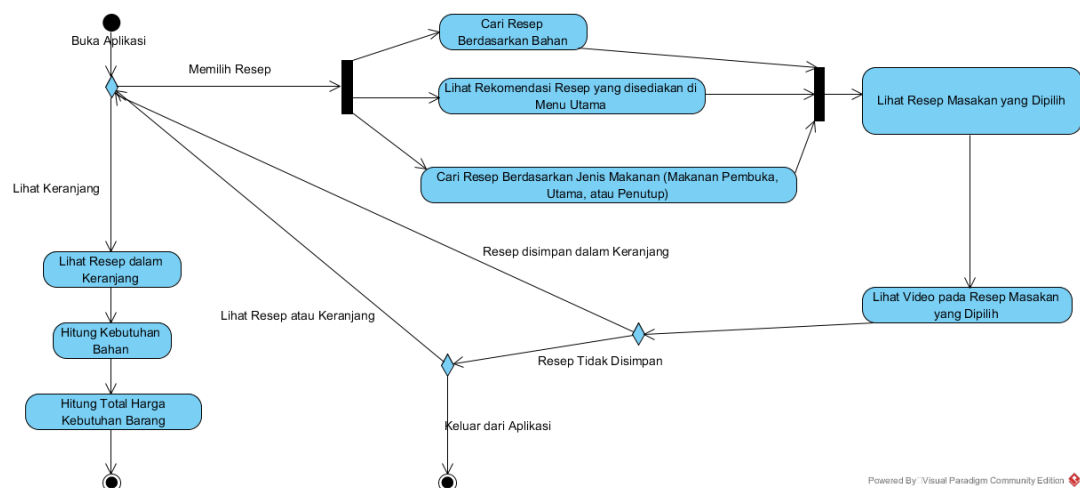


## BAB III

### IMPLEMENTASI PROGRAM

Sesuai dengan tahapan-tahapan pengembangan perangkat lunak yang tertera pada SDLC, maka penulis melakukan serangkaian kegiatan untuk menunjang pengembangan aplikasi resep masakan berbasis Android ini. Karena menggunakan SDLC Model Spiral, maka penulis akan melakukan beberapa tahapan yaitu identifikasi, desain, konstruksi dan pembangunan, serta evaluasi. Evaluasi akan dibahas pada Bab IV (Uji Coba dan Hasil Percobaan) Adapun alur dari aplikasi yang akan dibuat dijelaskan pada Gambar 3.1



