# I

# TINJAUAN PUSTAKA

## Penelitian terdahulu

Penelitian dengan tema *food waste* (sampah makanan) sudah dilakukan penelitian yang bersumber dari jurnal internasional/nasional, tugas akhir maupun aplikasi yang sudah terealisasi. Berikut adalah hasil penelitian terkait tema *food waste* (sampah makanan):

### Prastowo, Dirga Enggar 2017

Aplikasi *web food* yang dirancang dan dibuat oleh prastowo bertujuan untuk mengurangi sampah dari sisa makanan di Kota Surabaya. Pengembangan dilakukan dengan menggunakan *platform* *web* dan teknologi *google maps.* Studi menunjukkan bahwa *smartphone* memiliki kelemahan yang belum dimanfaatkan secara optimal (Prastowo, 2017).

### Ayesha Anzer, Hadeel A. Tabaza, dan Wedad Ahmed 2018

Penelitian yang dilakukan di negara United Arab Emirates ini yaitu dengan mengembangkan *aplikasi food waste reduction* berbasis android. munculnya penelitian ini dikarenakan banyak tempat makan (café, hotel, restoran, dll) yang membuang makanan baru padahal makanan tersebut belum disentuh oleh konsumen. Tujuan utama aplikasi ini yaitu sebagai penghubung orang yang memerlukan makanan dengan tempat makan (café, hotel, restoran, dll) yang makanannya masih tersisa banyak tapi masih layak dimakan lagi, belum menggunakan *location base service* untuk menemukan lokasi restoran menjadikan ini sebagai kelemahan dari aplikasi ini (Ayesha Anzer, 2018).

### Grace Phiri, Pip Trevorrow 2019

Dalam penelitian ini, aplikasi dibuat dengan berbasis android yang berkonsep mangkuk kadaluarsa dimana pengguna memasukkan tanggal kadaluarsa makanan dan mengeluarkan pemberitahuan saat makanan mendekati tanggal kadaluarsa. Kelemahan dari aplikasi ini adalah pemberitahuan dari kadaluarsanya makanan sering tidak dilihat (Grace Phiri., 2019).

### Pengembangan Aplikasi *Food Sharing* berbasis Android

Penelitian terkait mengurangi sampah makanan dan mengurangi kasus orang kelaparan. Pengembangan aplikasi ini menggunakan platform *mobile*. Aplikasi ini menjadi penghubung antara orang yang memiliki makanan berlebih dengan orang yang mau menerima kelebihan makanan agar makanan tidak terbuang sia-sia. Akan tetapi, aplikasi ini masih memiliki kekurangan yaitu dapat mengunggah foto selain foto makanan (Abdul Rozaq, et al., 2019).

### Aplikasi Olio

Merupakan aplikasi dari Inggris yang bertujuan untuk mengurangi limbah makanan, hal ini dapat Ini dilakukan dengan menghubungkan mereka yang memiliki kelebihan makanan dengan mereka yang membutuhkan atau ingin mengonsumsi makanan semacam itu. Mereka yang menyumbangkan kelebihan makanan dapat berupa perorangan atau perusahaan seperti  [restoran](https://translate.googleusercontent.com/translate_c?client=srp&depth=1&hl=id&prev=search&rurl=translate.google.com&sl=en&sp=nmt4&tl=id&u=https://en.m.wikipedia.org/wiki/Restaurant&usg=ALkJrhiKDPO-u7lSg5WbfFTj2PLZzGKqdw) , [kantin perusahaan](https://translate.googleusercontent.com/translate_c?client=srp&depth=1&hl=id&prev=search&rurl=translate.google.com&sl=en&sp=nmt4&tl=id&u=https://en.m.wikipedia.org/wiki/Cafeteria&usg=ALkJrhgGC0-0VMJcrDvYz38LgA4WoC--Aw) dan [fotografer makanan](https://translate.googleusercontent.com/translate_c?client=srp&depth=1&hl=id&prev=search&rurl=translate.google.com&sl=en&sp=nmt4&tl=id&u=https://en.m.wikipedia.org/wiki/Food_photography&usg=ALkJrhidRRCCAAC05G82EowwPoADvRIxzA) dan sumbangan dapat dilakukan secara berulang. Aplikasi ini memiliki kelemahan yaitu dapat mengunggah foto selain foto makanan di beranda untuk berbagi makanan (OLIO, 2020).

## Teori terkait

### Pengertian Java

Java adalah sebuah bahasa pemrograman scripting yang sering digunakan dalam pembuatan aplikasi berbasis handphone dan juga dapat digunakan untuk menyediakan akses objek yang disisipkan di aplikasi lain. Java berfungsi sebagai penambah tingkah laku agar widget dapat tampil lebih atraktif (Sallaby, et al., 2015).

