**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

# LEMBAR PENGESAHAN

SISTEM JUAL BELI SAYUR *ONLINE* BERBASIS *ANDROID*

MENGGUNAKAN *WEB SERVICE*

LAPORAN PRAKTIK KERJA LAPANG (PKL)

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 |
| Nigel Erico Pangestu | |  |
|  | |  |
|  |  |

# LEMBAR PENGESAHAN

SISTEM JUAL BELI SAYUR *ONLINE* BERBASIS *ANDROID*

MENGGUNAKAN *WEB SERVICE*

LAPORAN PRAKTIK KERJA LAPANG (PKL)

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 |
| Nigel Erico Pangestu |  |

Telah diperiksa dan disetujui oleh :

Pihak CV. SARANA UTAMA SOLUSINDO

**Menyetujui,**

**Pembimbing Praktik Kerja Lapang**

**Ardana Prakasita Devi, S.Kom**

# 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 Widhi Yahya, S.Kom., M.Sc. selaku dosen pembimbing yang telah menyediakan waktu untuk berdiskusi dan berbagai saran serta masukkannya.
7. Ardana Prakasita Devi, S.Kom 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 2014 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, 09 September 2018

Penulis

.

# DAFTAR ISI

[LEMBAR PENGESAHAN ii](#_Toc526098808)

[LEMBAR PENGESAHAN iii](#_Toc526098809)

[KATA PENGANTAR iv](#_Toc526098810)

[DAFTAR ISI vi](#_Toc526098811)

[DAFTAR TABEL ix](#_Toc526098812)

[DAFTAR GAMBAR xi](#_Toc526098813)

[DAFTAR LAMPIRAN xiii](#_Toc526098814)

[BAB 1 PENDAHULUAN 1](#_Toc526098815)

[1.1 Latar belakang 1](#_Toc526098816)

[1.2 Rumusan masalah 1](#_Toc526098817)

[1.3 Tujuan 1](#_Toc526098818)

[1.4 Manfaat 1](#_Toc526098819)

[1.5 Batasan masalah 1](#_Toc526098820)

[1.6 Sistematika pembahasan 2](#_Toc526098821)

[1.7 Pelaksanaan Praktek Kerja Lapang 3](#_Toc526098822)

[BAB 2 PROFIL PERUSAHAAN 4](#_Toc526098823)

[2.1 Sejarah dan Perkembangan CV Sarana Utama Solusindo (SASINDO) 4](#_Toc526098824)

[2.2 Visi, Misi dan Komitmen Perusahaan. 4](#_Toc526098825)

[2.2.1 Visi 4](#_Toc526098826)

[2.2.2 Misi 4](#_Toc526098827)

[2.2.3 Komitmen 5](#_Toc526098828)

[2.3 Struktur Organisasi CV SARANA UTAMA SOLUSINDO 5](#_Toc526098829)

[2.4 Ruang Lingkup CV SARANA UTAMA SOLUSINDO 5](#_Toc526098830)

[2.4.1 Solusi Bisnis TI (IT Business Solution) dan Konsultan IT (IT Consultant) 6](#_Toc526098831)

[2.4.2 Pendidikan dan Pelatihan (IT Training) 7](#_Toc526098832)

[BAB 3 LANDASAN KEPUSTAKAAN 8](#_Toc526098833)

[3.1 Sistem Penjualan. 8](#_Toc526098834)

[3.1.2 Sistem Jual Beli *Online*. 8](#_Toc526098835)

[3.2 Android 9](#_Toc526098836)

[3.2.1 Android Studio 10](#_Toc526098837)

[3.3 *Web Service* 11](#_Toc526098838)

[3.4 Laravel 12](#_Toc526098839)

[3.4.1 Model-View-Controller (MVC) 13](#_Toc526098840)

[3.5 *Unified Modeling Language (UML)* 13](#_Toc526098842)

[3.5.1 *Structured Diagram.* 14](#_Toc526098843)

[3.5.2 *Behavior Diagram*. 16](#_Toc526098844)

[BAB 4 METODOLOGI 20](#_Toc526098845)

[4.1 Penentuan Objek Penelitian 21](#_Toc526098846)

[4.2 Studi Literatur 21](#_Toc526098847)

[4.3 Analisa Kasus 22](#_Toc526098848)

[4.4 Perancangan Konsep dan Model 22](#_Toc526098849)

[4.5 Implementasi 22](#_Toc526098850)

[4.6 Pengujian dan Analisis 23](#_Toc526098851)

[4.7 Kesimpulan 23](#_Toc526098852)

[BAB 5 ANALISIS DAN PERANCANGAN 24](#_Toc526098853)

[5.1 Analisa Kebutuhan Sistem 25](#_Toc526098854)

[5.1.1 Deskripsi Umum Sistem 25](#_Toc526098855)

[5.1.2 Deskripsi Umum Perangkat Lunak 26](#_Toc526098856)

[5.1.3 Fungsi Perangkat Lunak 26](#_Toc526098857)

[5.1.4 Karakteristik Pengguna 27](#_Toc526098858)

[5.2 Analisa Kebutuhan Perangkat Lunak 28](#_Toc526098859)

[5.2.1 Kebutuhan Fungsional dan Non-Fungsional 28](#_Toc526098860)

[5.2.2 Pemodelan Kebutuhan 30](#_Toc526098861)

[5.3 Perancangan Perangkat Lunak 48](#_Toc526098862)

[5.3.1 Perancangan Algoritma 48](#_Toc526098863)

[5.3.2 Perancangan Antarmuka 53](#_Toc526098864)

[5.3.3.1 Perancangan Tampilan Halaman Login 54](#_Toc526098865)

[5.3.3.2 Perancangan Tampilan Halaman Register 55](#_Toc526098866)

[5.3.3.3 Perancangan Tampilan Halaman *Home User* 56](#_Toc526098867)

[5.3.3.4 Perancangan Tampilan Halaman Keranjang Belanja 57](#_Toc526098868)

[5.3.3.5 Perancangan Tampilan Halaman Transaksi 58](#_Toc526098869)

[5.3.3.6 Perancangan Tampilan Halaman Profil 59](#_Toc526098870)

[5.3.3.7 Perancangan Tampilan Gudang Sayur 60](#_Toc526098871)

[5.3.3.8 Perancangan Tampilan *Manage User* 61](#_Toc526098872)

[5.3.3.9 Perancangan Tampilan *Manage* Sayur 62](#_Toc526098873)

[5.3.3.10 Perancangan Tampilan *Manage* Transaksi 63](#_Toc526098874)

[BAB 6 IMPLEMENTASI 64](#_Toc526098875)

[6.1 Batasan Implementasi 65](#_Toc526098876)

[6.2 Spesifikasi Sistem 65](#_Toc526098877)

[6.2.1 Spesifikasi Perangkat Keras 65](#_Toc526098878)

[6.2.2 Spesifikasi Perangkat Lunak 66](#_Toc526098879)

[6.3 Implementasi Algoritma 66](#_Toc526098880)

[6.3.1 Implementasi Algoritma Login 67](#_Toc526098881)

[6.3.2 Implementasi Algoritma Tambah Data Sayur 68](#_Toc526098882)

[6.3.3 Implementasi Algoritma Edit Data Sayur 69](#_Toc526098883)

[6.4 Antarmuka 70](#_Toc526098884)

[BAB 7 PENGUJIAN DAN ANALISIS 80](#_Toc526098885)

[7.1 Pengujian 80](#_Toc526098886)

[7.1.1 Pengujian Validasi 80](#_Toc526098887)

[7.2 Analisis 98](#_Toc526098888)

[7.2.1 Analisis Hasil Pengujian Validasi 98](#_Toc526098889)

[BAB 8 PENUTUP 100](#_Toc526098890)

[8.1 Kesimpulan 100](#_Toc526098891)

[8.2 Saran 101](#_Toc526098892)

[DAFTAR PUSTAKA 102](#_Toc526098893)

# DAFTAR TABEL

[Tabel 1.1 Kegiatan 3](#_Toc526098413)

[Tabel 3.1 Notasi pada Structured diagram. 16](#_Toc526098414)

[Tabel 3.2 Notasi pada *Use Case Diagram* 17](#_Toc526098415)

[Tabel 3.3 Notasi pada *Sequence Diagram* 19](#_Toc526098416)

[Tabel 5.1 Karakter Pengguna 27](#_Toc526098417)

[Tabel 5.2 Kebutuhan Fungsional User 29](#_Toc526098418)

[Tabel 5.4 Skenario *Use Case* *Login* 32](#_Toc526098419)

[Tabel 5.5 Skenario *Use Case* Register 32](#_Toc526098420)

[Tabel 5.6 Skenario *Use Case* Lihat daftar sayur pada home 32](#_Toc526098421)

[Tabel 5.7 Skenario *Use Case* Melihat data sayur 33](#_Toc526098422)

[Tabel 5.8 Skenario *Use Case* melakukan pembelian sayur 33](#_Toc526098423)

[Tabel 5.9 Skenario *Use Case* Search sayur 33](#_Toc526098424)

[Tabel 5.10 Skenario *Use Case* Lihat Keranjang belanja 33](#_Toc526098425)

[Tabel 5.11 Skenario *Use Case* Edit jumlah sayur yang dibeli 33](#_Toc526098426)

[Tabel 5.12 Skenario *Use Case* Hapus daftar pembelian 34](#_Toc526098427)

[Tabel 5.13 Skenario *Use Case* pembayaran 34](#_Toc526098428)

[Tabel 5.14 Skenario *Use Case* Melengkapi Data pengiriman 34](#_Toc526098429)

[Tabel 5.15 Skenario *Use Case* Chatting 34](#_Toc526098430)

[Tabel 5.16 Skenario *Use Case* Transaksi 35](#_Toc526098431)

[Tabel 5.17 Skenario *Use Case* bukti pembayaran 35](#_Toc526098432)

[Tabel 5.18 Skenario *Use Case* Lihat sayur yang telah dikirm 35](#_Toc526098433)

[Tabel 5.19 Skenario *Use Case* lihat profile 35](#_Toc526098434)

[Tabel 5.20 Skenario *Use Case* Edit Profil 36](#_Toc526098435)

[Tabel 5.21 Skenario *Use Case* Tambah Data sayur 36](#_Toc526098436)

[Tabel 5.22 Skenario *Use Case* Lihat sayur pada admin dashboard 36](#_Toc526098437)

[Tabel 5.23 Skenario *Use Case* lihat data sayur 36](#_Toc526098438)

[Tabel 5.24 Skenario *Use Case* Hapus Data sayur 37](#_Toc526098439)

[Tabel 5.25 Skenario *Use Case* Edit sayur 37](#_Toc526098440)

[Tabel 5.26 Use Case Lihat daftar sayur pada Home 37](#_Toc526098441)

[Tabel 5.27 Use Case Melihat data sayur 37](#_Toc526098442)

