**LAPORAN PRAKTIK KERJA LAPANG (PKL)**

**CV SARANA UTAMA SOLUSINDO**

**SISTEM JUAL BELI SAYUR *ONLINE* BERBASIS *ANDROID***

**MENGGUNAKAN *WEB SERVICE***



**Disusun oleh:**

Kadek Dwi Aryasa 155150201111124

Gusti Agung Suastika adinata 155150201111273

Nigel Erico Pangestu

PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS BRAWIJAYA

MALANG

2018

# PENGESAHAN

**LAPORAN PRAKTIK KERJA LAPANGAN (PKL)**

Diajukan Untuk Memenuhi Sebagai Persyaratan Kurikulum

Program Studi Teknik Informatika

Jurusan Teknik Informatika

Universitas Brawijaya

SISTEM JUAL BELI SAYUR *ONLINE* BERBASIS *ANDROID*

MENGGUNAKAN *WEB SERVICE*

Dilaksanakan di :

CV SARANA UTAMA SOLUSINDO

Lantai 2 Jalan Anggrek garuda No. 49, kota malang

Tanggal :

1 Juli 2018 – 30 Agustus 2018

Disusun Oleh :

|  |  |
| --- | --- |
| Kadek Dwi Aryasa | 155150201111124 |
| Gusti Agung Suastika Adinata | 155150201111273 |

Telah diperiksa dan disetujui oleh :

Mengetahui, Menyetujui,

Kepala Jurusan Teknik Informatika Dosen Pembimbing PKL

Tri Astoto Kurniawan, S.T, M.T, Ph.D. Komang Candra Brata, S.Kom., M.T., M.Sc.

NIP: 19710518 200312 1 001 NIP: 201607 890711 1 000

# LEMBAR PENGESAHAN

**LAPORAN PRAKTIK KERJA LAPANGAN (PKL)**

Diajukan Untuk Memenuhi Sebagai Persyaratan Kurikulum

Program Studi Teknik Informatika

Jurusan Teknik Informatika

Universitas Brawijaya

SISTEM JUAL BELI SAYUR *ONLINE* BERBASIS *ANDROID*

MENGGUNAKAN *WEB SERVICE*

Dilaksanakan di :

CV SARANA UTAMA SOLUSINDO

Lantai 2 Jalan Anggrek garuda No. 49, kota malang

Tanggal :

1 Juli 2018 – 30 Agustus 2018

Disusun Oleh :

|  |  |
| --- | --- |
| Kadek Dwi Aryasa | 155150201111124 |
| Gusti Agung Suastika Adinata  Nigel Erico Pangestu | 155150201111273 |

Telah diperiksa dan disetujui oleh :

Pihak CV. SARANA UTAMA SOLUSINDO

Menyetujui, Menyetujui,

General Manager Pembimbing Lapangan

Sony Winarto Selvina Yustriasanti

# KATA PENGANTAR

*Assalamu’alaikum Wr. Wb.*

Puji syukur kehadirat Allah SWT yang telah memberikan rahmad, taufik dan hidayahnya kepada ilahi sehingga kami dapat menyelesaikan laporan praktek kerja lapang ini dengan baik. Adapun laporan praktek kerja lapang(PKL) yang kami buat ini yaitu berjudul “*Sistem Jual Beli Sayur Online Berbasis Android Menggunakan Web Service”.*

Sejak awal sampai dengan akhir penulisan ini, tidak sedikit bantuan yang kami terima dan karenanya dalam kesempatan ini kami ingin menyampaikan terima kasih kepada:

1. Allah S.W.T. atas berkah dan hidayah serta bimbingan-Nya dalam pelaksanaan kuliah kerja nyata.
2. Ibu, Bapak dan seluruh keluarga atas segenap dukungan yang diberikan.
3. Bapak Ir. Heru Nurwasito, M.Kom. selaku Pembantu Dekan I Bidang Akademik Fakultas Ilmu Komputer Universitas Brawijaya.
4. Bapak Tri Astoto Kurniawan, S.T, M.T, Ph.D. selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya.
5. Bapak Agus Wahyu Widodo, S.T, M.Cs. selaku Ketua Program Studi Informatika / Ilmu Komputer Fakultas Ilmu Komputer Universitas Brawijaya
6. Bapak Komang Candra Brata, S.Kom., M.T., M.Sc. selaku dosen pembimbing yang telah menyediakan waktu untuk berdiskusi dan berbagai saran serta masukkannya.
7. Selvina Yustriasanti selaku pembimbing praktik kerja lapang yang telah memberikan pengalaman, pengarahan dan saran dalam penyusunan laporan serta menyediakan kami kesempatan dalam PKL di CV. Sarana Utama Solusindo.
8. Teman-teman satu perjuangan Praktik Kerja Lapangan (PKL) di CV. Sarana Utama Solusindo yang meluangkan segenap waktu untuk saling berdiskusi.



1. Teman-teman satu angkatan Program Studi Informatika 2015 tercinta yang selalu memberikan informasi, semangat, dorongan dan bantuan pikiran.
2. Segenap civitas akademika Fakultas Ilmu Komputer (FILKOM) Universitas Brawijaya.

Demi kesempurnaan praktik kerja ini, penulis mengharapkan kritik dan saran yang bersifat membangun. Dan harapan kami, laporan Praktek Kerja Lapang (PKL) ini dapat bermanfaat bagi semua pihak.

*Wassalamu’alaikum Wr. Wb.*

Malang, 24 Oktober 2018

Penulis

.

# ABSTRAK

Sebagai manusia kita melakukan kegiatan jual beli untuk memenuhi kebutuhan sehati-hari kita seperti kebutuhan akan bahan pangan (makanan dan minuman). Di dalamnya terjadi interaksi dan proses persetujuan untuk melakukan sebuah transaksi. Jika dahulu kita melakukan proses transaksi dengan tatap muka, maka sekarang dengan adanya teknologi kita dapat melakukan transaksi dengan online atau melewati telepon genggam kita saja. Dengan berkembangnya aplikasi *smartphone* dan berkembangnya jenis transaksi kita dapat mempermudah interaksi antara pembeli dan pedagang. *Android* dapat menjawab perkembangan pengembangan aplikasi *smartphone* masa sekarang karena bersifat *open source* sehingga akan mudah untuk dikembangkan.

Pengembangan aplikasi *android* dalam sistem jual beli sayur, dapat mempermudah pembeli dan penjual untuk melakukan transaksi. Pengembangan aplikasi dengan android mempermudah pengembangan secara berlanjut karena menggunakan pemrograman berbasis objek. Dengan adanya sistem *web service* yang disematkan dalam aplikasi tersebut dapat membuat aplikasi berjalan secara *online* dapat dapat diakses kapan saja dan dimana saja. Untuk bagian aplikasi android menggunakan IDE (*Integrated Development Environment*) *android studio* untuk pengembangannya karena memiliki *library* dan forum yang dapat membantu pengembang. Dengan pengujian *black-box* (kotak hitam) pada fitur-fitur yang telah dibuat mendapatkan hasil jika aplikasi android yang dikembangkan dapat berjalan dengan baik dan tanpa masalah. *Android* sebagai *platform open souce* dapat berkembang dan memiliki *library* yang cukup untuk mempermudah proses pengembangan sistem kedepannya.

**Kata Kunci** : *android*, aplikasi *android*, *android studio*, *web services ,* sayur, transaksi, jual beli.

# DAFTAR ISI

[PENGESAHAN ii](#_Toc528129547)

[LEMBAR PENGESAHAN iii](#_Toc528129548)

[KATA PENGANTAR iv](#_Toc528129549)

[ABSTRAK vi](#_Toc528129550)

[DAFTAR ISI vii](#_Toc528129551)

[DAFTAR TABEL x](#_Toc528129552)

[DAFTAR GAMBAR xii](#_Toc528129553)

[DAFTAR LAMPIRAN xvi](#_Toc528129554)

[BAB 1 PENDAHULUAN 1](#_Toc528129555)

[1.1 Latar belakang 1](#_Toc528129556)

[1.2 Rumusan masalah 1](#_Toc528129557)

[1.3 Tujuan 1](#_Toc528129558)

[1.4 Manfaat 1](#_Toc528129559)

[1.5 Batasan masalah 1](#_Toc528129560)

[1.6 Sistematika pembahasan 2](#_Toc528129561)

[1.7 Pelaksanaan Praktek Kerja Lapang 3](#_Toc528129562)

[BAB 2 PROFIL PERUSAHAAN 4](#_Toc528129563)

[2.1 Sejarah dan Perkembangan CV Sarana Utama Solusindo (SASINDO) 4](#_Toc528129564)

[2.2 Visi, Misi dan Komitmen Perusahaan. 4](#_Toc528129565)

[2.2.1 Visi 4](#_Toc528129566)

[2.2.2 Misi 4](#_Toc528129567)

[2.2.3 Komitmen 5](#_Toc528129568)

[2.3 Struktur Organisasi CV SARANA UTAMA SOLUSINDO 5](#_Toc528129569)

[2.4 Ruang Lingkup CV SARANA UTAMA SOLUSINDO 5](#_Toc528129570)

[2.4.1 Solusi Bisnis TI (IT Business Solution) dan Konsultan IT (IT Consultant) 6](#_Toc528129571)

[2.4.2 Pendidikan dan Pelatihan (IT Training) 7](#_Toc528129572)

[BAB 3 LANDASAN KEPUSTAKAAN 8](#_Toc528129573)

[3.1 Sistem Penjualan. 8](#_Toc528129574)

[3.1.2 Sistem Jual Beli *Online*. 8](#_Toc528129575)

[3.2 *Android* 9](#_Toc528129576)

[3.2.1 *Android Studio* 11](#_Toc528129577)

[3.3 *Web Service* 12](#_Toc528129578)

[3.4 *Laravel* 13](#_Toc528129579)

[3.4.1 *Model-View-Controller* (MVC) 13](#_Toc528129580)

[3.5 *Unified Modeling Language (UML)* 14](#_Toc528129581)

