**EKSPLORASI JSON ADAPTER UNTUK RETROFIT PADA PEMBANGUNAN APLIKASI BERBASIS ANDROID**

**(Studi Kasus Publikasi informasi dan kegiatan di Studio Chendra PT Dinasti Motekar Grup)**

**TUGAS AKHIR**

Disusun untuk syarat kelulusan Seminar Kualifikasi Penelitian

di Program Studi Teknik Informatika, Universitas Pasundan Bandung

oleh:

Adhitya Sidiq Permana

nrp. 15.304.0049



**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS PASUNDAN BANDUNG**

**NOVEMBER 2020**

# LEMBAR PENGESAHAN LAPORAN TUGAS AKHIR

Telah diujikan dan dipertahankan dalam Sidang Sarjana Program Studi Teknik Informatika Universitas Pasundan Bandung, pada hari dan tanggal sidang sesuai berita acara sidang, tugas akhir dari:

Nama : Adhitya Sidiq Permana

Nrp : 15.304.0049

Dengan judul :

“**EKSPLORASI JSON ADAPTER UNTUK RETROFIT PADA PEMBANGUNAN APLIKASI BERBASIS ANDROID**

**(Studi Kasus Publikasi informasi dan kegiatan di Studio Chendra PT Dinasti Motekar Grup)**

Menyetujui,

Bandung,

Pembimbing Utama

(Wanda Gusdya, ST., MT.)

# LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR

Saya menyatakan dengan sesungguhnya bahwa :

1. Tugas akhir ini adalah benar-benar asli dan belum pernah diajukan untuk mendapatkan gelar akademik, baik di Universitas Pasundan Bandung maupun di Perguruan Tinggi lainnya.
2. Tugas akhir ini merupakan gagasan, rumusan dan penelitian saya sendiri, tanpa bantuan pihak lain kecuali arahan dari tim Dosen Pembimbing.
3. Dalam tugas akhir ini tidak terdapat karya atau pendapat orang lain, kecuali bagian-bagian tertentu dalam penulisan laporan Tugas Akhir yang saya kutip dari hasil karya orang lain telah dituliskan dalam sumbernya secara jelas sesuai dengan norma, kaidah, dan etika penulisan karya ilmiah, serta disebutkan dalam Daftar Pustaka pada tugas akhir ini.
4. Kakas, perangkat lunak, dan alat bantu kerja lainnya yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab saya, bukan tanggung jawab Universitas Pasundan Bandung.

Apabila di kemudian hari ditemukan seluruh atau sebagian laporan tugas akhir ini bukan hasil karya saya sendiri atau adanya plagiasi dalam bagian-bagian tertentu, saya bersedia menerima sangsi akademik, termasuk pencabutan gelar akademik yang saya sandang sesuai dengan norma yang berlaku di Universitas Pasundan, serta perundang-undangan lainnya.

|  |
| --- |
| Bandung, \_\_\_\_\_\_\_\_\_\_\_ |
| Yang membuat pernyataan |
|  |
|  |
|  |
|  |
|  |
| **(Adhitya Sidiq Permana)** |
| NRP. 15.304.0049 |

# ABSTRAK

# ABSTRACT

# KATA PENGANTAR

Ucapan dan rasa syukur penulis layangkan ke hadirat Ilahi Robbi, yang telah berkenan menguatkan penulis untuk membuat Laporan Tugas Akhir dengan judul “Eksplorasi JSON Adapter untuk Retrofit pada Pembangunan Pplikasi berbasis Android (Studi Kasus Publikasi Informasi dan Kegiatan di Studio Chendra PT Dinasti Motekar Grup)”

Adapun penulisan laporan ini bertujuan untuk memenuhi salah syarat kelulusan program stuidi strata 1 di Program Studi Teknik Informatika Universitas Pasundan.

Penulis menyadari laporan ini dapat terwujud berkat bantuan dan dorongan dari berbagai pihak. Maka pada kesempatan ini penulis sampaikan terima kasih yang sebesar-besarnya atas segala bantuan yang penulis terima baik secara moril maupun materil, sehingga penulis dapat menyelesaikan laporan ini kepada :

1. Kepada Orang Tua tersayang, dan keluarga yang selalu memberikan motivasi serta do’anya dalam pembuatan tugas akhir ini.
2. Kepada pembimbing utama, Bapak Wanda Gusdya, ST., MT
3. Koordinator Tugas Akhir dan Ketua Kelompok Keilmuan serta seluruh civitas akademika Teknik Informatika di UNIVERSITAS PASUNDAN BANDUNG, yang telah memberikan bekal ilmu selama penulis menimba ilmu.
4. Kepada teman-teman seperjuangan Universitas Pasundan Bandung yang tidak bisa semua penulis sebutkan.

Tiada gading yang tak retak, tiada gelombang tanpa ombak, segala kesalahan merupakan kelemahan dan kekurangan penulis. oleh karena itu, penulis harapkan kritik dan saran dari semua pihak demi perbaikan di masa yang akan datang.

Akhir kata, semoga penulisan laporan ini dapat bermanfaat bagi penulis dan bagi perkembangan ilmu Teknologi dimasa yang akan datang.

Bandung, \_\_\_\_\_\_\_\_\_\_\_

Penulis

# 

# DAFTAR ISI

[ABSTRAK i](#_Toc57613629)

[ABSTRACT ii](#_Toc57613630)

[KATA PENGANTAR iii](#_Toc57613631)

[DAFTAR ISI i](#_Toc57613632)

[DAFTAR TABEL iii](#_Toc57613633)

[DAFTAR GAMBAR iv](#_Toc57613634)

[BAB 1 PENDAHULUAN 1-1](#_Toc57613635)

[1.1. Latar Belakang 1-1](#_Toc57613636)

[1.2. Identifikasi Masalah 1-2](#_Toc57613637)

[1.3. Tujuan Tugas Akhir 1-2](#_Toc57613638)

[1.4. Lingkup Tugas Akhir 1-3](#_Toc57613639)

[1.5. Metodologi Tugas Akhir 1-3](#_Toc57613640)

[1.6. Sistematika Penulisan Tugas Akhir 1-5](#_Toc57613641)

[BAB 2 LANDASAN TEORI 2-1](#_Toc57613642)

[2.1. Aplikasi Mobile 2-1](#_Toc57613643)

[2.1.1. Perkembangan Aplikasi Mobile 2-1](#_Toc57613644)

[2.2. Sistem Operasi Android 2-3](#_Toc57613645)

[2.2.1. Aplikasi Android 2-3](#_Toc57613646)

[2.3. Web Service & REST API 2-3](#_Toc57613647)

[2.3.1. Metodologi REST API 2-4](#_Toc57613648)

[2.4. Retrofit Library 2-4](#_Toc57613649)

[2.5. Gson & Moshi 2-5](#_Toc57613650)

[2.6. Konsep MVC 2-5](#_Toc57613651)

[2.7. Software Development Life Cycle – Waterfall Model 2-5](#_Toc57613652)

[2.7.1. Tahapan – tahapan Waterfall Model 2-6](#_Toc57613653)

[2.8. Penelitian terdahulu terkait dengan topik Tugas Akhir 2-7](#_Toc57613654)

[BAB 3 SKEMA PENELITIAN 3-1](#_Toc57613655)

[3.1. Rancangan Penelitian 3-1](#_Toc57613656)

[3.2. Analisis Masalah dan Solusi Tugas Akhir 3-4](#_Toc57613657)

[3.3. Kerangka Pemikiran Teoritis 3-5](#_Toc57613658)

[3.3.1. Rancangan Analisis 3-5](#_Toc57613659)

[3.3.2. Kerangka Berfikir Teoritis 3-7](#_Toc57613660)

[3.4. Analisis Teknologi 3-7](#_Toc57613661)

[3.4.1. Android Studio 3-7](#_Toc57613662)

[3.4.2. Retrofit 3-8](#_Toc57613663)

[3.4.3. JSON 3-8](#_Toc57613664)

[3.4.4. Gson & Moshi 3-9](#_Toc57613665)

[3.4.5. Web Service 3-9](#_Toc57613666)

[3.5. Profile Penelitian 3-9](#_Toc57613667)

[3.5.1. Objek Penelitian 3-9](#_Toc57613668)

[3.5.2. Tempat Penelitian 3-10](#_Toc57613669)

[3.5.3. Struktur Organisasi 3-10](#_Toc57613670)

[BAB 4 ANALISIS DAN PERANCANGAN 4-1](#_Toc57613671)

[4.1. Analisis Sistem yang sedang berjalan 4-1](#_Toc57613672)

[4.2. Analisis Kebutuhan 4-1](#_Toc57613673)

[4.2.1. Menentukan Kebutuhan Dasar 4-1](#_Toc57613674)

[4.2.2. Analisis Pengguna 4-2](#_Toc57613675)

[4.2.3. Analisis Ranah 4-2](#_Toc57613676)

[4.2.4. Pendekatan untuk Pemodelan Spesifikasi Kebutuhan 4-3](#_Toc57613677)

[4.2.5. Pemodelan Berbasis Skenario 4-4](#_Toc57613678)

[4.2.6. Pemodelan data 4-26](#_Toc57613679)

[4.2.7. Pemodelan Berbasis Kelas 4-29](#_Toc57613680)

[4.2.8. Pemodelan Berbasis Perilaku 4-39](#_Toc57613681)

[DAFTAR PUSTAKA](#_Toc57613682)

# DAFTAR TABEL

[Tabel 2‑1. Daftar Penelitian Terdahulu 2-7](#_Toc57613581)

[Tabel 3‑1. Alur Penyelesaian Tugas Akhir 3-1](#_Toc57613582)

[Tabel 3‑2. Analisis manfaat implementasi dan perbandingan JSON Adapter Moshi dan Gson. 3-5](#_Toc57613583)

[Tabel 3‑3. Rancangan Analisis 3-5](#_Toc57613584)

[Tabel 3‑4. Penjelasan Rancangan Analisis 3-6](#_Toc57613585)

[Tabel 4‑1. Daftar Stakeholder 4-1](#_Toc57613586)

[Tabel 4‑2. Analisis Fungsional 4-2](#_Toc57613587)

[Tabel 4‑3. Daftar Pengguna 4-2](#_Toc57613588)

[Tabel 4‑4. Deskripsi hasil output analisis ranah 4-3](#_Toc57613589)

[Tabel 4‑5. Deskripsi Use Case 4-5](#_Toc57613590)

[Tabel 4‑6. Skenario Use Case menampilkan informasi workshop berupa foto atau video 4-6](#_Toc57613591)

[Tabel 4‑7. Skenario Use Case menginput pesan berupa teks untuk keluhan ke Customer Service 4-7](#_Toc57613592)

[Tabel 4‑8. Skenario Use Case menampilkan artikel fotografi berupa foto atau video 4-7](#_Toc57613593)

[Tabel 4‑9. Skenario Use Case menampilkan iklan layanan jasa dokumentasi berupa foto atau video 4-8](#_Toc57613594)

[Tabel 4‑10. Skenario Use Case Menampilkan portofolio gallery 4-9](#_Toc57613595)

[Tabel 4‑11. Skenario Use Case menginput registrasi order jasa dokumentasi 4-9](#_Toc57613596)

[Tabel 4‑12. Skenario Use Case Memposting video atau foto workshop 4-10](#_Toc57613597)

[Tabel 4‑13. Skenario Use Case Menginput jawaban berupa teks untuk menjawab keluhan dari 4-10](#_Toc57613598)

[Tabel 4‑14. Skenario Use Case Memposting video atau foto artikel fotografi 4-11](#_Toc57613599)

[Tabel 4‑15. Skenario Use Case Memposting video atau foto iklan layanan jasa dokumentasi 4-12](#_Toc57613600)

[Tabel 4‑16. Skenario Use Case Memposting foto untuk portofolio gallery 4-12](#_Toc57613601)

[Tabel 4‑17. Skenario Use Case Menampilkan data registrasi order jasa dokumentasi 4-13](#_Toc57613602)

[Tabel 4‑18. Objek Data 4-26](#_Toc57613603)

[Tabel 4‑19. Atribut data 4-27](#_Toc57613604)

[Tabel 4‑20. Relasi Objek Data 4-28](#_Toc57613605)

[Tabel 4‑21. Kelas - Kelas Analisis 4-29](#_Toc57613606)

[Tabel 4‑22. Atribut Kelas 4-32](#_Toc57613607)

[Tabel 4‑23. Perilaku Kelas 4-34](#_Toc57613608)

# DAFTAR GAMBAR

[Gambar 1‑1. Metodologi Tugas Akhir 1-4](#_Toc57613506)

[Gambar 2‑1. Pertumbuhan Aplikasi Mobile di setiap kategori di AS tahun 2009-2010 2-2](#_Toc57613507)

[Gambar 2‑2. Category aplikasi yang sering digunakan 2-2](#_Toc57613508)

[Gambar 2‑3. Web Service Flow 2-4](#_Toc57613509)

[Gambar 2‑4. Tahap pengembangan perangkat lunak pada waterfall model 2-6](#_Toc57613510)

[Gambar 3‑1. Fishbone Diagram Permasalahan 3-4](#_Toc57613511)

[Gambar 3‑2. Fishbone Diagram Solusi. 3-7](#_Toc57613512)

[Gambar 3‑3. Struktur Organisasi Dinasti Motekar Academy 3-10](file:///D:\Document\Work\Kuliah\1.TA\My%20TA\All\TAv7.docx#_Toc57613513)

[Gambar 4‑1. Analisis Ranah 4-3](#_Toc57613514)

[Gambar 4‑2. Elemen Model Analisis 4-4](#_Toc57613515)

[Gambar 4‑3. Use Case Diagram 4-5](#_Toc57613516)

[Gambar 4‑4. Swimlane Diagram pada UC-01 4-14](#_Toc57613517)

[Gambar 4‑5. Diagram Swimlane pada UC-02 4-15](#_Toc57613518)

[Gambar 4‑6. Swimlane Diagram pada UC-03 4-16](#_Toc57613519)

[Gambar 4‑7. Swimlane Diagram pada UC-04 4-17](#_Toc57613520)

[Gambar 4‑8. Swimlane Diagram pada UC-05 4-18](#_Toc57613521)

[Gambar 4‑9. Swimlane Diagram pada UC-06 4-19](#_Toc57613522)

[Gambar 4‑10. Swimlane Diagram pada UC-07 4-20](#_Toc57613523)

[Gambar 4‑11. Swimlane Diagram pada UC-08 4-21](#_Toc57613524)

[Gambar 4‑12. Swimlane Diagram pada UC-09 4-22](#_Toc57613525)

[Gambar 4‑13. Swimlane Diagram pada UC-10 4-23](#_Toc57613526)

[Gambar 4‑14. Swimlane Diagram pada UC-11 4-24](#_Toc57613527)

[Gambar 4‑15. Swimlane Diagram pada UC-12 4-25](#_Toc57613528)

[Gambar 4‑16. Diagram Kelas 4-39](#_Toc57613529)

[Gambar 4‑17. Diagram Sekuen Menampilkan infromasi workshop berupa Foto atau Video 4-40](#_Toc57613530)

[Gambar 4‑18. Diagram sekuen menginput pesan berupa teks untuk keluhan ke customer service 4-40](#_Toc57613531)

[Gambar 4‑19. Diagram Sekuen Menampilkan artikel fotografi berupa foto atau video 4-41](#_Toc57613532)

[Gambar 4‑20. Diagram Sekuen Menampilkan iklan layanan jasa dokumentasi berupa 4-41](#_Toc57613533)

[Gambar 4‑21. Diagram Sekuen Menampilkan portofolio gallery 4-42](#_Toc57613534)

[Gambar 4‑22. Diagram Sekuen Menginput registrasi order jasa dokumentasi 4-42](#_Toc57613535)

[Gambar 4‑23. Diagram Sekuen Memposting video atau foto workshop 4-43](#_Toc57613536)

[Gambar 4‑24. Diagram Sekuen Menginput jawaban berupa teks untuk menjawab keluhan dari 4-43](#_Toc57613537)

[Gambar 4‑25. Diagram Sekuen Memposting video atau foto artikel fotografi 4-44](#_Toc57613538)

[Gambar 4‑26. Diagram sekuen Memposting video atau foto iklan layanan jasa dokumentasi 4-44](#_Toc57613539)

[Gambar 4‑27. Diagram Sekuen Memposting foto untuk portofolio gallery 4-45](#_Toc57613540)

# PENDAHULUAN

Bab ini memberikan penjelasan umum mengenai tugas akhir yang penulis lakukan. Penjelasan tersebut meliputi latar belakang masalah, identifikasi masalah, tujuan, ruang lingkup, metodologi penyelesaian tugas akhir, dan sistematika penulisan.

## Latar Belakang

Selama lima tahun terakhir, perangkat *mobile*, seperti *smartphone*, dan tablet telah jauh lebih populer daripada perangkat *desk-based* tradisional seperti komputer pribadi dan laptop. Sejak iOS telah dirilis pada iPhone pada 2007, dan Android telah dirilis pada berbagai *smartphone* dan tablet pada tahun 2008, Sistem Operasi paling populer yang berjalan pada perangkat komputasi telah menjadi sistem operasi *mobile*. Dengan demikian, semakin banyak aplikasi yang berjalan pada Sistem Operasi *mobile* daripada Sistem Operasi *desktop*. [LIM14]

Pembangunan aplikasi berbasis android sering kali dilakukan setelah adanya sebuah *website* yang menampung data yang dibutuhkan. Sehingga dalam pembangunannya data yang dimuat selalu mengacu pada sebuah *website*. Dalam hal ini jika aplikasi dibuat dengan membuat skema *database* dari awal maka membuat pekerjaan bertambah juga kurang efektif penggunaannya.

*Web Service* adalah sistem perangkat lunak yang dirancang untuk mendukung interaksi mesin-ke-mesin yang interoperable melalui jaringan. Memiliki antarmuka yang dijelaskan dalam format yang dapat diproses mesin (khususnya WSDL). Sistem lain berinteraksi dengan layanan *Web* dengan cara yang ditentukan oleh deskripsinya menggunakan pesan SOAP, biasanya disampaikan menggunakan HTTP dengan serialisasi XML bersama dengan standar lain yang berhubungan dengan *Web*. [W3C04]

*REST Web* adalah subset dari WWW (berdasarkan HTTP) di mana agen menyediakan semantik antarmuka yang seragam - pada dasarnya membuat, mengambil, memperbarui dan menghapus - bukan antarmuka yang sewenang-wenang atau khusus aplikasi, dan memanipulasi sumber daya hanya dengan pertukaran dari representasi. Interaksi REST adalah "tanpa kewarganegaraan" dalam arti bahwa makna pesan tidak tergantung pada keadaan percakapan. [W3C04]

Pada arsitektur REST, *REST server* menyediakan resources yang direpresentasikan dalam bentuk format teks, JSON atau XML. JSON singkatan dari *JavaScript Object Notation* adalah suatu format ringkas pertukaran data yang sering digunakan untuk mentransmisikan data terstruktur melalui suatu koneksi jaringan pada suatu proses yang disebut serialisasi dan dapat direpresentasikan oleh berbagai bahsa pemrograman. *Resource* yang berbentuk format JSON yang disediakan *REST server* dapat dimanfaatkan oleh aplikasi android dengan library Retrofit.

