

BAB II

TINJAUAN PUSTAKA

II.1 Penelitian Terkait

Telah ada beberapa penelitian yang dilakukan terkait dengan penggunaan

Wireless Tethering pada perangkat *smartphone* android, diantaranya adalah :

Tabel II.1 Penelitian Terkait

No	Nama Peneliti	Judul Penelitian	Hasil Penelitian	Perbedaannya
1	Muhammad Zaelani (2017)	Aplikasi Mobile Pelayanan Donor Darah Berbasis Android Pada Unit Transfusi Darah PMI Kabupaten Tangerang	Penelitian ini bertujuan untuk aplikasi ini dapat melakukan kegiatan pelayanan tanpa harus bertatap muka antar admin dan customer, sehingga bisa didapatkan suatu informasi pelayanan, pelayanan tersebut mengenai pengajuan kegiatan donor darah, pengajuan piagam, stok darah, dan jadwal kegiatan	Dengan adanya teknologi android unit transfusi darah PMI kabupaten Tangerang tidak ingin ketinggalan teknologi tersebut, maka aplikasi ini dapat melakukan kegiatan pelayanan donor darah.
2	Castaka Agus Sugiatno (2017)	Studi Rancang Bangun Aplikasi Donor Darah Berbasis Mobile di PMI Kabupaten Bandung	Penelitian ini Berdasarkan hasil pengujian diperoleh informasi bahwa perangkat lunak yang dibuat berdasarkan perhitungan User Acceptance Test (UAT) dengan jumlah 20 responden telah sesuai dengan yang dibutuhkan oleh penggunayaitu dengan presentase kesesuaian sebesar 85%. Secara pengujian fungsional perangkat lunak yang dibangun telah sesuai dengan yang diharapkan	Donor darah adalah proses pemberian darah secara sukarela untuk maksud dan tujuan transfusi darah bagi orang lain yang membutuhkan. Berdasarkan data PMI Pusat di tahun 2013 kita masih kekurangan 600.000 liter stok darah. Proses penyampaian informasi terkait donor darah.

3	Nanda Aris (2018)	Aplikasi Donor Darah Dengan Fitur Realtime Broadcasting Android	Penelitian ini akan dibuat untuk mengetahui penerima donor darah dari suatu tempat. Dan pengguna tidak perlu lagi mencari mencari pendonor lagi.	Dengan adanya aplikasi yang mau dibuat pengambilan data pendonor yang jaraknya paling dekat yang membutuhkan pendonor
---	-------------------	---	--	---

II.2. Landasan Teori

II.2.1. Fitur Real Time

Waktu nyata (bahasa inggris : realtime) adalah kondisi pengoperasian dari suatu sistem perangkat keras dan perangkat lunak yang dibatasi oleh rentang waktu dan memiliki tenggat waktu (deadline) yang jelas, relatif terhadap waktu suatu peristiwa atau operasi terjadi. realtime menerapkan prioritas dalam pelaksanaan tugas dan toleransi waktu, sistem waktu nyata yang di ukur di ambil dari respon suatu plant terhadap perubahan input. Sistem dibatasi oleh suatu batasan yang tegas tetapi apabila batasan tersebut dilanggar, maka tidak akan berakibat fatal secara langsung tetapi mengakibatkan penurunan performansi dari sistem yang semakin menurun seiring dengan waktu yang berjalan, sampai suatu batasan waktu tertentu lalu performansi sistem akan benarbenar tidak berjalan (M. Ichwan ; 2012).

II.2.2. Broadcasting (Penyiaran)

Menurut disiplin ilmu komunikasi, broadcasting adalah cabang dari ilmu komunikasi yang berhubungan dengan penyiaran. Penyiaran (broadcasting) merupakan kegiatan penyelenggaraan siaran, yaitu rangkaian mata acara dalam

bentuk audio visual yang ditransmisikan dalam bentuk signal suara dan atau gambar-gambar.

Output penyiaran adalah siaran. Siaran merupakan benda abstrak yang sangat potensial untuk dipergunakan mencapai tujuan yang bersifat idial maupun material. Siaran merupakan hasil kerja kolektif yang memerlukan dana besar, banyak tenaga yang kreatif dan profesional serta sarana elektris canggih yang harganya relatif mahal.