[3.5.1 *Structured Diagram.* 15](#_Toc528129582)

[3.5.2 *Behavior Diagram*. 17](#_Toc528129583)

[*3.6* *Entity Relationship Diagram* 22](#_Toc528129584)

[BAB 4 METODOLOGI 24](#_Toc528129585)

[4.1 Penentuan Objek Penelitian 24](#_Toc528129586)

[4.2 Studi Literatur 25](#_Toc528129587)

[4.3 Analisa Kasus 25](#_Toc528129588)

[4.4 Perancangan Konsep dan Model 25](#_Toc528129589)

[4.5 Implementasi 26](#_Toc528129590)

[4.6 Pengujian dan Analisis 26](#_Toc528129591)

[4.7 Kesimpulan 26](#_Toc528129592)

[BAB 5 ANALISIS DAN PERANCANGAN 27](#_Toc528129593)

[5.1 Analisa Kebutuhan Sistem 27](#_Toc528129594)

[5.1.1 Deskripsi Umum Sistem 28](#_Toc528129595)

[5.1.2 Deskripsi Umum Perangkat Lunak 28](#_Toc528129596)

[5.1.3 Fungsi Perangkat Lunak 29](#_Toc528129597)

[5.1.4 Karakteristik Pengguna 29](#_Toc528129598)

[5.2 Analisa Kebutuhan Perangkat Lunak 30](#_Toc528129599)

[5.2.1 Kebutuhan Fungsional dan Non-Fungsional 31](#_Toc528129600)

[5.2.2 Pemodelan Kebutuhan 33](#_Toc528129601)

[5.3 Perancangan Perangkat Lunak 76](#_Toc528129602)

[5.3.1 Perancangan Algoritma 76](#_Toc528129603)

[5.3.2 Perancangan Antarmuka 80](#_Toc528129604)

[5.3.3.1 Perancangan Tampilan Halaman *Login* 81](#_Toc528129605)

[5.3.3.2 Perancangan Tampilan Halaman *Register* 82](#_Toc528129606)

[5.3.3.3 Perancangan Tampilan Halaman *Home User* 83](#_Toc528129607)

[5.3.3.4 Perancangan Tampilan Halaman Keranjang Belanja 84](#_Toc528129608)

[5.3.3.5 Perancangan Tampilan Halaman Transaksi 85](#_Toc528129609)

[5.3.3.6 Perancangan Tampilan Halaman Profil 86](#_Toc528129610)

[5.3.3.7 Perancangan Tampilan Gudang Sayur 87](#_Toc528129611)

[5.3.3.8 Perancangan Tampilan *Manage User* 88](#_Toc528129612)

[5.3.3.9 Perancangan Tampilan *Manage* Sayur 89](#_Toc528129613)

[5.3.3.10 Perancangan Tampilan *Manage* Transaksi 90](#_Toc528129614)

[5.3.3.11 Perancangan Tampilan *Login* Web 91](#_Toc528129615)

[5.3.3.12 Perancangan Tampilan *Manage* SayurWeb 92](#_Toc528129616)

[5.3.3.13 Perancangan Tampilan *Manage* TransaksiWeb 93](#_Toc528129617)

[5.3.3.14 Perancangan Tampilan *Manage User* Web 94](#_Toc528129618)

[5.3.3.14 Perancangan Tampilan *Manage User* Web 95](#_Toc528129619)

[5.3.3.15 Perancangan Tampilan Detail SayurWeb 96](#_Toc528129620)

[5.3.3.16 Perancangan Tampilan Detail TransaksiWeb 97](#_Toc528129621)

[BAB 6 IMPLEMENTASI 98](#_Toc528129622)

[6.1 Batasan Implementasi 99](#_Toc528129623)

[6.2 Spesifikasi Sistem 99](#_Toc528129624)

[6.2.1 Spesifikasi Perangkat Keras 99](#_Toc528129625)

[6.2.2 Spesifikasi Perangkat Lunak 100](#_Toc528129626)

[6.3 Implementasi Algoritma 100](#_Toc528129627)

[6.3.1 Implementasi Algoritma *Login* 100](#_Toc528129628)

[6.3.2 Implementasi Algoritma Tambah Data Sayur 101](#_Toc528129629)

[6.3.3 Implementasi Algoritma *Edit* Data Sayur 102](#_Toc528129630)

[6.4 Antarmuka 103](#_Toc528129631)

[BAB 7 PENGUJIAN DAN ANALISIS 112](#_Toc528129632)

[7.1 Pengujian 112](#_Toc528129633)

[7.1.1 Pengujian Validasi 112](#_Toc528129634)

[7.2 Analisis 130](#_Toc528129635)

[7.2.1 Analisis Hasil Pengujian Validasi 130](#_Toc528129636)

[BAB 8 PENUTUP 132](#_Toc528129637)

[8.1 Kesimpulan 132](#_Toc528129638)

[8.2 Saran 133](#_Toc528129639)

[DAFTAR PUSTAKA 134](#_Toc528129640)

[LAMPIRAN 136](#_Toc528129641)

# DAFTAR TABEL

[Tabel 1.1 Kegiatan 3](#_Toc528134013)

[Tabel 3.1 Notasi pada Structured diagram. 17](#_Toc528134014)

[Tabel 3.2 Notasi pada *Use Case Diagram* 18](#_Toc528134015)

[Tabel 3.3 Notasi pada *Sequence Diagram* 20](#_Toc528134016)

[Tabel 3.4 Notasi pada *Entity Relationship Diagram* 22](#_Toc528134017)

[Tabel 5.1 Karakter Pengguna 30](#_Toc528134018)

[Tabel 5.2 Kebutuhan Fungsional 31](#_Toc528134019)

[Tabel 5.4 Skenario *Use Case* *Sign in* 37](#_Toc528134020)

[Tabel 5.5 Skenario *Use Case* *Sign up* 37](#_Toc528134021)

[Tabel 5.6 Skenario *Use Case* Lihat daftar sayur pada *home* 38](#_Toc528134022)

[Tabel 5.7 Skenario *Use Case* Tambah keranjang belanja 38](#_Toc528134023)

[Tabel 5.8 Skenario *Use Case* hapus keranjang belanja 38](#_Toc528134024)

[Tabel 5.9 Skenario *Use Case* *Search* sayur 39](#_Toc528134025)

[Tabel 5.10 Skenario *Use Case* Lihat Keranjang belanja 39](#_Toc528134026)

[Tabel 5.11 Skenario *Use Case* *Edit* jumlah sayur pada keranjang. 39](#_Toc528134027)

[Tabel 5.12 Skenario *Use Case* Hapus daftar pembelian 40](#_Toc528134028)

[Tabel 5.13 Skenario *Use Case* Melengkapi Data pengiriman 40](#_Toc528134029)

[Tabel 5.14 Skenario *Use Case* Lihat transaksi. 40](#_Toc528134030)

[Tabel 5.15 Skenario *Use Case* Melihat detail transaksi. 41](#_Toc528134031)

[Tabel 5.16 Skenario *Use Case* lihat bantuan 41](#_Toc528134032)

[Tabel 5.17 Skenario *Use Case* lihat profil 41](#_Toc528134033)

[Tabel 5.18 Skenario *Use Case* *Edit* Profil 42](#_Toc528134034)

[Tabel 5.19 Skenario *Use Case* *Logout*. 42](#_Toc528134035)

[Tabel 5.20 Skenario *Use Case* lihat daftar sayur pada gudang. 43](#_Toc528134036)

[Tabel 5.21 Skenario *Use Case* Menambah sayur. 43](#_Toc528134037)

[Tabel 5.22 Skenario *Use Case* Hapus Data sayur 43](#_Toc528134038)

[Tabel 5.23 Skenario *Use Case* *Edit* data sayur 44](#_Toc528134039)

[Tabel 5.24 Skenario *Use Case* Melihat daftar *user* 44](#_Toc528134040)

[Tabel 5.25 Skenario *Use Case* Melihat data *user* 44](#_Toc528134041)

[Tabel 5.26 Skenario *Use Case* Hapus data *user* 45](#_Toc528134042)

[Tabel 5.27 Skenario *Use Case* Mengubah status transaksi. 45](#_Toc528134043)

[Tabel 5.28 Skenario *Use Case* Lihat transaksi *On progress*. 46](#_Toc528134044)

[Tabel 5.29 Skenario *Use Case* Lihat transaksi *Completed*. 46](#_Toc528134045)

[Tabel 5.30 Skenario *Use Case* *Login* pada web.. 46](#_Toc528134046)

[Tabel 5.31 Skenario *Use Case* Manajemen sayur. 47](#_Toc528134047)

[Tabel 5.32 Skenario *Use Case* tambah sayur. 47](#_Toc528134048)

[Tabel 5.33 Skenario *Use Case* lihat data sayur. 47](#_Toc528134049)

[Tabel 5.34 Skenario *Use Case* edit data sayur. 48](#_Toc528134050)

[Tabel 5.35 Skenario *Use Case* hapus data sayur. 48](#_Toc528134051)

[Tabel 5.36 Skenario *Use Case search* sayur. 48](#_Toc528134052)

[Tabel 5.37 Skenario *Use Case* manajemen transaksi. 49](#_Toc528134053)

[Tabel 5.38 Skenario *Use Case* lihat detail transaksi. 49](#_Toc528134054)

[Tabel 5.39 Skenario *Use Case* manajemen pengguna. 49](#_Toc528134055)

[Tabel 5.40 Skenario *Use Case* hapus pengguna. 50](#_Toc528134056)

[Tabel 5.41 Skenario *Use Case* tambah pengguna. 50](#_Toc528134057)

[Tabel 5.42 Skenario *Use Case logout*. 51](#_Toc528134058)

[Tabel 6.1 Spesifikasi Perangkat Keras 99](#_Toc528134059)

[Tabel 6.2 Spesifikasi Perangkat Lunak 100](#_Toc528134060)

[Tabel 7.1 Tabel Pengujian Validasi 113](#_Toc528134061)

