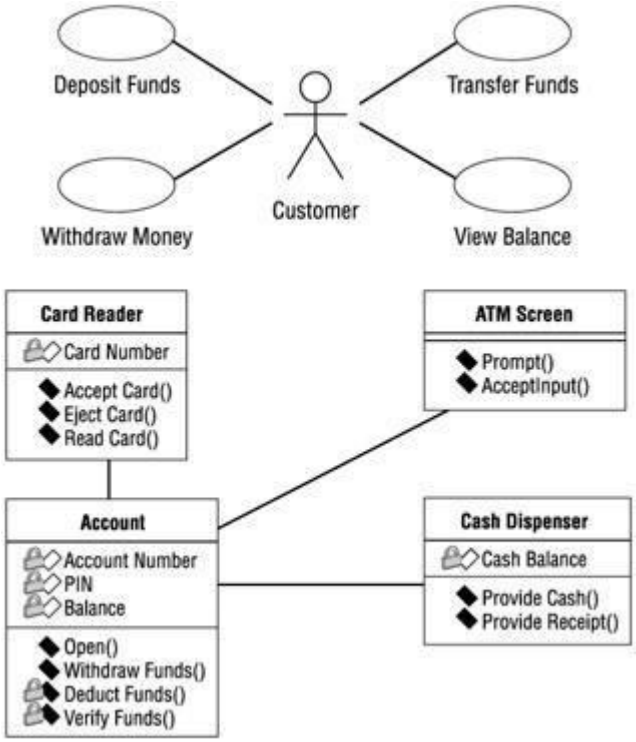
6. **Inheritas (*Inheritance*)** konsep dimana metode dan atribut yang ditentukan didalam sebuah objek kelas dapat diwariskan atau digunakan lagi oleh objek kelas lainnya. Sedangkan generalisasi merupakan teknik dimana atribut dan perilaku yang umum pada beberapa tipe kelas objek, dikelompokan kedalam kelasnya sendiri. Metode dan atribut kelas objek *supertype* kemudian diwariskan oleh kelas objek tersebut [7].

2.3 UML (Unified Modeling Language)

UML (Unified Modeling Language) adalah salah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat

analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek [8].

Pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language (UML).* UML muncul karena adanya kebutuhan pemodelan visual untuk mendefinisikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.



Gambar 2.1 Contoh Notasi UML

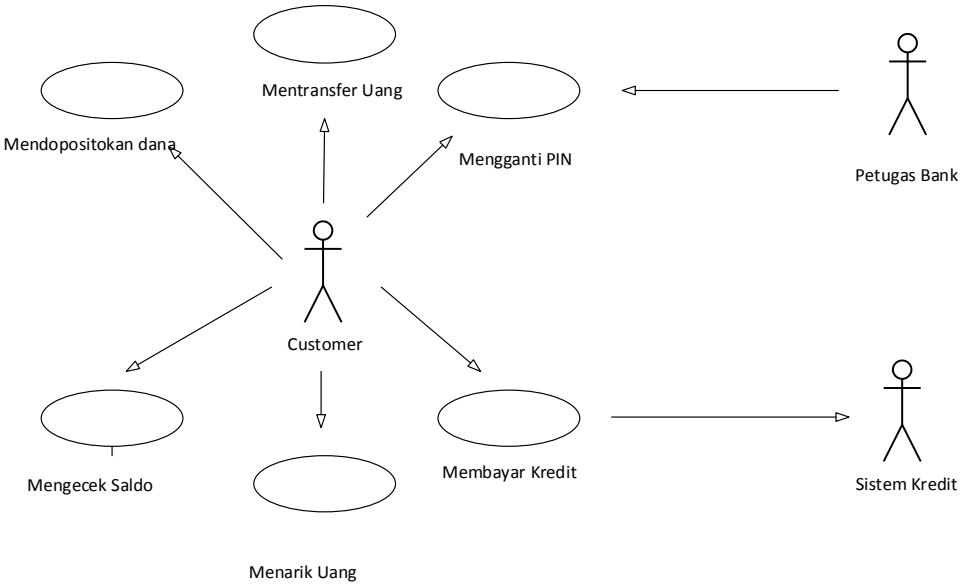
UML menyediakan beberapa diagram visual yang menunjukan berbagai jenis aspek dalam sistem. Ada beberapa diagram yang disediakan dalam UML antara lain :

1. Diagram *use case* (*use case diagram*)
2. Diagram aktivitas (*activity diagram*)

3. Diagram sekuensial (*sequence diagram)*
4. Diagram kolaborasi (*collaboration diagram*)
5. Diagram kelas (*class diagram*)
6. Diagram *statechart* (*statechart diagram*)
7. Diagram komponen (*component diagram*)
8. Diagram *deployment* (*deployment diagram*)

2.3.1 Diagram Use Case

Diagram *use case* menyajikan interaksi antara *use case* dan aktor. Dimana, aktor dapat berupa orang, peralatan, atau sistem lain yang berinteraksi dengan sistem yang sedang dibangun. *Use case* menggambarkan fungsionalitas sistem atau persyaratan yang harus dipenuhi sistem dari pandangan pemakai.