[Tabel 5.28 Use Case Melihat daftar user 38](#_Toc526098443)

[Tabel 5.29 Use Case Melihat data user 38](#_Toc526098444)

[Tabel 5.30 Use Case Hapus data user 38](#_Toc526098445)

[Tabel 5.31 Use Case Daftar Pembelian 38](#_Toc526098446)

[Tabel 5.32 Use Case Konfirmasi Pembayaran 39](#_Toc526098447)

[Tabel 6.1 Spesifikasi Perangkat Keras 67](#_Toc526098448)

[Tabel 6.2 Spesifikasi Perangkat Lunak 67](#_Toc526098449)

[Tabel 7.1 Tabel Pengujian Validasi 82](#_Toc526098450)

# DAFTAR GAMBAR

[Gambar 2.1 Struktur Organisasi 5](#_Toc525655798)

[Gambar 3.1. Illustrasi Android Stack 10](#_Toc525655799)

[Gambar 3.2. Class Diagram 14](#_Toc525655800)

[Gambar 3.3. Package diagram. 14](#_Toc525655801)

[Gambar 3.4. Use Case Diagram. 15](#_Toc525655802)

[Gambar 3.5. Activity Diagram. 16](#_Toc525655803)

[Gambar 3.6. Sequence Diagram. 16](#_Toc525655804)

[Gambar 3.7. Use Case Scenario. 17](#_Toc525655805)

[Gambar 4.1 Diagram Alir Metodologi 18](#_Toc525655806)

[Gambar 5.1 Diagram Alir Analisis dan Perancangan 21](#_Toc525655807)

[Gambar 5.2 Deskripsi Sistem 22](#_Toc525655808)

[Gambar 5.3 Use Case Diagram 28](#_Toc525655809)

[Gambar 5.4 Class Diagram 36](#_Toc525655810)

[Gambar 5.5 Activity Diagram “hapus data user” 37](#_Toc525655811)

[Gambar 5.6 Activity Diagram “search data user” 37](#_Toc525655812)

[Gambar 5.7 Activity Diagram “perbarui data sayur” 38](#_Toc525655813)

[Gambar 5.8 Activity Diagram “perbarui data sayur” 38](#_Toc525655814)

[Gambar 5.9 Activity Diagram “search sayur” 38](#_Toc525655815)

[Gambar 5.10 Activity Diagram “menambah sayur” 39](#_Toc525655816)

[Gambar 5.11 Activity Diagram “pesan sayur” 39](#_Toc525655817)

[Gambar 5.12 Activity Diagram “edit daftar pembelian” 40](#_Toc525655818)

[Gambar 5.13 Activity Diagram “hapus daftar pembelian” 40](#_Toc525655819)

[Gambar 5.14 Sequence Diagram “menambah sayur” 41](#_Toc525655820)

[Gambar 5.15 Sequence Diagram “perbarui data sayur” 42](#_Toc525655821)

[Gambar 5.16 Sequence Diagram “hapus sayur” 42](#_Toc525655822)

[Gambar 5.17 Sequence Diagram “hapus user” 43](#_Toc525655823)

[Gambar 5.18 Sequence Diagram “search sayur” 44](#_Toc525655824)

[Gambar 5.19 Sequence Diagram “search user” 44](#_Toc525655825)

[Gambar 5.20 Sequence Diagram “keranjang belanja” 45](#_Toc525655826)

[Gambar 5.21 Sequence Diagram “pesan sayur” 45](#_Toc525655827)

[Gambar 5.22 Diagram Alir Proses Login Sebagai Admin 46](#_Toc525655828)

[Gambar 5.23 Diagram Alir Proses Tambah Data Sayur 47](#_Toc525655829)

[Gambar 5.24 Diagram Alir Proses Lihat Data Sayur 48](#_Toc525655830)

[Gambar 5.25 Diagram Alir Proses Perbarui Data Sayur 49](#_Toc525655831)

[Gambar 5.26 Diagram Alir Proses Hapus Data Sayur 50](#_Toc525655832)

[Gambar 5.27 Perancangan Tampilan Halaman Login 51](#_Toc525655833)

[Gambar 5.28 Perancangan Tampilan Halaman Register 52](#_Toc525655834)

[Gambar 5.29 Perancangan Tampilan Halaman *Home User* 53](#_Toc525655835)

[Gambar 5.30 Perancangan Tampilan Halaman Keranjang Belanja 54](#_Toc525655836)

[Gambar 5.31 Perancangan Tampilan Halaman Transaksi 55](#_Toc525655837)

[Gambar 5.32 Perancangan Tampilan Halaman Profil 56](#_Toc525655838)

[Gambar 5.33 Perancangan Tampilan Halaman Gudang Sayur 57](#_Toc525655839)

[Gambar 5.34 Perancangan Tampilan Halaman Manage User 58](#_Toc525655840)

[Gambar 5.35 Perancangan Tampilan Halaman Manage Sayur 59](#_Toc525655841)

[Gambar 5.36 Perancangan Tampilan Halaman Manage Transaksi 60](#_Toc525655842)

[Gambar 6.1 Diagram Alir Implementasi 61](#_Toc525655843)

[Gambar 6.2 Kode Program Login 64](#_Toc525655844)

[Gambar 6.3 Kode Program Tambah Sayur 65](#_Toc525655845)

[Gambar 6.4 Kode Program Edit Data Sayur 66](#_Toc525655846)

[Gambar 6.5 Implementasi Halaman Login 67](#_Toc525655847)

[Gambar 6.6 Implementasi Halaman Register 68](#_Toc525655848)

[Gambar 6.7 Implementasi Halaman Home User 69](#_Toc525655849)

[Gambar 6.8 Implementasi Halaman Keranjang Belanja 70](#_Toc525655850)

[Gambar 6.9 Implementasi Halaman Transaksi 71](#_Toc525655851)

[Gambar 6.10 Implementasi Halaman Profil 72](#_Toc525655852)

[Gambar 6.11 Implementasi Halaman Gudang Sayur 73](#_Toc525655853)

[Gambar 6.12 Implementasi Halaman Manage User 74](#_Toc525655854)

[Gambar 6.13 Implementasi Halaman Manage Sayur 75](#_Toc525655855)

[Gambar 6.13 Implementasi Halaman Manage Transaksi 76](#_Toc525655856)

[Gambar 7.1 Diagram Alir Pengujian dan Analisis 77](#_Toc525655857)

# DAFTAR LAMPIRAN

1. PENDAHULUAN

## 1.1 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, petani sayur tentunya belum mempunyai suatu media yang menyediakan informasi tentang apa saja yang mereka perjual belikan secara digital. Tentunya kita 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 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 | April 2018 | | | | Mei 2018 | | | | Juni 2018 | | | | Juli - Agustus 2018 | | | |
| I | II | III | IV | I | II | III | IV | I | II | III | IV | I | II | III | IV |
| 1 | Pengenalan CV SASINDO secara garis besar dan ruang lingkup kerja |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Penentuan objek yang akan dibuat |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Pembagian tugas pengerjaan baik isi laporan dan kode sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Analisis kebutuhan user |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Implementasi Program |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Pembuatan Laporan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

* **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

1. 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



IT Engineer



Customer Service



Administration

& Marketing



**Power House:**

nd Floor,

2

49

th

Anggrek

Garuda Street

Malang, 65141

http://sasindo.id

cs.sasindo@gmail.com

##### 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*)

1. 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 code 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)

|  |  |  |  |
| --- | --- | --- | --- |
| 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*. |  |
| 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. |  |

Tabel 3.2 Notasi pada *Use Case Diagram*

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 3.3 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*. |  |
| 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 Scenario:



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

Sumber :

1. METODOLOGI

Metode penelitian ini akan membahas mengenai jual beli sayur *online* bersabasi 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 :



4.2

Studi Literatur



4.3

Analisa Kasus



4.4

Perancangan Konsep

dan Model



4.5

Implementasi



4.6

Pengujian

dan Analisis



4.7

Kesimpulan



4.1

Penentuan

O

bjek

P

enelitian

##### 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.

1. ANALISIS DAN PERANCANGAN

Perancangan aplikasi dalam pembuatan sistem sayur *online* diawali dengan analisis. Analisis dilakukan untuk mengetahui kebutuhan-kebutuhan sistem yang dijelaskan pada analisa kebutuhan sistem dan kebutuhan perangkat lunak, kemudian dilakukan perancangan sistem dari hasil analisa berupa perancangan perangkat lunak sepeti pada Gambar 5.1.



Analisis dan

Perancangan



Analisa

5.2

Kebutuhan

Perangkat Lunak



Kebutuhan

5.2.2

Fungsional dan Non



Pemodelan

5.2.2

Kebutuhan



Perancangan

5.3.3

Antarmuka User



5.3.1

Perancangan

Algoritma



5.3

Perancangan

Perangkat Lunak



5.1

Analisa

Kebutuhan Sistem



Deskripsi Umum

5.1.1

Sistem



5.1.2

Deskripsi Umum

Perangkat Lunak



Fungsi Perangkat

5.1.3

Lunak



Karakteristik

5.1.4



##### Gambar 5.1 Diagram Alir Analisis dan Perancangan

## 5.1 Analisa Kebutuhan Sistem

Pada tahap analisa kebutuhan sistem bertujuan untuk memodelkan informasi yang akan digunakan pada tahap perancangan. Analisa kebutuhan sistem yang diperlukan meliputi deskripsi umum sistem, deskripsi umum perangkat lunak, fungsi perangkat lunak, karakteristik pengguna, dan lingkungan operasi.

### 5.1.1 Deskripsi Umum Sistem

User dapat melakukan installasi aplikasi pada perangkat smartphone. Pada perangkat user dijalankan aplikasi sayur online , untuk menghubungkan sistem aplikasi sayur online ke web service dibutuhkan koneksi, melalui url web service yang menggandung kumpulan informasi, perintah, konfigurasi atau sintaks yang berguna membangun sebuah fungsi-fungsi tertentu dari aplikasi. Database digunakan sebagai media penyimpanan data dari aplikasi sayur online. Berikut adalah deskripsi item sistem dari aplikasi sayur online yang ditunjukan pada gambar 5.2

Admin



User

Sayur Online

Web Service

Database

##### Gambar 5.2 Deskripsi Sistem

### 5.1.2 Deskripsi Umum Perangkat Lunak

Perangkat lunak sayur online dibuat berbasis android menggunakan java dan xml yang dihubungkan ke web service. Sistem dapat berinteraksi dengan web service untuk menjalankan perintah user dan admin, seperti pada fitur lihat data dimana admin dapat melihat data sayur , data user dan transaksi yang dilakukan oleh user. Data sayur ditampilkan pada menu beranda admin. Admin dapat melakukan edit data melalui fitur edit yang disediakan pada masing-masing data sayur , data user , data transaksi, dan profil admin. Untuk melakukan hapus data, user dapat menghapus data dari sistem pada fitur hapus data yang ditampilkan. Admin juga dapat melakukan tambah sayur yang akan dijual dari gudang. Untuk fitur yang digunakan oleh user adalah melakukan edit profile , pembelian sayur , melihat data sayur , menyimpan barang belanja ke *cart ,* melihat detail transaksi dan ada fitur bantuan jika user bingung dengan aplikasi sayur online.