Di dalam broadcasting sendiri sebenarnya yang paling dominan dikaji adalah bagaimana membuat konten sebuah siaran mulai dari pra produksi, pasca produksi, jadi bukan sekedar belajar teori namun juga mempelajari praktek bagaimana membuat sebuah tayangan atau konten yang menarik dan enak dilihat atau didengar, menurut ilmu komunikasi, bagaimana pesan yang disampaikan sampai kepada khalayak umum (Rahmawati ; 2011).

II.2.3. Palang Merah Indonesia

Palang Merah Indonesia (PMI) Berdirinya Palang Merah di Indonesia sebenarnya sudah dimulai sejak masa sebelum Perang Dunia Ke-II. Saat itu, tepatnya pada tanggal 21 Oktober 1873 Pemerintah Kolonial Belanda mendirikan Palang Merah di Indonesia dengan nama Nederlands Rode Kruis Afdeling Indie (Nerkai), yang kemudian dibubarkan pada saat pendudukan Jepang. Perjuangan untuk mendirikan Palang Merah Indonesia sendiri diawali sekitar tahun 1932. Kegiatan tersebut dipelopori oleh Dr. RCL Senduk dan Dr Bahder Djohan. Rencana tersebut mendapat dukungan luas terutama dari kalangan terpelajar

Indonesia. Mereka berusaha keras membawa rancangan tersebut ke dalam sidang Konferensi Nerkai pada tahun 1940 walaupun akhirnya ditolak mentahmentah. Terpaksa rancangan itu disimpan untuk menunggu kesempatan yang tepat. Seperti tak kenal menyerah, saat pendudukan Jepang, mereka kembali mencoba untuk membentuk Badan Palang Merah Nasional, namun sekali lagi upaya itu mendapat halangan dari Pemerintah Tentara Jepang sehingga untuk kedua kalinya rancangan itu harus kembali disimpan. Tujuh belas hari setelah proklamasi kemerdekaan 17 Agustus 1945, yaitu pada tanggal 3 September 1945, Presiden Soekarno mengeluarkan perintah untuk membentuk suatu badan Palang Merah Nasional. Atas perintah Presiden, maka Dr. Buntaran yang saat itu menjabat sebagai Menteri Kesehatan Republik Indonesia Kabinet I, pada tanggal 5 September 1945 membentuk Panitia 5 yang terdiri dari: dr R. Mochtar (Ketua), dr. Bahder Djohan (Penulis), dan dr Djuhana; dr Marzuki; dr. Sitanala (anggota) (Sugiatno, 2017 : 12).

II.2.3.1 Darah

Pengertian Darah berasal dari bahasa Yunani yakni hemo, hemato dan haima yang berarti darah. Darah adalah cairan yang terdapat pada semua makhluk hidup (kecuali tumbuhan) tingkat tinggi yang berfungsi mengirimkan zat-zat dan oksigen yang dibutuhkan oleh jaringan tubuh, mengangkut bahan-bahan kimia hasil metabolisme, dan juga berfungsi sebagai pertahanan tubuh manusia terhadap virus atau bakteri. Darah manusia adalah cairan di dalam tubuh yang berfungsi untuk mengangkut oksigen yang diperlukan oleh sel-sel di seluruh tubuh. Darah

juga menyuplai jaringan tubuh dengan nutrisi, mengangkut zat-zat sisa metabolisme, dan mengandung berbagai bahan penyusun sistem imun yang bertujuan mempertahankan tubuh dari berbagai penyakit. Hormon-hormon dari sistem endokrin juga diedarkan melalui darah (Sugiatno, 2017 : 12).