Retrofit adalah *REST Client Library* (*Helper Library*) yang digunakan di Android dan Java untuk membuat permintaan HTTP dan juga untuk memproses respons HTTP dari REST API yang dibuat oleh Square, Retrofit juga dapat digunakan untuk menerima struktur data selain JSON, misalnya *SimpleXML* dan *Jackson*. [OSA17]

Memparsing Data ke bentuk JSON dalam pembangunan aplikasi berbasis android, sudah banyak JSON adapter yang dapat membantu memparsing data seperti Moshi dan Google Gson.

Gson adalah *Java library* yang dapat digunakan untuk mengubah *Java Objects* menjadi representasi JSON. Itu juga dapat digunakan untuk mengonversi string JSON ke objek Java yang setara. Gson dapat bekerja dengan objek Java yang arbitrer termasuk objek yang sudah ada sebelumnya yang tidak Anda miliki kode sumbernya. [GOO08]

Moshi adalah Modern *JSON Library* untuk Android dan Java. Dan itu bisa dengan mudah membuat serialisasi objek Java seperti JSON. [SQU15]

Perbandingan Kinerja Gson dan Moshi perlu diketahui agar kedepannya lebih dapat dipahami kinerja mana yang lebih cepat, efisien dan efektif dalam memparsing data ke dalam bentuk JSON yang nantinya akan digunakan dalam membangun aplikasi berbasis android menggunakan Retrofit yang akan digunakan untuk membuat aplikasi Publikasi informasi dan kegiatan di Studio Chendra PT Dinasti Motekar Grup. Metode Pembangunan Aplikasi ini mengadopsi metode pembangunan perangkat lunak *SDLC Waterfall Model*, dengan menerapkan metode yang ada maka diharapkan aplikasi Berbasis Android ini dapat dibangun dengan sempurna dan dapat dimanfaatkan untuk perbandingan penggunaan Gson dan Moshi juga untuk membuat aplikasi publikasi informasi dan kegiatan di Studio Chendra PT Dinasti Motekar Grup.

## Identifikasi Masalah

Berdasarkan latarbelakang yang telah dipaparkan sebelumnya, maka permasalahan yang dimunculkan pada tugas akhir ini adalah :

1. Bagaimana Mengimplementasikan JSON Adapter Gson dan Moshi dalam Pembangunan Aplikasi Berbasis Android?
2. Bagaimana Cara membandingkan JSON Adapter Gson dan Moshi dalam Pembangunan Aplikasi Berbasis Android?
3. Bagaimana Hasil Perbandingan JSON adapter Gson dan Moshi yang digunakan dalam Pembangunan Aplikasi Berbasis Android?

## Tujuan Tugas Akhir

Tujuan tugas akhir ini adalah :

1. Mengetahui cara mengimplementasikan JSON adapter Gson dan Moshi dalam pembangunan Aplikasi Berbasis Android
2. Mengetahui cara membandingkan JSON adapter Gson dan Moshi dalam pembangunan Aplikasi Berbasis Android.
3. Mengetahui Hasil perbandingan dari JSON adapter Gson dan Moshi yang digunakan dalam Pembangunan Aplikasi Berbasis Android.

## Lingkup Tugas Akhir

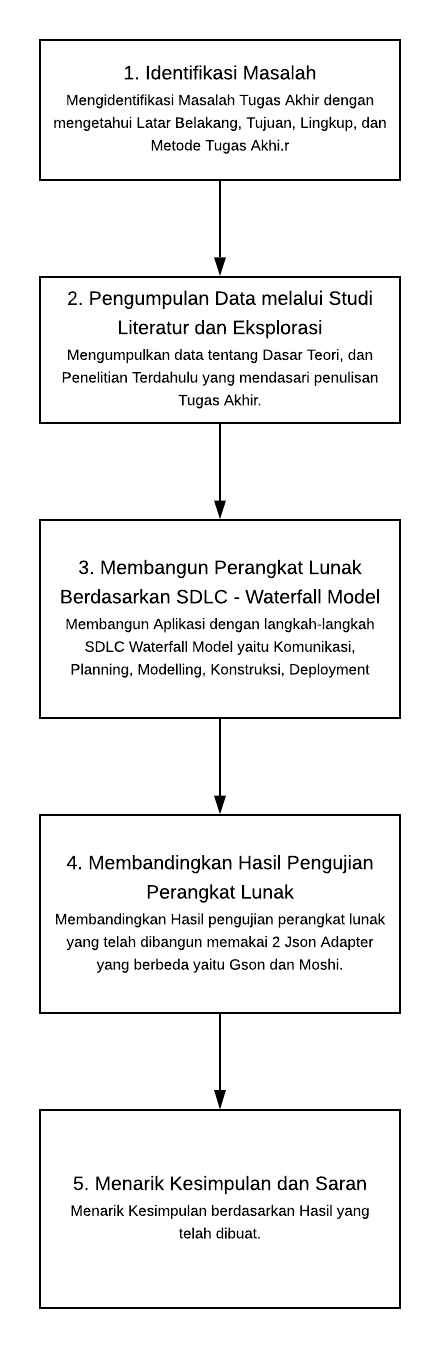
Penyelesaian Tugas Akhir dibatasi sebagai berikut :

1. Kasus yang dijadikan fokus penelitian adalah perbandingan dan pengimplementasian JSON adapter Aplikasi berbasis Android untuk publikasi informasi dan kegiatan di Studio Chendra PT Dinasti Motekar Grup.
2. Penelitian dibatasi hingga aplikasi telah selesai dibangun dan diketahui hasil Perbandingan JSON adapter Moshi dan Gson.
3. Metodologi yang digunakan adalah *Software Development Life Cycle* (SDLC) – *Waterfall Model*
4. Tidak melakukan pengukuran kepuasan pada aplikasi yang telah dibuat.

## Metodologi Tugas Akhir

Berikut langkah penyelesaian tugas akhir :

1. Melakukan Identifikasi Masalah
2. Melakukan pengumpulan data melalui studi literatur dan Eksplorasi
3. Membangun Perangkat Lunak berdasarkan metode *SDLC – Waterfall Model* yang langkah – langkahnya sebagai berikut :
   1. Komunikasi
   2. Planning
   3. Pemodelan
   4. Kontruksi Perangkat Lunak
   5. Deployment
4. Membandingkan Hasil Pengujian Perangkat Lunak
5. Menarik kesimpulan dan saran



**Gambar 1‑1. Metodologi Tugas Akhir**

## Sistematika Penulisan Tugas Akhir

Buku Tugas Akhir ditulis dengan mengikuti sistematika sebagaiberikut :

**Bab 1 : Pendahuluan**

Bab ini memberikan penjelasan umum mengenai tugas akhir yang penulis lakukan. Penjelasan tersebut meliputi latar belakang masalah, identifikasi masalah, tujuan, ruang lingkup, metodologi penyelesaian tugas akhir, dan sistematika penulisan.

**Bab 2 : Landasan Teori**

Dalam bab ini membahas beberapa teori yang mendasari penulisan dari tugas akhir ini. Teori umum yang digunakan meliputi teori pengembangan perangkat lunak, dan beberapa teori lainnya yang mendasari penyusunan tugas akhir ini.

**Bab 3 : Skema Penelitian**

Bab ini berisi kerangka pengerjaan tugas akhir, kerangka pemikiran teoritis,skema analisis yang dilakukan, penjelasan skema analisis yang telah dibuat, dan profil studi kasus.

**Bab 4 : Analisis dan Perancangan**

Bab ini menjelaskan proses awal pembangunan perangkat lunak yang terdiri dari analisis dan perancangan. Proses analisis terdiri dari pendefinisian perangkat lunak, pendefinisian kebutuhan yang harus dipenuhi perangkat lunak, dan analisis terhadap kebutuhan dengan menggunakan pemodelan spesifikasi kebutuhan dan pemodelan kebutuhan. Sedangkan proses perancangan perangkat lunak terdiri dari perancangan arsitektur, perancangan antarmuka, dan perancangan data.

**Bab 5 : Implementasi dan Pengujian**

Bab ini tahapan inti dan tahapan akhir dari pembangunan perangkat lunak, yaitu konstruksi perangkat lunak (coding) berdasarkan analisis dan perancangan yang telah dibuat serta pengujian dari hasil implementasi.

**Bab 6 : Kesimpulan dan Saran**

Bab ini menjelaskan mengenai kesimpulan yang didapatkan dari hasil penelitian yang dilakukan serta saran yang diberikan oleh penulis untuk pengembangan penelitian selanjutnya.

# LANDASAN TEORI

Dalam bab ini membahas beberapa teori yang mendasari penulisan dari tugas akhir ini. Teori umum yang digunakan meliputi teori pengembangan perangkat lunak, dan beberapa teori lainnya yang mendasari penyusunan tugas akhir ini.

## Aplikasi Mobile

Aplikasi *Mobile* terdiri dari perangkat lunak / kumpulan sebuah program yang berjalan di perangkat seluler dan bekerja dengan tugas tertentu untuk pengguna. Aplikasi *Mobile* merupakan hal yang baru dengan kemampuan *developing* *segment* yang cepat pada *global information* dan *comunication technology*. Aplikasi *Mobile* sangat mudah, *user friendly*, murah, dapat diunduh dan dijalankan di ponsel seperti apapun termasuk ponsel yang murah bahkan sampai *entry level*. Aplikasi *Mobile* memiliki kegunaan yang luas seperti *calling, messaging, browsing, chatting, social network communication, audio, video, games* dll. [IJE10]

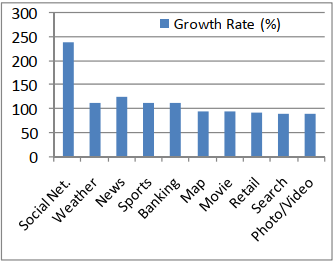
Sejumlah besar aplikasi *mobile* biasanya sudah diinstal sebelumnya di ponsel dan yang lainnya dapat diunduh oleh pengguna di internet dan menginstalnya di ponsel. Jumlah aplikasi *mobile* yang ada di market biasanya dibuat oleh berbagai macam *mobile application developer*, *publisher* dan *provider*. Dari segi teknis melihat berbagai aplikasi *mobile* yang dapat dijalankan di berbagai platform terkelola seperti iPhone, BlackBerry, Android, Symbian, Windows; juga beberapa mesin virtual seperti Java / J2ME, BREW, Flash Light, Silverlight. [IJE10]

Menurut area aplikasi, ada perbedaan kategori Aplikasi *Mobile* :

1. Komunikasi: Browsing Internet, email IMklien, Jejaring Sosial
2. Permainan: Teka-Teki / Strategi, Kartu / Kasino,Aksi petualangan
3. Multimedia: Penampil Grafik / Gambar, Penampil presentasi, Pemutar Video, Audio
4. Produktivitas: Kalender, Kalkulator, Buku Harian, Notepad / Memo / Pengolah Kata, Spreadsheet
5. Perjalanan: Panduan kota, Konverter mata uang, Penerjemah, GPS / Peta, Rencana Perjalanan / Jadwal, Cuaca
6. Utilitas: Manajer profil, Layar / Layar Idle penghemat, Buku alamat, Manajer tugas, Manajer panggilan, Manajer file [IJE10]

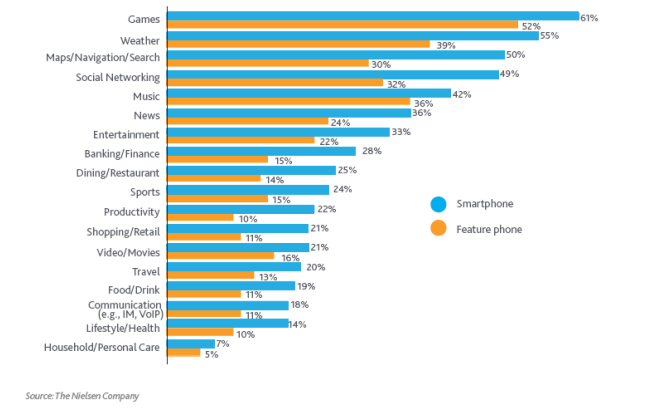
### Perkembangan Aplikasi Mobile

Menurut sebuah studi oleh MobiLens, jumlah pengguna aplikasi seluler di AS tumbuh sebesar 28% antara April 2009 dan April 2010. Berikut adalah tabel pertumbuhan tingkat semua kategori aplikasi dalam satu tahun hingga April 2010. [IJE10]



**Gambar 2‑1. Pertumbuhan Aplikasi Mobile di setiap kategori di AS tahun 2009-2010**

Perusahaan Nielsen, merilis survei pada bulan September 2010 telah menerbitkan laporan aplikasi *mobile* yang sering di download. Dalam survei itu mereka menunjukkan beberapa data statistik penting dari aplikasi *mobile* 30 hari. Menurut survei, gambar berikut mewakili tingkat penggunaan berbagai kategori aplikasi *mobile*. [IJE10]



**Gambar 2‑2. Category aplikasi yang sering digunakan**

Dari data-data diatas yang telah dirilis menunjukan bahwa pertumbuhan pengguna aplikasi *mobile* paling banyak dilakukan di media sosial dan kedua terbanyak untuk berita, ini menunjukan bahwa pengguna lebih sering mencari dan membagikan informasi. Bahkan dari data yang dirilis Nielsen penggunaan *Smartphone* untuk berbagai macam kegiatan sangat tinggi, daripada tipe-tipe perangkat seluler yang lainnya. Dan biasanya banyaknya *smartphone* di dukung oleh sistem Operasi Android.

## Sistem Operasi Android

Android adalah platform perangkat lunak dan sistem operasi untuk perangkat mobile, dibuat dari kernel Linux, dan dikembangkan oleh Google dan kemudian dijadikan *Open Source*. membuat pengembang memungkinkan untuk menulis kode yang dikelola bahasa Java, mengontrol perangkat melalui pustaka Java yang dikembangkan Google. Android tersedia secara *Open Source* yang dapat diunduh secara gratis dan menyediakan berbagai macam hal seperti sistem operasi, *middleware* dan aplikasi utama berbasis Linux dan Java. Google membeli pengembang Android pada tahun 2005, dan Android diresmikan pada tahun 2007. Google merilis kode Android sebagai open-source di bawah Lisensi Apache. Android memiliki banyak pengembang aplikasi di seluruh dunia. [INT15]

### Aplikasi Android

Aplikasi, yang memperluas fungsionalitas perangkat, dibangun menggunakan *Software Development kit* (SDK) dan, seringkali, menggunakan bahasa pemrograman Java yang memiliki akses lengkap ke API Android. Java dapat digabungkan dengan C / C ++, bersama dengan pilihan *runtime* *non-default* memungkinkan dukungan C ++ yang lebih baik, bahasa pemogram Go juga didukung sejak versi 1.4, yang juga dapat digunakan secara eksklusif meskipun dengan API Android yang terbatas. SDK mencakup aset perangkat alat pengembangan yang komprehensif, termasuk *debugger, software libraries, handset emulator based* di QEMU, dokumentasi, sample code, dan tutorial. Awalnya, Google mendukung *integrated development environment* (IDE) dengan Eclipse menggunakan *Android Development Tools* (ADT) *Plugin*, namun pada bulan Desember 2014, Google merilis Android Studio, berbasis IntelliJIDEA, sebagai IDE utamanya untuk aplikasi pengembangan Android. Alat pengembangan lainnya tersedia, termasuk *native development kit* (NDK) untuk aplikasi atau ekstensi di C atau C ++, Google App Inventor, lingkungan visual untuk pemrogram pemula, dan berbagai *cross platform mobile web applications frameworks*. Pada Januari 2014, Google meluncurkan *framework* berbasis Apache Cordova untuk porting Aplikasi web Chrome HTML 5 ke Android, dibungkus dalam *native application shell*. [INT15]

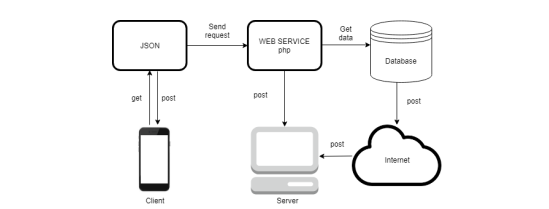
## Web Service & REST API

*Web Service* adalah Standarisasi distribusi berbagai macam data dan proses dari berbagai macam aplikasi dan *services* yang ada di internet. Aplikasi di internet biasanya di bangun dengan berbagai bahasa pemograman yang berbeda tergantung sistem operasi dan perangkat. Pemanfaatan *Web Service* adalah untuk mengatasi masalah pengintegrasian data dan proses saat pertukaran informasi antar platform, serta tingkat keamanan data dapat ditangani dengan baik. Arsitektur yang menjadi dasar *web service* adalah REST dan SOAP [INT19]

REST API didefinisikan sebagai alat arsitektur yang dirancang pada layanan web yang berfokus pada sumber daya sistem seperti mentransfer & meminta data menggunakan http. REST telah diterima secara luas oleh masyarakat karena simpel digunakan dalam paradigma *cloud seluler*. Layanan web REST bekerja dengan mengidentifikasi URL lalu memodifikasinya dengan perintah GET, POST, PUT atau DELETE yang dapat direpresentasikan pada beberapa bahasa pemrograman seperti XML atau JSON. Berbagai aplikasi dibuat dengan memanfaatkan REST API seperti catatan kesehatan pasien, indeks kesejahteraan manusia badan perlindungan lingkungan AS (HWBI) yang mencirikan kesejahteraan manusia menggunakan data sosial, ekonomi dan ekologi, *Smartmeter* air yang secara otomatis dapat mengumpulkan air penggunaan terperinci, dan penambangan aturan asosiasi layanan yang digunakan untuk menemukan pola yang sering dari data besar. Beberapa penelitian menunjukkan bahwa SOAP *based Service* memang sangat berat untuk dijalankan pada perangkat seluler sehingga *Restful Web Services* lebih cocok untuk aplikasi *mobile* bahkan menunjukkan bahwa REST dengan JSON telah menunjukkan kinerja yang lebih baik dibandingkan dengan SOAP. [INT19]

### Metodologi REST API

*Web services* dibangun dengan menggunakan bahasa pemrpograman PHP dan MySQL sebagai *database* nya menyimpan data dalam aplikasi di server internet. REST API diperlukan untuk mengelola akses ke data diweb dan tindakan ke aplikasi seluler android. [INT19]