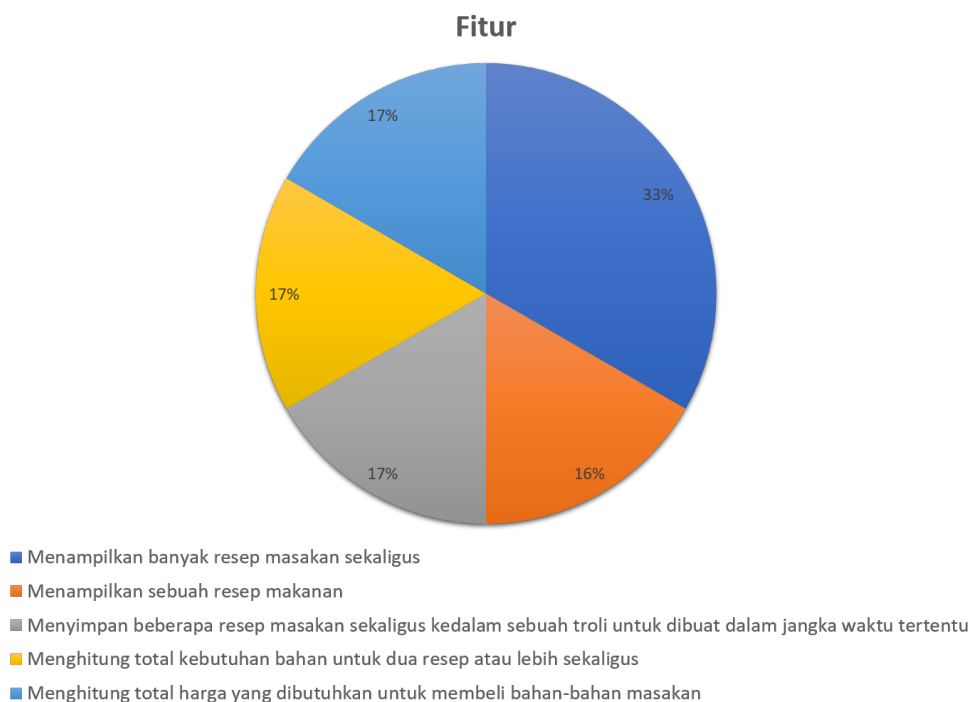
**Gambar 3.1:** Alur Kerja Aplikasi

#### 3.1 Identifikasi

Penulis mengumpulkan beberapa informasi yang dibutuhkan untuk membangun aplikasi ini dengan cara menyebarkan beberapa kuisisioner kepada ahli di bidang masakan, baik itu chef, asisten chef, serta pemilik usaha dibidang kuliner untuk mengetahui fitur apa sajakah yang seharusnya ada dalam aplikasi berbasis Android ini.

Penulis juga tidak lupa meminta pandangan kepada beberapa kaum awam yang berasal dari berbagai kalangan untuk mengetahui tentang pendapat masyarakat tentang fitur apa saja yang seharusnya terdapat pada aplikasi resep masakan ini.

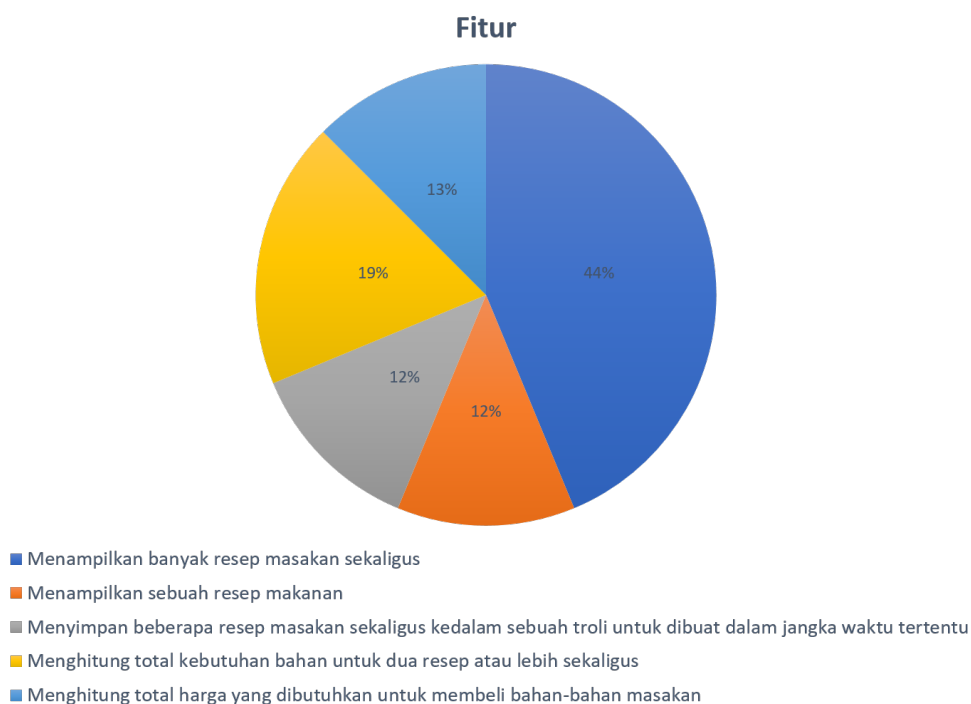
Dari 4 responden yang merupakan para ahli, dimana 2 dari 4 responden tersebut adalah chef, satu orang asisten chef, dan seorang pemilik usaha kuliner, fitur menampilkan banyak resep masakan sekaligus merupakan fitur yang paling banyak dipilih dengan nilai 2. Sedangkan fitur lain yang diusulkan penulis seperti menampilkan sebuah resep makanan, menyimpan beberapa resep masakan sekaligus kedalam sebuah troli untuk dibuat dalam jangka waktu tertentu, menghitung total kebutuhan bahan untuk dua resep atau lebih sekaligus, serta menghitung total harga yang dibutuhkan untuk membeli bahan-bahan masakan mendapatkan masing-masing nilai 1.