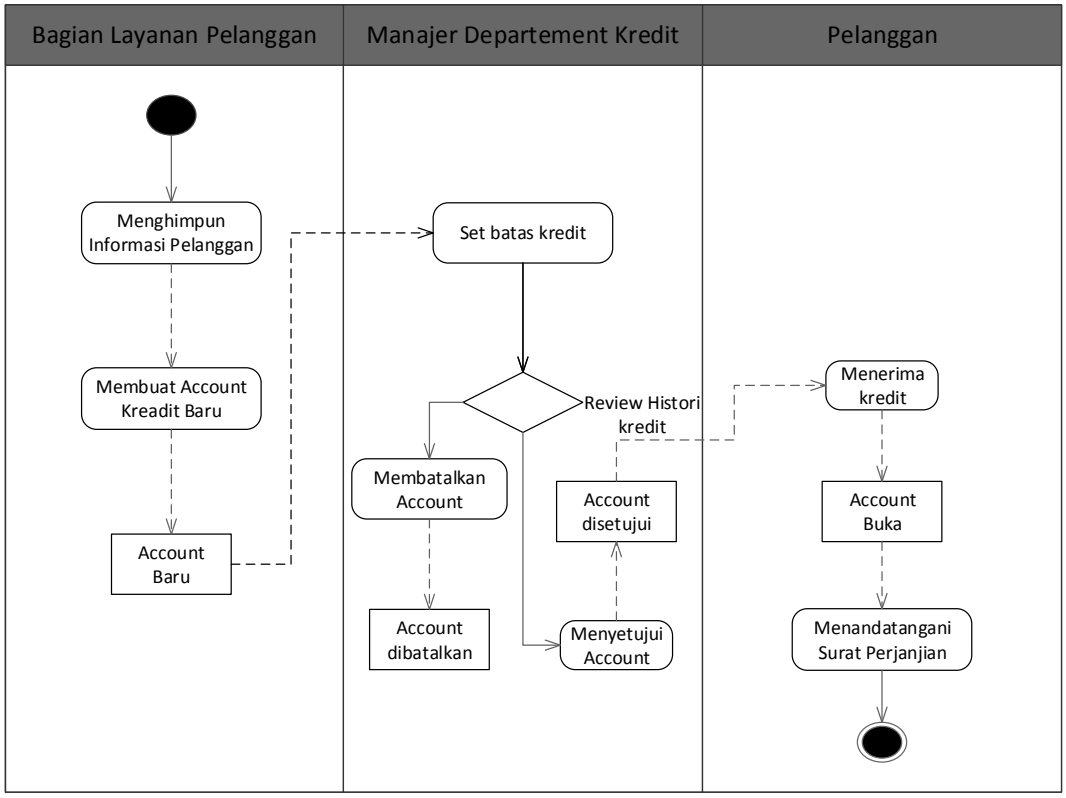
Gambar 2.2 Diagram Use case

Diagram use case ini menunjukan interaksi antara use case dan untuk sistem ATM. Pada contoh ini, aktor pelanggan menggunakan beberapa use case, antara lain: menarik uang, mendepositkan dana mentransfer uang, mengecek saldo,

membayar kredit, dan mengganti PIN. Petugas bank dapat menggati PIN pelanggan. Use case membayar kredit memberikan arah panah ke aktor Sistem Kredit, dimana merupakan aktor berupa sistem lain yang menerima informasi/data dari sistem ATM.

2.3.2 Diagram Aktivitas

Diagram aktivitas menggambarkan aliran fungsionalitas sistem. Pada tahap pemodelan bisnis, diagram aktivitas dapat digunakan untuk menunjukan aliran kerja bisnis. Dapat juga digunakan untuk menggambarkan aliran kejadian dalam *use case*.