**Gambar 2‑3. Web Service Flow**

## Retrofit Library

*Library* merupakan kumpulan *resource nonvolatile* yang digunakan oleh program computer dan sering digunakan untuk mengembangkan *software*. *Library* tersebut dapat berisikonfigurasi data, dokumentasi, data penunjang,maupun class. [CHE15]

Retrofit merupakan REST client library yang aman untuk android dan java. Dibuat oleh Square. Retrofit menyediakan cara yang aman untuk autentikasi dan interaksi dengan berbagai API lainnya. Sehingga memungkinkan pengiriman permintaan jaringan dengan OkHttp. Retrofit mengambil data JSON atau XML dari web API dan saat data diterima akan langsung diubah ke *Plain Old Java Object* (POJO). Sehingga harus ditentukan setiap *class* yang akan dipakai saat respon diterima. [COD16]

Retrofit juga bekerja dengan REST API menggunakan implementasi java interface, yang dapat dihasilkan dengan bantuan Rest Adapter. Implementasi dalam hal ini bertindak sebagai local instance dari layanan dan setiap panggilan sesuai dengan permintaan HTTP. [MAS15]

## Gson & Moshi

Gson adalah Java *library* yang dapat digunakan untuk mengubah Java *Objects* menjadi representasi JSON. Itu juga dapat digunakan untuk mengonversi string JSON ke objek Java yang setara. Gson dapat bekerja dengan objek Java yang arbitrer termasuk objek yang sudah ada sebelumnya yang tidak miliki kode sumbernya. [GOO08]

Moshi adalah Modern JSON *Library* untuk Android dan Java. Dan itu bisa dengan mudah membuat serialisasi objek Java seperti JSON [SQU15]

## Konsep MVC

*Model-View-Controller* atau MVC adalah sebuah metode untuk membuat sebuah aplikasi dengan memisahkan data (*Model*) dari tampilan (*View*) dan cara bagaimana memprosesnya (*Controller*). Dalam implementasi kebanyakan *framework* dalam aplikasi *web* adalah berbasis arsitektur MVC (*Model-View-Control*). [ROS11]. Definisi teknis dari arsitektur MVC dibagi menjadi  
tiga lapisan menurut Burbeck [BUR92] :

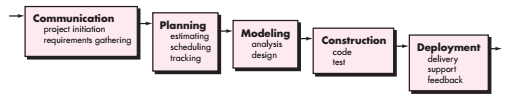
1. *Model*, digunakan untuk mengelola informasi dan memberitahu pengamat ketika ada  
   perubahan informasi. Hanya *model* yang mengandung data dan fungsi yang berhubungan  
   dengan pemrosesan data. Sebuah *model* meringkas lebih dari sekedar data dan fungsi yang  
   beroperasi di dalamnya. Pendekatan *model* yang digunakan untuk komputer *model* atau  
   abstraksi dari beberapa proses dunia nyata. Hal ini tidak hanya menangkap keadaan proses  
   atau sistem, tetapi bagaimana sistem bekerja. Sebagai contoh, programmer dapat menentukan *model* yang menjembatani komputasi *back-end* dengan *front-end* GUI (*graphical user interface*).
2. *View*, bertanggung jawab untuk pemetaan grafis ke sebuah perangkat. *View* biasanya memiliki hubungan satu banding satu dengan sebuah permukaan layar dan tahu bagaimana untuk membuatnya. *View* melekat pada model dan me-*render* isinya ke permukaan layar. Selain itu, ketika model berubah, *view* secara otomatis menggambar ulang bagian layar yang terkena perubahan untuk menunjukkan perubahan tersebut. Terdapat kemungkinan beberapa *view* pada *model* yang sama dan masing-masing *view* tersebut dapat me-*render* isi *model* untuk permukaan tampilan yang berbeda.
3. Controller, menerima input dari pengguna dan menginstruksikan model dan view untuk  
   melakukan aksi berdasarkan masukan tersebut. Sehingga, controller bertanggung jawab untuk pemetaan aksi pengguna akhir terhadap respon aplikasi. Sebagai contoh, ketika pengguna mengklik tombol atau memilih item menu, controller bertanggung jawab untuk menentukan bagaimana aplikasi seharusnya merespon.

## Software Development Life Cycle – Waterfall Model

*Waterfall Model* adalah model yang menunjukkan sistematika, pendekatan sekuensial untuk  
pengembangan perangkat lunak yang dimulai dengan spesifikasi kebutuhan pelanggan dan  
berkembang melalui perencanaan, pemodelan, konstruksi dan *deployment*, yang berpuncak pada  
dukungan berkelanjutan dari perangkat lunak yang sudah selesai. [PRE10]

### Tahapan – tahapan Waterfall Model

Di dalam tahapan *Waterfall Model* terdapat lima tahapan, yaitu tahap *communication*, *planning*, tahap *modeling*, tahap *construction*, dan tahap *deployment* seperti yang digambarkan pada Gambar 2-4. [PRE10]



**Gambar 2‑4. Tahap pengembangan perangkat lunak pada waterfall model**

#### **Communication**

Sebelum persyaratan *customer* dapat dianalisis, dimodelkan, atau ditentukan, mereka harus  
dikumpulkan melalui aktivitas komunikasi. *Customer* memiliki masalah yang mungkin terjadi dapat  
menerima solusi berbasis komputer. Permintaan *customer* tersebut ditanggapi untuk memberikan  
bantuan. Tetapi saat komunikasi telah dimulai, jalan dari komunikasi untuk membentuk suatu  
pemahaman sering penuh dengan lubang (kesulitan). Komunikasi yang efektif (di antara rekan-rekan  
teknis, dengan *customer* dan *stakeholder* lainnya, dan dengan manajer proyek) adalah salah satu  
kegiatan yang paling menantang yang akan dihadapi. Aktivitas komunikasi membantu untuk  
menentukan tujuan dan sasaran Anda secara keseluruhan (subjek, tapi berubah seiring berjalannya  
waktu). [PRE10]

#### **Planning**

Memahami tujuan dan sasaran pada tahap komunikasi tidak sama dengan mendefinisikan  
rencana selanjutnya. Kegiatan perencanaan mencakup seperangkat praktik manajemen dan teknis yang  
memungkinkan tim perangkat lunak untuk menentukan peta jalan ketika suatu proses bergerak menuju  
tujuan strategis dan tujuan taktis. Tidak ada cara mudah untuk menentukan masalah teknis yang tak  
terduga yang akan dihadapi. Namun, tim perangkat lunak yang baik harus merencanakan  
pendekatannya. Pada banyak proyek, perencanaan berlebihan memakan banyak waktu dan tidak  
membuahkan hasil (terlalu banyak hal berubah) [PRE10]

#### **Modelling**

Memahami tujuan dan sasaran pada tahap komunikasi tidak sama dengan mendefinisikan  
rencana selanjutnya. Kegiatan perencanaan mencakup seperangkat praktik manajemen dan teknis yang  
memungkinkan tim perangkat lunak untuk menentukan peta jalan ketika suatu proses bergerak menuju  
tujuan strategis dan tujuan taktis. Tidak ada cara mudah untuk menentukan masalah teknis yang tak  
terduga yang akan dihadapi. Namun, tim perangkat lunak yang baik harus merencanakan  
pendekatannya. Pada banyak proyek, perencanaan berlebihan memakan banyak waktu dan tidak  
membuahkan hasil (terlalu banyak hal berubah) [PRE10]

#### **Construction**

Kegiatan konstruksi mencakup seperangkat tugas pengkodean dan pengujian yang mengarah  
ke perangkat lunak operasional yang siap dikirim ke pelanggan atau pengguna akhir. Dalam pekerjaan  
rekayasa perangkat lunak modern, pengkodean dapat dilakukan dengan : pembuatan langsung kode  
sumber bahasa pemrograman (misalnya, Java), pembuatan otomatis kode sumber menggunakan  
representasi antara desain seperti komponen yang akan dibangun atau generasi otomatis kode yang  
dapat dijalankan menggunakan "bahasa pemrograman generasi keempat" (misalnya, Visual C++).  
Fokus awal pengujian adalah pada tingkat komponen, yang sering disebut pengujian unit. [PRE10].

#### **Deployment**

Kegiatan *deployment* mencakup tiga tindakan yaitu : pengiriman, dukungan, dan umpan balik.  
Setiap siklus *deployment* kepada costomer dan pengguna akhir akan ada peningkatan perangkat lunak  
dalam hal operasional yang menyediakan fungsi dan fitur yang dapat digunakan. Setiap siklus  
dukungan menyediakan dokumentasi dan *human assistance* untuk semua fungsi dan fitur yang diperkenalkan dalam semua siklus penggunaan. Setiap siklus umpan balik menyediakan tim perangkat  
lunak dengan panduan perangkat lunak untuk dapat modifikasi fungsi, fitur, dan mengetahui  
pendekatan yang diambil untuk perbaikan berikutnya. [PRE10].

## Penelitian terdahulu terkait dengan topik Tugas Akhir

Ada beberapa penelitian terdahulu yang terkait dengan tugas akhir ini yang dapat dilihat pada Tabel 2-1.

**Tabel 2‑1. Daftar Penelitian Terdahulu**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Judul** | **Peneliti** | **Hasil** |
| 1. | Rancang Bangun Aplikasi Pengelolaan Data Anggota Koordinator Olahraga Mahasiswa Universitas Pasundan Bandung | Eka Ardilah Febriyanti  [EAF19] | Aplikasi Pengelolaan Data Anggota Koordinator Olahraga |
| 2. | Perangkat Lunak Untuk Layanan Warga Berbasis Android Di Lingkungan Rukun Warga Menggunakan Konsep M-Government (Studi Kasus Rw 10 Kelurahan Sukabungah Kota Bandung) | Andri Nurul Akbar  [ANA17] | Aplikasi berbasis android di Lingkungan RW |
| 3. | Pembangunan Aplikasi Pakan Sapi Berbasis Android | Berta Erwin SLAM [SLA18] | Aplikasi android untuk membantu peternak menyusun ramsun dari bahan pakan ternak |
| 4. | Eksplorasi Python Untuk Penyesuaian  Modul Perkuliahan Dan Praktikum  Di Teknik Informatika Unpas | Nur Sofia Malikhah [NSM16] | Topik dan pembelajaran dengan bahasa python |
| 5. | Eksplorasi Python Menggunakan Framework Django (Studi Kasus : Sistem Perpustakaan Teknik Informatika UNPAS) | Anggiyasti Yaktining Putri [AYP16] | Sistem Aplikasi Perpustakaan |
| 6. | Rancang Bangun Aplikasi Pemesanan Tiket Di Kolam Renang Failda Sukanagara Cianjur Selatan | Muhamad Faisal Ilhami Akbar [MUH19] | Aplikasi Pemesanan Tiket Di Kolam Renang Failda Sukanagara Cianjur Selatan |

# SKEMA PENELITIAN

Bab ini berisi kerangka pengerjaan tugas akhir, kerangka pemikiran teoritis,skema analisis yang dilakukan, penjelasan skema analisis yang telah dibuat, dan profil studi kasus.

## Rancangan Penelitian

Rancangan penelitian ini memiliki tujuan untuk memudahkan dalam memahami secara singkat mengenai tahapan pengerjaan dan literatur yang digunakan dalam pengerjaan tugas akhir. Setiap tahapan yang dilakukan memiliki tujuan dan output. Sebagaimana dijelaskan di Tabel 3-1.

**Tabel 3‑1. Alur Penyelesaian Tugas Akhir**

| **Tahap dan hasil** | **Langkah penelitian** | **Literatur dan referensi** |
| --- | --- | --- |
| **Tahap 1** : Menentukan topik Tugas Akhir  **Hasil** : Mendapatkan Topik tugas akhir dan masalah yang telah teridentifikasi  **Kontibusi** : Berguna untuk pengumpulan data | Menentukan Topik Tugas Akhir  Mengidentifikasi Masalah |  |
| **Tahap 2 :** Studi literatur dan pengumpulan data  **Hasil** : Teori yang sudah dipahami dan kumpulan data yang didapat  **Kontribusi :** Berguna untuk tahap pengumpulan data dan spesifikasi kebutuhan | Mengumpulkan jurnal dan teori terkait  Memahami teori yang terkait  Melakukan observasi dan wawancara | 1. Wawancara,   [ACE19]   1. Perkembangan Aplikasi Android   [LIM14] [IJE10]   1. Aplikasi Android   [INT15]   1. Web Service & REST API   [W3C04][INT19]   1. Retrofit Library   [CHE15][COD16][MAS15]   1. Gson & Moshi  [GOO08][SQU15] |
| **Tahap 3 :** Menerapkan rancangan penelitian  **Hasil :** Alur penyelesaian tugas akhir  **Kontribusi** : Berguna untuk perancangan perangkat lunak | Rancangan Penelitian  1  1  Membuat Alur Penyelesaian TA  Analisis Persoalan TA  Membuat solusi TA | 1. Rancang Bangun Aplikasi Pengelolaan Data Anggota Koordinator Olahraga Mahasiswa Universitas Pasundan Bandung [EAF19] 2. Perangkat Lunak Untuk Layanan Warga Berbasis Android Di Lingkungan Rukun Warga Menggunakan Konsep M-Government (Studi Kasus Rw 10 Kelurahan Sukabungah Kota Bandung) [ANA17] 3. Pembangunan Aplikasi Pakan Sapi Berbasis Android [SLA18] 4. Eksplorasi Python Untuk Penyesuaian Modul Perkuliahan Dan Praktikum Di Teknik Informatika Unpas [NSM16] 5. Eksplorasi Python Menggunakan Framework Django (Studi Kasus : Sistem Perpustakaan Teknik Informatika UNPAS) [AYP16] |
| **Tahap 4 :** Melakukan perancangan perangkat lunak  **Hasil :** Mendapatkan spesifikasi perangkat lunak, kebutuhan fungsional dan non-fungsional  **Kontribusi :** Berguna untuk tahap implementasi (koding) dan testing. | |  | | --- | | **Analisis Kebutuhan** | | Membuat pemodelan berbasis skenario  Membuat pemodelan berbasis data  Membuat pemodelan berbasis kelas  Membuat pemodelan perilaku (Sequence Diagram |   2  2   |  | | --- | | **Perancangan** | | Membuat perancangan data  Membuat perancangan arsitektural  Membuat perancangan antarmuka | | 1. Observasi dan wawancara [ACE19] 2. Model Proses Waterfall [PRE10] 3. Rancang Bangun Aplikasi Pemesanan Tiket Di Kolam Renang Failda Sukanagara Cianjur Selatan [MUH19] |
| **Tahap 5 :** Pembuatan aplikasi,  **Hasil** **:** Aplikasi yang sudah jadi  **Kontribusi :** Berguna untuk perbandingan kinerja 2 Json adapter | Pembuatan aplikasi | 1. Model Proses Waterfall [PRE10] 2. Pembangunan Aplikasi Pakan Sapi Berbasis Android [SLA18] 3. Perangkat Lunak Untuk Layanan Warga Berbasis Android Di Lingkungan Rukun Warga Menggunakan Konsep M-Government (Studi Kasus Rw 10 Kelurahan Sukabungah Kota Bandung) [ANA17] |
| **Tahap 6 :** Perbandingan 2 Json Adapter  **Hasil : Hasil** perbandingan kinerja Gson dan Moshi  **Kontribusi :** Berguna untuk tahap pengambilan kesimpulan tugas akhir | Pengujian aplikasi  Pencatatan hasil pengujian kedua JSON adapter | 1. Unit Testing [GUR20] |
| **Tahap 7 :** Penutup  **Hasil :** Kesimpulan dan saran tugas akhir  **Kontribusi : -** | Kesimpulan dan saran |  |

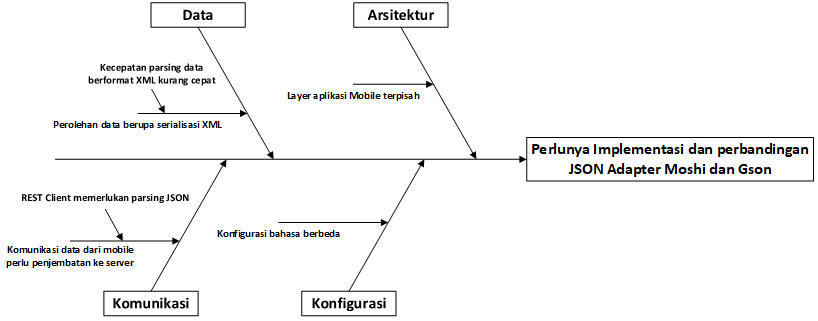
## Analisis Masalah dan Solusi Tugas Akhir

Analisis Masalah Dan Solusi TA bertujuan untuk mengetahui ketepatan solusi terhadap suatu  
permasalahan pada penelitian tugas akhir, yaitu pada Perlunya implementasi dan perbandingan *JSON Adapter* Moshi dan Gson. Dengan memberikan sebuah solusi diharapkan dapat menyelesaikan permasalahan-permasalahan yang ada. Sehingga permasalahan yang telah didefinisikan dapat diselesaikan dengan solusi yang sesuai.

Tahap analisis pada bagian ini menggunakan diagram sebab dan akibat (*cause and effect diagram*) atau biasa disebut dengan *Fishbone* Diagram. *Fishbone* diagram adalah alat untuk mengidentifikasi dan mengorganisir penyebab yang mungkin terjadi dari suatu masalah dalam format yang terstruktur [KEL95]. Terdapat 3 komponen utama pada *fishbone* diagram, yaitu sebagai berikut :

1. *Head* (Kepala), merupakan fokus masalah yang sedang diamati, dideskripsikan dalam kotak di bagian kepala pada diagram. *Head* terletak pada bagian kanan diagram.
2. *Spine* (Tulang Belakang), ditunjukkan dengan arah panah yang menunjukkan penyebab dari masalah yang terjadi pada bagian *head*.
3. *Bones* (Tulang atau Duri), menunjukkan kategori utama yang menyebabkan permasalahan terjadi. Jika terdapat bones yang lebih kecil, memperlihatkan kedalaman penyebab dari permasalahan tersebut. *Bones* yang saling terkoneksi memperlihatkan bahwa penyebab permasalahan tersebut saling berhubungan [KEL95].

Berikut merupakan Gambar 3-1 kerangka permasalahan untuk analisis Implementasi dan Perbandingan kedua *JSON Adapter* Moshi dan Gson.



**Gambar 3‑1. Fishbone Diagram Permasalahan**

Berikut ini merupakan Tabel 3-2 hasil analisis manfaat implementasi dan perbandingan *JSON Adapter* Moshi dan Gson. Terdapat masalah penyebab, dan solusi untuk menyelesaikan permasalahan.