**Gambar 3.2:** *Pie Chart* Hasil Kuisisioner Ahli

Untuk responden yang berasal dari kaum awam yang berjumlah 9 orang, dimana 2 dari 9 orang responden berprofesi sebagai guru, satu orang berprofesi sebagai

ibu rumah tangga, sedangkan sisanya adalah seorang karyawan swasta, fitur menampilkan banyak resep masakan sekaligus merupakan fitur yang paling banyak dipilih dengan nilai 7. Sementara fitur menghitung total kebutuhan bahan untuk dua resep atau lebih sekaligus berada pada peringkat 2 dengan nilai 3. Sedangkan fitur lain yang diusulkan penulis seperti menampilkan sebuah resep makanan, menyimpan beberapa resep masakan sekaligus kedalam sebuah troli untuk dibuat dalam jangka waktu tertentu, serta menghitung total harga yang dibutuhkan untuk membeli bahan-bahan masakan mendapatkan masing-masing nilai 2.



**Gambar 3.3:** *Pie Chart* Hasil Kuisisioner Awam

Masyarakat awam yang menjadi responden pada kuisisioner yang telah disebar oleh Penulis menyertakan beberapa pendapat maupun ide tentang fitur apa yang seharusnya dimiliki oleh sebuah sistem aplikasi resep masakan berbasis Android, diantaranya adalah:

- Memberikan deskripsi bahan-bahan makanan yang dibutuhkan juga dan bisa

didapatkan dimana

- Dalam sebuah resep akan lebih baik jika ada detail resep ini untuk berapa porsi, sehingga orang awam dapat mengetahui berapa takaran bahan-bahan masakan yang akan digunakan untuk porsi tertentu
- Menampilkan video tutorial masak, dan alternatif bahan masakan (bahan masakan pengganti untuk alergi misalnya)
- Aplikasi disertai video pembuatan masakan dan resepnya.

Sedangkan dari usulan maupun ide yang datang dari para professional atau ahli di bidang ini adalah sebagai berikut:

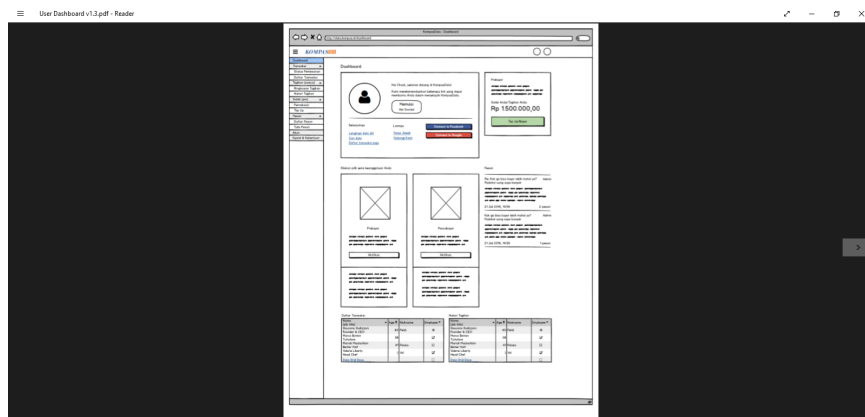
- Memberikan pilihan apa saja yang dapat dimasak dari suatu bahan-bahan tertentu
- Mengetahui kapan bahan akan habis
- Menampilkan metode memasak yang jelas sesuai dengan standar makanan yang akan dibuat
- Informasi jumlah takaran dalam sebuah resep haruslah jelas
- Mungkin bisa ditambah dengan foto atau video pembuatannya
- Memberikan tips dan trik
- Memberikan keterangan pada tulisan berbahasa asing untuk menambah pengetahuan. Contohnya *Mise En Place* artinya preparasi