Untuk mengelola seluruh informasi pada sistem, masuk ke dalam sistem melalui fitur login dengan menginputkan *username* serta *password* adminatau user dan keluar sistem melalui fiturlogout.

### 5.1.3 Fungsi Perangkat Lunak

Fungsi aplikasi sayur online ini 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.

### 5.1.4 Karakteristik Pengguna

Karakteristik pengguna digunakan untuk melihat fitur apa saja yang terdapat didalam aplikasi sayur online. Berikut adalah deskripsi fitur sistem dari aplikasi sayur online yang ditunjukan pada tabel 5.1.

Tabel 5.1 Karakter Pengguna

|  |  |  |
| --- | --- | --- |
| **No** | **Identifikasi Pengguna** | **Karakteristik** |
| 1 | User | Melakukan login  Melakukan register  Melihat data sayur  Menyimpang barang belanja ke cart  Membeli sayur  Melihat fitur bantuan  Melihat status transaksi pembelian  Mengubah profile  Melakukan logout |
| 2 | Admin | Melakukan Login  Melihat data sayur  Meng-edit data sayur  Menghapus data sayur  Menambah data sayur dari gudang  Melihat data user  Menghapus data user  Melihat status transaksi pembelian  Mengubah status transaksi pembelian  Mengubah profil  Melakukan logout |

## 5.2 Analisa Kebutuhan Perangkat Lunak

Pada tahap analisa kebutuhan perangkat lunak bertujuan untuk memodelkan kebutuhan fungsional dan non-fungsional dari perangkat lunak yang akan digunakan, serta menjelaskan mengenai kebutuhan antarmuka dan pemodelan dalam bentuk diagram *use case*, scenario *use case*, *sequence diagram* dan *class diagram*.

### 5.2.1 Kebutuhan Fungsional dan Non-Fungsional

1. **Kebutuhan Fungsional**

Kebutuhan fungsional adalah layanan sistem yang harus disediakan, bagaimana sistem bereaksi pada input tertentu dan bagaimana perilaku sistem pada situasi tertentu. Berikut adalah kebutuhan fungsional dari aplikasi sayur online yang ditunjukan pada tabel 5.2.

Tabel 5.2 Kebutuhan Fungsional

|  |  |  |  |
| --- | --- | --- | --- |
| No | Aktor | Nama Fungsi | Deskripsi |
| 1 | User/Admin | Login | Untuk menyeleksi pengguna yang memiliki autorisasi agar dapat mengakses ke dalam sistem dengan memasukkan username  dan password |
| 2 | User/Admin | Lihat daftar sayur | User/admin dapat melihat tampilan data sayur yang dijual dalam bentuk *grid layout*. |
| 3 | Admin | Lihat data user | Admin dapat melihat tampilan data User yang telah melakukan register |
| 4 | User /Admin | Lihat detail transaksi pembelian | Admin/user dapat melihat tampilan detail transaksi pembelian. |
| 5 | Admin | Mengubah status transaksi | Admin dapat melakukan merubah atau edit status transakasi  Pembelian jika barang sudah dikirim ataupun diterima |
| 6 | User/Admin | Lihat history transaksi | User/Admin dapat melihar daftar transaksi di dalam menu history, |
| 7 | User/Admin | Search sayur | User/Admin dapat mencari sayur yang diinginkan di halaman home. |
| 8 | Admin | Edit data sayur | Admin dapat melakukan meruba atau edit pada data sayur yang tersimpan dalam |
| 9 | Admin | Hapus data sayur | Admin dapat melakukan hapus terhadap data sayur yang  tersimpan dalam sistem |
| 10 | Admin | Menambah data sayur dari gudang | Admin dapat melakukan penambahan sayur yang akan dijual melalui gudang. |
| 11 | Admin | Hapus Data User | Admin dapat melakukan hapus terhadap data user yang  tersimpan dalam sistem |
| 12 | User /Admin | Edit profil | User/admin dapat melakukan edit profile untuk kelengkapan data dari pengguna user ataupun admin |
| 13 | User/Admin | Lihat profil | User/Admin dapat melihat profil |
| 14 | User | Register | User dapat melakukan pendaftaran pada aplikasi sayur online untuk membeli sayur yang tersedia. |
| 15 | User | Order sayur | User dapat memilih dan melakukan penyimpanan data sayur yang akan dibeli ke dalam fitur cart. |
| 16 | User | Melengkapi data transaksi | User dapat melakukan pembelian sayur dengan melengkapi data transaksi pada keranjang. |
| 17 | User | Edit jumlah sayur pada keranjang | User dapat mengubah jumlah sayur yang terdapat di dalam keranjang. |
| 18 | User | Hapus daftar pembelian | User dapat menghapus daftar sayur yang terdapat di dalam keranjang. |
| 19 | User | Melihat keranjang belanja. | User dapat melihat sayur yang dibeli di dalam keranjang |
| 20 | Admin/User | Logout | Admin/User keluar dari sistem |

1. **Kebutuhan Non-Fungsional**

Kebutuhan non-fungsional adalah batasan atau fungsi yang ditawarkan sistem seperti batasan waktu, batasan pengembangan proses dan standarisasi Berikut adalah kebutuhan non-fungsional dari aplikasi sayur online yang ditunjukan pada tabel 5.3.

**Tabel 5.3 Kebutuhan Non-Fungsional Sistem**

|  |  |  |
| --- | --- | --- |
| No | Kebutuhan | Deskripsi |
| 1 | Availability | Dengan menguji ketersedian sistem untuk dapat diakses dengan batas minimal waktu operasi 20 jam per hari tanpa downtime |
| 2 | Portability | Menguji bahwa aplikasi dapat dijalankan pada smartphone android |

### 5.2.2 Pemodelan Kebutuhan

#### 5.2.2.1 Use Case Diagram

Diagram *use case* digunakan untuk menggambarkan kebutuhan-kebutuhan dan fungsionalitas dari sistem serta digunakan untuk menunjukan aksi-aksi yang dilakukan oleh aktor terhadap sistem. Berikut adalah use case diagram dari aplikasi sayur online yang ditunjukan pada gambar 5.4.



##### Gambar 5.3 Use Case Diagram

#### 5.2.2.2 Use Case Skenario

Untuk menjelaskan diagram *use case* secara terperinci dalam skenario *use case*. Penggunaan skenario *use case* ini bertujuan untuk mendapatkan deskripsi secara global mengenai *use case*, kondisi awal dan akhir yang harus dipenuhi oleh *use case* setelah fungsionalitas selesai dijalankan. Dalam skenario ini akan diulas bagaimana tanggapan sistem terhadap aksi yang dilakukan oleh aktor. Berikut adalah skenario use case dari aplikasi sayur online yang ditunjukan pada tabel 5.4 – 5.32 .

Tabel 5.4 – 5.32 merupakan skenario *use case* pada aplikasi Sayur Online.

1. Guest menekan tombol sign in
2. Guest mengisikan data-data (no telp/email dan password) pada field yang telah disediakan

Guest menekan tombol sign in

Tabel 5.4 Skenario *Use Case* *Login*

|  |  |
| --- | --- |
| Use case *Login.* | |
| Tujuan | Masuk ke dalam sistem. |
| Aktor | Guest. |
| Kondisi awal | Halaman *Sign in.* |
| Skenario Utama | 1. Guest mengisi data pada kolom yang kosong, 2. Data yang dimasukkan tervalidasi 3. Menuju halaman home. |
| Skenario Alternatif | 1. Jika terdapat kolom kosong atau belum diisi sistem akan meminta melengkapi kolom sign in. |
| Kondisi akhir. | Menampilkan halaman home. |

Tabel 5.5 Skenario *Use Case* Register

|  |  |
| --- | --- |
| Use case *Register.* | |
| Tujuan | Mendaftar ke dalam sistem. |
| Aktor | Guest. |
| Kondisi awal | Halaman *Login.* |
| Skenario Utama | 1. Menekan tombol *sign up.* 2. Guest mengisi kolom dengan lengkap dan format yang benar. 3. Guest menekan tombol sign up. 4. Guest terdaftar dan dapat *login* pada halaman *sign in*. |
| Skenario Alternatif | 1. Jika terdapat kolom kosong atau belum diisi sistem akan meminta melengkapi kolom sign in. 2. Jika Mengisi dengan format yang salah maka muncul pesan “format salah” |
| Kondisi akhir. | Guest terdaftar dalam sistem. |

Tabel 5.6 Skenario *Use Case* Lihat daftar sayur pada home

|  |  |
| --- | --- |
| Use case melihat daftar sayur*.* | |
| Tujuan | Melihat daftar sayur. |
| Aktor | User/Admin. |
| Kondisi awal | Halaman *Home* |
| Skenario Utama | 1. Guest/User Melihat daftar sayur pada home. |
| Skenario Alternatif | - |
| Kondisi akhir. | Guest/User Melihat daftar sayur pada home. |

Tabel 5.7 Skenario *Use Case* *Order* sayur

|  |  |
| --- | --- |
| Use case *Order* sayur*.* | |
| Tujuan | *Order* Sayur |
| Aktor | User. |
| Kondisi awal | Halaman *Home* |
| Skenario Utama | 1. Guest/User Menekan tombol “Add” pada sayur yang ingin dibeli lalu muncul keranjang pada bagian bawah halaman. 2. Guest/User dapat mengurangi dan menambahkan sayur yang ingin dibeli. |
| Skenario Alternatif | - |
| Kondisi akhir. | Sayur yang ditambahkan muncul pada keranjang. |

Tabel 5.8 Skenario *Use Case* *Search* sayur

|  |  |
| --- | --- |
| Use case *Search* sayur*.* | |
| Tujuan | *Search* Sayur |
| Aktor | Guest/User/Admin. |
| Kondisi awal | Halaman *Home* |
| Skenario Utama | 1. Guest/User/Admin Mengisikan nama sayur yang ingin dicari pada kolom *Search*. |
| Skenario Alternatif | - |
| Kondisi akhir. | Menampilkan sayur yang dicari. |

Tabel 5.9 Skenario *Use Case* Lihat Keranjang belanja

|  |  |
| --- | --- |
| Use case lihat keranjang belanja*.* | |
| Tujuan | Melihat item pada keranjang belanja. |
| Aktor | Guest/User. |
| Kondisi awal | Halaman *Home* |
| Skenario Utama | 1. Guest/User Menekan tombol “Checkout” pada bottomsheet. |
| Skenario Alternatif | - |
| Kondisi akhir. | Menampilkan keranjang belanja. |