**Tabel 3‑2. Analisis manfaat implementasi dan perbandingan JSON Adapter Moshi dan Gson.**

|  |  |  |
| --- | --- | --- |
| **Masalah** | **Penyebab** | **Solusi** |
| Perlunya Implementasi dan perbandingan JSON Adapter Moshi dan Gson. | **Data**  Perolehan data berupa serialisasi XML yang kecepatan parsing data nya kurang cepat | Melakukan Implementasi dan perbandingan JSON Adapter Moshi dan Gson.untuk Parsing JSON dari REST Client |
| **Arsitektur**  Layer data aplikasi mobile berbeda dengan web yang semuanya menyatu dalam satu app, sedangkan untuk mobile karena keterbatasannya menjadikannya terpisah - pisah |
| **Komunikasi**  Komunikasi data dari aplikasi mobile perlu penjembatan ke server  Setelah dipilih penjembatan yaitu Rest Client, lalu memerlukan parsing JSON untuk menampilkan datanya |
| **Konfigurasi**  Konfigurasi penggunaan Bahasa yang berbeda pada tiap Rest Client dan JSON adapter |

## Kerangka Pemikiran Teoritis

Pada tahap kerangka pemikiran teoritis ini akan dijelaskan megenai tahapan yang dilakukan dalam Implementasi dan Perbandingan *JSON Adapter* Moshi dan Gson. Tahap ini berisi mengenai kerangka pemikiran teoritis yang di dalamnya menunjukkan tentang komponen-komponen pembentuk produk tugas akhir yang disertai dengan kajian konsep dan teori-teori yang mendasarinya.

### Rancangan Analisis

Rancangan analisis merupakan gambaran dari proses-proses analisis yang dilakukan. Dalam penggambaran rancangan analisis, terdapat input, proses, dan output. Input merupakan data yang akan dimanfaatkan pada proses analisis yang akan menghasilkan output. Rancangan analisis dapat dilihat pada Tabel 3-3.

**Tabel 3‑3. Rancangan Analisis**

| **Input** | **Analisis** | **Output** |
| --- | --- | --- |
|  | Communication  Menganalisis perangkat lunak yang akan dibangun |  |
|  | Mendefinisikan kebutuhan  perangkat lunak  Planning |  |
|  | Modelling  Membuat rancangan dari perangkat lunak yang akan dibangun |  |

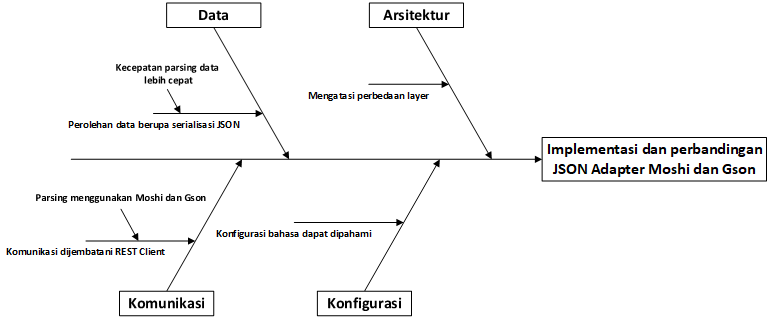
Rancangan analisis pada Table 3-3 dapat dijelaskan secara rinci pada Table 3-4

**Tabel 3‑4. Penjelasan Rancangan Analisis**

| **No.** | **Input** | **Analisis** | **Output** |
| --- | --- | --- | --- |
| 1. | Hasil wawancara dengan stakeholder | Menganalisis perangkat lunak yang akan dibangun | Lingkup perangkat lunak yang akan dibangun  Lingkup dari perangkat lunak yang akan dibangun bertujuan mendefinisikan lingkup pembangunan perangkat lunak yang akan dibangun terkait hasil wawancara yang dilakukan |
| 2. | Lingkup perangkat lunak yang akan dibangun | Mendefinisikan kebutuhan perangkat lunak | Kebutuhan perangkat lunak  Kebutuhan perangkat lunak tersebut mendeskripsikan fitur-fitur yang akan tersedia pada perangkat lunak demi menunjang penyelesaian masalah yang terjadi. |
| 3. | Kebutuhan perangkat lunak | Membuat rancangan dari perangkat lunak yang akan dibangun | Rancangan antarmuka dan pemodelan aplikasi yang akan dibangun  Pemodelan aplikasi bertujuan untuk merepresentasikan hasil analisis yang dilakukan berdasarkan kebutuhan perangkat lunak yang sudah ditentukan. Sedangkan untuk rancangan antarmuka bertujuan untuk membuat interaksi antara user dan aplikasi ketika digunakan. |

### Kerangka Berfikir Teoritis

Pada kerangfka berpikir teoritis akan dibuat solusi berdasarkan permasalahan yang telah diketahui diatas. Berdasarkan permasalahan diatas dibutuhkan solusi berupa Implementasi dan perbandingan JSON Adapter Moshi dan Gson. untuk Parsing JSON dari REST Client yang dapat mengatasi permasalahan tersebut. Seperti pada gambar di bawah ini.



**Gambar 3‑2. Fishbone Diagram Solusi.**

## Analisis Teknologi

Analisis teknologi bertujuan menentukan teknologi-teknologi yang sesuai untuk menyelesaikan permasalahan yang sesuai dengan topik tugas akhir. Teknologi-teknologi yang akan digunakan akan dipaparkan pada poin-poin berikut ini:

### Android Studio

Android Studio adalah Integrated Development Environment (IDE) resmi untuk pengembangan aplikasi Android, dibuat berdasarkan IntelliJ IDEA . Selain editor kode dan alat pengembang IntelliJ yang kuat, Android Studio menawarkan lebih banyak fitur yang meningkatkan produktivitas saat membuat aplikasi Android. [GOO20]

Alasan menggunakan Android Studio untuk membuat aplikasi mobile adalah karena android  
merupakan sistem operasi yang *open source*. Kode dengan sumber terbuka atau open source dan  
terdapat lisensi perizinan pada Android memungkinkan perangkat lunak untuk memodifikasi secara  
bebas dan didistribusikan oleh para pembuat perangkat atau pengembang aplikasi.

#### **Fitur – fitur Android Studio**

* Sistem *build* berbasis *Gradle* yang fleksibel
* *Emulator* yang cepat dan kaya fitur
* Lingkungan terpadu yang dapat mengembangkan untuk semua perangkat Android
* Membuat perubahan dan pergantian resource tanpa harus me *restart* app
* Template kode dan integrasi GitHub untuk membantu Anda membuat fitur aplikasi umum dan mengimpor kode sampel
* Alat dan kerangka pengujian yang ekstensif
* Alat Lint untuk mengetahui kinerja, kegunaan, kompatibilitas versi, dan masalah lainnya
* Dukungan C ++ dan NDK
* Dukungan bawaan untuk Google Cloud Platform , yang memudahkan integrasi Google Cloud *Messaging* dan App Engine [GOO20]

### Retrofit

Retrofit merupakan REST *client library* yang aman untuk android dan java. Dibuat oleh Square. Retrofit menyediakan cara yang aman untuk autentikasi dan interaksi dengan berbagai API lainnya. Sehingga memungkinkan pengiriman permintaan jaringan dengan OkHttp. Retrofit mengambil data JSON atau XML dari *web* API dan saat data diterima akan langsung diubah ke Plain Old Java Object (POJO). Sehingga harus ditentukan setiap *class* yang akan dipakai saat *response* diterima. [COD16]

Retrofit juga bekerja dengan REST API menggunakan implementasi Java interface, yang dapat dihasilkan dengan bantuan Rest Adapter. Implementasi dalam hal ini bertindak sebagai local instance dari layanan dan setiap panggilan sesuai dengan permintaan HTTP. [MAS15]

Alasan menggunakan retrofit adalah karena kemudahan dalam interaksi dengan API dan  
fleksibilitasnya yang memudahkan untuk membuat *custom service* sesuai yang kita inginkan.

### JSON

JSON singkatan dari *JavaScript Object Notation* adalah format pertukaran data (*lightweight data inter change format*), mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemprograman JavaScript, Standar ECMA-262 Edisi ke-3 –Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemprograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran-data. [JSO20]

Alasan utama menggunakan JSON (*Javascript Object Notation*) adalah karena JSON format pertukaran datanya yang ringan dan juga banyak didukung oleh bahasa pemrograman. Maka dari itu, penulis menggunakan JSON sebagai format pertukaran datanya.

### Gson & Moshi

Gson adalah *Java library* yang dapat digunakan untuk mengubah Java Objects menjadi representasi JSON. Itu juga dapat digunakan untuk mengonversi string JSON ke objek Java yang setara. Gson dapat bekerja dengan objek Java yang arbitrer termasuk objek yang sudah ada sebelumnya yang tidak Anda miliki kode sumbernya. [GOO08]

Moshi adalah Modern JSON Library untuk Android dan Java. Dan itu bisa dengan mudah membuat serialisasi objek Java seperti JSON [SQU15]

Alasan uatama menggunakan Gson dan moshi adalah karena penulis ingin membandingkan kinerja dari ke-2 JSON *Library* atau sering dipanggil JSON *adapter*, sehingga nanti dapat diketahui JSON *Adapter* mana yang lebih baik dari kinerjanya.

### Web Service

*Web service* merupakan sebuah perangkat lunak yang digunakan sebagai jembatan sehingga memungkinkan berbagai sistem dapat saling berkomunikasi tanpa terpengaruh dengan perbedaan platform.

Pada Implementasi dan perbandingan JSON Adapter Moshi dan Gson terdapat dua platform yang akan menggunakan *web service* untuk melakukan pertukaran data dengan *database* yang berada di sistem yaitu aplikasi *Web* yang telah dibangun dan Android yang akan dibangun. Dengan adanya *web service* kedua platform tersebut hanya perlu memanggil fungsi-fungsi yang sudah disediakan oleh *web service.*

## Profile Penelitian

Profil penelitian menjelaskan tentang objek penelitian, tempat dimana peneliti melakukan penelitian, dan hal-hal lain yang berkaitan dengan penelitian.

### Objek Penelitian

Di PT. Dinasti Motekar Grup terdapat bidang usaha salah satunya adalah unit usaha dokumentasi yang bernama Studio Chendra yang berfokus pada layanan jasa Dokumentasi, layanan jasa yang disediakan berupa Fotografi, Videografi, dan Desain Grafis.

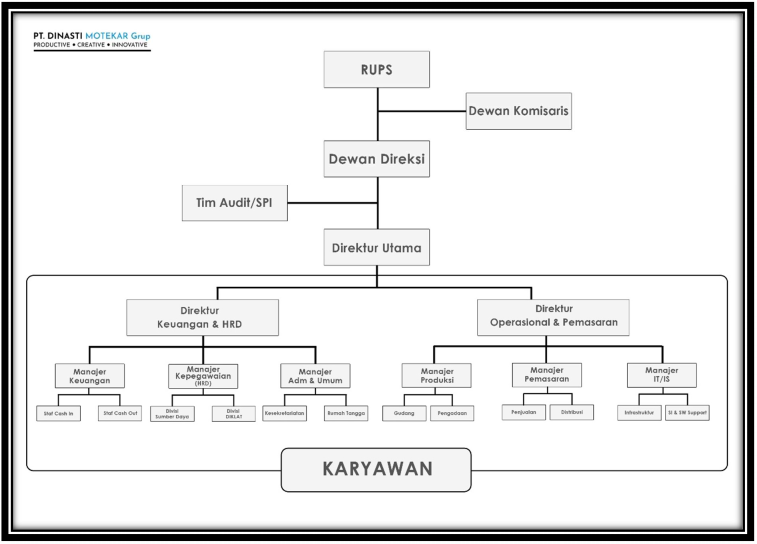
Layanan jasa Dokumentasi ini memiliki informasi dan kegiatan yang berjalan di sebuah website untuk menunjukan hasil pekerjaan serta informasi tentang perusahaan, dengan adanya website yang telah dibuat ini diharapkan calon client dapat mengaksesnya untuk melihat keseluruhuan informasi yang dibutuhkan tentang perusahaan ini.

Pada penelitian ini fokus peneliti adalah membangun sebuah aplikasi berbasis android yang menampung semua informasi yang dibutuhkan dari sebuah website yang telah dibuat, agar mobilitas akses informasi meningkat, serta mengimplentasikan dan melakukan perbandingan JSON adapter pada aplikasi yang akan dibangun nantinya.

### Tempat Penelitian

Tempat penelitian berada di Kota Cimahi tepatnya di Dinasti Motekar Academy yang beralamatkan di Jl. Rorojongrang Raya A12-12 Jawa Barat. Penelitian dilakukan pada bagian IT/IS. Dinasti Motekar Academy didirikan pada tanggal 17 Mei 2018 oleh Acep Hendra bersama Ferry Mulyanto dan Muhammad Tirta Mulia yang sama – sama berguru dengan Tung Desem Waringin, sebagai perwujudan dari konsep bebagi, dengan tagline “HayyunYuhyi” yang berarti Tumbuh dan Menumbuhkan.

### Struktur Organisasi

Berikut adalah gambar dari struktur organisasi tempat peneliti melakukan penelitian tugas akhir yaitu PT. Dinasti Motekar Grup. Sebagaimana dapat dilihat pada Gambar 3-3.

**Gambar 3‑3. Struktur Organisasi Dinasti Motekar Academy**

Bagian Infrastruktur, IT dan SI memiliki job desk sebagai berikut :

1. Mengelola Teknologi Informasi dan sistem computer.
2. Bertanggung jawab pada kesiapan dan ketersediaan sistem komputer / aplikasi dalam lingkungan perusahaan.
3. Membuat dan/atau implementasi semua sistem dan aplikasi
4. Merancang, mengelola dan mengawasi serta meng-evaluasi operasional dari sistem informasi (software dan aplikasi) dan pendukungnya (hardware, infrastruktur, telekomunikasi).
5. Membuat dan mengimplementasikan kebijakan dan prosedur TI (IT policy) termasuk kebijakan keamanan TI (IT security policy).
6. Berkerja sama dengan TI vendor untuk merancang , membuat dan mengimplementasikan sistem atau aplikasi jika diperlukan.
7. Membuat dan mengawasi anggaran TI (budget) dan expenditures.
8. Memberikan solusi Teknologi Informasi.
9. Bertanggung jawab pada penyediaan layanan infrastruktur termasuk aplikasi, jaringan komputer (LAN / WAN), keamanan Teknologi Informasi dan telekomunikasi.
10. Memberikan rekomendasi tentang solusi sistem informasi dan pendukungnya.
11. Memberikan orientasi kepada pegawai baru mengenai aplikasi atau system yang digunakan saat ini dan rencana atau strategi TI secara umum.
12. Memberikan arahan pada bawahan mengenai penggunaan dan solusi teknologi.
13. Pengawasan dan perawatan Teknologi Informasi.
14. Bekerja sama dengan senior manajemen untuk membuat, merancang pelayanan TI dalam dokumen Service Level Agreements.
15. Memberikan laporan bulanan kepada Direktur Keuangan mengenai semua aspek dari departemen TI (Teknologi Informasi).

Dalam kasus kali ini peneliti melakukan tahapan komunikasi dengan direktur utama yakni Acep Hendra, ST., C.Ht., CFDM., dikarenakan pada divisi Infarstruktur IT/IS masih belum memiliki manager.

# ANALISIS DAN PERANCANGAN

Bab ini menjelaskan proses awal pembangunan perangkat lunak yang terdiri dari analisis dan perancangan. Proses analisis terdiri dari pendefinisian perangkat lunak, pendefinisian kebutuhan yang harus dipenuhi perangkat lunak, dan analisis terhadap kebutuhan dengan menggunakan pemodelan spesifikasi kebutuhan dan pemodelan kebutuhan. Sedangkan proses perancangan perangkat lunak terdiri dari perancangan arsitektur, perancangan antarmuka, dan perancangan data.

## Analisis Sistem yang sedang berjalan

Saat ini informasi tentang Studio Chendra PT Dinasti Motekar Grup bisa diakses di *website* studiochendra.co.id, dan semua informasi perihal Studio Chendra ada semua di website tersebut. Calon client biasanya mengakses website untuk melihat hasil pekerjaan dari unit usaha Dokumentasi ini lalu menghubungi pihak perusahaan untuk menyewa jasanya. Dalam hal ini untuk mengakses *website* memerlukan perangkat yang bisa membuka browser seperti desktop dan lainnya. Dengan adanya perangkat berbasis android, mobilitas akses calon client dapat ditingkatkan dengan hanya membuka aplikasinya.

## Analisis Kebutuhan

Analisis kebutuhan akan menghasilkan spesifikasi-spesifikasi dari karekteristik-karakteristik operasional yang akan dimiliki oleh perangkat lunak yang akan dikembangkan. Pada tahap ini penulis selaku analis sistem, melakukan interaksi dengan calon pengguna, yaitu dengan pemilik (*owner*) Studio Chendra Dinasti Motekar Grup.

### Menentukan Kebutuhan Dasar

Pada tahap analisis kebutuhan, harus ditentukan kebutuhan aplikasi yang akan dibangun. Hal-hal apa saja yang harus ada dan bisa dilakukan oleh aplikasi. Berikut ini adalah kebutuhan fungsional diantaranya sebagai berikut :

#### **Analisis Stakeholder**

Berikut ini adalah stakeholder-stakeholder atau orang yang berkepentingan yang terlibat dalam publikasi informasi dan kegiatan di Studio Chendra PT Dinasti Motekar Grup. Dapat dilihat pada Table 4-1 diantaranya sebagai berikut :

**Tabel 4‑1. Daftar Stakeholder**

|  |  |  |
| --- | --- | --- |
| **No.** | **Stakeholder** | **Deskripsi** |
| 1. | Administrator | Administrator adalah orang yang menerima permintaan Dokumentasi di Studio Chendra PT Dinasti Motekar Grup. |
| 2. | Pengguna | Pengguna adalah orang yang menggunakan aplikasi dalam hal ini adalah calon customer yang ingin menggunakan jasa Dokumentasi. |

#### **Analisis Fungsional**

Berikut ini adalah analisis fungsional yang dibutuhkan oleh aplikasi yang akan dibangun, dapat dilihat pada table 4-2.