### Pengertian Extensible Markup Language (XML)

Extensible Markup Language (XML) merupakan kumpulan aturan untuk mendesain format teks, sehingga format teks lebih terstruktur dan lebih mudah dibaca oleh komputer. Pada dasarnya XML merupakan penyusun informasi, sehingga sebuah informasi menjadi terstruktur dan dapat dibaca dengan mudah oleh komputer serta informasi tersebut mudah diterima oleh pengguna. Adapun fungsi XML adalah sebagai media pembawa data/informasi (Sallaby, et al., 2015).

### Penegertian Android

Android adalah sebuah *toolkit software* yang baru untuk perangkat bergerak yang dibuat oleh Google dan *Open Handset Alliance*. Dalam beberapa tahun, android diharapkan dapat ditemukan dalam jutaan *handphone* dan berbagai perangkat bergerak, membuat android menjadi *platform* utama untuk pengembang aplikasi. Android adalah *platform* yang menggabungkan beberapa hal berikut:

1. Android merupakan sebuah platform yang berbasis linux dan open source. Pembuat handset menyukai hal ini karena bersifat gratis.
2. Sebuah arsitektur berbasis komponen. Bagian dari aplikasi android dapat digunakan sebagai bahan lain oleh developer.
3. Banyak built-in service yang tidak biasa. Servis berdasarkan lokasi menggunakan GPS atau cell tower triangulation yang membuat pengalaman pemakai terjadi bergantung lokasi (Sallaby, et al., 2015).

### Pengertian **Model-View-Presenter** (MVP)

**Model-View-Presenter** (MVP) adalah sebuah konsep arsitektur pengembangan aplikasi yang memisahkan antara tampilan aplikasi dengan proses bisnis yang bekerja pada aplikasi. Arsitektur ini akan membuat pengembangan aplikasi kita menjadi lebih terstuktur, mudah di-test dan juga mudah di-maintain.

Berikut penjelasan masing-masing layer pada **MVP**:

1. ***View*** merupakan layer untuk menampilkan data dan interaksi ke user. *View* biasanya berupa Activity, Fragment atau Dialog di Android. *View* ini juga yang langsung berkomunikasi dengan user.
2. **Model** merupakan *layer* yang menunjuk kepada objek dan data yang ada pada aplikasi.
3. ***Presenter*** merupakan *layer* yang menghubungkan komunikasi antara Model dan *View*. Setiap interaksi yang dilakukan oleh user akan memanggil *Presenter* untuk memrosesnya dan mengakses Model lalu mengembalikan responnya kembali kepada *View* (Sianturi, 2018).

### Pengertian PostgreSQL

PostgreSQL merupakan salah satu Object Relational Database Management System (ORDBMS) yang bersifat open source, yang berarti bahwa source code dari PostgreSQL dapat digunakan secara bebas. PostgreSQL mendukung Structured Query Language (SQL) yang memiliki kemampuan transactions, subqueries, triggers, dan lain-lain. PostgreSQL hadir dengan banyak fitur yang ditujukan untuk membantu pengembang membangun aplikasi, administrator untuk melindungi integritas data dan membangun lingkungan yang toleran terhadap kesalahan, dan membantu Anda mengelola data tidak peduli seberapa besar atau kecil kumpulan data tersebut. Selain gratis dan open source, PostgreSQL sangat dapat dikembangkan. Misalnya, kita dapat menentukan tipe data kita sendiri, membangun fungsi kustom, bahkan menulis kode dari bahasa pemrograman yang berbeda tanpa mengkompilasi ulang database kita (PostgreSQL, 2021).

### Pengertian Hasura GraphQL Engine

Hasura GraphQL Engine adalah sebuah server GraphQL ultra-cepat yang menyediakan **API GraphQL instan dan**real-time**diatas Postgres,** dengan **[pemicu](https://hasura.github.io/graphql-engine/event-triggers.html)**[webhook](https://hasura.github.io/graphql-engine/event-triggers.html) pada event basis data, dan **[skema jarak jauh](https://hasura.github.io/graphql-engine/remote-schemas.html)** untuk mengimplementasi logika bisnis (Hasura GraphQL Engine, 2021).

### Pengertian *application*

*Application Programming Interface* (API) merupakan suatu dokumentasi yang terdiri dari *Interface*, fungsi, kelas, struktur dan sebagainya untuk membangun sebuah perangkat lunak. Dengan adanya API ini, maka memudahkan programmer untuk membongkar suatu *software*, kemudian dapat dikembangkan atau diintegrasikan dengan perangkat lunak yang lain

*Application Programming Interface* (API) pada web service fokus sebagai upaya untuk menguji kemampuan pada tiap unit. Unit yang diuji secara spesifik akan digabungkan dengan unit lain hingga menjadi suatu *system* terintegrasi. Aplikasi *mobile* membutuhkan dukungan API sebagai penghubung dengan basis data, dimana aplikasi *mobile* dengan basis data tersebut tidak saling berhubungan secara langsung (Ramadhani, 2015).