Tabel 5.10 Skenario *Use Case* Edit jumlah sayur pada keranjang.

|  |  |
| --- | --- |
| Use case *edit* jumlah sayur pada keranjang*.* | |
| Tujuan | Mengubah jumlah sayur pada keranjang belanja. |
| Aktor | User. |
| Kondisi awal | Halaman *Checkout.* |
| Skenario Utama | 1. Menekan tombol “+” dan “-“ pada sayur yang diinginkan untuk mengubah jumlah sayur. |
| Skenario Alternatif | - |
| Kondisi akhir. | Jumlah sayur berubah. |

Tabel 5.11 Skenario *Use Case* Hapus daftar pembelian

|  |  |
| --- | --- |
| Use case hapus daftar pembelian*.* | |
| Tujuan | Menghapus daftar pembelian*.* |
| Aktor | User. |
| Kondisi awal | Halaman *Checkout.* |
| Skenario Utama | 1. Menekan tombol hapus atau tombol tong sampah pada sayur yang diinginkan dihapus. |
| Skenario Alternatif | - |
| Kondisi akhir. | Daftar pembelian terhapus. |

Tabel 5.12 Skenario *Use Case* Melengkapi Data pengiriman

|  |  |
| --- | --- |
| Use case melengkapi data pengiriman | |
| Tujuan | Melengkapi data pengiriman. |
| Aktor | User. |
| Kondisi awal | Halaman *Checkout.* |
| Skenario Utama | 1. User mengisi data tanggal pengiriman dan waktu pengiriman. 2. User mengisi alamat yang ingin dituju. 3. Jika sudah lengkap user dapat menekan tombol “Checkout” untuk melakukan transaksi. |
| Skenario Alternatif | - |
| Kondisi akhir. | Transaksi berhasil dan kembali ke halaman *home*. |

Tabel 5.13 Skenario *Use Case* Lihat *History* transaksi.

|  |  |
| --- | --- |
| Use case Lihat *History* transaksi. | |
| Tujuan | Melihat *history* transaksi. |
| Aktor | User/Admin. |
| Kondisi awal | Halaman *Home.* |
| Skenario Utama | 1. User/Admin Menekan menu *history*. |
| Skenario Alternatif | - |
| Kondisi akhir. | Menampilkan halaman *history.* |

Tabel 5.14 Skenario *Use Case* Melihat detail transaksi.

|  |  |
| --- | --- |
| Use case Lihat detail transaksi. | |
| Tujuan | Melihat detail transaksi. |
| Aktor | User/Admin. |
| Kondisi awal | Halaman *History.* |
| Skenario Utama | 1. User/Admin Menekan daftar transaksi pada halaman *history*. |
| Skenario Alternatif | - |
| Kondisi akhir. | Menampilkan detail transaksi. |

Tabel 5.15 Skenario *Use Case* lihat profil

|  |  |
| --- | --- |
| Use case Lihat profil. | |
| Tujuan | Melihat profil user. |
| Aktor | User/Admin. |
| Kondisi awal | Halaman *Home.* |
| Skenario Utama | 1. User/Admin Menekan menu profil. |
| Skenario Alternatif | - |
| Kondisi akhir. | Menampilkan halaman profil. |

Tabel 5.16 Skenario *Use Case* *Edit* Profil

|  |  |
| --- | --- |
| Use case *edit* profil. | |
| Tujuan | Melakukan edit profil. |
| Aktor | User/Admin. |
| Kondisi awal | Halaman Profil*.* |
| Skenario Utama | 1. User menekan icon camera untuk mengganti foto user 2. User dapat langsung mengedit nama, email, dan no telp pada field nama, email, dan no telp 3. User menekan tombol simpan |
| Skenario Alternatif | - |
| Kondisi akhir. | Profil berubah. |

Tabel 5.17 Skenario *Use Case* *Logout*.

|  |  |
| --- | --- |
| Use case *logout*. | |
| Tujuan | Keluar sistem. |
| Aktor | User/Admin. |
| Kondisi awal | Halaman Profil*.* |
| Skenario Utama | 1. User tombol logout. |
| Skenario Alternatif | - |
| Kondisi akhir. | Keluar dari sistem. |

Tabel 5.18 Skenario *Use Case* Menambah sayur.

|  |  |
| --- | --- |
| Use case *logout*. | |
| Tujuan | Keluar sistem. |
| Aktor | Admin. |
| Kondisi awal | Halaman Profil*.* |
| Skenario Utama | 1. Admin menekan tombol menu gudang. 2. Admin memilih sayur yang akan ditambahkan dengan menekan tombol “Add” 3. Muncul pesan “Apakah anda yakin ingin menambahkan sayur?” 4. JIka admin memilih “Iya” maka sayur ditambahkan ke halaman home. |
| Skenario Alternatif | - |
| Kondisi akhir. | Sayur ditambahkan ke halaman *home*. |

Tabel 5.19 Skenario *Use Case* Hapus Data sayur

|  |  |
| --- | --- |
| Use case hapus data sayur. | |
| Tujuan | Menghapus data sayur. |
| Aktor | Admin. |
| Kondisi awal | Halaman Home*.* |
| Skenario Utama | 1. Admin menekan tombol lihat pada sayur. 2. Admin menekan tombol hapus. |
| Skenario Alternatif | - |
| Kondisi akhir. | Sayur berhasil dihapus. |

Tabel 5.20 Skenario *Use Case* *Edit* sayur

|  |  |
| --- | --- |
| Use case *Edit* sayur. | |
| Tujuan | Melakukan edit data sayur. |
| Aktor | Admin. |
| Kondisi awal | Halaman Home*.* |
| Skenario Utama | 1. Admin menekan tombol lihat pada sayur. 2. Admin menekan tombol edit. 3. Mengisi data sayur. |
| Skenario Alternatif | - |
| Kondisi akhir. | Sayur berhasil di-*edit* |

Tabel 5.21 Skenario *Use Case* Melihat daftar user

|  |  |
| --- | --- |
| Use case Melihat daftar user. | |
| Tujuan | Melihat data user. |
| Aktor | Admin. |
| Kondisi awal | Halaman Home*.* |
| Skenario Utama | 1. Admin menekan tombol daftar user pada menu. |
| Skenario Alternatif | - |
| Kondisi akhir. | Menampilkan halaman daftar user. |

Tabel 5.22 Skenario *Use Case* Hapus data user

|  |  |
| --- | --- |
| Use case Hapus data user. | |
| Tujuan | Menghapus data user. |
| Aktor | Admin. |
| Kondisi awal | Halaman daftar user*.* |
| Skenario Utama | 1. Admin menekan tombol hapus pada daftar user. |
| Skenario Alternatif | - |
| Kondisi akhir. | Data user berhasil dihapus. |

Tabel 5.23 Skenario *Use Case* Mengubah status transaksi.

|  |  |
| --- | --- |
| Use case mengubah status transaksi. | |
| Tujuan | Mengubah status transaksi. |
| Aktor | Admin. |
| Kondisi awal | Halaman *history.* |
| Skenario Utama | 1. Admin memilih daftar transaksi yang ingin diubah. 2. Memilih status transaksi yang tersedia. |
| Skenario Alternatif | - |
| Kondisi akhir. | Status transaksi berhasil berubah. |

#### 5.2.2.3 Class Diagram

Class *diagram* menggambarkan struktur dan deskripsi class, package, dan

objek beserta hubungan satu sama lain seperti komposisi, asosiasi, pewarisan dan lain-lain yang menyusun sebuah perangkat lunak. *Class diagram* dirancang dari hasil analisis terhadap *use case* yang dimodelkan sebelumya. Berikut Gambar 5.4 merupakan class diagram aplikasi Sayur Online :

****

##### Gambar 5.4 Class Diagram

#### 5.2.2.4 Activity Diagram

Activity diagram merupakan alur kerja atau kegiatan dari sebuah sistem atau menu yang ada pada pernagkat lunak. Activity diagram juga digunakan untuk mendefinisikan urutan atau pengelompokan tampilan dari sistem/user interface dimana setiap aktivitas dianggap memiliki sebuah rancang antar muka tampilan serta rancang menu yang ditampilkan pada perangkat lunak. Gambar 5.5 merupakan activity diagram dari hapus user :



##### Gambar 5.5 Activity Diagram “hapus data user”

*Activity Diagram search user* merupakan alur kerja program untuk mencari dengan spesifik data user yang terdaftar. Berikut Gambar 5.6 activity diagram dari search user :



##### Gambar 5.6 Activity Diagram “search data user”

*Activity Diagram perbarui data sayur* merupakan alur kerja program untuk merubah/meng-*edit* data sayur yang dijual. Berikut Gambar 5.7 activity diagram dari perbarui data sayur :



##### Gambar 5.7 Activity Diagram “perbarui data sayur”

*Activity Diagram hapus sayur* merupakan alur kerja program untuk menghapus data sayur yang dijual. Berikut Gambar 5.8 activity diagram dari hapus sayur :



##### Gambar 5.8 Activity Diagram “hapus data sayur”

*Activity Diagram search* sayurmerupakan alur kerja program untuk mencari dengan spesifik data sayur yang dijual. Berikut Gambar 5.9 activity diagram dari search sayur :



##### Gambar 5.9 Activity Diagram “search sayur”

*Activity Diagram* menambahsayurmerupakan alur kerja program untuk menambah data sayur dari gudang *webservice*. Berikut Gambar 5.10 activity diagram dari menambah sayur :

****

##### Gambar 5.10 Activity Diagram “menambah sayur”

*Activity Diagram* pesansayurmerupakan alur kerja program untuk user/pengguna untuk memesan sayur yang telah dijual di aplikasi sayur online. Berikut Gambar 5.11 activity diagram dari pesan sayur :

##### Gambar 5.11 Activity Diagram “pesan sayur”

*Activity Diagram* edit daftar pembelianmerupakan alur kerja program untuk user/pengguna untuk mengubah menambah, mengurangi jumlah sayur yang akan dibeli yang tersimpan di keranjang belanja. Berikut Gambar 5.12 activity edit daftar pembelian :



##### Gambar 5.12 Activity Diagram “edit daftar pembelian”

*Activity Diagram* hapus daftar pembelianmerupakan alur kerja program untuk menghapus daftar sayur yang tersimpan di keranjang belanja. Berikut Gambar 5.13 activity hapus daftar pembelian :

##### Gambar 5.13 Activity Diagram “hapus daftar pembelian”

#### 5.2.2.5 Sequence Diagram

Sequence diagram digunakan untuk menggambarkan perilaku sebuah skenario. Kegunaannya untuk menunjukan rangkaian pesan yang dikirim antara objek juga interaksi antara objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. Gambar 5.14 merupakan sequence diagram menambah sayur :



##### Gambar 5.14 Sequence Diagram “menambah sayur”

*Sequence Diagram* perbarui data sayurmerupakan interaksi anatara object untuk memperbarui/meng-*edit* data sayur. Berikut Gambar 5.15 merupakan Sequence diagram perbarui data sayur :