**Tabel 4‑2. Analisis Fungsional**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Kode** | **Kebutuhan Fungsional** | **Deskripsi** |
| 1. | F-01 | Perangkat lunak dapat menampilkan informasi Workshop berupa video atau foto | Perangkat lunak dapat menampilkan informasi mengenai workshop yang diselenggarakan oleh PT Dinasti Motekar Grup. |
| 2. | F-02 | Perangkat lunak melayani chatting dari customer ke customer service | Perangkat lunak memiliki fitur chatting yang menerima keluhan atau pertanyaan tentang sesuatu berkaitan dengan Studio Chendra PT Dinasti Motekar Grup. |
| 3. | F-03 | Perangkat lunak dapat menampilkan artikel fotografi | Perangkat lunak dapat memiliki kemampuan untuk menampilkan berita yang sedang hot topik yang berkaitan dengan jasa Dinasti Motekar Grup. |
| 4. | F-04 | Perangkat Lunak dapat menampilkan iklan layanan dokumentasi berupa video atau foto | Iklan atau Promosi layanan di PT Dinasti Motekar Grup dapat dilihat melalui perangkat lunak. |
| 5. | F-05 | Perangkat lunak dapat menampilkan portofolio gallery | Perangkat Lunak memiliki kemampuan untuk menampilkan portofolio gallery tentang hasil pekerjaan yang telah dilakukan oleh jasa di PT Dinasti Motekar Grup |
| 6. | F-06 | Perangkat lunak dapat menerima input registrasi order jasa dokumentasi | Perangkat lunak dapat menerima input customer yang ingin melakukan registrasi order jasa fotografi |

### Analisis Pengguna

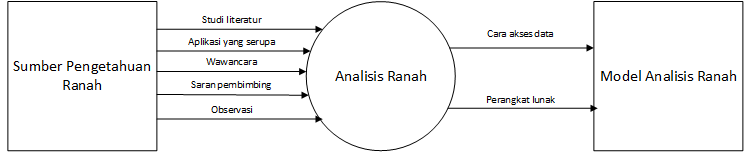
Analisis pengguna merupakan analisis yang digunakan untuk menentukan siapa saja user yang akan menggunakan aplikasi yang akan dibangun, dapat dilihat pada table 4-3.

**Tabel 4‑3. Daftar Pengguna**

|  |  |  |
| --- | --- | --- |
| **No.** | **User** | **Deskripsi** |
| 1. | Admin | Orang yang mengelola data dan informasi yang akan ditampilkan di aplikasi serta menerima order dari customer. |
| 2. | Customer | Orang yang melihat dan menggunakan aplikasi untuk melihat informasi serta memesan jasa fotografi |

### Analisis Ranah

Analisi ranah perangkat lunak adalah proses mengidentifikasi, menganalisis, dan menspesifikasi kebutuhan-kebutuhan umum dari suatu ranah perangkat lunak yang sifatnya spesifik biasanya agar suatu saat dapat digunakan pada berbagai proyek lain yang ada pada ranah aplikasi yang sama.



**Gambar 4‑1. Analisis Ranah**

Dibawah ini adalah deskripsi dari hasil output yang ada pada analisis ranah.

**Tabel 4‑4. Deskripsi hasil output analisis ranah**

|  |  |  |
| --- | --- | --- |
| **No.** | **Nama Output** | **Deskripsi** |
| 1. | Cara akses data | Hasil dari output cara akses data ini adalah menggunakan teknologi pertukaran data yaitu web service, dan JSON. |
| 2. | Perangkat Lunak | Hasil output dari perangkat lunak ini adalah dibangunnya perangkat lunak berbasis android untuk menampung informasi dan portofolio gallery. |

### Pendekatan untuk Pemodelan Spesifikasi Kebutuhan

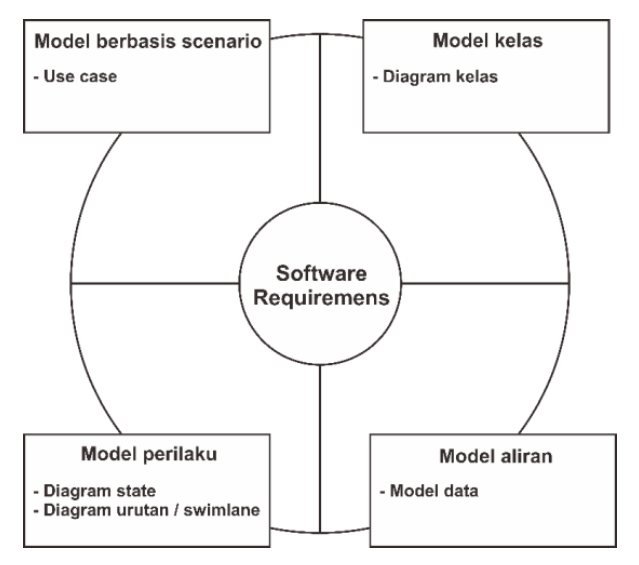
Suatu pandangan dari pemodelan spesifikasi-spesifikasi kebutuhan, yang dinamakan sebagai analisis tertruktur, memperlakukan data dan proses yang melakukan transformasi data tersebut segabai entitas yang terpisah. Jadi, pemodelan spesifikasi kebutuhan ini untuk memisahkan mana untuk keperluan data dan mana untuk proses. Pada sub bab ini akan dijelaskan mengenai perkakas yang  
digunakan dan menggambarkan dari spesifikasi kebutuhan yang akan dibuat.

#### **UML (Unified Modelling Process)**

Unified Modelling Language (UML) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Abstraksi konsep dasar UML terdiri dari structural classification, dynamic behavior, dan model management yang dapat dipahami main concepts sebagai term yang akan muncul pada saat membuat diagram dan view adalah kategori dari diagram tersebut. UML mendefinisikan diagram-diagram sebagai *Use case diagram, Class diagram, Statechart diagram, Activity diagram, Sequence diagram, Collaboration diagram, Component diagram, dan Deployment diagram*. Sebagian dari diagram-diagram itulah yang akan digunakan untuk pemodelan sistem atau perangkat lunak yang akan dibuat.

#### **Elemen – elemen Model Analisis**

Dibawah ini adalah elemen-elemen model analisis kebutuhan yang digunakan. Dapat dilihat pada gambar Gambar 4-2.



**Gambar 4‑2. Elemen Model Analisis**

Masing-masing unsur model kebutuhan pada Gambar 4-2 memperlihatkan permasalahan dari berbagai sudut pandang yang berbeda. Elemen-elemen berbasis skenario memperlihatkan bagaimana interaksi yang akan terjadi antara pengguna dengan sistem atau perangkat lunak yang akan dikembangkan. Elemen model berbasis kelas memodelkan objek-objek yang akan dimanipulasi oleh sistem, memodelkan operasi-operasi yang akan diterapkan pada objek-objek untuk melakukan manipulasi, memodelkan relasi (beberapa bersifat hierarkis) yang terjadi diantara objek satu dengan yang lainnya. Elemen-elemn perilaku memperlihatkan bagaimana event-event eksternal melakukan perubahan pada keadaan (state) sistem atau kelas-kelas yang ada didalamnya. Terakhir, elemen-elemen berorientasi aliran memperlihatkan sistem atau perangkat lunak yang bertindak sebagai pelaku transformasi informasi, memperlihatkan bagaimana objek-objek data ditansformasikan saat megnalir melintasi berbagai fungsi yang dimiliki sistem. [PRE10]

### Pemodelan Berbasis Skenario

Pemodelan berbasis skenario memperlihatkan bagaimana interaksi yang akan terjadi antara pengguna dengan sistem atau perangkat lunak yang akan dikembangkan. Pemodelan spesifikasi kebutuhan pengguna menggunakan UML (*Unified Modelling Language*) pada umumnya dimulai dengan pembuatan skenario-skenario dalam bentuk *use case-use case,* dan diagram *swimlane*. Diagram Use Case untuk aplikasi yang akan dibangun ada pada gambar 4-3 dibawah ini.

|  |
| --- |
| **Use Case Diagram** |
| **Gambar 4‑3. Use Case Diagram** |
| **Deskripsi** |
| Diagram Use Case ini digunakan untuk menggambarkan interaksi antara pengguna dan sistem atau perangkat lunak dimana aplikasi dapat menampilkan informasi workshop, menerima input pesan ke Customer Service, memperlihatkan artikel fotografi berupa foto dan video, menampilkan iklan layanan jasa dokumentasi, menampilkan portofolio gallery, juga dapat menerima input order jasa fotografi, yang semua itu dapat diakses oleh Customer, dan aplikasi dapat memposting video dan foto tentang workshop, menerima input jawaban dari Customer service, memposting video atau foto artikel fotografi, memposting iklan layanan jasa dokumentasi, memposting foto untuk portofolio gallery, menampilkan data registrasi order jasa dokumentasi yang dimana itu semua bisa diakses oleh admin. |

#### **Deskripsi Use Case**

Diagram Use Case pada Gambar 4-3 dapat dijelaskan secara lebih detail pada Tabel 4-5 dibawah ini.

**Tabel 4‑5. Deskripsi Use Case**

| **No.** | **Kode** | **Nama Use Case** | **Deskripsi** |
| --- | --- | --- | --- |
| 1. | UC-01 | Menampilkan informasi workshop berupa video atau foto | Use case ini digunakan untuk menampilkan informasi perihal workshop yang diselengarakan oleh PT Dinasti Motekar Grup berupa foto atau video. |
| 2. | UC-02 | Menginput pesan berupa teks untuk keluhan ke Customer Service | Use case ini digunakan untuk menerima input pesan berupa teks keluhan dari Customer. |
| 3. | UC-03 | Menampilkan artikel fotografi berupa foto atau video | Use case ini digunakan untuk menampilkan artikel berita tentang fotografi berupa foto atau video. |
| 4. | UC-04 | Menampilkan iklan layanan jasa dokumentasi berupa foto atau video | Use case ini digunakan untuk menampilkan ilklan jasa layanan dokumentasi di Studio Chendra PT Dinasti Motekar Grup berupa foto atau video. |
| 5. | UC-05 | Menampilkan portofolio gallery | Use case ini digunakan untuk menampilkan portofolio gallery jasa layanan dokumentasi di Studio Chendra PT Dinasti Motekar Grup. |
| 6. | UC-06 | Menginput registrasi order jasa dokumentasi | Use case ini digunakan untuk menerima input registrasi order jasa dokumentasi di Studio Chendra PT Dinasti Motekar Grup. |
| 7. | UC-07 | Memposting video atau foto workshop | Use Case ini digunakan untuk menerima posting video atau foto tentang workshop yang diselenggarakan di PT Dinasti Motekar Grup. |
| 8. | UC-08 | Menginput jawaban berupa teks untuk menjawab keluhan dari Customer | Use Case ini digunakan untuk menerima input pesan berupa teks jawaban dari Customer Service. |
| 9. | UC-09 | Memposting video atau foto artikel fotografi | Use case ini digunakan untuk menerima posting video atau foto tentang artikel fotografi. |
| 10. | UC-10 | Memposting video atau foto iklan layanan jasa dokumentasi | Use case ini digunakan untuk menerima posting video atau foto tentang iklan layanan jasa dokumentasi di Studio Chendra PT Dinasti Motekar Grup. |
| 11. | UC-11 | Memposting foto untuk portofolio gallery | Use case ini digunakan untuk menerima posting foto portofolio gallery di Studio Chendra PT Dinasti Motekar Grup. |
| 12. | UC-12 | Menampilkan data registrasi order jasa dokumentasi dari customer | Use case ini digunakan untuk menampilkan data registrasi order jasa dokumentasi Studio Chendra di PT Dinasti Motekar Grup. |

#### **Skenario Use Case**

Skenario *use case* merupakan gambaran proses berjalannya perangkat lunak berdasarkan *use case* yang telah didefinisikan. Skenario *use case* melibatkan aktor yang berinteraksi dengan *use case* tersebut serta aksi-reaksi yang terjadi antara aktor dengan sistem atau perangkat lunak. Berikut ini merupakan skenario *use case* aplikasi publikasi informasi dan kegiatan di Studio Chendra PT Dinasti Motekar Grup berbasis Android.

**Tabel 4‑6. Skenario Use Case menampilkan informasi workshop berupa foto atau video**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Identifikasi** | | | | | |
| **Nomor Use Case** | UC-01 | | | | |
| **Nama Use Case** | Menampilkan informasi workshop berupa foto dan video | | | | |
| **Tujuan** | Untuk menampilkan informasi workshop yang di selenggarakan di Dinasti Motekar Grup pada aplikasi berupa foto atau video. | | | | |
| **Aktor** | Customer | | | | |
| **Isi Use Case** | Lihat | | | | |
| **Skenario Utama** | | | | | |
| **Kondisi Awal** | Customer masuk ke menu informasi workshop | | | | |
| **Aksi Aktor** | | | | | **Reaksi Sistem** |
| 1. Customer masuk ke menu informasi workshop | | | | | 1. Sistem mengambil data informasi workshop |
| 1. Customer menekan informasi workshop yang diinginkan | | | | | 1. Sistem menampilkan informasi workshop berupa foto atau video |
|  | | | | | 1. Sistem menampilkan detail informasi workshop berupa foto atau video |
| **Kondisi Akhir** | | | Customer melihat detail informasi workshop berupa foto atau video | | |
| **Skenario Alternatif** | | | | | |
| **Kondisi Awal** | | | Customer masuk ke menu informasi workshop | | |
| **Aksi Aktor** | | | | **Reaksi Sistem** | |
| 1. Customer masuk ke menu informasi workshop | | | | 1. Sistem mengambil data informasi workshop | |
|  | | | | 1. Sistem menampilkan tidak ada workshop | |
| **Kondisi Akhir** | | Customer melihat pemberitahuan tidak ada informasi workshop | | | |

**Tabel 4‑7. Skenario Use Case menginput pesan berupa teks untuk keluhan ke Customer Service**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Identifikasi** | | | | | |
| **Nomor Use Case** | UC-02 | | | | |
| **Nama Use Case** | Menginput pesan berupa teks untuk keluhan ke Customer Service | | | | |
| **Tujuan** | Untuk menerima input pesan berupa teks dari customer yang akan disimpan dan dikirimkan ke customer service | | | | |
| **Aktor** | Customer | | | | |
| **Isi Use Case** | Tambah | | | | |
| **Skenario Utama** | | | | | |
| **Kondisi Awal** | Customer masuk ke menu kontak customer service | | | | |
| **Aksi Aktor** | | | | | **Reaksi Sistem** |
| 1. Customer masuk ke menu kontak customer service | | | | | 1. Sistem menampilkan tampilan chating |
| 1. Customer menginput pesan berupa teks | | | | | 1. Sistem mengecek input customer |
|  | | | | | 1. Sistem menyimpan input customer |
|  | | | | | 1. Sistem menampilkan pesan yang telah diinput di kolom chatting |
| **Kondisi Akhir** | | | Customer melihat pesan teks yang telah diinput di kolom chatting | | |
| **Skenario Alternatif** | | | | | |
| **Kondisi Awal** | | | Customer masuk ke menu kontak customer service | | |
| **Aksi Aktor** | | | | **Reaksi Sistem** | |
| 1. Customer masuk ke menu kontak customer service | | | | 1. Sistem menampilkan tampilan chating | |
| 1. Customer menginput pesan berupa teks | | | | 1. Sistem mengecek input customer | |
|  | | | | 1. Sistem menampilkan input teks tidak valid | |
| **Kondisi Akhir** | | Customer melihat pemberitahuan input teks tidak valid | | | |

**Tabel 4‑8. Skenario Use Case menampilkan artikel fotografi berupa foto atau video**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Identifikasi** | | | | | |
| **Nomor Use Case** | UC-03 | | | | |
| **Nama Use Case** | Menampilkan artikel fotografi berupa foto atau video | | | | |
| **Tujuan** | Untuk menampilkan artikel fotografi berupa foto atau video. | | | | |
| **Aktor** | Customer | | | | |
| **Isi Use Case** | Lihat | | | | |
| **Skenario Utama** | | | | | |
| **Kondisi Awal** | Customer masuk ke menu artikel fotografi | | | | |
| **Aksi Aktor** | | | | | **Reaksi Sistem** |
| 1. Customer masuk ke menu artikel fotografi | | | | | 1. Sistem mengambil data artikel fotografi |
| 1. Customer menekan artikel fotografi yang diinginkan | | | | | 1. Sistem menampilkan artikel fotografi berupa foto atau video |
|  | | | | | 1. Sistem menampilkan detail artikel fotografi berupa foto atau video |
| **Kondisi Akhir** | | | Customer melihat artilkel fotografi | | |
| **Skenario Alternatif** | | | | | |
| **Kondisi Awal** | | | Customer masuk ke menu artikel | | |
| **Aksi Aktor** | | | | **Reaksi Sistem** | |
| 1. Customer masuk ke menu artikel | | | | 1. Sistem mengambil data artikel fotografi | |
|  | | | | 1. Sistem menampilkan tidak ada artikel | |
| **Kondisi Akhir** | | Customer melihat pemberitahuan tidak ada artikel | | | |

**Tabel 4‑9. Skenario Use Case menampilkan iklan layanan jasa dokumentasi berupa foto atau video**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Identifikasi** | | | | | |
| **Nomor Use Case** | UC-04 | | | | |
| **Nama Use Case** | Menampilkan iklan layanan jasa dokumentasi berupa foto atau video | | | | |
| **Tujuan** | Untuk menampilkan iklan layanan jasa dokumentasi di Studio Chendra PT Dinasti Motekar Grup | | | | |
| **Aktor** | Customer | | | | |
| **Isi Use Case** | Lihat | | | | |
| **Skenario Utama** | | | | | |
| **Kondisi Awal** | Customer masuk ke menu iklan layanan jasa | | | | |
| **Aksi Aktor** | | | | | **Reaksi Sistem** |
| 1. Customer masuk ke menu iklan layanan jasa dokumentasi | | | | | 1. Sistem mengambil data iklan layanan jasa dokumentasi |
| 1. Customer menekan iklan layanan jasa dokumentasi yang diinginkan | | | | | 1. Sistem menampilkan iklan layanan jasa dokumentasi |
|  | | | | | 1. Sistem menampilkan detail iklan layanan jasa dokumentasi berupa foto atau video |
| **Kondisi Akhir** | | | Customer melihat detail iklan layanan jasa dokumentasi berupa foto atau video | | |
| **Skenario Alternatif** | | | | | |
| **Kondisi Awal** | | | Customer masuk ke menu iklan layanan jasa | | |
| **Aksi Aktor** | | | | **Reaksi Sistem** | |
| 1. Customer masuk ke menu iklan layanan jasa | | | | 1. Sistem mengambil data iklan layanan jasa dokumentasi | |
|  | | | | 1. Sistem menampilkan tidak ada layanan jasa | |
| **Kondisi Akhir** | | Customer melihat pemberitahuan tidak ada layanan jasa dokumentasi | | | |

**Tabel 4‑10. Skenario Use Case Menampilkan portofolio gallery**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Identifikasi** | | | | | |
| **Nomor Use Case** | UC-05 | | | | |
| **Nama Use Case** | Menampilkan portofolio gallery | | | | |
| **Tujuan** | Untuk menampilkan portofolio gallery tentang Studio Chendra PT Dinasti Motekar Grup | | | | |
| **Aktor** | Customer | | | | |
| **Isi Use Case** | Lihat | | | | |
| **Skenario Utama** | | | | | |
| **Kondisi Awal** | Customer masuk ke menu portofolio gallery | | | | |
| **Aksi Aktor** | | | | | **Reaksi Sistem** |
| 1. Customer masuk ke menu portofolio gallery | | | | | 1. Sistem mengambil data portofolio gallery |
| 1. Customer menekan foto yang diinginkan | | | | | 1. Sistem menampilkan portofolio gallery |
|  | | | | | 1. Sistem menampilkan detail foto |
| **Kondisi Akhir** | | | Customer melihat detail foto | | |
| **Skenario Alternatif** | | | | | |
| **Kondisi Awal** | | | Customer masuk ke menu portofolio gallery | | |
| **Aksi Aktor** | | | | **Reaksi Sistem** | |
| 1. Customer masuk ke menu portofolio gallery | | | | 1. Sistem mengambil data portofolio gallery | |
|  | | | | 1. Sistem menampilkan tidak ada foto | |
| **Kondisi Akhir** | | Customer melihat pemberitahuan tidak ada foto | | | |