Melalui data yang telah terhimpun, penulis memutuskan untuk memilih fitur yang banyak dipilih oleh responden, baik ahli maupun kaum awam, untuk diimplementasikan pada aplikasi yang akan dibuat oleh penulis, yaitu menampilkan banyak

resep masakan sekaligus, menghitung total kebutuhan bahan untuk dua resep atau lebih sekaligus, menyimpan beberapa resep masakan sekaligus kedalam sebuah troli untuk dibuat dalam jangka waktu tertentu, serta menghitung total harga yang dibutuhkan untuk membeli bahan-bahan masakan. Sementara, fitur tambahan yang diinisiasi oleh penulis adalah fitur mencari resep berdasarkan bahan masakan yang akan dibuat.

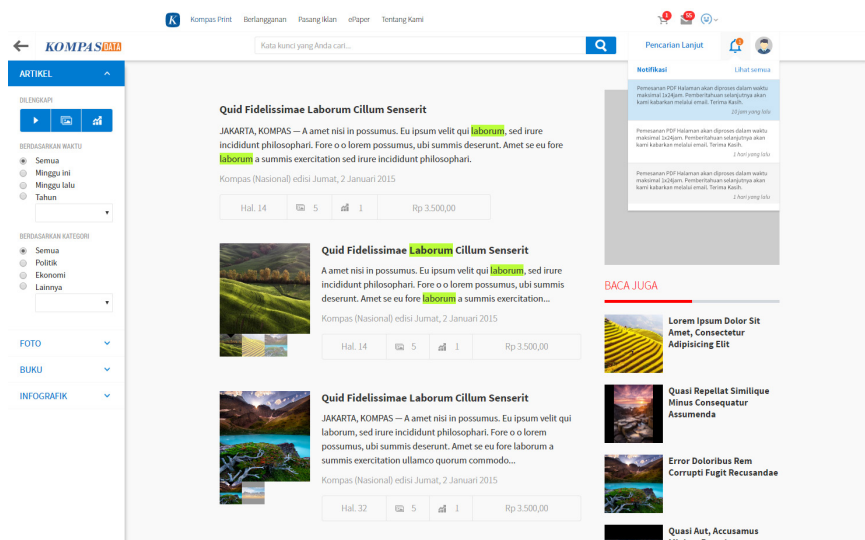
### 3.2 Desain

Dalam mengembangkan aplikasi ini, penulis mengembangkan desain aplikasi terlebih dahulu, mulai dari desain *Use Case*, *Entity Relationship Diagram (ERD)*, *Class Diagram*, serta *Activity Diagram*. Selain itu, penulis juga membuat desain *wireframe* dan *mock-up*. Penulis memutuskan untuk membuat kedua desain tersebut untuk memudahkan pengguna dalam mengeksekusi desain antarmuka dari aplikasi ini. Berdasarkan pengalaman penulis dalam mengikuti program Praktek Kerja Lapangan pada PT Kompas Media Nusantara (Harian Kompas), dimana penulis berperan sebagai *Front-End Developer*, perbedaan dari *wireframe* dan *mock-up* sendiri adalah pada tampilannya. *Wireframe* mengedepankan *User Experience* dari sebuah laman aplikasi, yaitu alur kerja dari laman aplikasi tersebut. Biasanya tampilan yang disuguhkan oleh *wireframe* tidak terlalu menarik dan hanya terlihat seperti kerangka dari tampilan aplikasi tersebut.