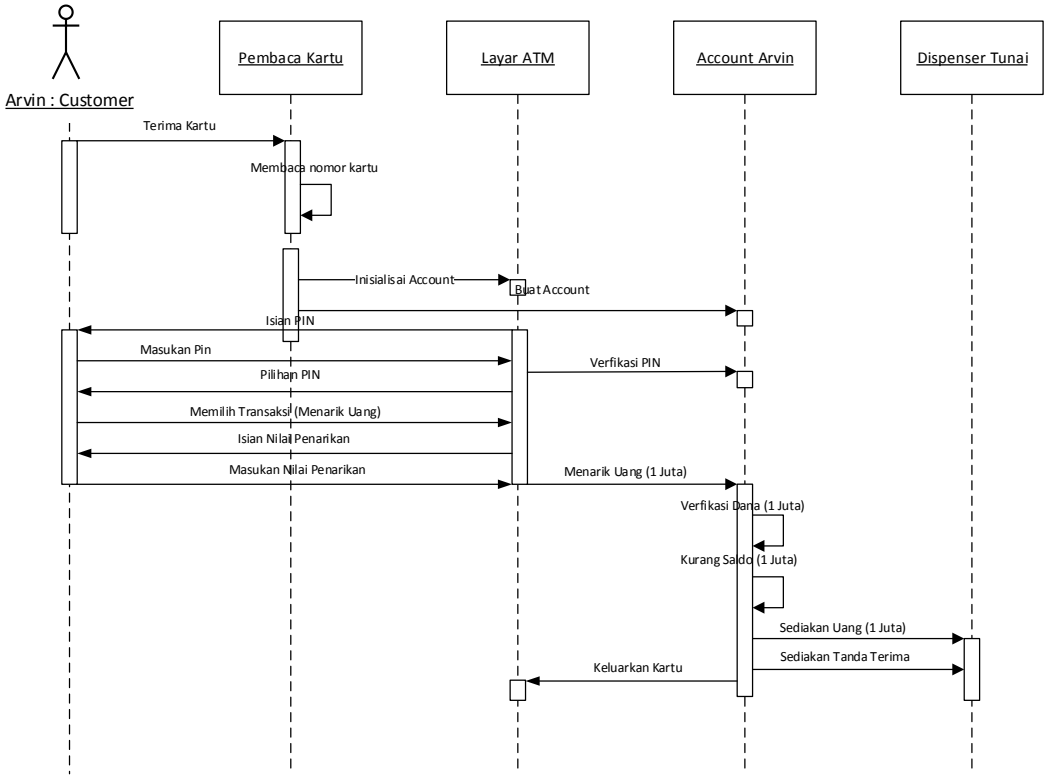
Gambar 2.3 Diagram Aktifitas

Dimana aktivitas dalam diagram dipresentasikan dengan bentuk bujur sangkar bersudut tidak lancip, yang di dalamnya berisi langkah-langkah apa saja yang terjadi dalam aliran kerja. Ada sebuah keadaan mulai yang menunjukan

dimulainya aliran kerja, dan sebuah keadaan selesai yang menunjukan akhir diagram, titik keputusan dipresentasikan dengan *diamond*.

2.3.3 Diagram Sekuensial

Diagram sekuensial digunakan untuk menunjukan aliran fungsionalitas dalam *use case*. Misalkan, dalam *use case* "menarik uang", mempunyai beberapa kemungkinan, seperti penarikan uang secara normal, percobaan penarikan uang tanpa kecukupan ketersediaan dana, penarikan dengan penggunaan PIN yang salah, dan lainnya.



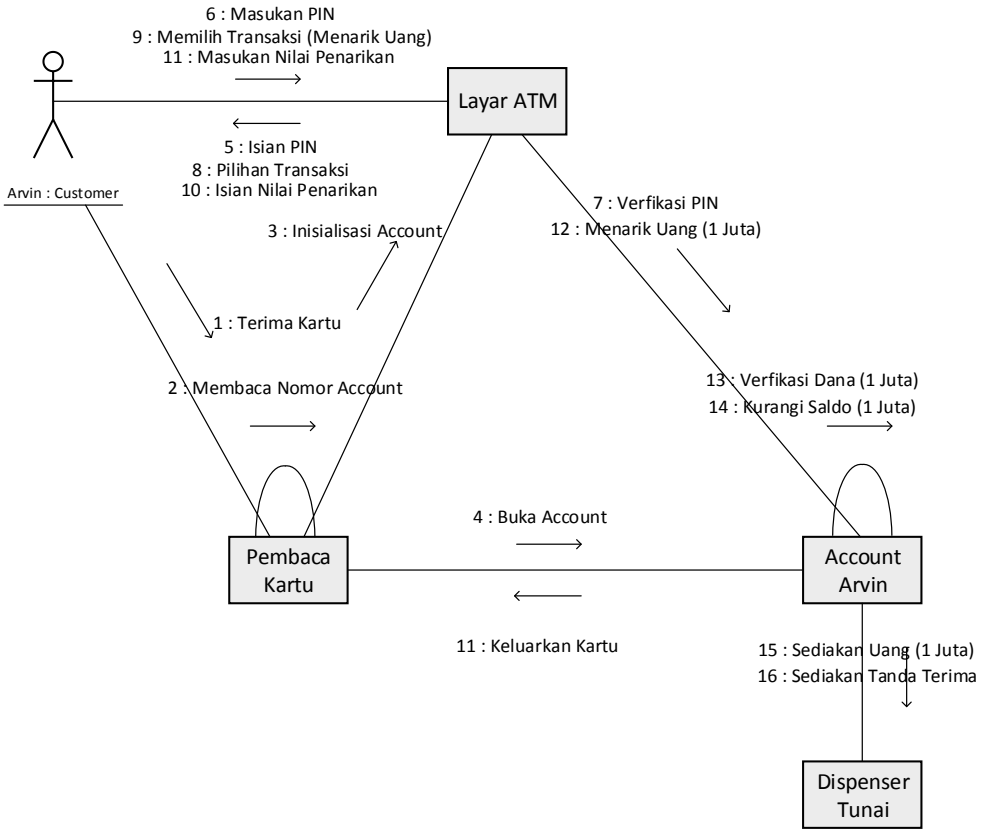
Gambar 2.4 Diagram Sekuensial

proses dalam use case diawali ketika seorang pelanggan yang bernama Arvin memasukan kartu ATM ke dalam pembaca kartu yang digambarkan oleh sebuah objek yang terdapat dibagian atau diagram. Kemudian pembaca kartu membaca nomor kartu, membuka account objek milik Arvin, dan menampilkan layar ATM

untuk meminta nomor PIN. Arvin measukan nomor PIN , layar membaca PIN dan melakukan verfikasi. Layar menampilkan beberapa pilihan dan Arvin memilih menu menarik uang. Layar kemudian menampilkan isian jumlah uang yang akan diambil Arvin, dan Arvin mengetik 1 juta rupiah.

2.3.4 Diagram Kolaborasi

Diagram kolaborasi menunjukan informasi yang sama persis dengan diagram sekuensial. Pada diagram sekuensial, keseluruhan interaksi berdasarkan urutan waktu, tetapi pada diagram kolaborasi, interaksi antara objek atau aktor ditunjukan dengan arah panah tanpa keterangan waktu.