##### Gambar 5.15 Sequence Diagram “perbarui data sayur”

*Sequence Diagram* hapus sayurmerupakan interaksi anatara object untuk menghapusdata sayur. Berikut Gambar 5.16 merupakan Sequence diagram hapus sayur :

##### Gambar 5.16 Sequence Diagram “hapus sayur”

*Sequence Diagram* hapus usermerupakan interaksi anatara object untuk menghapus user yang terdaftar. Berikut Gambar 5.17 merupakan Sequence diagram hapus user :

##### Gambar 5.17 Sequence Diagram “hapus user”

*Sequence Diagram* search sayurmerupakan interaksi anatara object untuk mencarai data sayur berdasarkan namasayur. Berikut Gambar 5.18 merupakan Sequence diagram search sayur :

##### Gambar 5.18 Sequence Diagram “search sayur”

*Sequence Diagram* search usermerupakan interaksi anatara object untuk mencarai data user berdasarkan namauser. Berikut Gambar 5.19 merupakan Sequence diagram search user :

##### Gambar 5.19 Sequence Diagram “search user”

*Sequence Diagram* keranjang belanja merupakan interaksi anatara object untuk menyimpan data sayur ke dalam kerajang belanja/*cart*. Berikut Gambar 5.20 merupakan Sequence diagram keranjang belanja :

##### Gambar 5.20 Sequence Diagram “keranjang belanja”

*Sequence Diagram* pesan sayurmerupakan interaksi anatara object untuk memesan sayur yang telah dijual di sayur online. Berikut Gambar 5.21 merupakan Sequence diagram pesan sayur :



##### Gambar 5.21 Sequence Diagram “pesan sayur”

## 5.3 **Perancangan Perangkat Lunak**

Dalam pembuatan Sistem Jual Beli Sayur Online ini diperlukan tahap perancangan perangkat lunak yang terdiri dari perancangan algoritma, perancangan data dan perancangan antrarmuka user. Masing-masing tahap tersebut dijelaskan pada bagian perancangan berikut ini.

### 5.3.1 Perancangan Algoritma

Pada perancangan algoritma ini dijelaskan mengenai alur algoritma dalam bentuk diagram alir yang merupakan salah satu cara penyajian algoritma yang lebih mudah dipahami untuk menjelaskan alur-alur proses login, lihat data sayur, tambah data sayur, edit data sayurl dan hapus data sayur.

#### 5.3.1.1 Diagram Alir Proses Login

Diagram alir proses login sebagai user ini merupakan gambaran secara visual dari algoritma login, Admin melakukan input username dan password. Diagram alir proses tersebut ditunjukan seperti pada Gambar 5.22 berikut ini.



##### Gambar 5.22 Diagram Alir Proses Login Sebagai Admin

#### 5.3.1.2 Diagram Alir Proses Tambah Data Sayur

Diagram alir proses lihat data Sayur ini merupakan gambaran secara visual dari algoritma proses tambah data Sayur, Diagram alir proses tersebut ditunjukan seperti pada Gambar 5.23 berikut ini.



##### Gambar 5.23 Diagram Alir Proses Tambah Data Sayur

#### 5.3.1.3 Diagram Alir Proses Lihat Data Sayur

Diagram alir proses lihat data ibu hamil ini merupakan gambaran secara visual dari algoritma proses lihat data Sayur, Diagram alir proses tersebut ditunjukan seperti pada Gambar 5.24 berikut ini.



##### Gambar 5.24 Diagram Alir Proses Lihat Data Sayur

#### 5.3.1.4 Diagram Alir Proses Perbarui Data Sayur

Diagram alir proses perbaru data sayur ini merupakan gambaran secara visual dari algoritma proses perbarui data sayur, admin melakukan perbarui data sayur pada form manage sayur. Diagram alir proses tersebut ditunjukan seperti pada Gambar 5.25 berikut ini.



##### Gambar 5.25 Diagram Alir Proses Perbarui Data Sayur

#### 5.3.1.5 Diagram Alir Proses Hapus Data Sayur

Diagram alir proses hapus data sayur ini merupakan gambaran secara visual dari algoritma proses hapus data sayur, Admin melakukan hapus data sayur pada form manage sayur. Diagram alir proses tersebut ditunjukan seperti pada Gambar 5.26 berikut ini.



##### Gambar 5.26 Diagram Alir Proses Hapus Data Sayur

### 5.3.2 Perancangan Antarmuka

Pengguna menggunakan aplikasi melalui sistem antarmuka. Perancangan antarmuka akan di terangkan pada bagian ini. Berikut ini adalah rancangan tampilan dari sistem yang akan dibuat.

### 5.3.3.1 Perancangan Tampilan Halaman Login

Tampilan halaman login sebagai user berisi logo, judul sistem, *field email* dan *password* serta tombol *Sign In* yang digunakan *admin* dan *user* untuk masuk ke dalam sistem setelah menginputkan *email* dan *password.* Pada Gambar 5.27 berikut ini menunjukkan halaman login*.*

**

##### Gambar 5.27 Perancangan Tampilan Halaman Login

### 5.3.3.2 Perancangan Tampilan Halaman Register

Tampilan halaman register berisi logo, judul sistem, *field* nama, no telp, email, password, konfiramsi password serta tombol *Sign Up* yang digunakan *user* untuk mendaftar ke dalam sistem setelah menginputkan *field* yang tersedia*.* Pada Gambar 5.28 berikut ini menunjukkan halaman *register.*

**

##### Gambar 5.28 Perancangan Tampilan Halaman Register

### 5.3.3.3 Perancangan Tampilan Halaman *Home User*

Tampilan halaman *Home User*  berisi tampilan yang muncul ketika user berhasil login. User dapat memilih dan melihat sayur sesuai yang dijaul oleh admin, di halaman home user memiliki fitur *bottom sheet layout* berfungsi untuk menyimpan sayur yang akan dibeli dan menentukan jumlah sayur yang akan dibeli. Terdapat Button add digunakan untuk menyimpan sayur yang akan dibeli ke bottom sheet layout, dan button checkout berfungsi sebagai menyimpan sayur yang akan dibeli ke database dan berpindah halaman ke keranjang belanja untuk melengkapi data pengiriman. Yang terakhir terdapat tombol search yang berfungsi untuk mencari data sayur secara spesifik. Pada Gambar 5.29 berikut ini menunjukkan halaman *Home User*.

##### Gambar 5.29 Perancangan Tampilan Halaman *Home User*

### 5.3.3.4 Perancangan Tampilan Halaman Keranjang Belanja

Tampilan halaman keranjang belanja berisi tampilan yang muncul ketika user menekan tombol checkout. Halaman ini akan menampilkan data sayur, total harga sayur yang akan dibeli dan untuk melengkapi data pengiriman, di halaman keranjang belanja ini terdapat button *order* untuk masuk ke proses pembelian sayur. Pada Gambar 5.30 berikut ini menunjukkan halamanKeranjang Belanja.

**

##### Gambar 5.30 Perancangan Tampilan Halaman Keranjang Belanja

### 5.3.3.5 Perancangan Tampilan Halaman Transaksi

Tampilan halaman Transaksi belanja berisi tampilan yang muncul ketika user menekan tombol order. Halaman ini akan menampilkan data sayur, total sayur yang sebelumnya telah di order,jika barang sudah terkirim makan data akan berpindah ke tab *completed*. Pada Gambar 5.31 berikut ini menunjukkan halaman tranasksi.

**

##### Gambar 5.31 Perancangan Tampilan Halaman Transaksi

### 5.3.3.6 Perancangan Tampilan Halaman Profil

Tampilan halamanProfil muncul ketika user menekan tombol yang ada dipojok kanan bawah yaitu tombol *profile*. Halaman ini akan menampilkan data user yang yang berhasil login. Dihalaman ini user dapat merubah nama ,email , no telepon dan foto. Pada Gambar 5.32 berikut ini menunjukkan halamanProfil.

**

##### Gambar 5.32 Perancangan Tampilan Halaman Profil

### 5.3.3.7 Perancangan Tampilan Gudang Sayur

Tampilan halamangudang sayur adalah menu ketika admin berhasi login. Halaman ini akan menampilkan data sayur yang ada digudang. Dihalaman ini terdapat tombol add yang berfungsi untuk menambahkan sayur yang ingin dijual dari gudang ,dan diatas juga terdapat tombol search untuk mencari sayur di gudang dengan spesifik. Pada Gambar 5.33 berikut ini menunjukkan halamanGudang Sayur.

******

##### Gambar 5.33 Perancangan Tampilan Halaman Gudang Sayur

### 5.3.3.8 Perancangan Tampilan *Manage User*

******Tampilan halaman *Manage* User adalah menu ketika admin berhasi login. Halaman ini akan menampilkan data user yang telah mendaftar. Dihalaman ini terdapat tombol lihat yang berfungsi untuk memanage user, dan diatas juga terdapat tombol search untuk mencari user dengan spesifik. Pada Gambar 5.34 berikut ini menunjukkan halaman *Manage* User.

##### Gambar 5.34 Perancangan Tampilan Halaman Manage User

### 5.3.3.9 Perancangan Tampilan *Manage* Sayur

Tampilan halaman *Manage* Sayur adalah menu ketika admin berhasi login. Halaman ini akan menampilkan data sayur yang telah dijual. Dihalaman ini terdapat tombol edit yang berfungsi untuk memanage data sayur, dan diatas juga terdapat tombol search untuk mencari sayur yang dijual dengan spesifik. Pada Gambar 5.35 berikut ini menunjukkan halaman *Manage* Sayur.



##### Gambar 5.35 Perancangan Tampilan Halaman Manage Sayur

### 5.3.3.10 Perancangan Tampilan *Manage* Transaksi

Tampilan halaman *Manage* Transaksi adalah menu ketika admin berhasi login. Halaman ini akan menampilkan data sayur yang dibeli oleh user. Dihalaman ini admin dapat mengedit transaki yang telah dilakukan oleh user. Pada Gambar 5.36 berikut ini menunjukkan halaman *Manage* Transaksi.



##### Gambar 5.36 Perancangan Tampilan Halaman Manage Transaksi

1. IMPLEMENTASI

Bab implementasi ini merupakan tahap pembuatan sistem yang telah dianalisa dan dirancang pada bab sebelumnya, seperti batasan implementasi, spesifikasi sistem, implementasi algoritma, dan implementasi antarmuka. Tujuan yang dicapai pada tahap ini adalah hasil perancangan sistem yang telah dibuat dapat dioperasikan. Pembahasan pada bab implementasi ditunjukkan dengan pohon implementasi seperti pada Gambar 6.1 berikut ini.