II.2.3.2. Donor Darah

Donor darah adalah proses pemberian darah secara sukarela untuk maksud dan tujuan transfusi darah bagi orang lain yang membutuhkan. Donor Darah Sukarela (DDS) adalah orang yang dengan sukarela mendonorkan darahnya. Mereka tidak tahu siapa yang akan menerima darahnya. Donor Darah Sukarela biasanya memiliki prevalensi IMLTD yang paling rendah, karena tidak ada alasan kuat untuk menutupi semua informasi yang dapat membuat pendonor ditolak untuk mendonorkan darahnya. Banyaknya pendonor DDS yang rutin donor darah, yang dapat membantu memenuhi kebutuhan darah setiap hari. Hal ini tentu sangat menguntungkan pasien yang pada saat itu membutuhkan transfusi darah, karena selalu tersedia darah sehat yang sudah diolah dan siap digunakan kapan pun. Pasien yang membutuhkan transfusi darah tidak perlu menunggu waktu lama untuk mendapatkan darah yang cocok dengannya atau menunggu proses pengolahan darah selesai (Sugiatno, 2017 : 12).

II.2.4. *Android*

Android merupakan sistem operasi berbasis *Linux* yang bersifat terbuka (*open source*) dan dirancang untuk perangkat selular layer sentuh seperti

smartphone dan komputer tablet. Android dikembangkan oleh Android, inc. dengan dukungan financial dari google yang kemudian dibeli pada tahun 2005. (Sherief Saibino, 2014 : 7).

1. Sejarah *Android*

Pada bulan oktober 2003 Android, inc. didirikan di Palo Alto, California, oleh Andy Rubin (pendiri Danger), Rich Miner (pendiri Wildfire Communications, inc), Nick Sears (mantan Vp T-Mobile), dan chris White (kepala desain dan pengembangan antar muka web IV) untuk mengembangkan Android yaitu untuk mengembangkan sebuah sistem operasi canggih yang ditujukan untuk kamera digital (Sherief Saibino, 2014 : 8).

2. Fundamental Android

Aplikasi *Android* ditulis dalam bahasa pemrograman *java*. Kode *java* dikompilasi bersama dengan data *file resource* yang dibutuhkan oleh aplikasi, dimana prosesnya di *package* oleh *tools* yang dinamakan “*apt tools*” ke dalam paket *Android* sehingga menghasilkan file dengan ekstensi *apk*. Empat jenis komponen aplikasi *Android*, yaitu :

1) *Activities*

Suatu *activity* akan menyajikan user *interface (UI)* kepada pengguna, sehingga pengguna dapat melakukan interaksi.

2) *Service*

Service tidakmemiliki *Graphic User Interface (GUI)*, tetapi *service* berjalan secara background.

3) *Broadcast receiver*

Broadcast receiver berfungsi untuk menerima dan bereaksi untuk menyampaikan notifikasi.

4) *Content Provider*

Content Provider membuat kumpulan aplikasi data secara spesifik sehingga bias digunakan oleh aplikasi lain.

3. Versi Android

Awal sistem android yang dirilis yaitu Android beta bulan November 2007. Sedangkan versi komersial pertama, android 1,0 dirilis pada September 2008. Sejak april 2009, versi Android yang dikembangkan diberi kode nama yang berdasarkan makanan pencuci mulut dan manis. (Sherief Saibino, 2014 : 15).

Tabel. II.1 *Versi Android*

Versi	Nama	Rillis	Catatan
1.0	<i>Android 1.0</i>	23 September 2008	Android pertama hanya untuk <i>smartphone</i>
1.1	<i>Android 1.1</i>	9 Februari 2008	
1.5	<i>Cupcake</i>	30 April 2009	Mulai pakai kode nama
1.6	<i>Donut</i>	15 September 2009	
2.0- 2.1	<i>Éclair</i>	26 Oktober 2009 (2.0) 12 Januari 2010 (2.1)	
2.2	<i>Froyo</i>	20 Mei 2010	
2.3	<i>Gingerbread</i>	6 Desember 2010	Digunakan pada <i>smartphone</i> jenis lama

3.0- 3.2	<i>Honeycomb</i>	22 Februari 2011 (3.0) 10 Mei 2011 (3.1) 15 Juli 2011 (3.2)	Hanya untuk tablet
4.0	<i>ICS (Ice Cream Sandwich)</i>	19 Oktober 2011	<i>Smartphone</i> dan tablet
4.1- 4.3	<i>Jelly Bean</i>	9 Juli 2012 (4.1) 13 November 2012 (4.2) 24 Juli 2013 (4.3)	<i>Update</i> untuk memperbaiki dan menambah <i>fitur</i> pada <i>ICS</i>
4.4	<i>Kit kat</i>	3 September 2013	
5.0	<i>Lollipop</i>	12 november 2014 (5.0) 9 Maret 2015 (5.1)	
6.0	<i>Marshmallow</i>	5 oktober 015	Terdapat <i>daze mode</i> , <i>Do Not Disturb mode</i> , mendukung <i>USB tipe C</i> , mendukung pembacaan <i>fingerprint</i> .