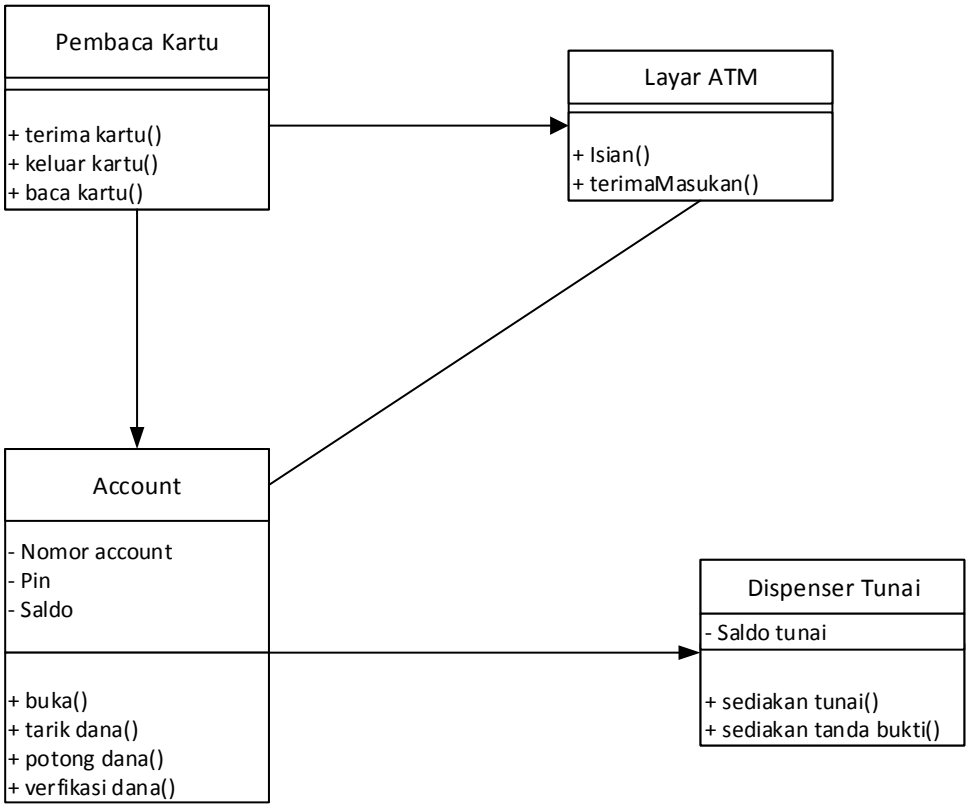
Gambar 2.5 Diagram Kolaborasi

Pada dasarnya diagram ini sama dengan diagram sekuensial, tetapi orang menggunakan diagram ini untuk alasan yang berbeda. Tenaga ahli jaminan kualitas dan arsitek sistem menggunakan diagram ini untuk melihat proses distribusi antar

objek. Diagram ini berbentuk seperti bintang, dengan beberapa objek yang berkomunikasi dengan sebuah objek pusat. Arsitek sistem menggunakan diagram ini untuk menyimpulkan bahwa sistem yang dibangun terlalu tergantung pada objek pusat, dan merancang ulang objek-objek untuk mendistribusikan proses secara merata.

2.3.5 Diagram Kelas

Diagram kelas menunjukan interaksi antar kelas dalam sistem. Sebagai contoh, nomor account milik Arvin adalah sebuah objek dari kelas *Account*. Kelas mengandung informasi dan tingkah laku yang berkaitan dengan informasi tersebut. Kelas *account* mengandung nomor PIN pengguna dan tingkah laku untuk mengecek PIN. Sebuah kelas pada diagram kelas untuk setiap tipe objek pada diagram sekuensial atau diagram kolaborasi.