##### Gambar 6.1 Diagram Alir Implementasi

## 6.1 Batasan Implementasi

Batasan implementasi dalam mengimplementasikan Sistem Jual Beli Sayur Online Berbasis Android Mengguanakan Webservice ini, diantaranya :

* *Input* sistem berupa database maupun data yang dilakukan input secaramanual oleh *user* dan *Admin.*
* *Output* yang diterima oleh user dan admin berupa hasil dari database yang telahdisimpan ke dalam sistem. Hasil database dapat ditampilkan berupa data.
* Paremeter yang digunakan yaitu :
  1. Pesan Sayur
  2. Keranjang Belanja
  3. Data Sayur

## 6.2 Spesifikasi Sistem

Dari hasil analisa dan perancangan pada bab sebelumnya akan dijadikan acuan dalam melakukan implementasi. Pengembangan sistem dilakukan dalam lingkungan implementasi yang terdiri dari perangkat keras dan perangkat lunak.

### 6.2.1 Spesifikasi Perangkat Keras

Sistem Jual Beli Sayur Online Berbasis Android Mengguanakan Webservice menggunakan komputer yang sesuai dengan spesifikasi perangkat keras yang dijelaskan pada Tabel 6.1 berikut

|  |  |  |
| --- | --- | --- |
|  | Tabel 6.1 Spesifikasi Perangkat Keras | |
|  |  |  |
| **Nama Komponen** |  | **Spesifikasi** |
| Processor |  | Intel(R) Core™ i5-4200u-1.8GHz Turbo 2.3GHz |
| Memori (RAM) |  | 8.00 GB RAM |
| Harddisk |  | 500 GB |

### 6.2.2 Spesifikasi Perangkat Lunak

Implementasi Sistem Jual Beli Sayur Online Berbasis Android Mengguanakan Webservice menggunakan sebuah komputer yang sesuai dengan spesifikasi perangkat lunak yang dijelaskan pada Tabel 6.2 berikut.

Tabel 6.2 Spesifikasi Perangkat Lunak

|  |  |
| --- | --- |
| **Nama Komponen** | **Spesifikasi** |
|  |  |
| Android Marshmallow 6.0 ke | Sistem operasi yang digunakan untuk menjalankan |
| atas | Sayur Online |
| Visual Studio Code dan | Sebagai editor untuk pemrograman Sayur Online |
| Android Studio |  |
| Android Emulator | Sebagai media untuk uji coba Sayur Online |

## 6.3 Implementasi Algoritma

Dalam implementasi Sistem Jual Beli Sayur Online Berbasis Android Mengguanakan Webservice ini memiliki beberapa proses, diantaranya login sebagai admin dan user, lihat data sayu, tambah data Sayur, edit data Sayur, dan hapus data Sayur.

### 6.3.1 Implementasi Algoritma Login

|  |  |
| --- | --- |
| 1 | private void checkLogin(final String email, final String password) { |
| 2 | StringRequest stringRequest = new StringRequest(Request.Method.POST, |
|  | **API\_URL, new Response.Listener<String>() {** |
| 3 | **public void** onResponse(String response) { |
| 4 | String json = response.toString(); |
| 5 | **try{JSONObject jsonObject = new JSONObject(json);** |
| 6 | **if**(jsonObject.getString(**"status"**).equals(**"failed"**)){ |
| 7 | Toast.*makeText*(getApplicationContext(),**"Email atau password salah"**, |
|  | Toast.LENGTH\_SHORT).show(); } |
| 8 | **else**{ if(jsonObject.getString("tipe").equals("admin")) { |
| 9 | **Intent intent = new Intent(getApplicationContext(), AdminHome.class);** |
| 10 | startActivity(intent); |
| 11 | finish();} |

Proses login ini sangat penting agar pengguna dapat mengakses atau masuk ke dalam sistem sayur online. Ketika admin dan user menjalankan aplikasi, maka akan menampilkan halaman login sistem yang berisi logo, judul sistem, field email dan password. User dan admin diminta untuk menginputkan email dan password sesuai yang tersimpan dalam sistem sebelumnya. Kemudian user mengklik tombol sign in untuk masuk ke dalam sistem. Berikut ini Gambar 6.2 merupakan algoritma kode program masuk sebagai pengguna beserta penjelasnnya.

##### Gambar 6.2 Kode Program Login

Penjelasan Kode :

* Baris 1 : Membuat Fungsi Cek Login dengan parameter email dan password
* Baris 2 : Membuat perintah String request dengan parameter Request data dari API
* Baris 3-4 : Membuat Fungsi Onresponse dengan paramameter String Respon
* Baris 5 : Membuat fungsi JSON object
* Baris 6-7 : Jika status login sama dengan failed / salah maka akan muncul error Email atau password salah
* Baris 8-11 : Jika status login sama dengan tipe admin maka akan intent / berpindah halaman ke Admin Home . Selesai.

### 6.3.2 Implementasi Algoritma Tambah Data Sayur

|  |  |
| --- | --- |
| 1 | **private void** showDialog(**final** Context context, **final** String idSayur){ |
| 2 | StringRequest stringRequest = **new** StringRequest(Request.Method.***POST***, |
|  | **API\_URL + "/tambah-sayur", new Response.Listener<String>() {** |
| 3 | **public void** onResponse(String response) { |
| 4 | Intent mIntent = **new** Intent(context, AdminHome.**class**); |
| 5 | **context.startActivity(mIntent);}},** |
| 6 | **new Response.ErrorListener() {** |
| 7 | **public void** onErrorResponse(VolleyError error) { |
|  | error.printStackTrace(); } }) |

Pada proses implementasi tambah data Sayur, Admin terlebih dahulu melakukan login untuk masuk ke dalam sistem. Kemudian Admin memilih menu data sayur. Sistem akan menampilkan form sayur gudang. Admin dapat memilih data yang akan ditambahkan atau dijual, jika sudah Admin dapat menyimpan data dengan mengklik tombol add. Data akan ditambahkan ke dalam database sistem pada webservice. Berikut ini Gambar 6.3 merupakan algoritma kode program tambah data ibu hamil beserta penjelasnnya.

##### Gambar 6.3 Kode Program Tambah Sayur

Penjelasan Kode :

* Baris 1 : Membuat Fungsi show dialog dengan parameter context dan idsayur
* Baris 2 : Membuat perintah String request dengan parameter Request data dari API
* Baris 3 : Membuat Fungsi Onresponse dengan paramameter String Respon
* Baris 4-5 : Data akan berpindah dari gudang ke data sayur yang akan dijual
* Baris 6-7 : Membuat fungsi error respon dengan parameter volley error

### 6.3.3 Implementasi Algoritma Edit Data Sayur

Pada implementasi proses edit Sayur, Admin terlebih dahulu melakukan login setelah itu memilih menu data Sayur. Sistem akan menampilkan data sayur yang dijual. Admin dapat merubahdata yang ada, jika sudah lengkap admin dapat menyimpan data dengan mengklik tombol edit. Data akan disimpan ke dalam database sistem pada webservice. Berikut ini Gambar 6.4 merupakan algoritma kode program edit data ibu hamil beserta penjelasnnya.

|  |  |
| --- | --- |
| 1 | **public void** ubahsayur (**final** String ID,**final** Context context){ |
| 2 | StringRequest updateReq = **new** StringRequest(Request.Method.***POST***, |
|  | **API\_URL\_ubahsayur,** |
| 3 | **new** Response.Listener<String>() { |
| 4 | **public void** onResponse(String response) { |
| 5 | Toast.*makeText*(getApplicationContext(), **"Edit Sayur Berhasil"**, |
|  | **Toast.LENGTH\_SHORT).show();** |
|  | startActivity( **new** Intent(Adminubahsayur.**this**,AdminHome.**class**)); }}, |
| 6 | **new** Response.ErrorListener() { |
| 7 | **public void** onErrorResponse(VolleyError error) { |
| 8 | Toast.makeText(Adminubahsayur.this, "pesan : Gagal Edit sayur", |
|  | Toast.LENGTH\_SHORT).show();} }) |

##### Gambar 6.4 Kode Program Edit Data Sayur

Penjelasan Kode :

* Baris 1 : Membuat Fungsi Ubah sayur dengan parameter id dan context
* Baris 2 : Membuat perintah String request dengan parameter Request data dari API
* Baris 3-4 : Membuat Fungsi Onresponse dengan paramameter String Respon
* Baris 5 : Data akan berhasil diubah dan akan menampilkan alert edit sayur berhasil dan akan berpindah from ke admin home
* Baris 6-8 : Jika data error atau tidak sama dengan field yang ada di database makan akan muncel pesan error “gagal edit sayur”

## Antarmuka

Implementasi antarmuka pada Sistem Jual Beli Sayur Online Berbasis Android Mengguanakan Webservice ini digunakan sebagai perantara antara pengguna dan sistem untuk berinteraksi. Berikut merupakan hasil implementasi antarmuka dari Sistem Jual Beli Sayur Online Berbasis Android Mengguanakan Webservice.

**a. Implementasi Halaman Login**

Tampilan halaman login sebagai user berisi logo, judul sistem, *field email* dan *password* serta tombol *Sign In* yang digunakan *admin* dan *user* untuk masuk ke dalam sistem setelah menginputkan *email* dan *password .* Pada Gambar 6.5 berikut ini menunjukkan halaman login*.*

**

##### Gambar 6.5 Implementasi Halaman Login

**b. Implementasi Halaman Register**

Tampilan halaman register berisi logo, judul sistem, *field* nama, no telp, email, password, konfiramsi password serta tombol *Sign Up* yang digunakan *user* untuk mendaftar ke dalam sistem setelah menginputkan *field* yang tersedia*.* Pada Gambar 6.6 berikut ini menunjukkan halaman *register*.

##### Gambar 6.6 Implementasi Halaman Register

**c. Implementasi Halaman *Home User***

Tampilan halaman *Home User*  berisi tampilan yang muncul ketika user berhasil login. User dapat memilih dan melihat sayur sesuai yang dijaul oleh admin, di halaman home user memiliki fitur *bottom sheet layout* berfungsi untuk menyimpan sayur yang akan dibeli dan menentukan jumlah sayur yang akan dibeli. Terdapat Button add digunakan untuk menyimpan sayur yang akan dibeli ke bottom sheet layout, dan button checkout berfungsi sebagai menyimpan sayur yang akan dibeli ke database dan berpindah halaman ke keranjang belanja untuk melengkapi data pengiriman. Yang terakhir terdapat tombol search yang berfungsi untuk mencari data sayur secara spesifik. Pada Gambar 6.7 berikut ini menunjukkan halaman *Home User*.



##### Gambar 6.7 Implementasi Halaman Home User

**d. Implementasi Halaman Keranjang Belanja**

Tampilan halaman keranjang belanja berisi tampilan yang muncul ketika user menekan tombol checkout. Halaman ini akan menampilkan data sayur, total harga sayur yang akan dibeli dan untuk melengkapi data pengiriman, di halaman keranjang belanja ini terdapat button *order* untuk masuk ke proses pembelian sayur. Pada Gambar 6.8 berikut ini menunjukkan halamanKeranjang Belanja.