Sumber: (Sherief Saibino, 2014 : 15)

Aplikasi android merupakan kumpulan dari beberapa activity yang tergabung secara bebas dengan yang lainnya. Yang mana kumpulan tersebut memiliki satu activity utama untuk memanggil Activity secara bertumpuk. Activity adalah suatu komponen pada aplikasi Android yang menyediakan tampilan, sehinggadapat berinteraksi dengan melakukansesuatu, seperti dial telepon, mengambil foto, mengirim email, atau melihat peta. Setiap mendapatkan tampilan/layout sebagai antarmuka dengan user.

Android Development Tools (ADT) adalah plugin untuk Eclipse IDE yang dirancang khusus untuk memberikan integrated environment yang kuat untuk membuat aplikasi android. ADT memberikan kemampuan kepada Eclipse untuk membuat projek baru Android secara tepat, membuat aplikasi User Interface, menambahkan komponen berdasarkan Android Framework API, melakukan

debugging aplikasi yang dibuat dengan menggunakan Android SDK Tools dan bahkan melakukan distribusi aplikasi yang dibuat. Pembuatan aplikasi android dengan Eclipse beserta ADT sangat dianjurkan karena merupakan cara tercepat untuk memulai membuat projek Android. Dengan disediakannya project setup, serta tools yang sudah terintegrasi (Hendra Nugraha, 2015 : 21-22).

II.2.5. *Android Studio*

Android Studio adalah Lingkungan Pengembangan Terpadu - Integrated Development Environment (IDE) untuk pengembangan aplikasi Android, berdasarkan IntelliJ IDEA. Konsep yang dimiliki Android Inc ternyata menggugah minat Google untuk memilikinya. Pada bulan Agustus 2005, Akhirnya Android Inc diakuisisi oleh Google Inc. seluruh sahamnya dibeli oleh Google. Banyak yang memperkirakan nilai pembelian Android Inc oleh Google adalah sebesar USD 50 juta. saat itu banyak yang berspekulasi bahwa akuisisi ini adalah langkah awal yang dilakukan Google untuk masuk ke pasar mobile phone. Maka dari situlah banyak pengembang sistem maupun software berlomba lomba untuk membuat atau merancang sistem Android menggunakan software – software yang support dengan Android, dan sebagai contoh disini kita pengenalan Android Studio. Selain merupakan editor kode IntelliJ dan alat pengembang yang berdaya guna, Android Studio menawarkan fitur lebih banyak untuk meningkatkan produktivitas Anda saat membuat aplikasi Android, misalnya:

- Sistem versi berbasis Gradle yang fleksibel
- Emulator yang cepat dan kaya fitur

- Lingkungan yang menyatu untuk pengembangan bagi semua perangkat Android
- Instant Run untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru
- Template kode dan integrasi GitHub untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh
- Alat pengujian dan kerangka kerja yang ekstensif
- Alat Lint untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah-masalah lain
- Dukungan C++ dan NDK
- Dukungan bawaan untuk Google Cloud Platform, mempermudah pengintegrasian Google Cloud Messaging dan App Engine

Laman ini berisi pengantar dasar fitur-fitur Android Studio. Untuk memperoleh rangkuman perubahan terbaru, lihat Catatan Rilis Android Studio (google developer ; 2016).