**Tabel 4‑11. Skenario Use Case menginput registrasi order jasa dokumentasi**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Identifikasi** | | | | | |
| **Nomor Use Case** | UC-06 | | | | |
| **Nama Use Case** | Menginput registrasi order jasa dokumentasi | | | | |
| **Tujuan** | Untuk menerima input dan menyimpan registrasi order jasa dokumentasi | | | | |
| **Aktor** | Customer | | | | |
| **Isi Use Case** | Lihat | | | | |
| **Skenario Utama** | | | | | |
| **Kondisi Awal** | 1. Customer masuk ke menu registrasi order 2. Customer menekan button order di detail iklan layanan jasa dokumentasi dan masuk menu registrasi order | | | | |
| **Aksi Aktor** | | | | | **Reaksi Sistem** |
| 1. Customer masuk ke menu registrasi order | | | | | 1. Sistem menampilkan form registrasi order layanan jasa dokumentasi |
| 1. Customer menginput registrasi order | | | | | 1. Sistem mengecek input customer |
| 1. Customer menekan button registrasi | | | | | 1. Sistem menyimpan input customer |
|  | | | | | 1. Sistem menampilkan input berhasil |
| **Kondisi Akhir** | | | Customer melihat pemberitahuan registrasi order berhasil | | |
| **Skenario Alternatif** | | | | | |
| **Kondisi Awal** | | | 1. Customer masuk ke menu registrasi order 2. Customer menekan button order di detail iklan layanan jasa dokumentasi dan masuk menu registrasi order | | |
| **Aksi Aktor** | | | | **Reaksi Sistem** | |
| 1. Customer masuk ke menu registrasi order | | | | 1. Sistem menampilkan form registrasi order layanan jasa dokumentasi | |
| 1. Customer menginput registrasi order | | | | 1. Sistem mengecek input customer | |
| 1. Customer menekan button registrasi | | | | 1. Sistem menampilkan input tidak valid | |
| **Kondisi Akhir** | | Customer melihat pemberitahuan input tidak valid | | | |

**Tabel 4‑12. Skenario Use Case Memposting video atau foto workshop**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Identifikasi** | | | | | |
| **Nomor Use Case** | UC-07 | | | | |
| **Nama Use Case** | Memposting video atau foto workshop | | | | |
| **Tujuan** | Untuk menambakan video atau foto workshop yang di selenggarakan di Studio Chendra PT Dinasti Motekar Grup | | | | |
| **Aktor** | Admin | | | | |
| **Isi Use Case** | Tambah | | | | |
| **Skenario Utama** | | | | | |
| **Kondisi Awal** | Admin masuk ke menu tambah workshop | | | | |
| **Aksi Aktor** | | | | | **Reaksi Sistem** |
| 1. Admin masuk ke menu tambah workshop | | | | | 1. Sistem menampilkan form tambah workshop |
| 1. Admin menginput informasi workshop | | | | | 1. Sistem mengecek input admin |
|  | | | | | 1. Sistem menyimpan input admin |
|  | | | | | 1. Sistem menampilkan input berhasil |
| **Kondisi Akhir** | | | Admin melihat pemberitahuan informasi Workshop telah berhasil ditambahkan | | |
| **Skenario Alternatif** | | | | | |
| **Kondisi Awal** | | | Admin masuk ke menu tambah workshop | | |
| **Aksi Aktor** | | | | **Reaksi Sistem** | |
| 1. Admin masuk ke menu tambah workshop | | | | 1. Sistem menampilkan form tambah workshop | |
| 1. Admin menginput informasi workshop | | | | 1. Sistem mengecek input admin | |
|  | | | | 1. Sistem menampilkan input tidak valid | |
| **Kondisi Akhir** | | Admin melihat pemberitahuan input tidak valid | | | |

**Tabel 4‑13. Skenario Use Case Menginput jawaban berupa teks untuk menjawab keluhan dari Customer**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Identifikasi** | | | | | |
| **Nomor Use Case** | UC-08 | | | | |
| **Nama Use Case** | Menginput jawaban berupa teks untuk menjawab keluhan dari Customer | | | | |
| **Tujuan** | Untuk menerima input pesan berupa teks yang akan disimpan dan dikirimkan ke customer | | | | |
| **Aktor** | Admin | | | | |
| **Isi Use Case** | Tambah | | | | |
| **Skenario Utama** | | | | | |
| **Kondisi Awal** | Admin masuk ke menu respond customer | | | | |
| **Aksi Aktor** | | | | | **Reaksi Sistem** |
| 1. Admin masuk ke menu respond Customer | | | | | 1. Sistem mengambil data list chatting customer |
|  | | | | | 1. Sistem menampilkan tampilan list chatting customer |
| 1. Admin memilih customer | | | | | 1. Sistem menampilkan tampilan chating dengan customer |
| 1. Admin menginput pesan berupa teks | | | | | 1. Sistem mengecek input admin |
|  | | | | | 1. Sistem menyimpan input admin |
|  | | | | | 1. Sistem menampilkan pesan yang telah diinput di kolom chatting |
| **Kondisi Akhir** | | | Admin melihat pesan teks yang telah diinput di kolom chatting | | |
| **Skenario Alternatif 1** | | | | | |
| **Kondisi Awal** | | | Admin masuk ke menu respond customer | | |
| **Aksi Aktor** | | | | **Reaksi Sistem** | |
| 1. Admin masuk ke menu respond Customer | | | | 1. Sistem mengambil data list chatting customer | |
|  | | | | 1. Sistem menampilkan tidak ada pesan | |
| **Kondisi Akhir** | | Admin melihat pemberitahuan tidak ada pesan dari customer | | | |
| **Skenario Alternatif 2** | | | | | |
| **Kondisi Awal** | | | Admin masuk ke menu respond customer | | |
| **Aksi Aktor** | | | | **Reaksi Sistem** | |
| 1. Admin masuk ke menu respond Customer | | | | 1. Sistem mengambil data list chatting customer | |
|  | | | | 1. Sistem menampilkan tampilan list chatting customer | |
| 1. Admin memilih customer | | | | 1. Sistem menampilkan tampilan chating dengan customer | |
| 1. Admin menginput pesan berupa teks | | | | 1. Sistem mengecek input admin | |
|  | | | | 1. Sistem menampilkan input tidak valid | |
| **Kondisi Akhir** | | Admin melihat pemberitahuan input tidak valid | | | |

**Tabel 4‑14. Skenario Use Case Memposting video atau foto artikel fotografi**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Identifikasi** | | | | | |
| **Nomor Use Case** | UC-09 | | | | |
| **Nama Use Case** | Memposting video atau foto artikel fotografi | | | | |
| **Tujuan** | Untuk menambakan video atau foto artikel yang berkaitan dengan fotografi | | | | |
| **Aktor** | Admin | | | | |
| **Isi Use Case** | Tambah | | | | |
| **Skenario Utama** | | | | | |
| **Kondisi Awal** | Admin masuk ke menu tambah artikel fotografi | | | | |
| **Aksi Aktor** | | | | | **Reaksi Sistem** |
| 1. Admin masuk ke menu tambah artikel fotografi | | | | | 1. Sistem menampilkan form tambah artikel fotografi |
| 1. Admin menginput informasi artikel fotografi | | | | | 1. Sistem mengecek input admin |
|  | | | | | 1. Sistem menyimpan input admin |
|  | | | | | 1. Sistem menampilkan input berhasil |
| **Kondisi Akhir** | | | Admin melihat pemberitahuan artikel telah berhasil ditambahkan | | |
| **Skenario Alternatif** | | | | | |
| **Kondisi Awal** | | | Admin masuk ke menu tambah artikel | | |
| **Aksi Aktor** | | | | **Reaksi Sistem** | |
| 1. Admin masuk ke menu tambah artikel | | | | 1. Sistem menampilkan form tambah artike; | |
| 1. Admin menginput informasi artikel fotografi | | | | 1. Sistem mengecek input admin | |
|  | | | | 1. Sistem menampilkan input tidak valid | |
| **Kondisi Akhir** | | Admin melihat pemberitahuan input tidak valid | | | |

**Tabel 4‑15. Skenario Use Case Memposting video atau foto iklan layanan jasa dokumentasi**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Identifikasi** | | | | | |
| **Nomor Use Case** | UC-10 | | | | |
| **Nama Use Case** | Memposting video atau foto iklan layanan jasa dokumentasi | | | | |
| **Tujuan** | Untuk menambakan video atau foto iklan layanan jasa dokumentasi yang ada di Studio Chendra PT Dinasti Motekar Grup. | | | | |
| **Aktor** | Admin | | | | |
| **Isi Use Case** | Tambah | | | | |
| **Skenario Utama** | | | | | |
| **Kondisi Awal** | Admin masuk ke menu tambah iklan layanan jasa dokumentasi | | | | |
| **Aksi Aktor** | | | | | **Reaksi Sistem** |
| 1. Admin masuk ke menu tambah iklan layanan jasa dokumentasi | | | | | 1. Sistem menampilkan form tambah iklan layanan jasa dokumentasi |
| 1. Admin menginput informasi iklan layanan jasa dokumentasi | | | | | 1. Sistem mengecek input admin |
|  | | | | | 1. Sistem menyimpan input admin |
|  | | | | | 1. Sistem menampilkan input berhasil |
| **Kondisi Akhir** | | | Admin melihat pemberitahuan iklan layanan jasa dokumentasi telah berhasil ditambahkan | | |
| **Skenario Alternatif** | | | | | |
| **Kondisi Awal** | | | Admin masuk ke menu tambah iklan layanan jasa dokumentasi | | |
| **Aksi Aktor** | | | | **Reaksi Sistem** | |
| 1. Admin masuk ke menu tambah iklan layanan jasa | | | | 1. Sistem menampilkan form tambah iklan layanan jasa | |
| 1. Admin menginput informasi artikel fotografi | | | | 1. Sistem mengecek input admin | |
|  | | | | 1. Sistem menampilkan input tidak valid | |
| **Kondisi Akhir** | | Admin melihat pemberitahuan input tidak valid | | | |

**Tabel 4‑16. Skenario Use Case Memposting foto untuk portofolio gallery**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Identifikasi** | | | | | |
| **Nomor Use Case** | UC-11 | | | | |
| **Nama Use Case** | Memposting foto untuk portofolio gallery | | | | |
| **Tujuan** | Untuk menambakan foto yang akan ditampilkan di portofolio gallery | | | | |
| **Aktor** | Admin | | | | |
| **Isi Use Case** | Tambah | | | | |
| **Skenario Utama** | | | | | |
| **Kondisi Awal** | Admin masuk ke menu tambah foto portofolio gallery | | | | |
| **Aksi Aktor** | | | | | **Reaksi Sistem** |
| 1. Admin masuk ke menu tambah foto portofolio | | | | | 1. Sistem menampilkan form tambah foto portofolio |
| 1. Admin menginput informasi foto | | | | | 1. Sistem mengecek input admin |
|  | | | | | 1. Sistem menyimpan input admin |
|  | | | | | 1. Sistem menampilkan input berhasil |
| **Kondisi Akhir** | | | Admin melihat pemberitahuan foto telah berhasil ditambahkan | | |
| **Skenario Alternatif** | | | | | |
| **Kondisi Awal** | | | Admin masuk ke menu tambah foto portofolio gallery | | |
| **Aksi Aktor** | | | | **Reaksi Sistem** | |
| 1. Admin masuk ke menu tambah foto portofolio | | | | 1. Sistem menampilkan form tambah foto portofolio | |
| 1. Admin menginput foto | | | | 1. Sistem mengecek input admin | |
|  | | | | 1. Sistem menampilkan input tidak valid | |
| **Kondisi Akhir** | | Admin melihat pemberitahuan input tidak valid | | | |

**Tabel 4‑17. Skenario Use Case Menampilkan data registrasi order jasa dokumentasi dari customer**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Identifikasi** | | | | | |
| **Nomor Use Case** | UC-12 | | | | |
| **Nama Use Case** | Menampilkan data registrasi order jasa dokumentasi dari customer | | | | |
| **Tujuan** | Untuk menampilkan data registrasi order yang telah dilakukan oleh customer | | | | |
| **Aktor** | Admin | | | | |
| **Isi Use Case** | Lihat | | | | |
| **Skenario Utama** | | | | | |
| **Kondisi Awal** | Admin masuk ke menu registrasi order jasa dokumentasi | | | | |
| **Aksi Aktor** | | | | | **Reaksi Sistem** |
| 1. Admin masuk ke menu registrasi order jasa dokumentasi | | | | | 1. Sistem mengambil data list registrasi order customer |
| 1. Admin memilih detail registrasi order customer yang diinginkan | | | | | 1. Sistem menampilkan list registrasi order customer |
|  | | | | | 1. Sistem menampilkan detail registrasi order customer |
| **Kondisi Akhir** | | | Admin melihat detail registrasi order customer | | |
| **Skenario Alternatif** | | | | | |
| **Kondisi Awal** | | | Admin masuk ke menu registrasi order jasa dokumentasi | | |
| **Aksi Aktor** | | | | **Reaksi Sistem** | |
| 1. Admin masuk ke menu registrasi order jasa dokumentasi | | | | 1. Sistem mengambil data list registrasi order customer | |
|  | | | | 1. Sistem menampilkan tidak ada registrasi order | |
| **Kondisi Akhir** | | Admin melihat pemberitahuan ridak ada registrasi order | | | |

#### **Aliran Aktivitas**

Aliran aktivitas digunakan untuk mengetahui alur dari aktivitas-aktivitas yang ada. Untuk membuat aliran aktivitas ini akan digunakan diagram *Swimlane*. Diagram *Swimlane* menjelaskan aliran aktivitas yang dijelaskan oleh use case dan pada saat yang sama menunjukkan aktor mana (jika ada beberapa aktor yang terlibat dalam use case tertentu) atau kelas analisis sebagai tanggung jawab atas tindakan yang dijelaskan oleh sebuah aktivitas [PRE10].

##### **Diagram Swimlane pada Use Case UC-01**

|  |
| --- |
| **Diagram Swimlane Use Case UC-01** |
| **Gambar 4‑4. Swimlane Diagram pada UC-01** |
| **Deskripsi** |
| Diagram Swimlane ini menggambarkan aliran aktivitas dari Use Case UC-01 yang dimulai dari memilih menu informasi workshop dan diakhiri dengan melihat detail informasi workshop berupa foto atau video. |

##### **Diagram Swimlane pada Use Case UC-02**

|  |
| --- |
| **Diagram Swimlane Use Case UC-02** |
| **Gambar 4‑5. Diagram Swimlane pada UC-02** |
| **Deskripsi** |
| Diagram Swimlane ini menggambarkan aliran aktivitas dari Use Case UC-02 yang dimulai dari memilih menu kontak customer service dan diakhiri dengan melihat pesan teks di kolom chatting. |

##### **Diagram Swimlane pada Use Case UC-03**

|  |
| --- |
| **Diagram Swimlane Use Case UC-03** |
| **Gambar 4‑6. Swimlane Diagram pada UC-03** |
| **Deskripsi** |
| Diagram Swimlane ini menggambarkan aliran aktivitas dari Use Case UC-03 yang dimulai dari memilih menu artikel dan diakhiri dengan melihat detail artikel fotografi berupa foto atau video. |

##### **Diagram Swimlane pada Use Case UC-04**

|  |
| --- |
| **Diagram Swimlane Use Case UC-04** |
| **Gambar 4‑7. Swimlane Diagram pada UC-04** |
| **Deskripsi** |
| Diagram Swimlane ini menggambarkan aliran aktivitas dari Use Case UC-04 yang dimulai dari memilih menu iklan layanan jasa dokumentasi dan diakhiri dengan melihat detail iklan layanan jasa dokumentasi berupa foto atau video. |

##### **Diagram Swimlane pada Use Case UC-05**

|  |
| --- |
| **Diagram Swimlane Use Case UC-05** |
| **Gambar 4‑8. Swimlane Diagram pada UC-05** |
| **Deskripsi** |
| Diagram Swimlane ini menggambarkan aliran aktivitas dari Use Case UC-05 yang dimulai dari memilih menu portofolio gallery dan diakhiri dengan melihat detail foto. |

##### **Diagram Swimlane pada Use Case UC-06**

|  |
| --- |
| **Diagram Swimlane Use Case UC-06** |
| **Gambar 4‑9. Swimlane Diagram pada UC-06** |
| **Deskripsi** |
| Diagram Swimlane ini menggambarkan aliran aktivitas dari Use Case UC-06 yang dimulai dari memilih menu registrasi order atau menekan button order di detail ilkan layanan jasa dokumentasi dan diakhiri dengan melihat pemberitahuan registrasi order berhasil. |

##### **Diagram Swimlane pada Use Case UC-07**

|  |
| --- |
| **Diagram Swimlane Use Case UC-07** |
| **Gambar 4‑10. Swimlane Diagram pada UC-07** |
| **Deskripsi** |
| Diagram Swimlane ini menggambarkan aliran aktivitas dari Use Case UC-07 yang dimulai dari memilih menu tambah workshopi dan diakhiri dengan melihat pemberitahuan workshop berhasil ditambahkan. |

##### **Diagram Swimlane pada Use Case UC-08**

|  |
| --- |
| **Diagram Swimlane Use Case UC-08** |
| **Gambar 4‑11. Swimlane Diagram pada UC-08** |
| **Deskripsi** |
| Diagram Swimlane ini menggambarkan aliran aktivitas dari Use Case UC-08 yang dimulai dari memilih menu respond customer dan diakhiri dengan melihat pesan di kolom chatting. |

##### **Diagram Swimlane pada Use Case UC-09**

|  |
| --- |
| **Diagram Swimlane Use Case UC-09** |
| **Gambar 4‑12. Swimlane Diagram pada UC-09** |
| **Deskripsi** |
| Diagram Swimlane ini menggambarkan aliran aktivitas dari Use Case UC-09 yang dimulai dari memilih menu tambah artikel dan diakhiri dengan melihat pemberitahuan artikel fotografi telah berhasil ditambahkan. |

##### **Diagram Swimlane pada Use Case UC-10**

|  |
| --- |
| **Diagram Swimlane Use Case UC-10** |
| **Gambar 4‑13. Swimlane Diagram pada UC-10** |
| **Deskripsi** |
| Diagram Swimlane ini menggambarkan aliran aktivitas dari Use Case UC-10 yang dimulai dari memilih menu tambah iklan layanan jasa dokumentasi dan diakhiri dengan melihat pemberitahuan iklan layanan jasa dokumentasi telah berhasil ditambahkan. |