# DAFTAR GAMBAR

[Gambar 2.1 Struktur Organisasi 5](#_Toc528134450)

[Gambar 3.1. Illustrasi *Android* *Stack* 10](#_Toc528134451)

[Gambar 3.2. *Class Diagram* 16](#_Toc528134452)

[Gambar 3.3. *Package diagram*. 16](#_Toc528134453)

[Gambar 3.4. *Use Case Diagram*. 18](#_Toc528134454)

[Gambar 3.6. *Sequence Diagram*. 20](#_Toc528134455)

[Gambar 3.7. *Use Case Scenario.* 21](#_Toc528134456)

[Gambar 3.7. *Use Case Scenario*. 22](#_Toc528134457)

[Gambar 4.1 Diagram Alir Metodologi 24](#_Toc528134458)

[Gambar 5.1 Diagram Alir Analisis dan Perancangan 27](#_Toc528134459)

[Gambar 5.2 Deskripsi Sistem 28](#_Toc528134460)

[Gambar 5.3 *Entity Relationship Diagram* Untuk Aplikasi Sayur *Online* 34](#_Toc528134461)

[Gambar 5.4 *Use Case Diagram* Untuk Platform *Android* 35](#_Toc528134462)

[Gambar 5.3 *Use Case Diagram* Untuk Platform Web 36](#_Toc528134463)

[Gambar 5.4 *Class Diagram* Untuk Platform *Android* Pada Bagian *User* 52](#_Toc528134464)

[Gambar 5.5 *Class Diagram* Untuk Platform *Android* Pada Bagian Admin 53](#_Toc528134465)

[Gambar 5.6 *Class Diagram* Untuk Platform *Android* Pada *Package* Admin Bagian 1 54](#_Toc528134466)

[Gambar 5.7 *Class Diagram* Untuk Platform *Android* Pada *Package* Admin Bagian 2 54](#_Toc528134467)

[Gambar 5.8 Detail *Class Diagram* Untuk Platform *Android* Pada *Package* *Adapter* 55](#_Toc528134468)

[Gambar 5.9 Detail *Class Diagram* Untuk Platform *Android* Pada *Package* *Fragment* 56](#_Toc528134469)

[Gambar 5.10 Detail *Class Diagram* Untuk Platform *Android* Pada *Package Server* 56](#_Toc528134470)

[Gambar 5.11 Detail *Class Diagram* Untuk Platform *Android* Pada *Package Singleton* 56](#_Toc528134471)

[Gambar 5.12 Detail *Class Diagram* Untuk Platform *Android* Pada *Package drawersayur* 57](#_Toc528134472)

[Gambar 5.13 *Class Diagram* Untuk Platform Web Untuk Proses Pada Website 58](#_Toc528134473)

[Gambar 5.14 *Class Diagram* Untuk Platform Web Untuk Proses *Web Service* 58](#_Toc528134474)

[Gambar 5.15 Detail *Class Diagram* Untuk Platform Web Pada *Folder Controller* 59](#_Toc528134475)

[Gambar 5.16 Detail *Class Diagram* Untuk Platform Web Pada *Folder API* 59](#_Toc528134476)

[Gambar 5.17 Detail *Class Diagram* Untuk Platform Web Pada *Folder App* 60](#_Toc528134477)

[Gambar 5.18 *Activity Diagram* “hapus data *user*” 60](#_Toc528134478)

[Gambar 5.19 *Activity Diagram* “*search* data *user*” 61](#_Toc528134479)

[Gambar 5.20 *Activity Diagram* “perbarui data sayur” 61](#_Toc528134480)

[Gambar 5.21 *Activity Diagram* “hapus data sayur” 62](#_Toc528134481)

[Gambar 5.22 Activity Diagram “*search* sayur” 62](#_Toc528134482)

[Gambar 5.23 Activity Diagram “menambah sayur” 63](#_Toc528134483)

[Gambar 5.24 Activity Diagram “pesan sayur” 63](#_Toc528134484)

[Gambar 5.25 *Activity Diagram* “melihat daftar transaksi” 64](#_Toc528134485)

[Gambar 5.26 *Activity Diagram* “melihat daftar transaksi” 64](#_Toc528134486)

[Gambar 5.27 *Activity Diagram* “*edit* profil” 65](#_Toc528134487)

[Gambar 5.28 *Activity Diagram* “tambah sayur pada web” 65](#_Toc528134488)

[Gambar 5.29 *Activity Diagram* “*search* sayur pada web” 66](#_Toc528134489)

[Gambar 5.30 *Activity Diagram* “ubah sayur pada *web*” 66](#_Toc528134490)

[Gambar 5.31 *Activity Diagram* “tambah *user* pada web” 67](#_Toc528134491)

[Gambar 5.32 *Activity Diagram* “hapus *user* pada web” 67](#_Toc528134492)

[Gambar 5.33 *Activity Diagram* “lihat data transaksi pada web” 68](#_Toc528134493)

[Gambar 5.34 Activity Diagram “Ubah data transaksi pada web” 68](#_Toc528134494)

[Gambar 5.35 *Sequence Diagram* “menambah sayur” 69](#_Toc528134495)

[Gambar 5.36 *Sequence Diagram* “perbarui data sayur” 69](#_Toc528134496)

[Gambar 5.37 *Sequence Diagram* “hapus sayur” 70](#_Toc528134497)

[Gambar 5.38 *Sequence Diagram* “hapus *user*” 70](#_Toc528134498)

[Gambar 5.39 *Sequence Diagram* “*search* sayur” 71](#_Toc528134499)

[Gambar 5.40 *Sequence Diagram* “*search user*” 71](#_Toc528134500)

[Gambar 5.41 *Sequence Diagram* “keranjang belanja” 72](#_Toc528134501)

[Gambar 5.42 *Sequence Diagram* “pesan sayur” 72](#_Toc528134502)

[Gambar 5.43 *Sequence Diagram* “Tambah sayur web” 73](#_Toc528134503)

[Gambar 5.44 *Sequence Diagram* “*Search* sayur web” 73](#_Toc528134504)

[Gambar 5.45 *Sequence Diagram* “ubah sayur web” 74](#_Toc528134505)

[Gambar 5.46 *Sequence Diagram* “hapus *user* web” 74](#_Toc528134506)

[Gambar 5.47 *Sequence Diagram* “tambah *user* web” 75](#_Toc528134507)

[Gambar 5.48 *Sequence Diagram* “lihat transaksi web” 75](#_Toc528134508)

[Gambar 5.49 Diagram Alir Proses Login Sebagai Admin 76](#_Toc528134509)

[Gambar 5.50 Diagram Alir Proses Tambah Data Sayur 77](#_Toc528134510)

[Gambar 5.51 Diagram Alir Proses Lihat Data Sayur 78](#_Toc528134511)

[Gambar 5.52 Diagram Alir Proses Perbarui Data Sayur 79](#_Toc528134512)

[Gambar 5.53 Diagram Alir Proses Hapus Data Sayur 80](#_Toc528134513)

[Gambar 5.54 Perancangan Tampilan Halaman *Login* 81](#_Toc528134514)

[Gambar 5.55 Perancangan Tampilan Halaman *Register* 82](#_Toc528134515)

[Gambar 5.56 Perancangan Tampilan Halaman *Home User* 83](#_Toc528134516)

[Gambar 5.57 Perancangan Tampilan Halaman Keranjang Belanja 84](#_Toc528134517)

[Gambar 5.58 Perancangan Tampilan Halaman Transaksi 85](#_Toc528134518)

[Gambar 5.59 Perancangan Tampilan Halaman Profil 86](#_Toc528134519)

[Gambar 5.60 Perancangan Tampilan Halaman Gudang Sayur 87](#_Toc528134520)

[Gambar 5.61 Perancangan Tampilan Halaman *Manage* *User* 88](#_Toc528134521)

[Gambar 5.62 Perancangan Tampilan Halaman Manage Sayur 89](#_Toc528134522)

[Gambar 5.63 Perancangan Tampilan Halaman *Manage* Transaksi 90](#_Toc528134523)

[Gambar 5.64 Perancangan Tampilan Halaman *Login* Web 91](#_Toc528134524)

[Gambar 5.65 Perancangan Tampilan Halaman *Manage* Sayur Web 92](#_Toc528134525)

[Gambar 5.66 Perancangan Tampilan Halaman *Manage* Transakasi Web 93](#_Toc528134526)

[Gambar 5.67 Perancangan Tampilan Halaman *Manage* *User* Web 94](#_Toc528134527)

[Gambar 5.68 Perancangan Tampilan Halaman *Manage* *User* Web 95](#_Toc528134528)

[Gambar 5.69 Perancangan Tampilan Halaman Detail Sayur Web 96](#_Toc528134529)

[Gambar 5.70 Perancangan Tampilan Halaman Detail Transaski Web 97](#_Toc528134530)

[Gambar 6.1 Diagram Alir Implementasi 98](#_Toc528134531)

[Gambar 6.2 Kode Program *Login* 101](#_Toc528134532)

[Gambar 6.3 Kode Program Tambah Sayur 102](#_Toc528134533)

[Gambar 6.4 Kode Program *Edit* Data Sayur 103](#_Toc528134534)

[Gambar 6.5 Implementasi Halaman *Login* 104](#_Toc528134535)

[Gambar 6.6 Implementasi Halaman *Register* 104](#_Toc528134536)

[Gambar 6.7 Implementasi Halaman *Home User* 105](#_Toc528134537)

[Gambar 6.8 Implementasi Halaman Keranjang Belanja 106](#_Toc528134538)

[Gambar 6.9 Implementasi Halaman Transaksi 107](#_Toc528134539)

[Gambar 6.10 Implementasi Halaman Profil 108](#_Toc528134540)

[Gambar 6.11 Implementasi Halaman Gudang Sayur 109](#_Toc528134541)

[Gambar 6.12 Implementasi Halaman *Manage User* 109](#_Toc528134542)