### *Entity Relational Diagram(ERD)*

*ERD (Entity Relationship Diagram)* adalah model teknik pendekatan yang menyatakan atau menggambarkan hubungan suatu model. Didalam hubungan ini tersebut dinyatakan yang utama dari *ERD* adalah menunjukan objek data *(Entity)* dan hubungan *(Relationship),* yang ada pada *Entity* berikutnya (Fridayanthie & Mahdiati, 2016).

Menurut Simarmata dalam buku yang berjudul perancangan basisdata Entity RelationShip Diagram (ERD) adalah alat pemodelan data utama dan akan mambantu mengorganisasi data dalam suatu proyek ke dalam entitas-entitas dan menentukan hubungan antar entitas. Proses memungkinkan analis menghasilkan struktur basis data dapat disimpan dan diambil secara efisien. Simbol-simbol dalam ERD (Entity Relationship Diagram) adalah sebagai berikut:

1. Entitas: suatu yang nyata atau abstrak yang mempunyai karakteristik dimana kita akan menyimpan data.
2. Atribut: ciri umum semua atau sebagian besar instansi pada entitas tertentu.
3. Relasi: hubungan alamiah yang terjadi antara satu atau lebih entitas.
4. Link: garis penghubung atribut dengan kumpulan entitas dan kumpulan entitas dengan relasi (Simarmata, 2010).

### *Use Case Diagram*

*Use case* diagram merupakan pemodelan untuk kelakuakn (behavior) *system* informasi yang akan dibuat. Use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam *system* informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut (Fridayanthie & Mahdiati, 2016). Simbol-simbol yang digunakan dalam Use Case Diagram yaitu:

**Tabel 2.1.** Usecase Diagram

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama Simbol** | **Keterangan** |
| https://lh4.googleusercontent.com/IUKnp-feZXrh2twyGIKnPH4CSx3FLENHRcoePdBp8t9cF0oCWaScru1EMs5O6sKya1l02_XeWjKD6iWsBpa1w-lSBAasouJAMKNEw3oOTuaD2ODCSMl5O46ppUeEP7M9cDXGAadi49iEGJIaTg | Use Case | Urutan aksi-aksi yang ditampilkan oleh *system*, menghasilkan suatu hasil yang terukur bagi suatu actor |
| https://lh5.googleusercontent.com/F7vUf_rfOa7nRdhypfizZIox1e-WYl1EXk9fgS5zbFk2uBJvFI1YENf972xGwo894-Y0LwlWYMX96TUIjrldE4AnKyn5lWl1ERBvOZjKWfwbxbon9-dJzGZYbBi3kKWvYqtqf9vUWuFLUCdxnQ | Aktor / Pengguna | Segala sesuatu yang  Berinteraksi dengan *system*  untuk pertukaran informasi. |
| https://lh4.googleusercontent.com/UgC6Xp59ga_DtETG-f_-G6tgalrINhgcQWX3IJGg_TWfQUB3DShu_FXTaakzUeMscTjibsyHPbKdc20vz12iXrZpOqnkqc0UEVsJtVomcOKr01wsD-fCbSymjeRkg7J_J4vx4is4U_6ex3Uf4w | Generalisasi | Hubungan generalisasi dan spesialisasi (umum - khusus) antara dua buah use case dimana fungsi yang satu merupakan fungsi yang lebih umum dari lainnya arah panah mengarah pada usecase yang menjadi generalisasinya (umum) . |
| https://lh5.googleusercontent.com/DMqDqObgZyTpdP5YGqmyJkbI-AH8T-7hCtLTi1LKsLhLG79oKYZXuF42jnaXvtG2vEXDTWHzYmRFq32jSm-rWk8aIFGqKpx254uMqXasjjJ-gE7S-SMj5Qoxrm87QaSt6j5POKU7tjSV1M3f8A | Assosiasi | Asosiasi adalah komunikasi antara aktor dan use case yang berpartisipasi pada use case diagram atau use case yang memiliki interaksi dengan aktor. Asosiasi merupakan simbol yang digunakan untuk menghubungkan link antar element. |
| https://lh4.googleusercontent.com/0e5HoNjRyrbZPO6Wonjia37D9G1dRegY1bJ_lssIF7q2mHN1rUft3oeBNvX7M8FU1wFMv06fNPdiEW_X_A8xYR3fbhau2p3lh4zMQuN0mze1MEUxo1u4oPr8J9S8yDnLXkMV368G5gRtMXdx5w | *Include* | Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan membutuhkan use case ini untuk menjalankan fungsinya |
| https://lh3.googleusercontent.com/EXenpvRoF0yY1IgGTCr1gh1Pr-Neu3bBAS0-6AaXmIL4ZqwhmJX912h79glBJPsKaFUfm-lDVhLZP7zL2NE8w09aqe826mhzyPyDvgsxOiJAdyI8ySowXwyy70l96tq-U9NGYsD08BKzdzxrpQ | *Extend* | Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri meski tanpa use case tambahan itu arah |