##### **Diagram Swimlane pada Use Case UC-11**

|  |
| --- |
| **Diagram Swimlane Use Case UC-11** |
| **Gambar 4‑14. Swimlane Diagram pada UC-11** |
| **Deskripsi** |
| Diagram Swimlane ini menggambarkan aliran aktivitas dari Use Case UC-11 yang dimulai dari memilih menu tambah foto portofolio dan diakhiri dengan melihat pemberitahuan foto telah berhasil ditambahkan ke portofolio gallery. |

##### **Diagram Swimlane pada Use Case UC-12**

|  |
| --- |
| **Diagram Swimlane Use Case UC-12** |
| **Gambar 4‑15. Swimlane Diagram pada UC-12** |
| **Deskripsi** |
| Diagram Swimlane ini menggambarkan aliran aktivitas dari Use Case UC-12 yang dimulai dari memilih menu registrasi order jasa dokumentasi customer dan diakhiri dengan melihat detail registrasi order jasa dokumentasi customer. |

### Pemodelan data

Kebutuhan-kebutuhan perangkat lunak mencakup di dalamnya kebutuhan-kebutuhan untuk membuat, memperluas, atau bersinggungan dengan basis data. Untuk itu, harus didefinisikan semua objek data yang akan diproses di dalam sistem atau perangkat lunak, mendefinisikan relasi antar objek, dan mendefinisikan informasi lain terkait relasi. [PRE10]

#### **Objek Data**

Sebuah objek data sesungguhnya adalah representasi dari informasi komposit yang harus dipahami oleh perangkat lunak. Memiliki sejumlah property atau atribut yang berbeda. Suatu objek data dapat saja berupa entitas eksternal (sesuatu yang menghasilkan atau menggunakan informasi), laporan atau tampilan, panggilan telepon atau suatu alarm, suatu peran misalnya wiraniaga, suatu unit organisasi, suatu tempat, atau suatu struktur contohnya berkas, [PRE10]

**Tabel 4‑18. Objek Data**

|  |  |  |
| --- | --- | --- |
| **No.** | **Nama Objek** | **Deskripsi** |
|  | Menu | Objek ini digunakan untuk mengelola menu yang akan digunakan pada aplikasi |
|  | User | Objek ini digunakan untuk mengambil, menyimpan dan menampilkan akun dari user |
|  | Workshop | Objek ini digunakan untuk mengambil, menyimpan, dan menampilkan informasi Workshop |
|  | Chat Customer Service | Objek ini digunakan untuk mengambil, menyimpan, dan menampilkan chatting keluhan customer dan jawaban keluhan dari admin. |
|  | Artikel Fotografi | Objek ini digunakan untuk mengambil, menyimpan, dan menampilkan informasi artikel fotografi |
|  | Iklan Layanan Jasa Dokumenatsi | Objek ini digunakan untuk mengambil, menyimpan, dan menampilkan informasi iklan layanan jasa fotografi |
|  | Portofolio Gallery | Objek ini digunakan untuk mengambil, menyimpan, dan menampilkan portofolio gallery |
|  | Registrasi Order | Objek ini digunakan untuk mengambil, menyimpan, dan menampilkan informasi registrasi order layanan jasa dokumentasi |
|  | User Level | Objek ini digunakan untuk menyimpan data level-level pengguna yang ada. |
|  | Hak akses | Objek ini digunakan untuk menentukan hak akses setiap pengguna aplikasi. |

#### **Atribut Data**

Atribut data pada dasarnya mendefinisikan property-properti yang dimiliki oleh objek data dan pada saranya memiliki tiga karakteristik yang berbeda. Pertama, memberi nama objek dari objek data, mendeskripsikan objek,atau merujuk pada objek lain yang ada pada table yang lain. Berikut ini adalah daftar atribut data dari objek data yang sudah dibuat sebelumnya.

**Tabel 4‑19. Atribut data**

| **No.** | **Objek Data** | **Atribut Data** | **Deskripsi** |
| --- | --- | --- | --- |
|  | Menu | id\_menu | Atribut ini digunakan untuk primary key pada tabel. |
| title\_menu | Atribut ini digunakan untuk memberi nama pada menu. |
| url | Atribut ini digunakan untuk memberi nama url pada menu. |
| is\_mainmenu | Atribut ini digunakan untuk mengecek apakah menu adalah menu utama atau sub menu. |
| Is\_active | Atribut ini digunakan untuk menentukan menu ini aktif atau tidak. |
| order\_num | Atribut ini digunakan untuk Menyusun urutan menu. |
|  | User | Id\_user | Atribut ini digunakan untuk primary key pada tabel |
| nama\_user | Atribut ini digunakan untuk memberi nama user |
| username\_user | Atribut ini digunakan untuk menyimpan username. |
| password\_user | Atribut ini digunakan untuk autentikasi saat login |
| email\_user | Atribut ini digunakan untuk autentikasi saat login |
| no\_hp\_user | Atribut ini digunakan untuk menyimpan no hp. |
| Id\_user\_level | Atribut ini digunakan untuk menentukan level akses pada user. |
|  | Workshop | id\_workshop | Atribut ini digunakan untuk primary key pada tabel. |
| judul\_workshop | Atribut ini digunakan untuk, memberi nama judul workshop. |
| deskripsi\_workshop | Atribut ini digunakan untuk mendeskripsikan workshop |
| gambar\_workshop | Atribut ini digunakan untuk menambahkan gambar sebagai poster workshop |
| video\_workshop | Atribut ini digunakan untuk menambahkan video workshop. |
|  | Chat Customer Service | id\_chat | Atribut ini digunakan untuk primary key pada tabel. |
| id\_user\_1 | Atribut ini digunakan untuk mengetahui pengguna chat pertama untuk bertanya |
| id\_user\_2 | Atribut ini digunakan untuk mengetahui pengguna chat kedua untuk menjawab |
| Id\_user\_level | Atribut ini digunakan untuk mengetahui dan menentukan level akses user |
| teks\_chat\_1 | Atribut ini digunakan untuk menyimpan pesan teks pertanyaan pengguna pertama |
| teks\_chat\_2 | Atribut ini digunakan untuk menyimpan pesan teks jawaban pengguna kedua |
|  | Artikel Fotografi | id\_artikel | Atribut ini digunakan untuk primary key pada tabel. |
| judul\_artikel | Atribut ini digunakan untuk, memberi nama judul artikel. |
| deskripsi\_artikel | Atribut ini digunakan untuk mendeskripsikan artikel |
| gambar\_artikel | Atribut ini digunakan untuk menambahkan gambar artikel |
| video\_artikel | Atribut ini digunakan untuk menambahkan video artikel. |
|  | Iklan Layanan Jasa Dokumentasi | id\_iklan | Atribut ini digunakan untuk primary key pada tabel. |
| judul\_iklan | Atribut ini digunakan untuk, memberi nama judul iklan. |
| deskripsi\_iklan | Atribut ini digunakan untuk mendeskripsikan iklan |
| gambar\_iklan | Atribut ini digunakan untuk menambahkan gambar iklan |
| video\_iklan | Atribut ini digunakan untuk menambahkan video iklan. |
|  | Portofolio Gallery | id\_foto | Atribut ini digunakan untuk primary key pada tabel. |
| judul\_foto | Atribut ini digunakan untuk, memberi nama judul foto. |
| deskripsi\_foto | Atribut ini digunakan untuk mendeskripsikan foto |
| foto\_gallery | Atribut ini digunakan untuk menambahkan foto |
|  | Registrasi Order | id\_registrasi | Atribut ini digunakan untuk primary key pada tabel. |
| Id\_user | Atribut ini digunakan untuk, mengetahui profile user yang melakukan registrasi |
| id\_iklan | Atribut ini digunakan untuk mengetahui jasa layanan apa yang akan disewa |
| tgl\_registrasi | Atribut ini digunakan untuk mengetahui tanggal registrasi dilakukan |
|  | User Level | id\_user\_level | Atribut ini digunakan untuk primary key pada tabel |
| nama\_level | Atribut ini digunakan untuk memberi nama level untuk pemberian hak akses pada user |
|  | Hak Akses | id\_hak\_akses | Atribut ini digunakan untuk primary key pada tabel |
| id\_user\_level | Atribut ini digunakan untuk mengetahui level user |
| id\_menu | Atribut ini digunakan untuk menentukan menu apa saja yang bisa diakses |

#### **Relasi**

Objek-objek data saling terhubung satu dengan yang lainnya dengan menggunakan berbagai cara yang berbeda [PRE10].

**Tabel 4‑20. Relasi Objek Data**

| **No.** | **Objek Data** | **Relasi Data** | **Kardinalitas** | **Deskripsi** |
| --- | --- | --- | --- | --- |
|  | Chat Customer Service |  | one to many | Relasi ini digunakan untuk satu chat customer service dapat dilakukan banyak user |
|  | one to one | Relasi ini digunakan untuk satu chat customer service dapat memilik banyak user level |
|  | Hak Akses |  | one to one | Relasi ini digunakan untuk memberi satu hak pada satu menu |
|  | many to one | Relasi ini digunakan untuk satu user level memiliki banyak hak akses pada aplikasi |
|  | Registrasi Order |  | many to one | Relasi ini digunakan untuk satu user dapat Memiliki banyak registrasi order |
|  | one to many | Relasi ini digunakan untuk satu registrasi order dapat memiliki banyak Iklan Layanan Jasa Dokumentasi |
|  | User Level |  | one to many | Relasi ini digunakan untuk satu user level dapat memiliki banyak user |

### Pemodelan Berbasis Kelas

Pemodelan berbasis kelas pada dasarnya memperlihatkan objek-objek yang akan dimanipulasi oleh sistem atau perangkat lunak, memperlihatkan operasi-operasi yang akan diterapkan pada objekobjek untuk menghasilkan umbas tertentu pada manipulasi objek, memperlihatkan relasi-relasi antar objek [PRE10]. Pada pemodelan berbasis kelas ini akan mengidentifikasi kelas-kelas analisis kemudian menentukan atribut dan perilaku setiap kelas dan terakhir akan membuat diagram kelas.

#### **Mengidentifikasi Kelas – Kelas Analisis**

Proses mengidentifikasi kelas dengan cara memeriksa skenario penggunaan sistem atau perangkat lunak yang telah dikembangkan sebelumnya sebagai bagian dari model-model kebutuhan. [PRE10]

**Tabel 4‑21. Kelas - Kelas Analisis**

| **No.** | **Nama Kelas** | **Jenis** | **Deskripsi** |
| --- | --- | --- | --- |
|  | Activity\_workshop | View | Kelas ini digunakan untuk tampilan pada halaman workshop |
|  | Activity\_iklan\_layanan\_jasa | View | Kelas ini digunakan untuk tampilan pada halaman iklan layanan jasa |
|  | Activity\_portofolio\_gallery | View | Kelas ini digunakan untuk tampilan pada halaman portofolio gallery |
|  | Activity\_chat\_customer\_service | View | Kelas ini digunakan untuk tampilan pada halaman chat customer service |
|  | Activity\_artikel\_fotografi | View | Kelas ini digunakan untuk tampilan pada halaman artikel fotografi |
|  | Activity\_registrasi\_order | View | Kelas ini digunakan untuk tampilan pada halaman registrasi order |
|  | Activity\_menu | View | Kelas ini digunakan untuk tampilan pada menu |
|  | Activity\_login | View | Kelas ini digunakan untuk tampilan pada halaman login |
|  | Activity\_posting\_workshop | View | Kelas ini digunakan untuk tampilan pada halaman posting workshop |
|  | Activity\_posting\_iklan\_layanan\_jasa | View | Kelas ini digunakan untuk tampilan pada halaman posting iklan layanan jasa |
|  | Activity\_posting\_foto\_portofolio | View | Kelas ini digunakan untuk tampilan pada halaman posting foto portofolio |
|  | Activity\_respond\_customer | View | Kelas ini digunakan untuk tampilan pada halaman respond customer |
|  | Activity\_posting\_artikel\_fotografi | View | Kelas ini digunakan untuk tampilan pada halaman posting artikel fotografi |
|  | Activity\_add\_registrasi\_order | View | Kelas ini digunakan untuk tampilan pada halaman add registrasi order |
|  | WorkshopActivity | Controller | Kelas ini digunakan untuk menghubungkan tampilan workshop dengan model workshop |
|  | IklanLayananJasaActivity | Controller | Kelas ini digunakan untuk menghubungkan tampilan iklan layanan jasa dengan model iklan layanan jasa |
|  | PortofolioGalleryActivity | Controller | Kelas ini digunakan untuk menghubungkan tampilan portofolio gallery dengan model portofolio gallery |
|  | ChatCustomerServiceActivity | Controller | Kelas ini digunakan untuk menghubungkan tampilan chat customer service dengan model chat customer service |
|  | ArtikelFotografiActivity | Controller | Kelas ini digunakan sebagai controller untuk halaman artikel fotografi |
|  | RegistrasiOrderActivity | Controller | Kelas ini digunakan untuk menghubungkan tampilan registrasi order dengan model registrasi order |
|  | MenuActivity | Controller | Kelas ini digunakan sebagai controller untuk menu |
|  | LoginActivity | Controller | Kelas ini digunakan sebagai controller untuk halaman login |
|  | PostWorkshopActivity | Controller | Kelas ini digunakan untuk menghubungkan tampilan post workshop dengan model workshop |
|  | PostIklanLayananJasaActivity | Controller | Kelas ini digunakan untuk menghubungkan tampilan post iklan layanan jasa dengan model iklan layanan jasa |
|  | PostFotoPortofolioActivity | Controller | Kelas ini digunakan untuk menghubungkan tampilan post foto portofolio dengan model portofolio gallery |
|  | RespondCustomerActivity | Controller | Kelas ini digunakan untuk menghubungkan tampilan respond customer dengan model chat customer service |
|  | PostArtikelFotografiActivity | Controller | Kelas ini digunakan untuk menghubungkan tampilan porst artikel fotografi dengan model artikel fotografi |
|  | AddRegistrasiOrderActivity | Controller | Kelas ini digunakan untuk menghubungkan tampilan add registrasi order dengan model registrasi order |
|  | WorkshopModel | Model | Kelas ini digunakan untuk berinteraksi dengan database dan mengambil atau menyimpan data workshop |
|  | IklanLayananJasaModel | Model | Kelas ini digunakan untuk berinteraksi dengan database dan mengambil atau menyimpan data Iklan Layanan Jasa |
|  | PortofolioGalleryModel | Model | Kelas ini digunakan untuk berinteraksi dengan database dan mengambil atau menyimpan data Portofolio Gallery |
|  | ChatCustomerServiceModel | Model | Kelas ini digunakan untuk berinteraksi dengan database dan mengambil atau menyimpan data chat customer dengan admin |
|  | ArtikelFotografiModel | Model | Kelas ini digunakan untuk berinteraksi dengan database dan mengambil atau menyimpan data artikel fotografi |
|  | RegistrasiOrderModel | Model | Kelas ini digunakan untuk berinteraksi dengan database dan mengambil atau menyimpan data registrasi order |
|  | UserModel | Model | Kelas ini digunakan untuk berinteraksi dengan database dan mengambil atau menyimpan data user |
|  | MenuModel | Model | Kelas ini digunakan untuk berinteraksi dengan database dan mengambil atau menyimpan data menu |

#### **Menentukan Atribut-Atribut Kelas**

Atribut mendeskripsikan kelas-kelas yang telah terpilih untuk dimasukan dalam model spesifikasi kebutuhan sistem atau perangkat lunak. Atribut-atribut merupakan sesuatu yang mendefinikan kelas yang mengklasifikasikan makna suatu kelas dalam konteks ruang permasalahan yang telah diketahui sebelumnya.

**Tabel 4‑22. Atribut Kelas**

| **No.** | **Nama Kelas** | **Nama Atribut** |
| --- | --- | --- |
|  | Activity\_workshop | judulWorkshopText |
| deskripsiWorkshopText |
| images |
| video |
|  | Activity\_iklan\_layanan\_jasa | judulIklanText |
| deskripsiIklanText |
| images |
| video |
|  | Activity\_portofolio\_gallery | images |
|  | Activity\_chat\_customer\_service | userText |
| messageText |
| btnSend |
|  | Activity\_artikel\_fotografi | judulArtikelText |
| deskripsiArtikelText |
| images |
| video |
|  | Activity\_registrasi\_order | userText |
| judulIklanText |
|  | Activity\_menu | titleText |
| isActive |
| isMainMenu |
|  | Activity\_login | usernameText |
| passwordText |
| btnLogin |
|  | Activity\_posting\_workshop | judulWorkshopText |
| deskripsiWorkshopText |
| formImages |
| formVideo |
| btnPost |
|  | Activity\_posting\_iklan\_layanan\_jasa | judulIklanText |
| deskripsiIklanText |
| formImages |
| formVideo |
| btnPost |
|  | Activity\_posting\_foto\_portofolio | formImages |
| btnPost |
|  | Activity\_respond\_customer | userText |
| messageText |
| btnSend |
|  | Activity\_posting\_artikel\_fotografi | judulArtikelText |
| deskripsiArtikelText |
| formImages |
| formVideo |
| btnPost |
|  | Activity\_add\_registrasi\_order | userText |
| judulIklanText |
| btnConfirm |
|  | WorkshopActivity | - |
|  | IklanLayananJasaActivity | - |
|  | PortofolioGalleryActivity | - |
|  | ChatCustomerServiceActivity | - |
|  | ArtikelFotografiActivity | - |
|  | RegistrasiOrderActivity | - |
|  | MenuActivity | - |
|  | LoginActivity | username |
| password |
| btnLogin |
|  | PostWorkshopActivity | - |
|  | PostIklanLayananJasaActivity | - |
|  | PostFotoPortofolioActivity | - |
|  | RespondCustomerActivity | - |
|  | PostArtikelFotografiActivity | - |
|  | AddRegistrasiOrderActivity | - |
|  | WorkshopModel | idWorkshop |
| judulWorkshop |
| deskripsiWorkshop |
| gambarWorkshop |
| videoWorkshop |
|  | IklanLayananJasaModel | idIklan |
| judulIklan |
| deskripsiIklan |
| gambarIklan |
| videoIklan |
|  | PortofolioGalleryModel | idFoto |
| judulFoto |
| deskripsiFoto |
| foto |
|  | ChatCustomerServiceModel | idChat |
| idUser1 |
| idUser2 |
| message1 |
| message2 |
|  | ArtikelFotografiModel | idArtikel |
| judulArtikel |
| deskripsiArtikel |
| gambarArtikel |
| videoArtikel |
|  | RegistrasiOrderModel | idRegistrasi |
| idUser |
| idIklan |
| tglRegistrasi |
|  | UserModel | idUser |
| namaUser |
| usernameUser |
| passwordUser |
| emailUser |
| noHpUser |
| idUserLevel |
|  | MenuModel | idMenu |
| titleMenu |
| url |
| isMainMenu |
| isActive |
| orderNum |

#### **Menentukan Perilaku**

Operasi mendefinisikan perilaku suatu objek. Meskipun ada berbagai jenis operasi, mereka umumnya dapat dibagi menjadi empat kategori besar: operasi yang memanipulasi data dalam beberapa cara (misal, Menambahkan, menghapus, memformat ulang, memilih), operasi yang melakukan perhitungan, operasi yang menanyakan tentang keadaan suatu objek, dan operasi yang memantau objek untuk terjadinya peristiwa pengendalian [PRE10].