[Gambar 6.13 Implementasi Halaman *Manage* Sayur 110](#_Toc528134543)

[Gambar 6.13 Implementasi Halaman *Manage* Transaksi 111](#_Toc528134544)

[Gambar 7.1 Diagram Alir Pengujian dan Analisis 112](#_Toc528134545)

# DAFTAR LAMPIRAN

[1. FOTO DOKUMENTASI KEGIATAN 136](#_Toc528134547)

[2. *USER* MANUAL 136](#_Toc528134548)

[1.1 Admin 136](#_Toc528134549)

[1.1.1 Bagian I – *Login* 136](#_Toc528134550)

[1.1.2 Bagian II – Menambahkan sayur. 139](#_Toc528134551)

[1.1.3 Bagian III – *Edit* sayur. 141](#_Toc528134552)

[1.1.4 Bagian IV – Hapus sayur 142](#_Toc528134553)

[1.1.5 Bagian V – Melihat dan menghapus daftar *user*. 143](#_Toc528134554)

[1.1.6 Bagian VI – Melihat dan mengubah status transaksi. 145](#_Toc528134555)

[*2.2.* *User* 147](#_Toc528134556)

[1.2.1 Bagian I – Login dan *Register*. 147](#_Toc528134557)

[1.2.2 Bagian II – Memesan sayur 149](#_Toc528134558)

[1.2.3 Bagian III – Melihat transaksi 150](#_Toc528134559)

[1.2.4 Bagian IV – *Edit* profil 152](#_Toc528134560)

[1.2.5 Bagian V – *Logout* 152](#_Toc528134561)

PENDAHULUAN

## Latar belakang

Sistem Informasi merupakan salah satu hal yang terpenting dalam era globalisasi saat ini. Dengan pesatnya pertumbuhan teknologi informasi, maka mengharuskan kita untuk turut serta dalam mengikuti perkembangan tersebut. Seiring dengan perkembangan teknologi informasi, kebutuhan akan informasi yang cepat, tepat dan akurat sangat diperlukan. Karena itu, keberadaan sistem yang terkomputerisasi yang handal sudah menjadi kebutuhan mutlak bagi perusahaan dalam proses bisnisnya. Sistem informasi berbasis android merupakan salah satu bentuk pemanfaatan *handphone* untuk untuk menghasilkan sebuah sistem informasi penjualan yang dapat mengelola data dan melakukan pembelian secara *online*.

Dalam pelayanan informasi, dalam kasus ini petani sayur tentunya belum mempunyai suatu media yang menyediakan layanan untuk menginformasikan tentang apa saja yang mereka jual secara kepada konsumen. Tentunya kita sebagai konsumen harus mendatangi pasar untuk mengetahuinya dan untuk melakukan pembelian sayur. Dengan masalah tersebut akan menimbulkan ketidak-efisenan waktu dalam melakukan kegiatan berbelanja dan melakukan kegiatan sehari-hari. Pembuatan sistem informasi dan penjualan sayur berbasis android adalah satu cara untuk mempermudah hal tersebut.

Dalam perancangan proyek tim ini, penulis dan tim menggunakan bahasa pemrograman yang digunakan oleh *Android studio* yaitu *java* dan *xml* untuk *front end* dan *webservice php framework laravel* untuk *back end*. Dengan menggunkan *Android studio* dan *php framework laravel* memudahkan penulis dan tim untuk pembuatan aplikasi ini sesuai dengan proses bisnis yang telah dijelaskan oleh pemilik dari CV. Sarana Utama Solusindo.

## 1.2 Rumusan masalah

Berdasarkan permasalahan yang telah dijelaskan pada latar belakang maka disusun rumusan masalah sebagai berikut :

1. Bagaimanakah merancang aplikasi sayur *online* berbasis android menggunakan *webservice* ?
2. Bagaimanakah mengimplementasikan aplikasi sayur *online* berbasis android menggunakan *webservice* ?

## 1.3 Tujuan

Tujuan dari pembuatan aplikasi ini adalah sebagai berikut:

1. Merancang aplikasi sayur *online* berbasis android.
2. Mengimplementasikan aplikasi sayur *online* berbasis android.

## 1.4 Manfaat

Penulisan dokumentasi ini memilki beberapa manfaat diantaranya :

1. Mempermudah konsumen untuk membeli sayur secara digital menggunakan aplikasi sayur *online* sehingga dapat membantu dalam melakukan kegiatan berbelanja tanpa harus mendatangi pasar.
2. Mempermudah petani sayur dalam menjual produknya melalui aplikasi sayur *online* ke konsumen.

## 1.5 Batasan masalah

Batasan masalah pada pengembangan ini sebagai berikut :

1. Tidak ada forum diskusi untuk aplikasi ini
2. Pembuatan aplikasi sayur *online* berbasis android ini menggunakan bahasa pemograman *java* , *xml* dan *php framework laravel* untuk *webservice*
3. Aplikasi ini memiliki website untuk mengolah data sayur

## 1.6 Sistematika pembahasan

Sistematika penulisan pada pengembangan aplikasi ini direncanakan sebagai berikut :

**BAB 1 PENDAHULUAN**

Bab ini berisi tentang latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, sistematika penulisan laporan, dan pelaksanaan PKL.

**BAB 2 PROFIL PERUSAHAAN**

Bab ini membahas tentang sejarah dan perkembangan serta profil CV. Sarana Utama Solusindo, Visi, Misi, Komitmen CV. Sarana Utama Solusindo, Struktur organisasi CV. Sarana Utama Solusindo beserta tugas, ruang lingkup di CV. Sarana Utama Solusindo.

**BAB 3 LANDASAN KEPUSTAKAAN**

Bab ini berisi tentang dasar teori perangkat lunak dalam perancangan, pengembangan dan pengujian.

**BAB 4 METODOLOGI**

Bab ini dibahas mengenai penerapan aplikasi Jual Beli Sayur *Online* dengan layanan *web service* secara detail dengan langkah-langkah pengerjaan serta tampilan gambar-gambar dari implementasi yang dilakukan.

**BAB 5 ANALISIS DAN PERANCANGAN**

Bab ini menguraikan analisis kebutuhan serta perancangan sistem yang menjadi objek studi kasus pembuatan sistem.

**BAB 6 IMPLEMENTASI**

Bab ini berisi tentang tahap-tahap implementasi yang telah dilakukan dalam pengembangan aplikasi.

**BAB 7 PENGUJIAN DAN ANALISIS**

Bab ini berisi tentang tahap-tahap pengujian fungsionalitas dari sistem dan analisis hasil pengujian.

**BAB 8 PENUTUP**

Bab ini menguraikan kesimpulan yang diperoleh dari pembuatan dan pengujian sistem informasi yang dibuat dalam pembuatan sistem ini ini serta saran-saran untuk pengembangan lebih lanjut.

## 1.7 Pelaksanaan Praktek Kerja Lapang

* **Waktu**

PKL dilaksanakan mulai tanggal 2 April 2018 hingga tanggal 31 Agustus 2018. Adapun kegiatan yang dilakukan selama pelaksanaan PKL di CV Sarana Utama Solusindo dapat dilihat pada Tabel 1.1 berikut ini:

Tabel 1.1 Kegiatan

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| NO | Kegiatan | Juli 2018 | | | | Agustus 2018 | | | |
| I | II | III | IV | II | III | IV | I |
| 1 | Pengenalan CV SASINDO secara garis besar dan ruang lingkup kerja | X |  |  |  |  |  |  |  |
| 2 | Penentuan objek yang akan dibuat | X |  |  |  |  |  |  |  |
| 3 | Pembagian tugas pengerjaan baik isi laporan dan kode sistem | X |  |  |  |  |  |  |  |
| 4 | Analisis kebutuhan *user* | X | X | X |  |  |  |  |  |
| 5 | Implementasi Program | X | X | X | X | X | X | X | X |
| 6 | Pembuatan Laporan | X | X | X | X | X | X | X | X |

* **Tempat**

PKL dilaksanakan di:

Nama Instansi : CV Sarana Utama Solusindo (SASINDO)

Alamat : Lantai 2 Jalan Anggrek Garuda 49 , Kota Malang

Kode Pos : 65141

E-mail : cs.sasindo@gmail.com

Website : http://sasindo.id

BAB 2 PROFIL PERUSAHAAN

## 2.1 Sejarah dan Perkembangan CV Sarana Utama Solusindo (SASINDO)

CV. Sarana Utama Solusindo (SASINDO) adalah salah satu perusahan berbasis Teknologi Informasi (TI) yang mendedikasikan diri untuk peningkatan fungsi TI pada setiap sendi kehidupan khususnya di Indonesia. Perusahaan ini memiliki beberapa ruang lingkup bisnis utama yaitu solusi bisnis IT / IT *Business Solution* (IT-Busol), Konsultan TI / IT *Consultant* ( IT-C), Pendidikan dan Pelatihan IT/ *IT Education and Training* (IT-ET)

CV. Sarana Utama Solusindo didirikan atas dasar pemikiran dan keinginan yang sangat kuat, bahwa sudah saatnya untuk menerapkan TI di dalam setiap sendi kehidupan, karena TI dapat membantu setiap pekerjaan manusia. Diharapkan dengan menerapkan TI untuk membatu kerja manusia maka hasil dari pekerjaan akan lebih baik jika dibandingkan dengan bekerja tanpa menggunakan TI.

## 2.2 Visi, Misi dan Komitmen Perusahaan.

CV. Sarana Utama Solusindo yang bergerak dalam bidang teknologi memiliki visi, misi dan komitmen agar perusahaannya terus berkembang dan meningkatkan sendi kehidupan dengan menggunakan teknologi.

## 2.2.1 Visi

Menjadi perusahaan TI yang terdepan dalam penerapan TI dalam setiap sendi kehidupan.

## 2.2.2 Misi

1. Memberikan edukasi kepada masyarakat tentang pentingnya TI untuk kehidupan.
2. Memberikan solusi terbaik dalam setiap masalah yang dapat diselesaikan dengan TI.
3. Ikut serta memberikan kontribusi dalam perkembangan di dunia TI.