II.2.6. Pengertian Perancangan

Perancangan sistem adalah penentuan proses data yang diperlukan oleh sistem baru. Jika sistem itu berbasis komputer, rancangan dapat menyertakan spesifikasi jenis peralatan yang digunakan. tahap-tahap perancangan sistem informasi adalah sebagai berikut :

- a. Menyiapkan rancangan sistem yang terinci

- b. Mengidentifikasi berbagai alternatif konfigurasi sistem
- c. Mengevaluasi berbagai alternatif konfigurasi sistem
- d. Memilih konfigurasi terbaik e. Menyiapkan usulan penerapan
- e. Menyetujui atau menolak penerapan sistem

“System design is the specification or construction of a technical , computer based solution for the business requirements identified in a system analysis”.

Yang diterjemahkan sebagai berikut: “Perancangan sistem adalah spesifikasi atau perwujudan dari solusi teknis berbasis komputer untuk kebutuhan bisnis yang diidentifikasi di sistem analisis”. *“System design is the process of preparing detail specifications for development of a new information system”.* Yang diterjemahkan sebagai berikut: “Perancangan sistem adalah suatu proses detail spesifikasi untuk mengembangkan sebuah sistem informasi yang baru”. Berdasarkan pendapat-pendapat diatas, dapat disimpulkan bahwa perancangan sistem adalah proses mengimplementasikan hasil-hasil dari analisis sistem ke dalam suatu rancangan sistem yang baru (Henny Hendarti ; 2009 : 158).

Model perancangan sesungguhnya adalah modal objek yang mendeskripsikan realisasi fisik *use case* dengan cara berfokus pada bagaimana spesifikasi-spesifikasi kebutuhan fungsional dan *non-fungsional*, bersama dengan batasan-batasan lain yang berhubungan dengan lingkungan implemenatasi, memiliki imbas langsung pada pertimbangan-pertimbangan pada aktivitas-aktivitas yang dilakukan pada tahap implementasi. Tambahannya, model perancangan sesungguhnya secara langsung bertindak sebagai abstraksi implementasi sistem/perangkat lunak dan dengan sendirinya model perancangan

suatu saat nanti akan menjadi asupan bagi aktivitas-aktivitas selanjutnya yang kelak akan terdefinisi pada tahap implementasi (Adi Nugroho ; 2010 : 212).

II.2.7. Pengertian Aplikasi

Program aplikasi adalah program siap pakai atau program yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain. Aplikasi juga diartikan sebagai penggunaan atau penerapan suatu konsep yang menjadi pokok pembahasan atau sebagai program komputer yang dibuat untuk menolong manusia dalam melaksanakan tugas tertentu. Aplikasi software yang dirancang untuk penggunaan praktisi khusus, klasifikasi luas ini dapat dibagi menjadi 2 (dua) yaitu:

- a. Aplikasi *software* spesialis, program dengan dokumentasi tergabung yang dirancang untuk menjalankan tugas tertentu.
- b. Aplikasi paket, suatu program dengan dokumentasi tergabung yang dirancang untuk jenis masalah tertentu (Rahmatillah ; 2011: 3).

II.2.7.1. Perkembangan Teknologi Komunikasi dan Informasi

Perkembangan teknologi informasi dan komunikasi (TIK) terakhir mengalami kemajuan yang sangat pesat. Hal tersebut dikarena oleh kuatnya era globalisasi, di mana komputer dan internet dengan sifatnya yang dinamis merupakan fasilitas yang telah mendominasi berbagai aktivitas kehidupan, sehingga aktivitas pendidikan, perkantoran, komersial dan industri secara mutlak memerlukan ketersediaan fasilitas tersebut. Karakteristik masyarakat itu dikenal

dengan istilah masyarakat berbasis pengetahuan, artinya komunitas yang menguasai pengetahuan, berpeluang untuk memanfaatkan informasi serta menjadikan informasi sebagai nilai tambah dalam kehidupannya (Syaad Patmanthara ; 2012).

II.2.8. UML (Unified Modelling Language)

UML merupakan kependekan dari *Unified Modeling Language* yaitu diagram dan metode standar untuk memodelkan dan merepresentasikan *object oriented software* dan sistem bisnis. (Mulyani, 2016 : 243).