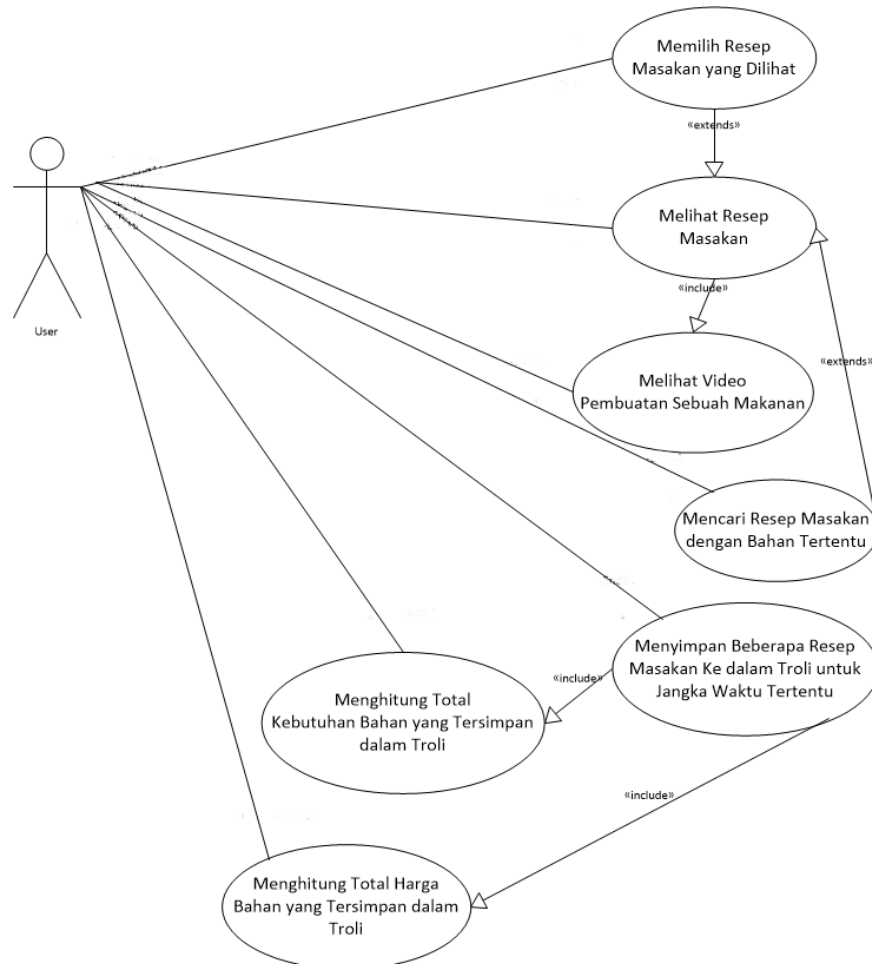
**Gambar 3.4:** Contoh *Wireframe*

Sedangkan *mock-up* adalah sebuah rancangan laman antarmuka aplikasi yang bersifat dekoratif dengan dekorasi yang sudah didefinisikan secara jelas oleh pembuatnya. *Mock-up* menonjolkan sisi *User Interface* dari sebuah laman antarmuka aplikasi. Berbeda dengan *wireframe* yang hanya mengedepankan alur kerja dari sebuah laman, *mock-up* adalah pengembangan lebih lanjut dari *wireframe* sehingga sisi estetika dari sebuah rancangan antarmuka aplikasi lebih menonjol. *Mock-up* memudahkan seorang Front-End Developer untuk mengembangkan dan mengeksekusi sebuah laman antarmuka aplikasi.