##### Gambar 6.8 Implementasi Halaman Keranjang Belanja

**e. Implementasi Halaman Transaksi**

Tampilan halaman Transaksi belanja berisi tampilan yang muncul ketika user menekan tombol order. Halaman ini akan menampilkan data sayur, total sayur yang sebelumnya telah di order,jika barang sudah terkirim makan data akan berpindah ke tab *completed*. Pada Gambar 6.9 berikut ini menunjukkan halaman tranasksi.

##### Gambar 6.9 Implementasi Halaman Transaksi

**f. Implementasi Halaman Profil**

Tampilan halamanProfil muncul ketika user menekan tombol yang ada dipojok kanan bawah yaitu tombol *profile*. Halaman ini akan menampilkan data user yang yang berhasil login. Dihalaman ini user dapat merubah nama ,email ,no telepon dan foto. Pada Gambar 6.10 berikut ini menunjukkan halamanProfil.



##### Gambar 6.10 Implementasi Halaman Profil

**g. Implementasi Halaman Gudang Sayur**

Tampilan halamangudang sayur adalah menu ketika admin berhasi login. Halaman ini akan menampilkan data sayur yang ada digudang. Dihalaman ini terdapat tombol add yang berfungsi untuk menambahkan sayur yang ingin dijual dari gudang ,dan diatas juga terdapat tombol search untuk mencari sayur di gudang dengan spesifik. Pada Gambar 6.11 berikut ini menunjukkan halamanGudang Sayur.

##### Gambar 6.11 Implementasi Halaman Gudang Sayur

**h. Implementasi Halaman *Manage* User**

Tampilan halaman *Manage* User adalah menu ketika admin berhasi login. Halaman ini akan menampilkan data user yang telah mendaftar. Dihalaman ini terdapat tombol lihat yang berfungsi untuk memanage user, dan diatas juga terdapat tombol search untuk mencari user dengan spesifik. Pada Gambar 6.12 berikut ini menunjukkan halaman *Manage* User.



##### Gambar 6.12 Implementasi Halaman Manage User

**i. Implementasi Halaman Manage Sayur**

Tampilan halaman *Manage* Sayur adalah menu ketika admin berhasi login. Halaman ini akan menampilkan data sayur yang telah dijual. Dihalaman ini terdapat tombol edit yang berfungsi untuk memanage data sayur, dan diatas juga terdapat tombol search untuk mencari sayur yang dijual dengan spesifik. Pada Gambar 6.13 berikut ini menunjukkan halaman *Manage* Sayur.

##### Gambar 6.13 Implementasi Halaman Manage Sayur

**j. Implementasi Halaman *Manage* Transaki**

Tampilan halaman *Manage* Transaksi adalah menu ketika admin berhasi login. Halaman ini akan menampilkan data sayur yang dibeli oleh user. Dihalaman ini admin dapat mengedit transaki yang telah dilakukan oleh user. Pada Gambar 6.14 berikut ini menunjukkan halaman *Manage* Transaksi.



##### Gambar 6.13 Implementasi Halaman Manage Transaksi

1. PENGUJIAN DAN ANALISIS

Pada bab pengujian dan analisis akan menjelaskan tahapan dalam melakukan pengujian dan analisa terhadap Sistem Jual Beli Sayur Online Berbasis Android dengan *Web Service.* Pengujian dilakukan dengan validasi untuk tiap kasus menggunakan *Black – Box testing*. Kemudian dilakukan tahap analisis dari hasil pengujian yang telah dilakukan. Berikut Gambar 7.1 merupakan alur tahap pengujian dan analisis.



Pengujian dan

Analisis



7.1

Pengujian



7.1.1

Pengujian

Validasi



7.2

Analisis



7.2.2

Analisis Hasil

Pengujian

Validasi

##### Gambar 7.1 Diagram Alir Pengujian dan Analisis

## 7.1 Pengujian

Pada pengujian sistem yang dilakukan menggunakan tahap pengujian validasi. Pengujian validasi berfungsi untuk menentukan apakah hasil dari pengembangan sistem telah sesuai dengan kebutuhan.

### 7.1.1 Pengujian Validasi

Pengujian digunakan untuk memastikan bahwa sistem berfungsi sesuai perancangan sistem dan memenuhi kebutuhan user. Pengujian yang digunakan pada sistem ini adalah teknik pengujian *black box*. Pada pengujian *black box* digunakan metode pengujian validasi bertujuan untuk mengetahui apakah sistem yang dikembangkan sesuai dengan kebutuhan perangkat lunak yang sudah ditentukan sebelumnya. Berikut Tabel 7.1 merupakan pengujian validasi dari aplikasi SIMKES.

Tabel 7.1 Tabel Pengujian Validasi

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Test Name** | | **Test Case** | | **Expected Result** | **Result** | | **Status** | |
|  |  |  | | **Pengujian Login web** | | |  | |  |
| 1 | Pengujian Login web | | Pengujian dengan menggunakan username/email yang benar. Kemudian user melakukan klik tombol login | | Login berhasil dan masuk ke halaman home web. | Login berhasil dan masuk ke halaman home web. | | Valid | |
|  |  | | | | | | | | |
| 2 | Pengujian Login web | | Pengujian dengan menggunakan username/email yang salah. Kemudian user melakukan klik tombol login | | Login gagal dan dikembalikan  ke halaman  login,  menampilkan pesan “Email atau password tidak cocok” | Login gagal dan dikembalikan  ke halaman  login,  menampilkan pesan “Email atau password tidak cocok” | | Valid | |
|  |  | | | | | | | | |
|  | **Pengujian tambah, edit, hapus data sayur web** | | | | | | | | |
| 3 | Pengujian menambah data sayur, melihat data sayur yang ditambahkan. | | Pengujian dengan mengklik menu tambah sayur, lalu user menambah data sayur dengan lengkap. | | Data berhasil ditambahkan pada menu manage sayur. Dan melihat seluruh data sayur yang ditambahkan pada Menu manage sayur | Data berhasil ditambahkan pada menu manage sayur. Dan melihat seluruh data sayur yang ditambahkan pada Menu manage sayur | | Valid | |
|  |  | | | | | | | | |
| 4 | Pengujian  Melihat data sayur. | | Pengujian dengan mengklik sayur yang ingin dilihat. | | Menampilkan  Data lengkap tentang sayur yang ditambahkan. | Menampilkan  Data lengkap tentang sayur yang ditambahkan. | | Valid | |
|  | | | | | | | | | |
| 5 | Pengujian  Mengedit data sayur. | | Pengujian dengan mengklik menu edit sayur pada detail sayur. Lalu user mengisi data yang diinginkan dengan lengkap. | | Berhasil diedit dan tampil pada daftar sayur. | Berhasil diedit dan tampil pada daftar sayur. | | Valid | |
|  | | | | | | | | | |
| 6 | Pengujian  Hapus data sayur | | Pengujian dengan melakukan klik pada tombol hapus pada detail sayur. | | Akan muncul modal yang berisi “apakah anda yakin ingin menghapus sayur?” jika dipilih “Yakin” maka data sayur akan dihapus. | Akan muncul modal yang berisi “apakah anda yakin ingin menghapus sayur?” jika dipilih “Yakin” maka data sayur akan dihapus. | | Valid | |
|  | | | | | | | | | |
|  | **Melihat dan menghapus data user web.** | | | | | | | | |
| 7 | Melihat data user dan menghapus data user. | | Pengujian dengan memilih menu “Manage user” maka akan menampilkan seluruh data user. data user dapat dihapus dengan menekan tombol hapus pada kolom user. | | Menampilkan data user yang login. Jika menekan tombol hapus maka akan menampilkan modal “Apakah anda yakin ingin menghapus dta user?”, jika dipilih yakin maka data user dihapus. | Menampilkan data user yang login. Jika menekan tombol hapus maka akan menampilkan modal “Apakah anda yakin ingin menghapus dta user?”, jika dipilih yakin maka data user dihapus. | | Valid | |
|  |  | | | | | | | | |
|  | **Pengujian Lihat data transaksi web.** | | | | | | | | |
| 8 | Pengujian  Lihat Data  transaksi | | Pengujian dengan mengklik menu “Manage transaksi” | | Menampilkan halaman lihat data transaksi | Menampilkan halaman lihat data transaksi | | Valid | |
|  |  | | | | | | | | |
|  | **Pengujian Sign In, dan Sign Up pada mobile** | | | | | | | | |
| 9 | Pengujian  Sign in, dan sign up. | | Pengujian dengan mengisi data pada form sign in dan sign up. | | Masuk ke halaman home. | Masuk ke halaman home. | | Valid | |
|  | C:\Users\aryasa\Desktop\Screenshot_1534941623.png C:\Users\aryasa\Desktop\Screenshot_1534941946.png | | | | | | | | |
| 10 | Pengujian  Sign in, dan sign up. | | Pengujian dengan mengisi form sign in dan sign up dengan data yang salah. | | Menampilkan toast “password dan email salah” jika sign in gagal. | Menampilkan toast “password dan email salah” jika sign in gagal. | | Valid | |
| C:\Users\aryasa\Desktop\Screenshot_1534942190.png C:\Users\aryasa\Desktop\Screenshot_1534942264.png | | | | | | | | | |
| **Pengujian Lihat daftar sayur dan melakukan order.** | | | | | | | | | |
| 11 | Pengujian  Lihat daftar sayur dan melakukan order. | | Pengujian dengan mengklik menu home dan menambah dengan klik tombol “Add” lalu menekan tombol checkout pada bottomsheet order dilakukan dengan mengisi data pada halaman checkout. | | Menampilkan *bottom sheet* yang berisikan sayur yang ditambahkan.  Setelah menekan tombol “checkout” maka akan kembali kepada tampilan home. | Menampilkan *bottom sheet* yang berisikan sayur yang ditambahkan.  Setelah menekan tombol “checkout” maka akan kembali kepada tampilan home. | | Valid | |
| C:\Users\aryasa\Desktop\Screenshot_1534942682.png C:\Users\aryasa\Desktop\Screenshot_1534942726.png  C:\Users\aryasa\Desktop\Screenshot_1534942893.png | | | | | | | | | |
| 12 | Melihat histori transaksi. | | Pengujian dengan memilih menu histori . | | Menampilkan data transaksi yang telah dilakukan yang sedang “on progress” atau sudah “completed”. | Menampilkan data transaksi yang telah dilakukan yang sedang “on progress” atau sudah “completed”. | | Valid | |
|  | C:\Users\aryasa\Desktop\Screenshot_1534943071.png | | | | | | | | |
| **Pengujian Edit profil.** | | | | | | | | | |
| 13 | Pengujian  Edit profil. | | Pengujian dengan memilih menu profil lalu mengisi data dengan data beru lalu di klik simpan. | | Data profil akan disimpan dan muncul toast  “Berhasil  menyimpan data” | Data profil akan disimpan dan muncul toast  “Berhasil  menyimpan data” | | Valid | |
| C:\Users\aryasa\Desktop\Screenshot_1534943254.png | | | | | | | | | |
|  | **Pengujian search sayur.** | | | | | | | | |
| 14 | Pengujian  Tambah  Data Balita | | Pengujian dengan menginput nama sayur yang ingin dicari pada menu search pada halaman home. | | Mengetik nama sayur dan menampilkan sayur yang dicari. | Mengetik nama sayur dan menampilkan sayur yang dicari. | | Valid | |
|  | C:\Users\aryasa\Desktop\Screenshot_1534943490.png | | | | | | | | |
|  | **Pengujian Menambah sayur yang dijual dari gudang.** | | | | | | | | |
| 15 | Pengujian  Menambah sayur yang dijual dari gudang**.** | | Pengujian dengan mengklik tombol “Add” pada menu gudang. | | Menampilkan pop-up. Jika memilih “Ya” maka sayur akan hilang dari gudang dan berpindah ke halaman sayur yang dijaul. | Menampilkan pop-up. Jika memilih “Ya” maka sayur akan hilang dari gudang dan berpindah ke halaman sayur yang dijaul. | | Valid | |
| C:\Users\aryasa\Desktop\Screenshot_1534944136.png C:\Users\aryasa\Desktop\Screenshot_1534944133.png  C:\Users\aryasa\Desktop\Screenshot_1534944199.png C:\Users\aryasa\Desktop\Screenshot_1534944203.png | | | | | | | | | |
| **Pengujian Melihat daftar user, Detail dan hapus user.** | | | | | | | | | |
| 16 | Pengujian  Daftar user , detail user, dan hapus user. | | Pengujian dengan mengklik menu daftar user. Mengklik tombol lihat maka menampilkan detail user. Melakukan klik hapus jika ingin menghapus user. | | Menampilkan daftar user yang terdaftar. Jika melakukan klik “lihat” maka menampilkan halaman detail user. Menampilkan pop-up jika ingin menghapus user jika memilih “Ya” maka data user akan dihapus. | Menampilkan daftar user yang terdaftar. Jika melakukan klik “lihat” maka menampilkan halaman detail user. Menampilkan pop-up jika ingin menghapus user jika memilih “Ya” maka data user akan dihapus. | | Valid | |
|  | C:\Users\aryasa\Desktop\Screenshot_1534944370.png C:\Users\aryasa\Desktop\Screenshot_1534944374.png C:\Users\aryasa\Desktop\Screenshot_1534944628.png | | | | | | | | |
| **Melihat data transaksi dari user.** | | | | | | | | | |
| 17 | Pengujian  Melihat data transaksi yang dilakukan. | | Pengujian dengan mengklik menu transaksi dan melakukan klik pada transaksi untuk melihat detail transaksi. | | Menampilkan list transaksi yang sudah dilakukan.  Menampilkan detail transaksi jika mengklik data transaksi. | Menampilkan list transaksi yang sudah dilakukan.  Menampilkan detail transaksi jika mengklik data transaksi. | | Valid | |
| C:\Users\aryasa\Desktop\Screenshot_1534944698.png C:\Users\aryasa\Desktop\Screenshot_1534944967.png | | | | | | | | | |
| **Pengujian search sayur (admin)** | | | | | | | | | |
| 18 | Pengujian  Search sayur pada admin. | | Pengujian dengan mengisi nama sayur yang diinginkan pada search. | | Menampilkan data sayur yang diinginkan. | Menampilkan data sayur yang diinginkan.” | | Valid | |
| C:\Users\aryasa\Desktop\Screenshot_1534945076.png | | | | | | | | | |
| **Pengujian Logout.** | | | | | | | | | |
| 19 | Pengujian  Logout. | | Pengujian dengan mengklik tombol logout pada bagian profil maka akan kembali kepada halaman home. | | Menampilkan halaman home dan user logout.  Icon profil juga akan berubah. | Menampilkan halaman home dan user logout. Icon profil juga akan berubah. | | Valid | |
| C:\Users\aryasa\Desktop\Screenshot_1534945398.png C:\Users\aryasa\Desktop\Screenshot_1534945446.png | | | | | | | | | |