(Mulyani, 2016 : 244) Beberapa fungsi dan kegunaan dari *UML* yaitu :

1. *Visualizing*

Visualizing adalah sebagai alat komunikasi konseptual model antara tim pengembang sistem (sistem analis dengan programmer)

2. *Specifying*

Specifying adalah sebagai *tools* yang digunakan untuk memodelkan sistem secara tepat dan jelas.

3. *Constructing*

Constructing adalah *UML* sebagai bahasa grafis mampu melakukan *mapping* dan konseptual model kedalam bahasa pemrograman.

4. *Documenting*

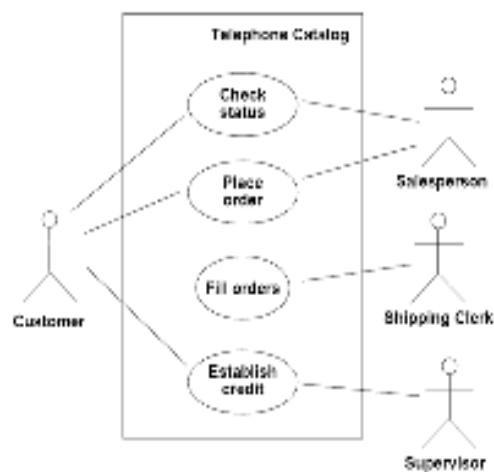
Documenting adalah *UML* digunakan sebagai *tools* untuk melakukan dokumentasi teknis sebuah sistem.

Diagram-diagram yang terdapat dalam *UML* sangat banyak, berikut ini beberapa diagram yang sering digunakan dalam pengembangan sistem yaitu :

1. *Use Case Model*

Use case model merupakan kumpulan diagram dan text yang saling bekerja sama untuk mendokumentasikan bagaimana *user* (aktor) berinteraksi dengan sistem. *Use case model* terdiri dari beberapa diagram :

- 1) *Use case diagram* yaitu diagram yang menggambarkan dan merepresentasikan aktor, *use cases*, dan *dependencies* suatu proyek dimana tujuan dan diagram ini adalah untuk menjelaskan konsep hubungan antara sistem dengan dunia luar. *Use case diagram* dapat dilihat pada gambar II.1 seperti berikut.



Gambar II.1 *Use Case Diagram*
(Sumber : Mulyani ; 2016)

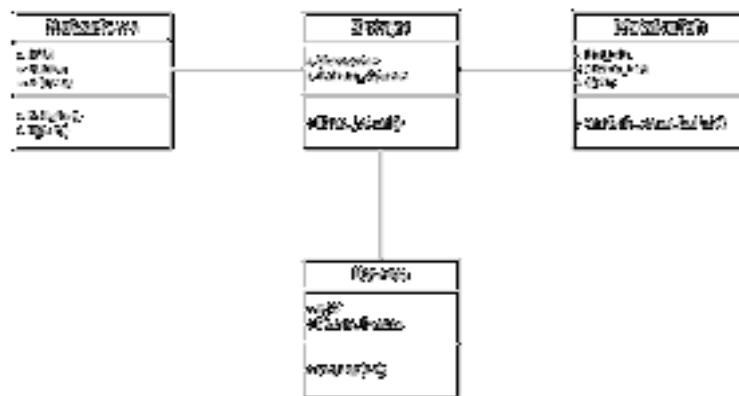
- 2) *Use case narrative* yaitu deskripsi yang menjelaskan use case diagram. Pada *use case diagram* sistem hanya digambarkan secara sederhana menggunakan simbol *use case* yang berhubungan

(*relationship*) dengan aktor, sehingga terkadang diperlukan deskripsi yang menjelaskan dan proses tersebut.

3) *Use case scenario* yaitu pemecahan kemungkinan logika pada *use case diagram*.

2. *Class Diagram*

Class diagram adalah diagram yang digunakan untuk merepresentasikan kelas, komponen-komponen kelas dan hubungan antara masing-masing kelas serta mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara mereka. *Class diagram* juga menunjukkan *property* dan operasi sebuah kelas serta batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut. *UML* menggunakan istilah fitur sebagai istilah umum yang meliputi *property* dan operasi sebuah kelas. *Class diagram* dapat dilihat pada gambar II.2 seperti berikut.