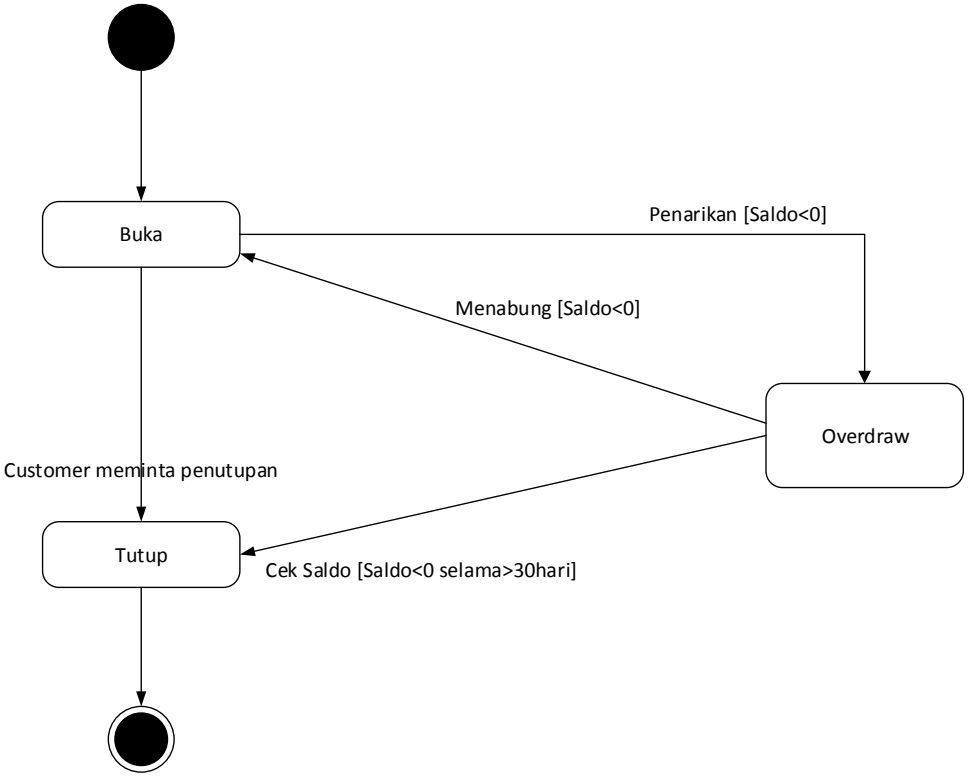


Gambar 2.6 Diagram Kelas

Diagram diatas menunjukan hubungan antar kela-kelas yang diimplemntasikan oleh use case “menarik uang”. Diagram ini terdiri atas empat kelas: Pembaca kartu, account, layar ATM, dan dispenser tunai. Sebuah kelas dalam bentuk bujur sangkar yang terbagi dalam tiga bagian. Bagian pertama menunjukan nama kelas. Bagian kedua menunjukan anggota kelas yang memuat informasi atau atribut, misalnya pada kelas account mempunyai tiga informasi yaitu nomor account, pin dan saldo. Bagian ketiga menunjukan operasi-operasi dari sebuah kelas, dimana operasi dari sebuah kelas adalah tingkah laku yang disediakan oleh kelas.

2.3.6 Diagram Statechart

Diagram *statechart* menyediakan sebuah cara untuk memodelkan bermacam-macam keadaan yang mungkin dialami oleh sebuah objek. Jika dalam diagram kelas menunjukan gambaran statis kelas-kelas dan relasinya, diagram *statechart* digunakan untuk memodelkan laku dinamik sistem.

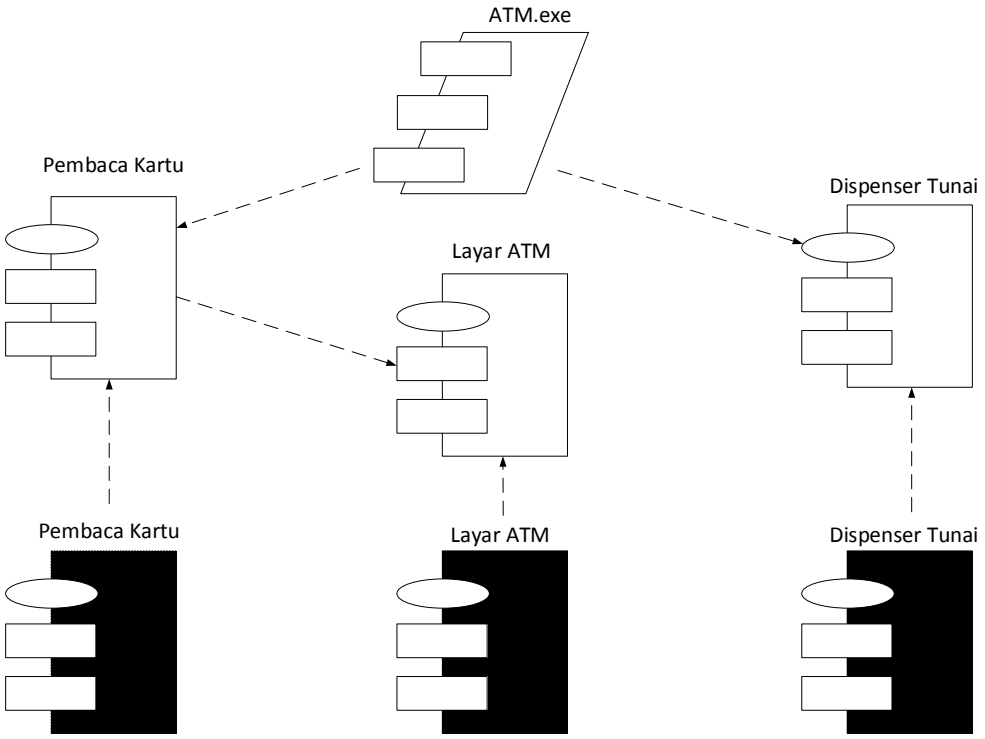


Gambar 2.7 Diagram Statechart

Diagram ini menunjukan kegiatan objek, misalkan sebuah Account di bank dapat eksis dalam beberapa keadaan yang berbeda. Seperti dapat buka, tutup, atau kondisi overdraw atau keadaan dimana jumlah pengambilan lebih besar dari simpanan yang ada. Pada gambar 2.7 terlihat keadaan sebuah account dapat eksis dan dapat bergerak dari suatu keadaan dan keadaan lainnya. Misalnya, saat account berada pada kondisi buka dan pelanggan meminta penutupan, maka account bergerak ke kondisi diisi tutup. Jika account ada pada kondisi buka dan pelanggan membuat penarikan, account akan bergerak ke kondisi *Overdawn*, jika saldo terakhir setelah dikurangi jumlah penarikan kurang dari nol.