**Gambar 3.5:** Contoh *Mock-Up*

### 3.2.1 Use Case Diagram

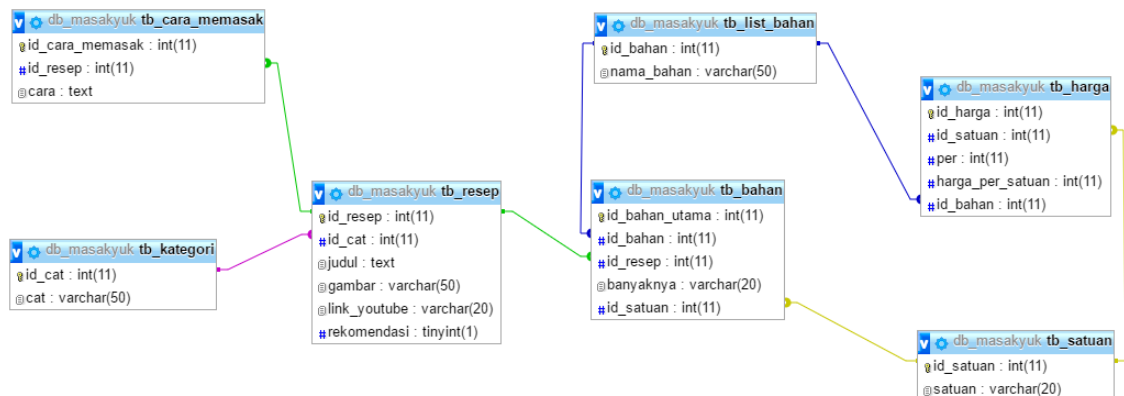


*Use Case Diagram* tersebut menunjukkan bagaimana pengguna aplikasi (*user*) berinteraksi dengan aplikasi yang dikembangkan oleh peneliti. Dalam *use case* tersebut dijelaskan bahwa *user* dapat memilih resep masakan yang ingin dilihat pada menu utama untuk kemudian melihat resep masakan sekaligus melihat video cara membuat makanan yang terdapat dalam resep tersebut. Selain itu, *user* juga dapat mencari resep masakan berdasarkan bahan tertentu sesuai dengan keinginan *user*. Setelah dilihat, *user* dapat menyimpan resep masakan tersebut ke dalam Trolis, dimana pada aplikasi ini disebut sebagai *Wishlist*. *User* dapat menyimpan sampai 3 resep masakan sekaligus dalam *Wishlist*. Resep yang ada di dalam *wishlist* dapat dihitung total ke-

butuhan bahannya dan juga dapat menghitung total kebutuhan harga untuk membeli bahan-bahan tersebut.

### 3.2.2 Entity Relationship Diagram

ERD pada aplikasi ini berfungsi untuk menyimpan resep, kategori resep, bahan-bahan yang ada pada resep, cara memasak, serta harga pada tiap-tiap bahan yang tercantum dalam resep. Untuk itu, penulis membuat beberapa tabel untuk menunjang penyimpanan data dari tiap elemen tersebut.



**Gambar 3.6:** Desain ERD Aplikasi yang Dikembangkan Penulis

Penulis menyediakan tabel resep untuk menyimpan data-data yang berhubungan dengan resep. Kemudian, penulis juga menyediakan tabel bahan untuk menyimpan bahan-bahan yang terdapat pada sebuah resep serta tabel list atau daftar bahan untuk menyimpan data bahan-bahan secara umum yang dijadikan sebagai referensi dari tabel bahan. Penulis juga menyediakan tabel harga dan satuan untuk melengkapi tabel bahan. Tabel satuan dibuat karena adanya kemungkinan sebuah bahan dapat memiliki banyak jenis satuan, sedangkan tabel harga dibuat untuk menghitung harga bahan yang tertera dalam tabel bahan dengan kalkulasi yang berbeda bergantung pada satuannya. Selain itu, penulis juga membuat tabel cara memasak untuk