## 2.2.3 Komitmen

TI dapat membantu menyelesaikan masalah manusia dalam kehidupan, untuk itu TI harus menjadi bagian dari kehidupan manusia.

## 2.3 Struktur Organisasi CV SARANA UTAMA SOLUSINDO

Berikut ini struktur organisasi dalam CV. Sarana Utama Solusindo dijelaskan pada Gambar 2.1 :

Director

Administration & Marketing

IT Engineer

Customer Service

**Power House:**

2nd Floor, 49th Anggrek Garuda Street Malang, 65141

http://sasindo.id

##### Gambar 2.1 Struktur Organisasi

## 2.4 Ruang Lingkup CV SARANA UTAMA SOLUSINDO

Beberapa pekerjaan di bidang TI yang menjadi ruang lingkup bisnis CV. Sarana Utama Solusindo adalah:

1. Solusi Bisnis TI (IT *Business Solution*)
2. Konsultan TI (IT *Consultant*)
3. Pendidikan & Pelatihan TI (IT *Training*)

### 2.4.1 Solusi Bisnis TI (IT Business Solution) dan Konsultan IT (IT Consultant)

CV. Sarana Utama Solusindo siap menjadi partner anda dalam pengembangan TI baik itu dalam hal pengembangan perangkat lunak (*software*) maupun pengembangan perangkat keras (*hardware*). Beberapas fokus dari CV. Sarana Utama Solusindo adalah:

1. Pengembangan Sistem Informasi Manajemen (*Management Information Sistem*)
   * Sistem Informasi Puskesmas (SIMAS) – *on going*

Sistem informasi ini merupakan sistem informasi alur proses pengobatan yang ada di puskesmas.

* + Sistem Informasi Geografis Puskesmas (SIMAS) – *on going* Sistem informasi ini berbasis geografis dengan melakukan pemetaan wilayah menurut alamat dari pasien. Pemetaan dilakukan berdasarkan poin-poin tertentu yang ingin diketahui salah satunya berdasarkan penyakit yang diderita pasien.
  + *Command Center* (Ngalam *Command Center*) – *on going* Sistem informasi yang berfungsi untuk mengetahui informasi semua kegiatan yang ada di beberapa dinas pemerintahan secara terpusat.
  + Smart City – *on going*

Sistem yang digunakan untuk memudahkan kegiatan yang ada di dalam pemerintahan yang berbasis teknologi.

1. Pengembangan Aplikasi Dekstop & Perangkat Bergerak (Dekstop &

*Mobile Application*)

* + Aplikasi Antrean Puskesmas – *on going*

1. Pengembangan Perangkat Cerdas (*Smart Device*)
   * Deteksi Dini Bahaya Banjir dan Banjir Rob (*Flood Early Warning System*) – *completed*
   * Deteksi Dini Bahaya Tsunami (*Tsunami Early Warning System*) – *completed*
   * Deteksi Dini Bahaya Tanah Longsor (*Landslide Early Warning System*) – *on going*
   * Pemantauan Cuaca dan Klimatologi – *completed*

### 2.4.2 Pendidikan dan Pelatihan (IT Training)

CV. Sarana Utama Solusindo siap memberikan pelatihan dan pendampingan dalam proses penelitian baik yang dilakukan oleh mahasiswa maupun pihak dari luar / umum.

Beberapa pelatihan yang dilakukan oleh CV. Sarana Utama Solusindo adalah:

1. Pelatihan penelitian berbasis Kecerdasan Buatan (*Artificial Intelligence*)
   * *Data Mining*
   * *Forecasting* (Peramalan)
   * *Decission Support System* (Sistem Pendukung Keputusan)
   * *Expert System* (Sistem Pakar)
   * *Optimization Algorithm* (Algoritma Optimasi)
   * *Computer Vision*
   * dll
2. Pelatihan penelitian berbasis Rekayasa Perangkat Lunak (*Software*

*Engineering*)

* + Aplikasi Dekstop (*Dekstop Application*)
  + Aplikasi Perangkat Bergerak (*Mobile Application*)
  + A0070likasi Berbasis Web (*Web Application*)

BAB 3 LANDASAN KEPUSTAKAAN

## 3.1 Sistem Penjualan.

Kata sistem berasal dari bahasa Yunani yaitu *systema*, *systema* sendiri memiliki pengertian himpunan bagian atau komponen yang independen namun saling berhubungan secara teratur dan merupakan suatu keseluruhan yang utuh. Sistem terdiri atas struktur dan proses. Struktur sistem merupakan unsur-unsur dan elemen yang membentuk sebuah sistem itu sendiri, sedangkan proses sistem menjelaskan bagaimana cara kerja setiap unsur atau elemen yang terdapat pada sistem dapat mencapai tujuan yang telah ditentukan. Setiap sistem merupakan bagian dari sistem lain yang lebih besar dan terdiri atas berbagai sistem yang lebih kecil, yang disebut Subsistem. (Rusdiana. H.A & Irfan, Moch., 2014).

Dengan demikian dapat disimpulkan bahwa sistem merupakan kumpulan atau himpunan dari beberapa bagian yang memiliki hubungan dan keterkaitan dan juga kerja sama dalam satu kesatuan untuk mencapai tujuan dari sistem tersebut. Sebuah sistem dibuat dengan tujuan untuk mencapai sebuah hasil atau tujuan tertentu dalam ruang lingkup yang sempit.

### 3.1.2 Sistem Jual Beli *Online*.

Sistem Jual Beli *Online* merupakan sistem jual beli atau transaksi yang dilakukan dengan bantuan teknologi informasi antara satu atau dua lebih organiasi atau antara sebuah organisasi dengan satu atau lebih pelanggan akhir antara satu atau lebih jaringan komputer. (Rusdiana. H.A & Irfan, Moch., 2014).

Keuntungan utama melakukan transaksi secara *online* adalah menghemat biaya serta dapat menjangkau pelanggan yang lebih luas. Selain itu, keuntungan lain dari transaksi secara *online*, yaitu:

1. Distribusi yang lebih murah dari dokumen dan produk digital;
2. Kemampuan memberikan layanan dukungan kepada pelanggan;
3. Kanal pemasaran yang baru;
4. Memiliki kemampuan untuk menarik pelanggan baru;
5. Menyediakan satu titik lokasi kontak untuk bermacam-macam

produk dan jasa;

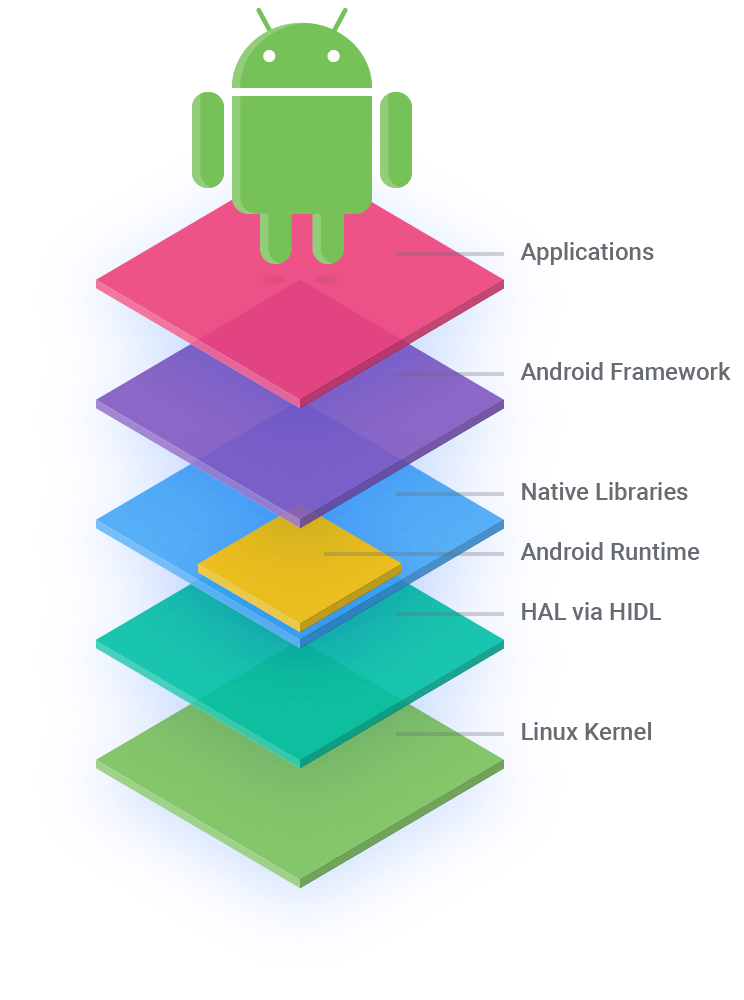
1. Dapat digunakan sebagai media riset pasar.

Selain keuntungan atau kelebihan tersebut transaksi secara *online* juga memiliki beberapa kelemahan, yaitu :

1. Keamanan akses dan transmisi;
2. Beban *traffic* yang terlalu *overload*;
3. Kesulitan sensor dan kesulitan mengukur kinerja dari situs, keberhasilannya akan diukur dengan jumlah orang yang mengunjungi atau diukur dengan cara lain.

## 3.2 *Android*

*Android* adalah sebuah perangkat lunak atau *platform* *open source* untuk perangkat seluler dan merupakan sebuah proyek aplikasi *open source* yang dipimpin pengembangannya oleh *Google*. *Google* menawarkan beberapa program dalam pengembangan sebuah sistem android salah satunya adalah *Android Open Source Project* (AOSP) yaitu sebuah wadah informasi dan *source code* yang diperlukan untuk membuat varian kustomisasi dari *stack* *Android*, perangkat port dan aksesoris ke dalam platform *Android*, dan memastikan kompabilitas sebuah perangkat untuk menjaga ekosistem dari pengembangan *Android* memiliki lingkungan yang sehat dan stabil untuk para penggunanya (Android, 2018).