**Tabel 4‑23. Perilaku Kelas**

| **No.** | **Nama Kelas** | **Jenis Kelas** | **Operasi** | **Perilaku** |
| --- | --- | --- | --- | --- |
|  | WorkshopModel | Model | getData(id) | Mengambil data workshop dari database |
| getDataWorkshop() | Mengambil data workshop dan menghilangkan duplikasi data workshop |
| insert(data) | Menambahkan data workshop ke database |
|  | IklanLayananJasaModel | Model | getData(id) | Mengambil data iklan layanan jasa dari database |
| getDataIklanLayananJasa() | Mengambil data iklan layanan jasa dan menghilangkan duplikasi data. |
| insert(data) | Menambahkan data iklan layanan jasa ke database |
|  | PortofolioGalleryModel | Model | getData(id) | Mengambil data portofolio gallery dari database |
| getDataPortofolioGallery() | Mengambil data portofolio gallery dan menghilangkan duplikasi data. |
| insert(data) | Menambahkan data portofolio gallery ke database |
|  | ChatCustomerServiceModel | Model | getData(id) | Mengambil data chat customer service dari database |
| getDataChatCustomerService() | Mengambil data chat customer service dan menghilangkan duplikasi data. |
| insert(data) | Menambahkan data chat customer service ke database |
|  | ArtikelFotografiModel | Model | getData(id) | Mengambil data artikel fotografi dari database |
| getDataArtikelFotografi() | Mengambil data artikel fotografi dan menghilangkan duplikasi data. |
| insert(data) | Menambahkan data artikel fotografi ke database |
|  | RegistrasiOrderModel | Model | getData(id) | Mengambil data registrasi order dari database |
| getDataRegistrasiOrder() | Mengambil data registrasi order dan menghilangkan duplikasi data. |
| insert(data) | Menambahkan data registrasi order ke database |
|  | UserModel | Model | getData(id) | Mengambil data user dari database |
| getDataUser() | Mengambil data user dan menghilangkan duplikasi data. |
| insert(data) | Menambahkan data user ke database |
|  | MenuModel | Model | getData(id) | Mengambil data menu dari database |
| getDataMenu() | Mengambil data menu dan menghilangkan duplikasi data. |
| insert(data) | Menambahkan data menu ke database |
|  | WorkshopActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data workshop |
| read(id) | Menyediakan data detail workhshop yang diperoleh dari web service |
|  | IklanLayananJasaActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data iklan layanan jasa |
| read(id) | Menyediakan data detail iklan layanan jasa yang diperoleh dari web service |
|  | PortofolioGalleryActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data portofolio gallery |
| read(id) | Menyediakan data detail portofolio gallery yang diperoleh dari web service |
|  | ChatCustomerServiceActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data chat customer service |
| read(id) | Menyediakan data detail chat customer service yang diperoleh dari web service |
| add() | Menyediakan data yang diperlukan untuk view Activity\_chat\_customer\_service |
| addAction() | Menghubungkan view dengan web service untuk menambahkan data pesan teks |
| \_rules() | Menambahkan peraturan saat akan menambahkan data pesan teks |
|  | ArtikelFotografiActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data artikel fotografi |
| read(id) | Menyediakan data detail artikel fotografi yang diperoleh dari web service |
|  | RegistrasiOrderActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data registrasi order |
| read(id) | Menyediakan data detail registrasi order yang diperoleh dari web service |
|  | MenuActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data menu |
| read(id) | Menyediakan data detail menu yang diperoleh dari web service |
|  | LoginActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data yang dibuthkan untuk login |
| cekLogin() | Memeriksa keseuaian username dan password |
| logout() | Mengarahakn kembali ke view login |
|  | PostWorkshopActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data workshop |
| add() | Menyediakan data yang diperlukan untuk view Activity\_posting\_workshop |
| addAction() | Menghubungkan view dengan web service untuk menambahkan data workshop |
| \_rules() | Menambahkan peraturan saat akan menambahkan data workshop. |
|  | PostIklanLayananJasaActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data Iklan Layanan Jasa |
| add() | Menyediakan data yang diperlukan untuk view Activity\_posting\_iklan\_layanan\_jasa |
| addAction() | Menghubungkan view dengan web service untuk menambahkan data iklan layanan jasa |
| \_rules() | Menambahkan peraturan saat akan menambahkan data iklan layanan jasa |
|  | PostFotoPortofolioActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data portofolio gallery |
| add() | Menyediakan data yang diperlukan untuk view Activity\_posting\_foto\_portofolio |
| addAction() | Menghubungkan view dengan web service untuk menambahkan data foto portofolio gallery |
| \_rules() | Menambahkan peraturan saat akan menambahkan data portofolio gallery |
|  | RespondCustomerActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data chat customer service |
| read(id) | Menyediakan data detail chat customer service yang diperoleh dari web service |
| add() | Menyediakan data yang diperlukan untuk view Activity\_respond\_customer |
| addAction() | Menghubungkan view dengan web service untuk menambahkan data pesan teks |
| \_rules() | Menambahkan peraturan saat akan menambahkan data pesan teks |
|  | PostArtikelFotografiActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data artikel fotografi |
| add() | Menyediakan data yang diperlukan untuk view Activity\_posting\_artikel\_fotografi |
| addAction() | Menghubungkan view dengan web service untuk menambahkan data artikel fotografi |
| \_rules() | Menambahkan peraturan saat akan menambahkan data artikel fotografi |
|  | AddRegistrasiOrderActivity | Controller | index() | Menghubungkan view dengan web service untuk mengambil data registrasi order |
| add() | Menyediakan data yang diperlukan untuk view Activity\_add\_registrasi\_order |
| addAction() | Menghubungkan view dengan web service untuk menambahkan data registrasi order |
| \_rules() | Menambahkan peraturan saat akan menambahkan data registrasi order |
|  | Activity\_workshop | View | - | - |
|  | Activity\_iklan\_layanan\_jasa | View | - | - |
|  | Activity\_portofolio\_gallery | View | - | - |
|  | Activity\_chat\_customer\_service | View | - | - |
|  | Activity\_artikel\_fotografi | View | - | - |
|  | Activity\_registrasi\_order | View | - | - |
|  | Activity\_menu | View | - | - |
|  | Activity\_login | View | - | - |
|  | Activity\_posting\_workshop | View | - | - |
|  | Activity\_posting\_iklan\_layanan\_jasa | View | - | - |
|  | Activity\_posting\_foto\_portofolio | View | - | - |
|  | Activity\_respond\_customer | View | - | - |
|  | Activity\_posting\_artikel\_fotografi | View | - | - |
|  | Activity\_add\_registrasi\_order | View | - | - |

#### **Perancangan Kelas**

Perancangan kelas disini adalah merancang diagram kelas dari kelas-kelas yang telah didefinisikan sebelumnya beserta atribut dan perilakunya.

******

**Gambar 4‑16. Diagram Kelas**

Gambar diatas merupakan diagram kelas yang sudah dirancang dari mulai kelas-kelas analisis sampai penentuan perilaku. Class diagram diatas adalah rancangan untuk aplikasi yang menggunakan web service sebagai pertukaran datanya.

### Pemodelan Berbasis Perilaku

Model perilaku pada dasarnya menggambarkan bagaimana perangkat lunak akan berperilaku  
dalam menanggapi *event-event* yang datang dari arah luar atau bagaimana perangkat lunak akan berperilaku terhadap tindakan yang muncul dari luar [PRE10]. Pemodelan berbasis perilaku akan digambarkan dengan *sequence* diagram.

##### Diagram Sekuen Menampilkan infromasi workshop berupa Foto atau Video

|  |
| --- |
| **Menampilkan infromasi workshop berupa Foto atau Video** |
| **Gambar 4‑17. Diagram Sekuen Menampilkan infromasi workshop berupa Foto atau Video** |
| **Deskripsi** |
| Diagram ini menggambarkan interaksi antara aktor customer dengan sistem dimana setelah memilih menu workhop maka sistem akan menampilkan data workshop. |

##### Diagram Sekuen Menginput pesan berupa teks untuk keluhan ke Customer Service

|  |
| --- |
| **Menginput pesan berupa teks untuk keluhan ke Customer Service** |
| **Gambar 4‑18. Diagram sekuen menginput pesan berupa teks untuk keluhan ke customer service** |
| **Deskripsi** |
| Diagram ini menggambarkan interaksi antara aktor customer dengan sistem dimana setelah memilih menu menu kontak customer service sistem menampilkan data chat lalu customer menginput pesan teks maka sistem akan menyimpannya dan menampilkan pesan tersebut di kolom chat. |

##### Diagram Sekuen Menampilkan artikel fotografi berupa foto atau video

|  |
| --- |
| **Menampilkan artikel fotografi berupa foto atau video** |
| **Gambar 4‑19. Diagram Sekuen Menampilkan artikel fotografi berupa foto atau video** |
| **Deskripsi** |
| Diagram ini menggambarkan interaksi antara aktor customer dengan sistem dimana setelah memilih menu artikel maka sistem akan menampilkan data artikel fotografi. |

##### Diagram Sekuen Menampilkan iklan layanan jasa dokumentasi berupa foto atau video

|  |
| --- |
| **Menampilkan iklan layanan jasa dokumentasi berupa foto atau video** |
| **Gambar 4‑20. Diagram Sekuen Menampilkan iklan layanan jasa dokumentasi berupa foto atau video** |
| **Deskripsi** |
| Diagram ini menggambarkan interaksi antara aktor customer dengan sistem dimana setelah memilih menu iklan layanan jasa maka sistem akan menampilkan data iklan layanan jasa. |

##### Diagram Sekuen Menampilkan portofolio gallery

|  |
| --- |
| **Menampilkan portofolio gallery** |
| **Gambar 4‑21. Diagram Sekuen Menampilkan portofolio gallery** |
| **Deskripsi** |
| Diagram ini menggambarkan interaksi antara aktor customer dengan sistem dimana setelah memilih menu portofolio gallery maka sistem akan menampilkan data portofolio gallery. |

##### Diagram Sekuen Menginput registrasi order jasa dokumentasi

|  |
| --- |
| **Menginput registrasi order jasa dokumentasi** |
| **Gambar 4‑22. Diagram Sekuen Menginput registrasi order jasa dokumentasi** |
| **Deskripsi** |
| Diagram ini menggambarkan interaksi antara aktor customer dengan sistem dimana setelah memilih menu registrasi order maka sistem akan menampilkan form data registrasi order lalu customer menginput order dan menekan button konfirmasi maka sistem akan menyimpan dan menampilkan pemberitahuan order berhasil. |

##### Diagram Sekuen Memposting video atau foto workshop

|  |
| --- |
| **Memposting video atau foto workshop** |
| **Gambar 4‑23. Diagram Sekuen Memposting video atau foto workshop** |
| **Deskripsi** |
| Diagram ini menggambarkan interaksi antara aktor admin dengan sistem dimana setelah memilih menu post workshop maka sistem akan menampilkan form data informasi workshop lalu admin menginput informasi workshop dan menekan button post maka sistem akan menyimpan dan menampilkan pemberitahuan workshop berhasil ditambahkan. |

##### Diagram Sekuen Menginput jawaban berupa teks untuk menjawab keluhan dari Customer

|  |
| --- |
| **Menginput jawaban berupa teks untuk menjawab keluhan dari Customer** |
| **Gambar 4‑24. Diagram Sekuen Menginput jawaban berupa teks untuk menjawab keluhan dari Customer** |
| **Deskripsi** |
| Diagram ini menggambarkan interaksi antara aktor admin dengan sistem dimana setelah memilih menu menu kontak respond customer sistem menampilkan data chat lalu admin menginput pesan teks maka sistem akan menyimpannya dan menampilkan pesan tersebut di kolom chat. |

##### Diagram Sekuen Memposting video atau foto artikel fotografi

|  |
| --- |
| **Memposting video atau foto artikel fotografi** |
| **Gambar 4‑25. Diagram Sekuen Memposting video atau foto artikel fotografi** |
| **Deskripsi** |
| Diagram ini menggambarkan interaksi antara aktor admin dengan sistem dimana setelah memilih menu post artikel fotografi maka sistem akan menampilkan form data informasi artikel fotografi lalu admin menginput informasi artikel fotografi dan menekan button post maka sistem akan menyimpan dan menampilkan pemberitahuan artikel fotografi berhasil ditambahkan. |

##### Diagram Sekuen Memposting video atau foto iklan layanan jasa dokumentasi

|  |
| --- |
| **Memposting video atau foto iklan layanan jasa dokumentasi** |
| **Gambar 4‑26. Diagram sekuen Memposting video atau foto iklan layanan jasa dokumentasi** |
| **Deskripsi** |
| Diagram ini menggambarkan interaksi antara aktor admin dengan sistem dimana setelah memilih menu post iklan layanan jasa maka sistem akan menampilkan form data informasi iklan layanan jasa lalu admin menginput informasi iklan layanan jasa dan menekan button post maka sistem akan menyimpan dan menampilkan pemberitahuan iklan layanan jasa berhasil ditambahkan. |

##### Diagram Sekuen Memposting foto untuk portofolio gallery

|  |
| --- |
| **Memposting foto untuk portofolio gallery** |
| **Gambar 4‑27. Diagram Sekuen Memposting foto untuk portofolio gallery** |
| **Deskripsi** |
| Diagram ini menggambarkan interaksi antara aktor admin dengan sistem dimana setelah memilih menu post foto portofolio maka sistem akan menampilkan form data foto lalu admin menginput foto dan menekan button post maka sistem akan menyimpan dan menampilkan pemberitahuan foto berhasil ditambahkan. |

##### Diagram Sekuen Menampilkan data registrasi order jasa dokumentasi dari customer

|  |
| --- |
| **Menampilkan data registrasi order jasa dokumentasi dari customer** |
|  |
| **Deskripsi** |
| Diagram ini menggambarkan interaksi antara aktor admin dengan sistem dimana setelah memilih menu registrasi order customer maka sistem akan menampilkan data registrasi order customer |

##### 

# DAFTAR PUSTAKA

|  |  |
| --- | --- |
| [ACE19]  [ANA17]  [AYP16]  [BUR19]  [CHE15]  [COD16]  [EAF19]  [GOO20]  [GOO08]  [GUR20]  [INT19]  [JSO20]  [KEL95]  [INT15]  [KUC19]  [LIM15]  [NSM16]  [MAS15]  [IJE10]  [MUH19]  [OSA17]  [PRE10]  [ROS11]  [SAT11]  [SLA18]  [SQU15]  [W3C04] | Acep, H. (2019). Dinasti Motekar Academy.  Akbar, A. N. (2017). Perangkat Lunak Untuk Layanan Warga Berbasis Android Di Lingkungan Rukun Warga Menggunakan Konsep M-Government (Studi Kasus Rw 10 Kelurahan Sukabungah Kota Bandung).  Anggiyasti, Y. P. (2016). Eksplorasi Python Menggunakan Framework Django (Studi Kasus : Sistem Perpustakaan Teknik Informatika UNPAS).  Burbeck, S. (1992). *Applications Programming in Smalltalk-80 (TM) How to use Model-View-Controller (MVC). Vol. 2.5.* ParcPlace Systems, Inc.  Chen, H. (2015). *Management of contextual information for data.*  Codepath. (2016). *Consuming APIs with Retrofit*. Retrieved from Codepath: https://guides.codepath.com/android/Con  Febriyanti, E. A. (2019). Rancang Bangun Aplikasi Pengelolaan Data Anggota Koordinator Olahraga Mahasiswa Universitas Pasundan Bandung.  Google, JetBrains. (2020, 11 11). *Android Studio Intro.* Retrieved from developer.android.com: https://developer.android.com/studio/intro  Google.Inc. (2008). *Github of Google Gson*. Retrieved from Github.com: https://github.com/google/gson  Guru99. (2020, 11 10). *Unit Testing Guide.* Retrieved from Guru99: https://www.guru99.com/unit-testing-guide.html  Ika Oktavia Suzanti, N. F. (2019). International Conference on Science and Technology. *REST API Implementation on Android Based Monitoring*.  JSON. (2020, 11 11). *Intoducing JSON.* Retrieved from json.org: https://www.json.org/json-en.html  Kelleher, K. C. (1995). *Cause and Effect Diagram : Plain and.* Joiner Associates Inc USA.  Kirthika.B, P. V. (2015). International Journal of Trend in Research and Development. *Android Operating System: A Review*, 260.  Kucherenko, S. M. (2019). *Learn Spring for Android Application Development.* Brimingham: Packt.  Lim, S. H. (2015). Experimental Comparison of Hybrid and Native Applications for Mobile System. *International Journal of Multimedia and Ubiquitous Engineering*.  Malikhah, N. S. (2016). Eksplorasi Python Untuk Penyesuaian.  Maskov. (2015). *Implementing REST Client for Android.* Helsinki: Metropolia University of Applied Sciences.  Md. Rashedul Islam, M. R. (2010). International Journal of Engineering & Technology IJET-IJENS Vol: 10 No: 06. *Mobile Application and Its Global Impact*, 104.  Muhammad, F. I. (2019). Rancang Bangun Aplikasi Pemesanan Tiket Di Kolam Renang Failda Sukanagara Cianjur Selatan.  Osahon, G. (2017). *Consuming REST API using Retrofit Library in Android.* Retrieved from android.jlelse.eu: https://android.jlelse.eu/: https://android.jlelse.eu/consuming-rest-api-using-retrofit-library-in-android-ed47aef01ecb  Pressman, R. S. (2010). *Software Engineering Practitioner’s Approach. 2010. 7th Ed.* New York: McGraw-hill.  Rosmala, D. M. (2011). *Komparasi Framework MVC (Codeigniter, dan CakePHP) pada Aplikasi Berbasis Web (Studikasus: Sistem Informasi Perwalian Di Jurusan Informatika Institut Teknologi Nasional).* Bandung: ITENAS.  Satzinger. (2011). *Systems Analysis and Design in a Changing World, 6th Edition. Course Technology.*  Slam, B. E. (2018). Pembangunan Aplikasi Pakan Sapi Berbasis Android.  Square Inc. (2015). *Github of Square Moshi.* Retrieved from Github.com: https://github.com/square/moshi  W3C. (2004). W3C Working Group Note. *Web Service Architecture*. |