### Twilio

Twilio adalah sebuah *platform* berbasis *cloud* yang menawarkan pengembang *API* kuat untuk layanan telepon untuk membuat dan menerima panggilan telepon. Dengan menggunakan *API* kita dapat mengirim dan menerima pesan teks pemrograman yang sangat nyaman bagi pengembang yang perlu menggunakan layanan telepon dan pesan *SMS* misalnya untuk memvalidasi akun menggunakan ponsel dan *reset password* (Twilio, 2021)*.*

### *Location Based Service*(LBS)

*Location Based Service* (LBS) merupakan layanan informasi berbasis lokasi yang dapat diakses menggunakan perangkat bergerak melalui jaringan Internet dan seluler serta memanfaatkan kemampuan penunjuk lokasi pada piranti *mobile*. Dengan kata lain LBS merupakan kemampuan sebuah perangkat bergerak dengan bantuan GPS untuk menunjukkan suatu letak lokasi*. Location Based Service* (LBS) memberikan kemungkinan komunikasi dan interaksi dua arah. Oleh karena itu pengguna memberitahu penyedia layanan untuk mendapatkan informasi yang dia butuhkan, dengan referensi posisi pengguna tersebut (Iqbal, et al., 2015).

### PHP *(Hypertext Preprocessor).*

PHP *(Hypertext Preprocessor)* adalah sebuah bahasa pemrograman server *side scripting* yang bersifat *open source.* Sebagai sebuah *scripting language*, PHP menjalankan instruksi pemrograman saat proses *runtime*. Hasil dari instruksi tentu akan berbeda tergantung data yang diproses.PHP merupakan bahasa pemrograman *server-side*, maka script dari PHP nantinya akan diproses di server (PHP, 2021).

### *Activity diagram*

Activity Diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah *system* atau proses bisnis (Fridayanthie & Mahdiati, 2016). Simbol-simbol yang digunakan dalam activity Diagram yaitu:

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Simbol** | | **Keterangan** |
| 1. |  |  | Kondisi Awal  Menunjukan awal dari suatu diagram aktivitas |
| https://lh5.googleusercontent.com/jHu-NYaHde3tQ43_YlJVPwiP7-Klhen28wZrqmQsuBGBHDGSx6dMNC54llwJwSMxvQ7TI56TcNNthOygseXIbrjs9OThsbgZnmKwPIFDXp0eQuRWlspH5pkbUBwpaDV06w3uvqlLZBsIcavCEQ | |
| 2. | https://lh6.googleusercontent.com/9HjUNmaCyaTk3EXVYwxWKXiQ8t37gSIMKKB2uvVnqJSp_WQtJRIV-OwSqYT0EAfaSTTquFqhSVqJXUcJ6q3x6uTzlXwS3yQoQMuQSLWaaVrNLqT74Yszo0HFOauhjiQSzoEG5c_IeyI2eQ4vRA | | Kondisi Akhir  Menunjukan akhir dari suatu diagram aktivitas |
| 3. | https://lh4.googleusercontent.com/DY0Q93ZTsWJWpFARwssvcpyWcFMCDEp3mnhpssCuUUp_dE36F-X9lwYRD9JLTJhJ0tO8SxdfWczDCJpGAz4Vb6y3HJ59jqglCS0YcN3XvkiHfp_1KzN_q-vfAEXuzv_vS4Pe9xEZXL4tAS4-Ig | | Kondisi Transisi  Menunjukan kondisi transisi antar aktivitas |
| 4. | https://lh4.googleusercontent.com/SY2j0XyIZ2RXsWnsEcc_mNXaCkTV6nDWQwf5oQeIpmPTUYAH1aaC6z4sQBoOnNkCIxwUlsjaTclQQgCIPdT8KpytMrL7E6Ct9i9IAxrweJL1IS3vKNdM9rxCtcx__FbXfNj-vztoYrYx7EOUvQ | | Swimlame  Menunjukan aktor dari diagram aktivitas yang  dibuat |
| 5. |  | | Aktivitas menunjukan aktivitas -aktivitas yang terdapat pada diagram aktivitas |
|  |
| 6. |  | | Pengecekan Kondisi  Menunjukan pengecekan terhadap suatu kondisi |

**Tabel 2.2** Activity Diagram

# 