##### Gambar 3.1. Illustrasi *Android* *Stack*

Sumber : Android (2018)

*Android* diimplementasikan dalam bentuk tumpukan arsitektur perangkat lunak atau sebuah *Software Stack* yang terdiri dari kernel Linux, perangkat *Runtime* dan *libraries* yang sesuai, *Framework* aplikasi dan sekumpulan aplikasi. Kernel Linux bekerja sebagai jembatan antara perangkat keras dan lapisan atas, ART melakukan penerjemahan *bytecode* menjadi perintah yang dikenali oleh *processor* perangkat, *Android Libraries* menyediakan bantuan untuk menangani perintah-perintah umum (*String*, jaringan, dan manipulasi file),*Native libraries* adalah sekumpulan layanan yang merupakan dari bahasa C/C++, *Android Framework* adalah serangkaian layanan yang secara kolektif membentuk lingkungan untuk menjalankan dan mengelola android, lalu yang terakhir aplikasi yang merupakan sekumpulan aplikasi android *native* dengan implementasi tertentu.Aplikasi yang terdapat dan dibuat pada android sebagian besar ditulis dalam Bahasa *Java* dan dikompilasi ke dalam format *bytecode* pada lingkungan pengembangan aplikasi dengan Android Studio dan ketika aplikasi dijalankan pada perangkat maka bytecode tadi akan dikompilasi oleh *ART* agar dapat dibaca oleh *CPU*. Tujuan utama dari pembuatan arsitektur *Android* yang seperti itu adalah untuk meningkatkan kinerja dan efisiensi, baik dalam mengeksekusi aplikasi yang telah dibuat dan dalam penerapan penggunaan kembali dan pengembangan dalam desain aplikasi android yang dibuat. (Smyth, 2015)

Tujuan Android adalah untuk menghindari kegagalan dalam pengembangan yaitu dimana satu pemegang industri mempunyai kendali dan membatasi inovasi yang muncul dari pengguna dan pengembang lain yang merupakan sebuah prinsip dalam pengembangan *open source* untuk tetap membuka dan memberikan ruang untuk segala inovasi. Untuk itu, Android adalah sistem operasi yang berkualitas produksi untuk menjembatani pembuatan produk-produk perangkat lunak yang baik bagi konsumen, lengkap dengan *source code* yang dapat disesuaikan dan dikembangkan dan juga dapat dipindah hampir ke semua perangkat dan memiliki dokumentasi publik yang tersedia untuk semua orang dan dapat diakses secara mudah.

### 3.2.1 *Android Studio*

*Android Studio* adalah *Integrated Development Environment* (IDE) resmi untuk pengembangan aplikasi *Android*, yang berpedoman dengan *IntelliJ IDEA* (Android Developer,2018). Selain memiliki *tools editor* dan pengembangan yang kuat dari *IntelliJ*, *Android* *Studio* juga menawarkan banyak fitur yang dapat digunakan dalam meningkatkan produktivitas pengguna saat membuat dan mengembangkan aplikasi *Android*, seperti:

1. Sistem bangun berbasis *Gradle* yang fleksibel
2. Emulator yang kaya dan kaya fitur
3. Lingkungan terpadu tempat Anda dapat mengembangkan untuk semua perangkat *Android*
4. Instant Run untuk mendorong perubahan pada aplikasi yang sedang berjalan tanpa membangun *APK* baru
5. Template kode dan integrasi *GitHub* untuk membantu Anda membuat fitur aplikasi umum dan mengimpor kode sampel
6. Alat dan kerangka pengujian ekstensif
7. Alat serat untuk menangkap kinerja, kegunaan, kompatibilitas versi, dan masalah lainnya
8. Dukungan *C ++* dan *NDK*
9. Dukungan bawaan untuk *Google Cloud Platform*, sehingga mudah untuk mengintegrasikan *Google Cloud Messaging* dan *App Engine.*

Dengan segala fitur yang telah diberikan dalam *IDE* android studio, hal tersebut memudahkan para pengembang perangkat lunak androiddalam melakukan penggembangan yang lebih baik lagi. Untuk mendapatkan informasi terkait pengembangan dengan *android studio* juga sangat mudah karena telah banyak situs dan banyak pengembang yang menggunakan Android Studio sebagai media pengembangan aplikasi android. Pengembang android juga dibantu dengan banyaknya *library* yang disediakan untuk mengembangkan sebuah aplikasi yang lebih luas dan lebih bermanfaat lagi.

## 3.3 *Web Service*

*Web services* adalah suatu sistem perangkat lunak yang didesain untuk mendukung interaksi mesin ke mesin pada suatu jaringan. Ia mempunyai suatu interface yang diuraikan dalam suatu format machine-processible seperti *WSDL* (*Web Service Description Language*). Sistem lain yang berinteraksi dengan *Web* service dilakukan melalui interface/antar muka menggunakan pesan seperti pada *SOAP*. Pada umumnya pesan ini melalui *HTTP* dan *XML* yang merupakan salah satu standard web. (W3C,2018)

*Web Service* dapat diakses secara *remote* oleh berbagai perangkat dengan sebuah perantara tertentu. Secara umum,*web service* dapat diidentifikasikan dengan menggunakan *URL* seperti hanya web pada umumnya. Namun yang membedakan *web service* dengan web pada umumnya adalah interaksi yang diberikan oleh *web service*. Berbeda dengan *URL* *web* pada umumnya, *URL* *web service* hanya menggandung kumpulan informasi, perintah, konfigurasi atau sintaks yang berguna membangun sebuah fungsi-fungsi tertentu dari aplikasi yang kemudian akan diolah lagi oleh perangkat maupun aplikasi yang terhubung.

## 3.4 *Laravel*

*Laravel* adalah sebuah *framework* aplikasi web PHP *open source* dengan sintaksis yang lebih ekspresif dan ditujukan untuk pengembangan aplikasi web mengikuti pola arsitektur *MVC* (Chen, et al.,2017). *Laravel* melakukan pengembangan yang mempermudah tugas-tugas sederhana dan umum dengan kerangka yang kuat dan mudah dipahami pada sebagian besar proyek web, seperti otentikasi, *routing, session* dan *caching.* Dalam pengembangannya, *Laravel* menggabungkan beberapa fitur-fitur dari *framework* lain yang pernah kita lihat sebelumnya termasuk *framework* yang diterapkan dalam bahasa lain, seperti *Ruby on Rails*, *ASP*.*NET* *MVC*, dan *Sinatra.*

*Laravel* dapat diakses secara mudah, lebih kuat dan menyediakan *tools* yang memadai untuk pengembangan aplikasi yang besar dan kuat. *Laravel* memiliki susunan yang baik sebagai sebuah kontrol, sistem migrasi yang baik, dan dukungan pengujian unit yang terintegrasi secara ketat memberi bantuan yang baik bagi pengembang untuk mengembangkan aplikasi dengan baik. (Laravel,2018)

### ***3.4.1 Model-View-Controller* (MVC)**

*Model-View-Controller* merupakan sebuah pola arsitektur yang membagi kode dalam 3 basis kode program, yaitu:

1. *Model*

*Model* merupakan sebuah basis dimana pengembang dapat melakukan manipulasi data. *Model* sendiri berada di antara data dan aplikasi. Data dapat disimpan dalam berbagai jenis sistem *database* seperti *MySQL*, *file* *XML*, atau bahkan *Excel*.

1. *View*

*View* merupakan kumpulan kode yang bertugas untuk menampilkan sebuah tampilan visual data yang diterima *Controller* dari *Model*. Sedangkan dalam *Laravel* menggunakan template Blade untuk tampilan dengan menggunakan kode PHP seperti biasa.

1. *Controller*.

*Controller* bertugas menangani permintaan dan meneruskan data dari *Model* menuju *View*. Jadi dapat dianggap *Controller* merupakan sebuah pengguhubung antara *Model* dan *View*.

Pola arsitektur MVC memungkinkan aplikasi web memiliki banyak pandangan berbeda dari satu model umum (Armel, 2014). Dalam pengembangan MVC, satu *Model* untuk satu tabel akan dapat dibuat dan melalui satu *Model* tersebut, beberapa *View* dapat dibuat. MVC dengan cepat menjadi standar industri yang digunakan dalam setiap pengembangan web secara modern. Banyak framework menjadikan MVC untuk memisahkan logika antar lapisan aplikasi.

## 3.5 *Unified Modeling Language (UML)*

*Modelling Language* merupakan alat perancangan sistem yang berorientasi pada objek. *UML* diagram memiliki tujuan utama untuk membantu tim pengembangan proyek untuk dapat berkomunikasi, mengeksplorasi potensi desain, dan memvalidasi desain arsitektur perangkat lunak atau pembuat program (UML Diagrams,2018). *UML* mempunyai tiga kategori utama yaitu struktur diagram, *behaviour* diagram dan *interaction* diagram. Dimana masing-masing kategori tersebut memiliki diagram yang menjelaskan arsitektur sistem dan saling terintegrasi.