2.3.7 Diagram Komponen

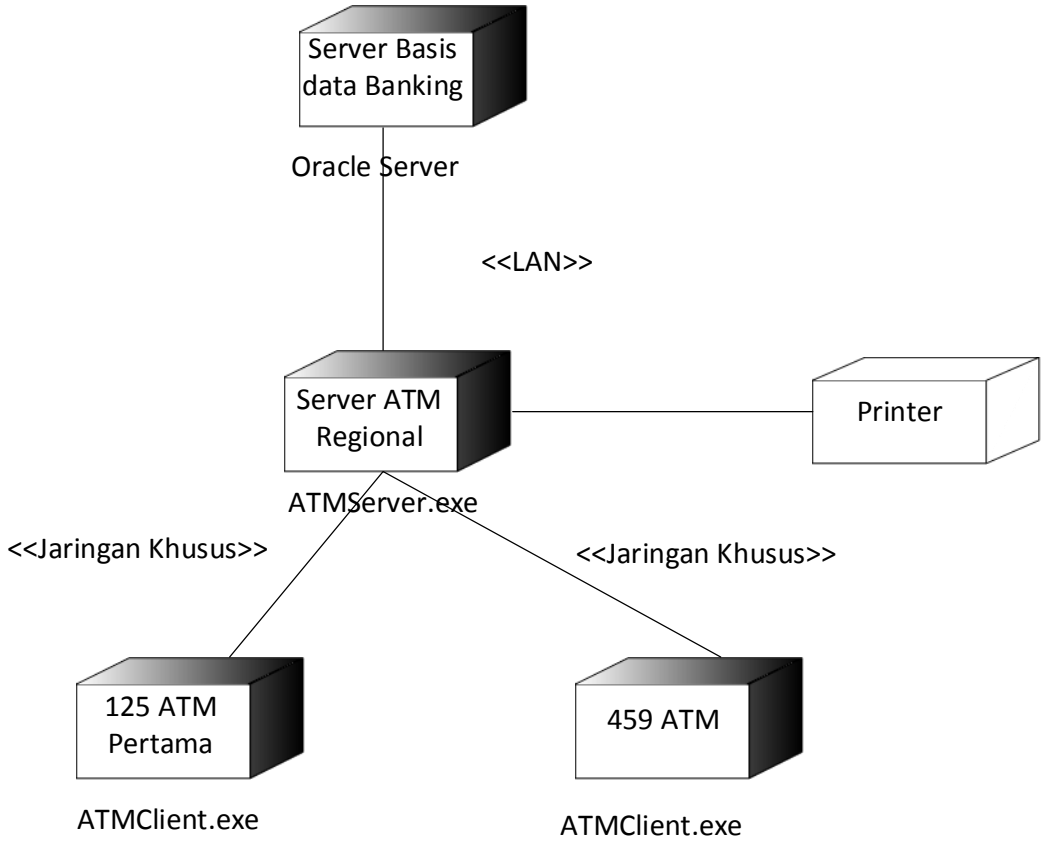
Diagram komponen menunjukan model secara fisik komponen perangkat lunak pada sistem dan hubungannya antar mereka. Ada dua tipe komponen dalam diagram yaitu komponen *excutable* dan kode pustaka. Masing – masing kelas dalam model akan dipetakan ke sebuah komponen kode pustaka. Setelah komponen dibuat, mereka ditambahkan dalam diagram komponen dengan memberikan relasi antara komponen-komponen.



Gambar 2.8 Diagram Komponen

2.3.8 Diagram Deployment

Diagram *deployment* adalah fisik dari sistem seperti komputer dan perangkat dan bagaimana hubungannya dengan yang lain. View ini digambarkan dalam deployment diagrams dan digunakan untuk pengembang, pengintegrasi, dan penguji [8].



Gambar 2.9 Diagram Deployment

2.4 Adobe

Adobe Systems (NASDAQ: ADBE) adalah perusahaan perangkat lunak yang bergerak dibidang grafis, animasi video, dan pengembangan web. Adobe adalah salah satu perusahaan perangkat lunak yang paling besar didunia. Perusahaan ini didirikan pada tahun 1982 dan berpusat di San Jose, California, A.S. Beberapa dari produk-produknya yang terkenal dan mendunia adalah Photoshop, Flash, dan Acrobat.

Adobe telah berkembang pesat dalam beberapa tahun terakhir baik dari segi penjualan atau pun jumlah pengguna. Pada tahun 2005, perusahaan Macromedia telah dibeli oleh Adobe [9].

2.4.1 Adobe Flash

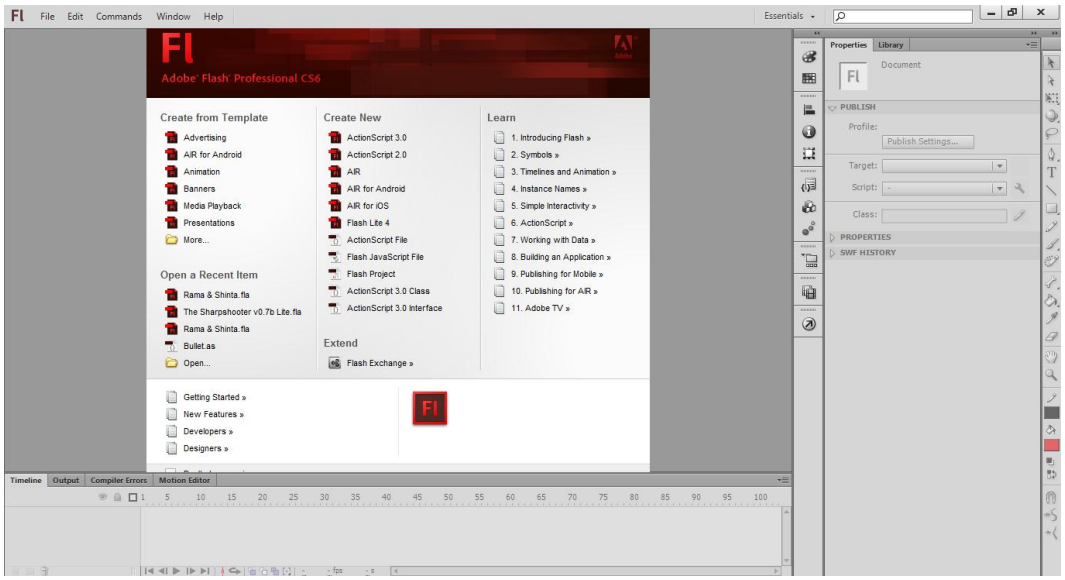
Adobe Flash (dahulu bernama Macromedia Flash) adalah salah satu perangkat lunak komputer yang merupakan produk unggulan Adobe Systems. Adobe Flash digunakan untuk membuat gambar vektor maupun animasi gambar tersebut. Berkas yang dihasilkan dari perangkat lunak ini mempunyai file extension .swf dan dapat diputar dipenjelajah web yang telah dipasangi Adobe Flash Player. Flash menggunakan bahasa pemrograman bernama ActionScript yang muncul pertama kalinya pada Flash 5 [9].