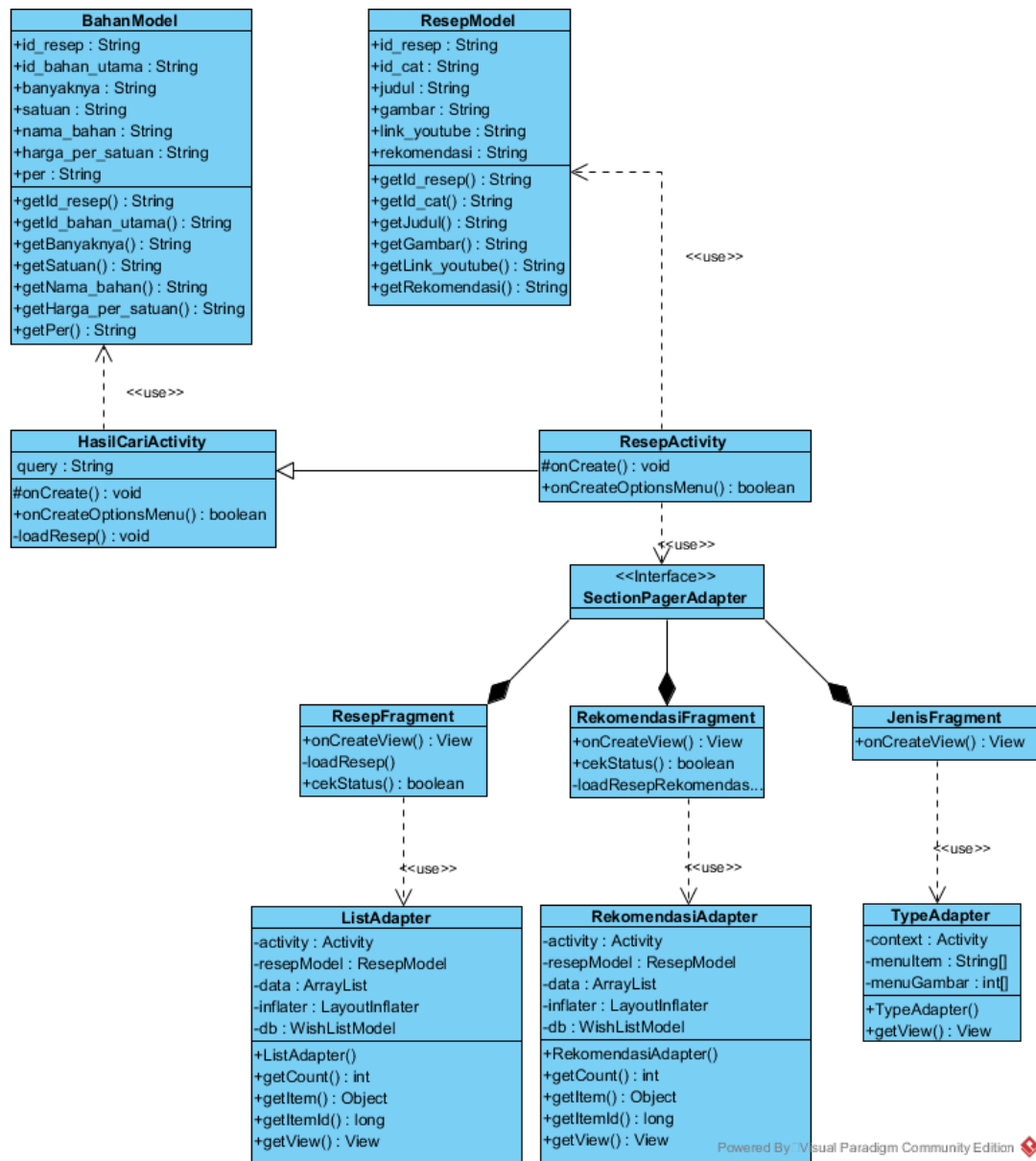
menyimpan cara-cara memasak berdasarkan resep yang ada. Terakhir, penulis membuat tabel kategori untuk mengelompokkan resep berdasarkan jenisnya yaitu Makanan Pembuka, Makanan Utama, dan Makanan Penutup.

Tabel resep berelasi *one-to-many* dengan tabel cara memasak dan tabel bahan karena satu resep dapat memiliki banyak cara memasak dan banyak bahan. Sedangkan tabel kategori berelasi *one-to-many* juga karena satu kategori dapat memiliki banyak resep. Tabel list bahan berelasi *one-to-many* dengan tabel bahan karena bahan yang terdapat pada tabel list bahan dapat dimasukkan lebih dari satu kali pada tabel bahan yang secara langsung menunjang tabel resep. Tabel Satuan berelasi *one-to-many* dengan tabel harga dan tabel bahan karena tabel harga dapat memiliki banyak satuan serta tabel bahan yang juga sama, yakni dapat memiliki lebih dari satu satuan.