Tujuan utama pemanfaatan *UML* menurut Sugrue J. 2009 adalah sebagai berikut:

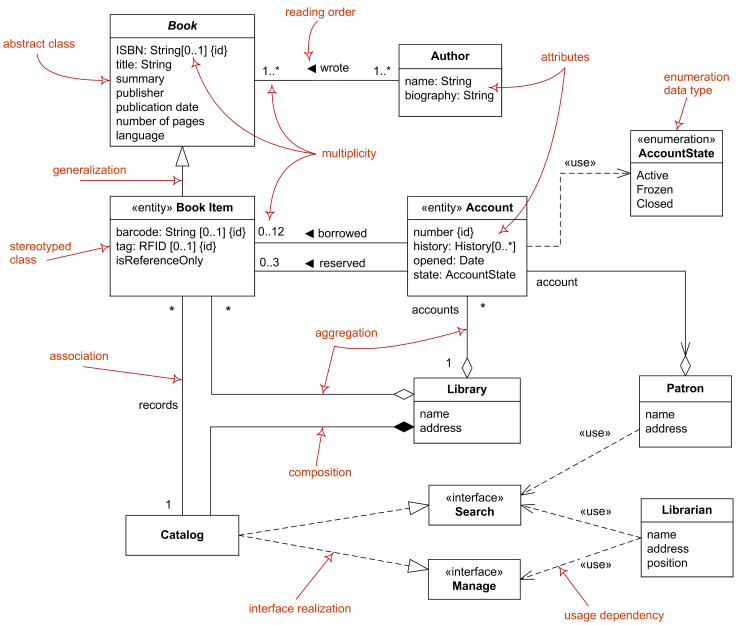
1. Menyediakan bagi pengguna (analisis dan desain sistem) suatu bahasa pemodelan visual yang ekspresif sehingga mereka dapat mengembangkan dan melakukan pertukaran model data yang bermakna.
2. Menyediakan mekanisme yang spesialisasi untuk memperluas konsep inti.
3. Karena merupakan bahasa pemodelan visual dalam proses pembangunannya maka UML bersifat independen terhadap bahasa pemrograman tertentu.
4. Memberikan dasar formal untuk pemahaman bahasa pemodelan.
5. Mendorong pertumbuhan pasar terhadap penggunaan alat desain sistem yang berorientasi objek (OO).
6. Mendukung konsep pembangunan tingkat yang lebih tinggi seperti kolaborasi, kerangka, pola dan komponen terhadap suatu sistem.
7. Memiliki integrasi praktik terbaik.

### 3.5.1 *Structured Diagram.*

Diagram dengan model terstruktur menggambarkan organisasi suatu sistem dalam hal komponen-komponen yang membentuk sistem tersebut seperti kelas, objek, metode yang digunakan serta hubungan antar komponen tersebut dalam sistem. (Sommerville, 2011) Struktur diagram yang digunakan dalam pembuatan aplikasi ini, yaitu :

#### *Class Diagram.*

*Class diagram* Menggambarkan struktur statis dari kelas dalam sistem yang dibuat dan menggambarkan atribut, operasi dan hubungan antara kelas. Berikut merupakan contoh dari *Class Diagram* :

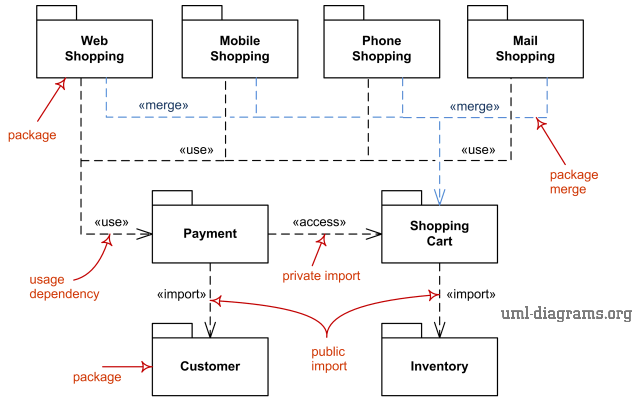


##### Gambar 3.2. *Class Diagram*

Sumber : UML Diagrams(2018)

#### *Package Diagram.*

*Package Diagram* biasanya digunakan untuk menggambarkan tingkat organisasi yang tinggi dari suatu proyek *software.* Dalam penggunaan notasi pada *package diagram* memiliki persamaan dengan penggunaan pada *class diagram*. Berikut merupakan contoh dari *Pakcage Diagram* :



##### Gambar 3.3. *Package diagram*

Sumber : UML Diagrams (2018)

Tabel 3.1 Notasi pada Structured diagram.

|  |  |  |  |
| --- | --- | --- | --- |
| No | Tipe | Deskripsi | Simbol |
| 1 | *Class* | Notasi yang digunakan untuk menggambarkan sebuah objek, atribut, maupun operasi. |  |
| 2 | *Association* | Notasi yang digunakan untuk menunjukkan hubungan antar kelas, banyak kelas atau dengan kelas itu sendiri. |  |
| 3 | *Package* | Notasi yang digunakan untuk menunjukkan package atau kompulan dari kelas yang dibungkus dalam package. |  |
| 4 | *Generalization* | Notasi yang digunakan untuk menunjukkan hubungan antar kelas yang menggunakan pewarisan dari kelas utama ke kelas anak. |  |
| 5 | *Dependency* | Notasi yang menunjukkan hubungan kelas yang bergantung pada kelas lain. |  |
| 6 | *Agregation* | Notasi yang menunjukkan hubungan kelas yang terdiri dari beberapa komponen yang lebih kecil. |  |

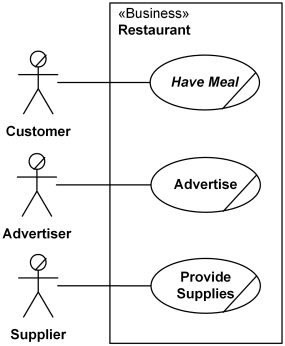
Sumber : Dennis (2012)

### **3.5.2 *Behavior Diagram***.

Menggambarkan ciri-ciri *behavior*/metode/ fungsi dari sebuah sistem secara dinamis atau *business process*. *Behavior Diagram* menunjukkan apa yang terjadi atau apa yang seharusnya terjadi ketika suatu sistem merespons atau menerima suatu stimulus atau *trigger* dari lingkungannya (Sommerville,2011). *Behavior diagram* yang digunakan dalam pembuatan aplikasi ini, yaitu :

#### *Use Case Diagram*

Menggambarkan untuk menggambarkan serangkaian tindakan *actor*, *use case* dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor. Setiap *use case* harus memberikan beberapa hasil yang dapat diamati dan berharga kepada para aktor atau stakeholder lain dari sistem. Berikut merupakan contoh dari *Use Case Diagram* :



##### Gambar 3.4. *Use Case Diagram*

Sumber : UML Diagrams (2018)

Tabel 3.2 Notasi pada *Use Case Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| No | Tipe | Deskripsi | Simbol |
| 1 | *Actor* | Notasi yang digunakan untuk menggambarkan seseorang (*user*) atau sistem lain yang menggunakan fitur atau fungsi yang berada dalam sistem. Perlu dicatat bahwa *actor* berinteraksi dengan *use case* tetapi tidak memiliki kontrol atas *use case*. |  |

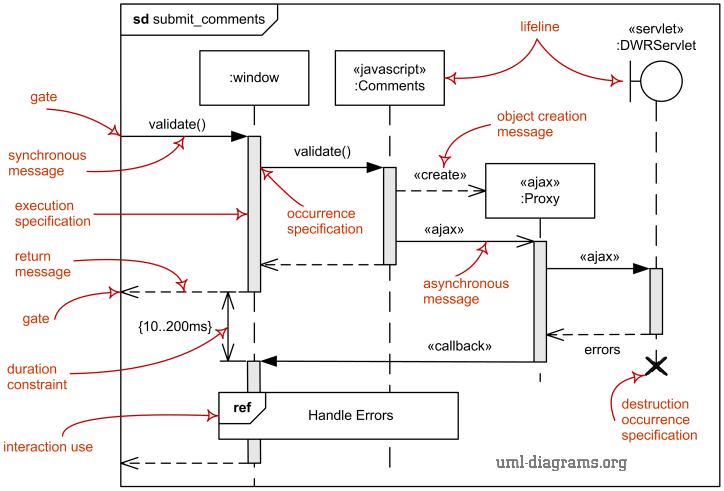
Tabel 3.3 Notasi pada *Use Case Diagram* (lanjutan)

|  |  |  |  |
| --- | --- | --- | --- |
| 2 | *Use case* | Notasi yang digunakan untuk menggambarkan fungsionalitas atau fitur yang disediakan oleh sistem yang saling bertukar pesan antar unit dengan *actor* , yang dinyatakan dalam bentuk kata kerja. |  |
| 3 | *Asociation* | Notasi yang digunakan untuk menunjukkan hubungan antar siapa dan apa yang meminta interaksi secara langsung antar *use case* maupun *actor*. |  |
| 4 | *extend* | *Extend*, merupakan perluasan dari *use case* lain jika dalam kondisi tertentu atau memiliki syarat. |  |
| 5 | *include* | *Include*, merupakan di dalam *use case* lain atau pemanggilan *use case* oleh *use case* lain. |  |

Sumber : Dennis (2012)

#### *Sequence Diagram*

*Sequence diagram* adalah jenis diagram interaksi yang paling umum, yang berfokus pada pertukaran pesan antara sejumlah *lifetime* (Dennis ,et al., 2012). Sequence diagram menggambarkan interaksi dengan berfokus pada urutan pesan yang dipertukarkan, bersama dengan spesifikasi kemunculannya yang sesuai pada *lifetime*. Berikut merupakan contoh dari *Sequence Diagram* :



##### Gambar 3.6. *Sequence Diagram*

Sumber : UML Diagrams(2018)

Tabel 3.4 Notasi pada *Sequence Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| No | Tipe | Deskripsi | Simbol |
| 1 | *Actor* | Notasi yang digunakan untuk menggambarkan seseorang (*user*) atau sistem lain yang menggunakan fitur atau fungsi yang berada dalam sistem. |  |
| 2 | *Object* | Notasi yang digunakan untuk menggambarkan *object* yang berpartisipasi dalam sebuah *sequence* dengan mengirim pesan atau menerima pesan. |  |
| 3 | *Lifeline* | Notasi yang digunakan untuk menunjukkan lama hidup suatu *object* dalam *sequence*. |  |