Sebelum tahun 2005, Flash dirilis oleh Macromedia. Flash 1.0 diluncurkan pada tahun 1996 setelah Macromedia membeli program animasi vektor bernama Future Splash. Versi terakhir yang diluncurkan dipasaran dengan menggunakan nama ‘Macromedia’ adalah Macromedia Flash 8. Pada tanggal 3 Desember 2005 Adobe System mengakuisisi Macromedia dan seluruh produknya, sehingga nama Macromedia Flash berubah menjadi Adobe Flash.

Adobe Flash merupakan sebuah program yang didesain khusus oleh Adobe dan program aplikasi standar authoring tool profesional yang digunakan untuk membuat animasi dan bitmap yang sangat menarik untuk keperluan pembangunan situs web yang interaktif dan dinamis. Flash didesain dengan kemampuan untuk membuat animasi 2 dimensi yang andal dan ringan sehingga Flash banyak digunakan untuk membangun dan memberikan efek animasi pada website, CD Interaktif dan yang lainnya. Selain itu, aplikasi ini juga dapat digunakan untuk membuat animasi logo, movie, game, pembuatan navigasi pada situs web, tombol animasi, banner, menu interaktif, interaktif form isian, e-card, screen saver, dan pembuatan aplikasi-aplikasi web lainnya. Dalam Flash, terdapat teknik-teknik membuat animasi, fasilitas action script, filter, custom easing dan dapat memasukan video lengkap dengan fasilitas playback FLV. Keunggulan yang dimiliki oleh Flash ini adalah ia mampu diberikan sedikit code pemrograman baik yang berjalan sendiri

untuk mengatur animasi yang ada di dalamnya atau digunakan untuk berkomunikasi dengan program lain seperti HTML, PHP dan Database dengan pendekatan XML, dapat dikolaborasikan dengan web, karena mempunyai keunggulan antara lain kecil dalam ukuran file outputnya [9].

Movie-movie Flash memiliki ukuran file yang kecil dan dapat ditampilkan dengan ukuran layar yang dapat disesuaikan dengan keinginan. Aplikasi Flash merupakan sebuah standar aplikasi industri perancangan animasi web dengan peningkatan pengaturan dan perluasan kemampuan integrasi yang lebih baik. Banyak fitur baru dalam Flash yang dapat meningkatkan kreativitas dalam pembuatan isi media yang kaya dengan memanfaatkan kemampuan aplikasi tersebut secara maksimal. Fitur-fitur baru ini membantu kita lebih memusatkan perhatian pada desain yang dibuat secara cepat, bukannya memusatkan pada cara kerja dan penggunaan aplikasi tersebut. Flash juga dapat digunakan untuk mengembangkan secara cepat aplikasi-aplikasi web yang kaya dengan pembuatan script tingkat lanjut. Didalam aplikasinya juga tersedia sebuah alat untuk men-debug script. Dengan menggunakan code hint mempermudah dan mempercepat pembuatan dan pengembangan isi ActionScript secara otomatis [9].