## 7.2 Analisis

Proses analisis dilakukan untuk mendapatkan hasil dari kesimpulan dari pengujian validasi yang telah dilakukan pada Sistem Jual Beli Sayur Berbasis Android dengan *Web Service*. Tahap analisis bertujuan sebagai perbaikan pada program yang telah dikembangan apabila terdapat ketidaksesuaian fungsi pada sistem.

### 7.2.1 Analisis Hasil Pengujian Validasi

Pengujian yang telah dilakukan dengan melihat daftar kebutuhan sistem dari bab analisis dan perancangan sebagai acuan pengujian. Dengan menggunakan teknik pengujian *black-box* menunjukkan validitas 100% terhadap hasil fungsi pada *expected result* dan hasil fungsi pada *result*, sehingga apa yang diharapkan sebelumya sesuai dengan hasil apa yang didapatkan. Dengan demikian Sistem Jual Beli Sayur Berbasis Android dengan *Web Service* dapat dinyatakan sesuai dengan daftar kebutuhan yang telah diuraikan dalam dokumentasi dari tahap analisa, perancangan dan implementasi.

1. PENUTUP

## 8.1 Kesimpulan

Berdasarkan hasil analisis, perancangan, implementasi dan pengujian yang dilakukan, maka dapat diperoleh kesimpulan sebagai berikut :

1. Dalam tahap pengembangan Sistem Jual Beli Sayur Berbasis Android dengan *Web Service* yang dikembangkan menggunakan android studio merupakan framework khusus untuk pengembangan aplikasi berbasis android terpenuhi sesuai dengan spesifikasi kebutuhan. Sehingga akan memudahkan jika dilakukan pengembangan lebih lanjut dengan penggunaan yang mudah dijangkau.
2. Dengan Sistem Jual Beli Sayur Berbasis Android dengan *Web Service* ini, menggunakan pelayanan *Web Service* dan *database* terpusat pada sebuah server. Untuk menyimpan data-data sayur dan pengguna yang dikelola dan telah terbukti dapat memenuhi kebutuhan informasi jual-beli yang dibutuhkan. Sehingga dapat memudahkan pelayanan penjualan sayur online berbasis *web service*.
3. Berdasarkan hasil pengujian dan analisis yang telah dilakukan pada Sistem Jual Beli Sayur Berbasis Android dengan *Web Service* ini telah dibuat sesuai dengan perancangan dan diimplementasikan dari fungsi dan komponen-komponen yang telah ditentukan. Sehingga sistem ini dapat memenuhi semua daftar kebutuhan. Dengan demikian tahap perancangan dan pengembangan Sistem Jual Beli Sayur Berbasis Android dengan *Web Service* dapat dinyatakan berhasil.

## 8.2 Saran

Saran yang diberikan dalam pengembangan Sistem Jual Beli Sayur Berbasis Android dengan *Web Service* kedepannya adalah sebagai berikut :

1. Untuk pengembangan lebih lanjut, dapat ditambahkan fitur-fitur baru yaitu seperti transaksi dengan pembayaran menggunakan ATM atau transfer pembayaran melalui bank yang telah ditentukan, menambah fitur chatting dengan user dengan admin, penambahan *notifikasi*, penambahan fungdi agar pada saat keluar dari aplikasi data sayur yang tersimpan pada keranjang tidak menghilang, penambahan fungsi untuk melakukan penambahan *user* pada bagian web sayurin dan juga dilakukan perbaikan-perbaikan pada beberapa fitur agar prosesnya dapat lebih sederhana dan *user friendly*.
2. Dapat dilanjutkan pengembangannya pada platform lain agar pelayanan dapat merata kepada seluruh konsumen atau pelanggan.

# DAFTAR PUSTAKA

Android, 2018. *About the Android Open Source Project*. [online] Tersedia di : <

https://source.android.com/ > [Diakses 11 September 2018].

Android Developer, 2018. *Meet Android Studio*.[online] Tersedia di : < https://

developer.android.com/studio/intro/#top\_of\_page> [Diakses 10 September

2018].

Armel, J. 2014. *Web application development with Laravel PHP Framework version 4.*

S1*.*Helsinski : Metropolia University. Tersedia di : <http://www.theseus.fi/

bitstream/handle/10024/74052/Author.pdf?sequence=1&isAllowed=y>

[Diakses pada 18 september 2018]

Bell, D. 2003. *UML basics: An introduction to the Unified Modeling Language* [e-

book]. IBM : Developer works. Tersedia di : IBM Developer Works < https://

www.ibm.com/developerworks/rational/library/769.html > [Diakses pada 20

agustus 2018 ].

Bittner, K.2006. Introduction To Writing Good Use Cases*. Development Conference*

*2006.*[Pdf] IBM Rational Software. Tersedia di : <http://www-07.ibm.com

/shared\_downloads2/software/rsdc2006/ra\_day\_1/WritingGoodUseCases.pdf

>[Diakses pada 18 September 2018]

Chen , X., Ji , Z., Yu , F. & Zhan , Y. 2017. Restful API Architecture Based on Laravel

Framework. *Journal of Physics: Conference Series,* [e-journal] 910 (012016).

Tersedia di : IOP Publishing <http://iopscience.iop.org/article/10.1088/1742-6596/910/1/012016> [Diakses pada 18 september 2018] **.**

Dennis, A., Wixom, B.H. & Roth , R.M., 2012. *System Analysis & Design*. 5th ed. USA :

Jhon Wiley & Sons, Inc.

Laravel , 2018*. Introduction*. [online] Tersedia di : <https://laravel.com/docs/4.2/

introduction /> [Diakses 11 September 2018].

Marlindawati, 2014. *Aplikasi Pemesanan Menu Makanan Di Rumah Makan Berbasis*

*Web Service Menggunakan Mobile Android*. S1. Universitas binadarma. Tersedia di : <http://eprints.binadarma.ac.id/2002/1/Jurnalayu\_inayah.pdf>

[diakses pada 10 september 2018]

Rusdiana. H.A & Irfan, Moch., 2014. *Sistem informasi manajemen*. Bandung: Pustaka

Setia.

Sommerville, I., 2011. *Software engineering*. 9th ed. London: Addison-Wesley.

Smyth, N. 2015. Android Development Essential. 6th ed [e-book]. Tersedia di :

ebookfrenzy <https://www.ebookfrenzy.com/pdf\_previews/AndroidStudio6

EssentialsPreview.pdf> [Diakses pada 5 Juli 2018].

UML Diagram, 2018.*The Unified Modeling Language*.[online] Tersedia di :

<https://www.uml-diagrams.org/ > [Diakses 11 September 2018].

W3C. 2004. *Web Service Glossary*.[online] Tersedia di : <https://www.w3.org/TR/ws-

gloss/> [Diakses 3 Agustus 2018].