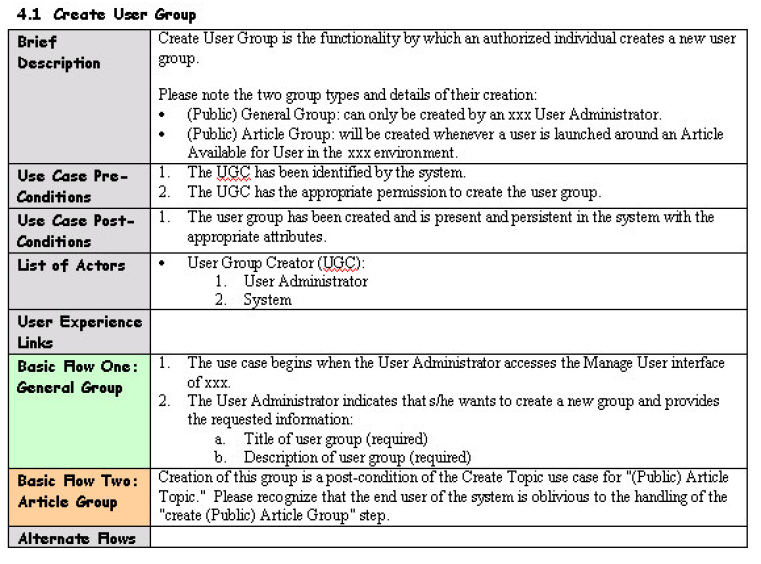
Tabel 3.5 Notasi pada *Sequence Diagram* (lanjutan)

|  |  |  |  |
| --- | --- | --- | --- |
| 4 | *Focus Control* | Notasi yang digunakan untuk menunjukkan bahwa sebuah *object* sedang aktif dan menjalankan tugasnya. |  |
| 5 | *Message* | Notasi yang digunakan untuk menunjukkan bahwa sebuahpesan sedang dikirim dari suatu *object* ke *object* yang lain. |  |

Sumber : Dennis (2012)

#### *Use Case Scenario*

Menggambarkan *Use Case* secara detail dengan segala skenario yang ditetapkan dan yang mungkin terjadi. *Use Case* skenario berguna saat nanti dalam proses pengujian. Berikut merupakan contoh dari *Use Case* Skenario:

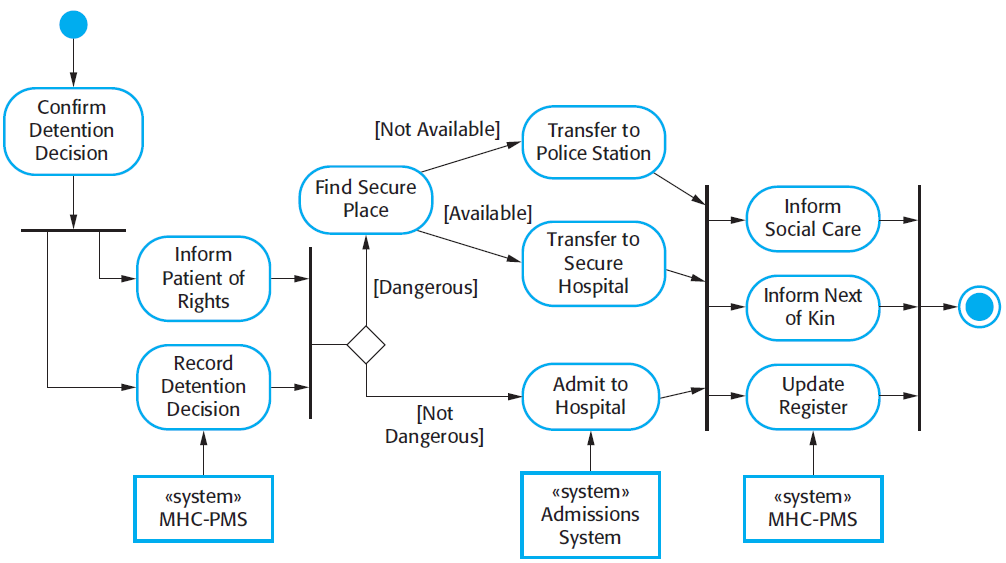


##### Gambar 3.7. *Use Case Scenario.*

Sumber : Bittner (2006)

#### *Activity Diagram*

Menggambarkan aktifitas-aktifitas, objek, *state*, transisi *state* dan *event*. Dengan kata lain *activity diagram* menunjukkan aktivitas yang terlibat dan yang dilaksanakan ketika menjalankan suatu proses atau pemrosesan data (Sommerville,2011). *Activity diagram* dimaksudkan untuk menunjukkan aktifitas yang terjadi pada suatu proses yang dijalankan di dalam sistem. Awal proses ditunjukan dengan lingkaran berisi warna dan akhir proses ditunjukkan dengan lingkaran yang berisi warna di dalam lingkaran lain. Kegiatan atau aktifitas yang harus dijalani ditandai dengan persegi bersudut bulat. Tanda panah menunjukan perpindahan aktifitas ke aktifitas yang lainnya.



##### Gambar 3.7. *Activity diagram*.

Sumber : Sommerville (2011)

## *Entity Relationship Diagram*

*Entity Relationship Diagram* (ERD) merupakan sebuah diagram yang menggambarkan tentang bagaiamana sebuah informasi dibuat , disimpan dan digunakan dalam sebuah sistem (Dennis, 2012). Dalam ERD kita dapat melihat informasi yang digunakan dalam sistem dan keterkaitannya dengan informasi lainnya. Informasi yang memiliki kesamaan ditempatkan dalam kotak yang disebut entitas. Antar entitas diberikan garis penghubung yang menggambarkan hubungan antar entitas dan diberikan simbol yang menggambarkan aturan yang berlaku antar entitas dalam sistem.

Tabel 3.6 Notasi pada *Entity Relationship Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| No | Tipe | Deskripsi | Simbol |
| 1 | *Entity* | Notasi yang digunakan untuk menggambarkan seseorang (*user*) atau tempat atau benda, memiliki *identifier* dan memiliki lebih dari 1 data. |  |
| 2 | *Attribute* | Notasi yang digunakan untuk menggambarkanbagian dari sebuah *entity*. |  |
| 3 | *Relation* | Notasi yang digunakan untuk menggambarkan hubungan antar *entity,* merupakan kata kerja, memiliki cardinality (1:1, 1:N, M:N), memiliki modal (*null, not null*). |  |

Sumber : Dennis (2012)

BAB 4 METODOLOGI

Metode penelitian ini akan membahas mengenai jual beli sayur *online* berbasis *android* menggunakan *webservice*. Tahapan metodologi penelitian yang digunakan pada laporan PKL ini adalah penentuan objek, studi literatur, analisa kasus, konsep dan desain sistem, implementasi sistem, pengujian dan analisis, serta penarikan kesimpulan.

Adapun diagram alir metodologi tersebut dijelaskan pada Gambar 4.1 berikut :

4.1 Penentuan Objek Penelitian

4.2 Studi Literatur

4.3 Analisa Kasus

4.4 Perancangan Konsep dan Model

4.5 Implementasi

4.6 Pengujian dan Analisis

4.7 Kesimpulan

##### Gambar 4.1 Diagram Alir Metodologi

## 4.1 Penentuan Objek Penelitian

Merupakan langkah awal yang bertujuan menentukan objek penentuan yang bisa didapat dari kegiatan PKL yang dilakukan di CV. Sarana Utama Solusindo yang kemudian bisa dijadikan suatu yang akan dikerjakan. Melakukan diskusi awal dengan *software developer* CV. Sarana Utama Solusindo yang bersangkutan, kemudian diberikan daftar objek apa saja yang dapat dijadikan sebagai objek untuk dikerjakan pada PKL ini.

## 4.2 Studi Literatur

Dalam tahap studi literatur, dilakukan studi pustaka dan literatur untuk mendapatkan informasi yang berhubungan dengan pengembangan sistem, sehingga informasi tersebut diharapkan dapat mempermudah dan membantu dalam melaksanakan pengembangan sistem. Literatur-literatur yang digunakan diperoleh dari buku, *paper* nasional maupun internasional dan dokumentasi internet.

## 4.3 Analisa Kasus

Analisa kasus ini digunakan untuk mengetahui informasi mengenai data sayur yang dibutuhkan melalui proses *input* data yang dilakukan oleh pihak penjual atau tengkulak sayur , gudang sayur akan digunakan untuk menyediakan sayur yang akan diberikan kepada tengkulak atau penjual sayur, pembeli yang akan membeli dan melihat sayur yang dijual ,dengan data tersebut kelompok kami mengembangkan pengolahan data tersebut ke dalam suatu aplikasi sayur *online*, serta didukung pengumpulan data secara tidak langsung melalui internet.

## 4.4 Perancangan Konsep dan Model

Proses perancangan konsep dan model sistem dibagi menjadi beberapa tahapan yaitu:

1. Menganalisa kebutuhan sistem dan menerjemahkan ke dalam fungsi fungsi pada aplikasi.
2. Menentukan objek aplikasi berdasarkan kebutuhan yang ada.
3. Pembuatan desain *interface*.

Pada tahap ini dilakukan proses perancangan sistem untuk mengimplementasikan ke dalam aplikasi. Adapun proses-proses yang dilakukan dalam tahap perancangan ini adalah analisis kebutuhan perangkat lunak dan perancangan sayur *online* berbasis *android* menggunakan web *service*.

## 4.5 Implementasi

Pada tahap ini dilakukan proses pengimplementasian rancangan sistem yang telah ditentukan pada tahap sebelumnya menjadi suatu sistem. Implementasi sistem dibuat dengan menggunakan *php framework laravel*. Kemudian untuk *user* interface menggunakan android studio dengan bahasa pemograman *java* dan *xml*.

## 4.6 Pengujian dan Analisis

Pengujian yang akan digunakan yaitu pengujian *Black Box*. Pengujian *Black Box* merupakan pengujian fungsionalitas sistem untuk mengetahui apakah sistem telah berjalan dengan benar sesuai dengan parameter yang telah ditentukan. Uji coba ini dilakukan oleh tim internal dan juga dilakukan oleh calon pengguna untuk mengetahui kelebihan dan kekurangan sistem.

## 4.7 Kesimpulan

Penarikan kesimpulan dilakukan setelah semua tahapan perancangan, implementasi, dan pengujian sistem aplikasi telah dilaksanakan. Kesimpulan diambil melalui hasil pengujian dan analisis terhadap sistem yang dibuat.