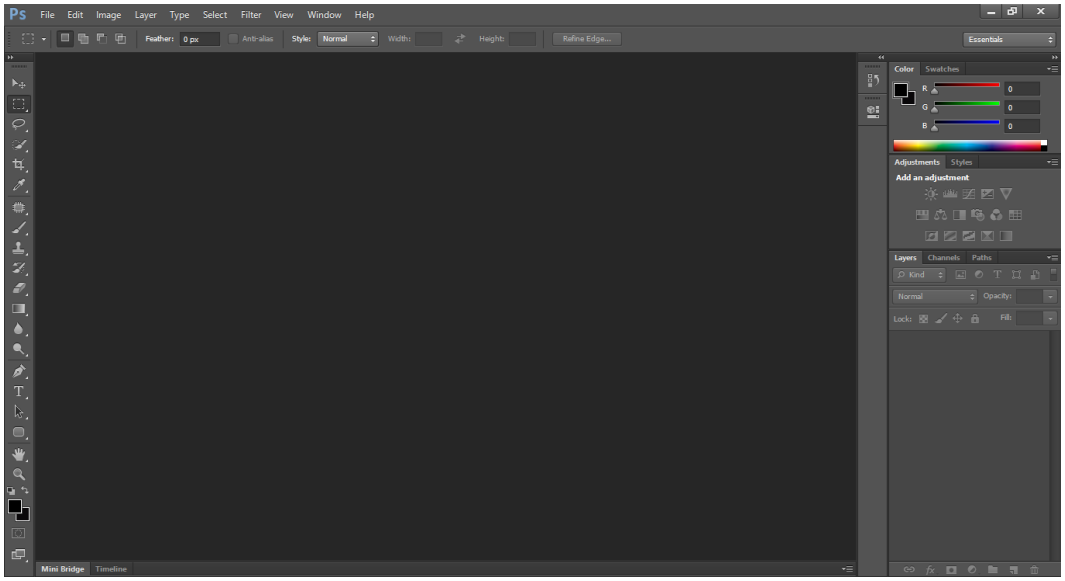


Gambar 2.10 Tampilan Utama Adobe Flash

2.4.2 Adobe Photoshop

Adobe Photoshop adalah software yang dibuat oleh perusahaan Adobe System, yang di khususkan untuk pengeditan foto atau gambar dan pembuatan effect. Perangkat lunak ini banyak digunakan oleh Fotografer Digital dan perusahaan iklan sehingga dianggap sebagai pemimpin pasar (market leader) untuk perangkat lunak pengolah gambar. Meskipun pada awalnya Photoshop dirancang untuk menyunting gambar untuk cetakan berbasis-kertas, Photoshop yang ada saat ini juga dapat digunakan untuk memproduksi gambar untuk *World Wide Web*. Beberapa versi terakhir juga menyertakan aplikasi tambahan, *Adobe ImageReady*, untuk keperluan tersebut.

Photoshop juga memiliki hubungan erat dengan beberapa perangkat lunak penyunting media, animasi, dan *authoring* buatan-*Adobe* lainnya. File format asli *Photoshop*, PSD, dapat diekspor ke dan dari *Adobe ImageReady*. *Adobe Illustrator*, *Adobe Premiere Pro, After Effects dan Adobe Encore DVD* untuk membuat DVD profesional, menyediakan penyuntingan gambar non-linear dan layanan special *effect* seperti *background*, tekstur, dan lain-lain untuk keperluan televisi, film, dan situs web [9]

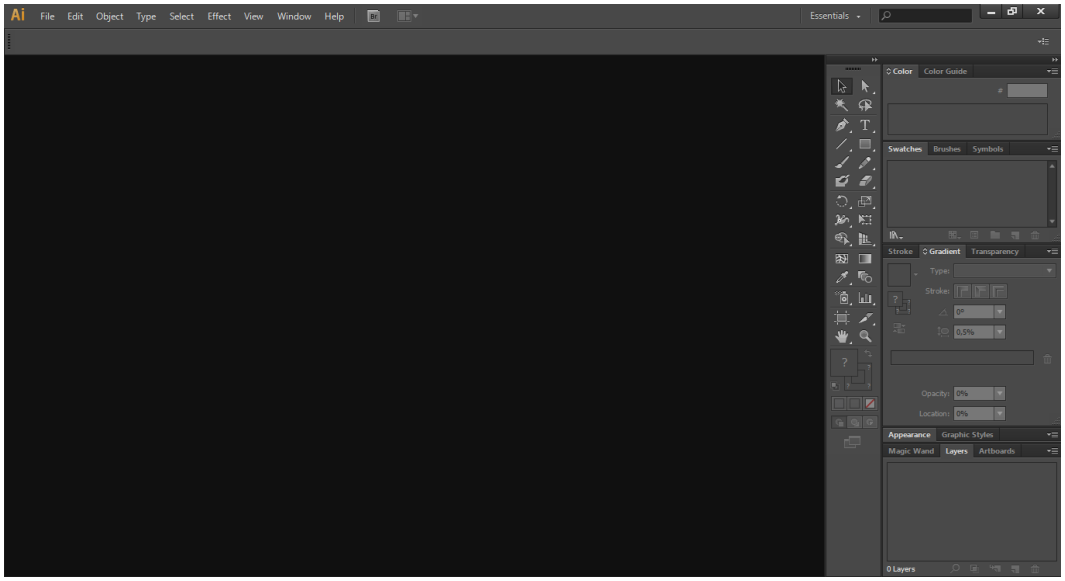


Gambar 2.11 Tampilan Utama Adobe Photoshop

2.4.3 Adobe Illustrator

Adobe Illustrator adalah sebuah perangkat lunak yang dipasarkan dan dikembangkan oleh Adobe *Systems*. Software ini berfungsi sebagai editor pengolah grafis vektor (*graphics vector editor*) sama halnya seperti Corel Draw, Macromedia Freehand, dan lain-lain. Aplikasi ini banyak digunakan desainer profresional, mulai desainer iklan, majalah, cover, banner sampai desainer web.

Adobe Illustrator sudah dipasarkan sejak tahun 1988 dan hingga sekarang sudah mengeluarkan versi terbaru dari AI yaitu Adobe Illustrator CS6. Adobe Illustrator CS6 ini pun mendukung format warna RGB (Red, Green, Blue) dan CMYK (Cyan, Magenta, Yellow, Key – Black) yang biasa digunakan untuk proses cetak-mencetak. Namun komputer anda harus memenuhi spesifikasi kebutuhan minimal (Minimum Requirements) agar kinerja AI dapat berfungsi dengan baik.



Gambar 2.12 Tampilan Utama Adobe Illustrator

2.5 ActionScript

Adalah bahasa yang dikenali dan digunakan dalam Flash untuk berbagai hal dalam movie [9].

1. ActionScript 1.0 : Flash 5 dan Flash MX(6)
2. ActionScript 2.0 : Flash MX 2004(7) dan Flash (8)
3. ActionScript 3.0 : Flash CS3(9), Flash CS4(10), Flash CS6(11)