Gambar II.2 *Class Diagram*
(Sumber : Mulyani ; 2016)

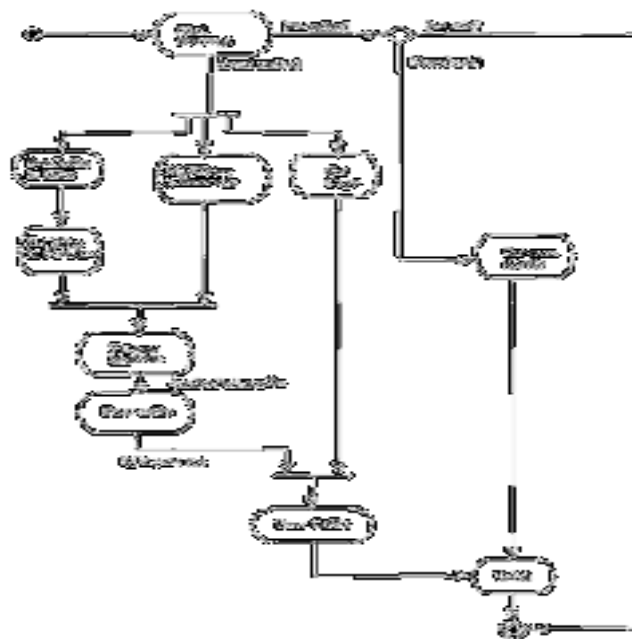
3. *Object Diagram*

Object diagram adalah diagram yang digunakan untuk menggambarkan dan merepresentasikan objek dan hubungan antar objek tersebut, selain itu *object*

diagram juga dapat digunakan sebagai diagram untuk menunjukkan sebuah konfigurasi contoh dan sebuah objek.

4. *Activity Diagram*

Activity diagram adalah diagram *UML* yang digunakan untuk menggambarkan alur aktivitas dari satu proses. *Activity diagram* memungkinkan siapapun yang melakukan proses untuk memilih urutan dalam melakukannya. Hal ini penting untuk pemodelan bisnis karena proses-proses sering muncul secara paralel. Ini juga berguna pada algoritma yang bersamaan, dimana urutan-urutan independen dapat melakukan hal-hal secara paralel. *Activity diagram* dapat dilihat pada gambar II.3 seperti berikut.

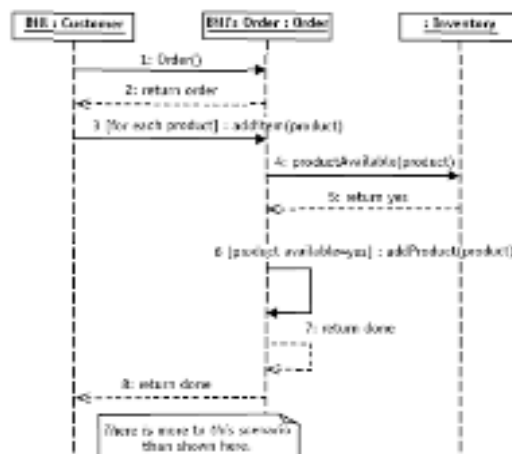


Gambar II.3 *Activity Diagram*
(Sumber : Mulyani ; 2016)

5. *Sequence Diagram*

Sequence diagram adalah diagram yang menggambarkan interaksi antar objek. *Sequence diagram* secara khusus menjabarkan *behavior* sebuah

skenario tunggal. Diagram tersebut menunjukkan sejumlah objek contoh dan pesan-pesan yang melewati objek ini dalam sebuah *use case*.



Gambar II.4 Sequence Diagram
(Sumber : Mulyani ; 2016)

II.9. Normalisasi (*Normalization*)

Normalisasi adalah salah satu cara untuk meminimalisir pengulangan data (*data redundancy*), normalisasi akan diperlukan jika ada indikasi bahwa tabel yang kita buat tidak baik (terjadi pengulangan informasi, potensi inkonsistensi data pada operasi perubahan, tersembunyinya informasi tertentu dan lain sebagainya) dan diperlukan supaya jika tabel-tabel yang didekomposisi kita gabung kembali dapat menghasilkan tabel awal sebelum didekomposisi, sehingga diperoleh tabel yang baik. Hasil dari normalisasi adalah himpunan-himpunan data (tabel-tabel) dalam bentuk normal (normal form) (Mulyani, 2016 : 132).

Beberapa kegunaan normalisasi adalah :

1. Meminimalisir pengulangan data (*data redundancy*)
2. Memudahkan identifikasi *entity* objek.

Beberapa bentuk normal yaitu :

1. Bentuk Normal I (*First Normal Form / 1-NF*)
 Suatu relasi memenuhi *1-NF* jika dan hanya jika setiap *attribute* dan relasi tersebut hanya memiliki nilai tunggal dalam 1 (satu) baris *record* (memisahkan group berulang).
2. Bentuk Normal II (*Second Normal Form/2-NF*)
 Suatu relasi memenuhi *2-NF* jika dan hanya jika memenuhi *1-NF* dan Setiap *attribute* yang bukan kunci utama tergantung secara fungsional terhadap semua attribute kunci dan bukan hanya sebagian attribute (menghilangkan ketergantungan fungsional pada sebagian/salah satu key).
3. Bentuk Normal III (*Third Normal Form/3-NF*)
 Suatu relasi memenuhi *3-NF* jika dan hanya jika memenuhi *2-NF* dan setiap *attribute* bukan kunci tidak tergantung secara fungsional kepada *attribute* bukan kunci yang lain dalam relasi tersebut (menghilangkan ketergantungan transitif pada yang bukan *key*).
4. Bentuk Normal IV (*Fourth Normal Form/4-NF*)
 Suatu relasi memenuhi *4-NF* jika dan hanya jika memenuhi *BCNF* dan tabel tersebut tidak boleh memiliki lebih dari sebuah *multivalued attribute*. Untuk setiap *multivalued attribute (MVD)* juga harus merupakan *functional dependencies*.

Beberapa key dalam Normalisasi :

1. *Superkey* adalah sejumlah *attribute entity* yang dapat digunakan untuk mengidentifikasi objek secara unik.

2. *Candidate key* adalah *superkey* dengan jumlah attribute minimal dan dapat berdiri sendiri.
3. *Primary key* adalah *superkey* yang dipilih oleh database administrator.
4. *Foreign key* adalah *attribute* disuatu relasi (*tabel*) yang menjadi *primary key* di relasi (*label*) lain.

II.10. Basis Data (*Database*)

Database adalah kumpulan dari semua data yang diperlukan oleh sistem. Dengan menggunakan database, beberapa aplikasi berbeda bisa saling terintegrasi, misalnya aplikasi keuangan, aplikasi kepegawaian dengan penggajian atau aplikasi persediaan. (Mulyani, 2016 : 148).

Menurut Mulyani (2016 : 158) menyatakan suatu bangunan basis data memiliki jenjang sebagai berikut:

1. Karakter, merupakan bagian data terkecil yang berupa angka, huruf, atau karakter khusus yang membentuk sebuah *item* data atau *field*. Contoh A,B,X,Y,2,1,2,9,0,=,<,> dan sebagainya.
2. *Field/item*, merupakan representasi suatu atribut dan *record* (rekaman/tupel) yang sejenis yang menunjukkan suatu *item* dari data. Contoh *field* nama (berisi data nama-nama pegawai) dan lain sebagainya.
3. *Record/rekaman/tupel*: Kumpulan dari field membentuk suatu record atau rekaman. Record menggambarkan suatu unit data individu yang tertentu. Contoh: file pegawai, dimana tiap-tiap *record* berisi kumpulan data nama, alamat, departemen, yang dapat mewakili tiap-tiap data.

4. *File*, merupakan kumpulan dari *record-record* yang menggambarkan satu kesatuan data yang sejenis. Contoh *file* pegawai berisi data tentang semua yang berhubungan dengan pegawai seperti nama pegawai, alamat pegawai, departemen, yang dapat mewakili tiap-tiap data.
5. *Database*, merupakan kumpulan dari *file* atau *tabel* yang membentuk suatu *database*. Contoh *database* pegawai berisi file pegawai dan sebagainya.