### 3.2.3 Class Diagram

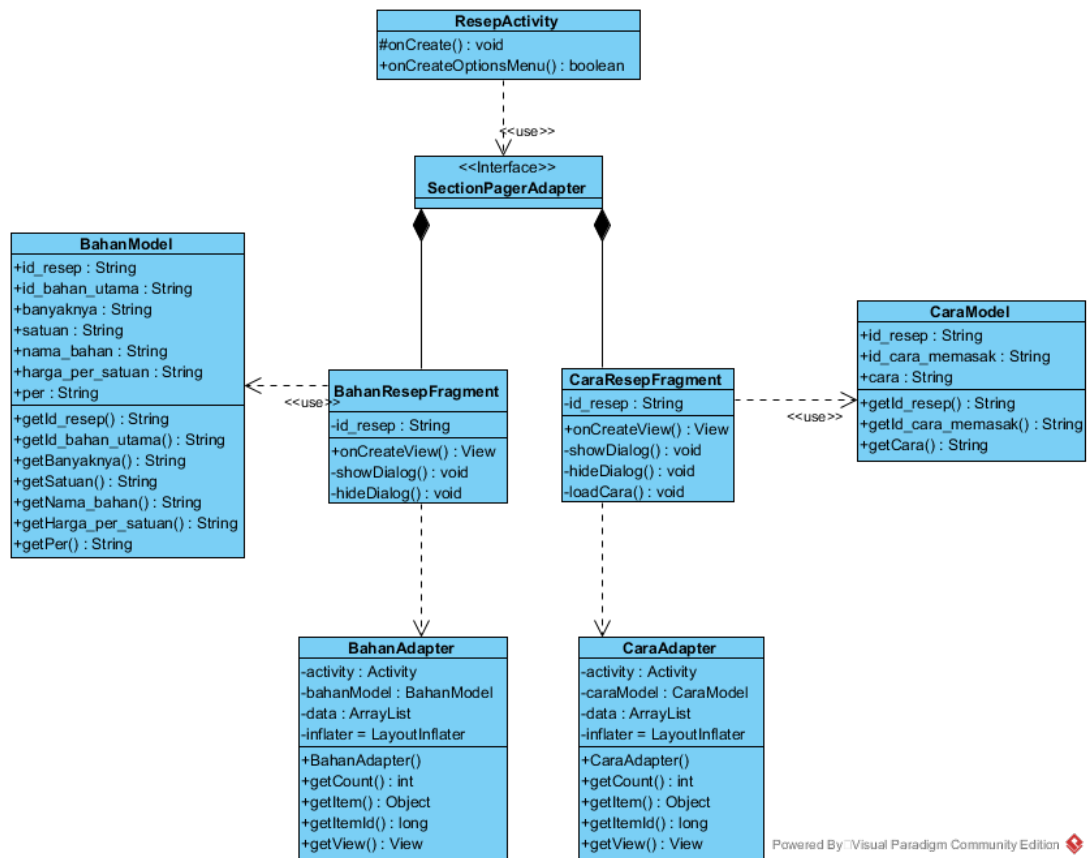
Pada pengembangan aplikasi resep masakan ini, penulis memecah *class diagram* menjadi tiga bagian, yaitu *class diagram* menu utama, *class diagram* detail resep, dan *class diagram wishlist*.

Terdapat dua model yang diakses pada *class diagram* Menu Utama. Resep Model digunakan untuk menyediakan data resep yang diolah pada *controller* Resep Fragment, Rekomendasi Fragment, dan Jenis Fragment. Bahan Model digunakan untuk menyediakan data resep berdasarkan bahan yang dicari oleh *user*. *Controller* selanjutnya mengalirkan data yang telah diolah kepada *view* yang terdapat pada masing-masing *controller*. Desain class diagram Menu Utama dapat dilihat pada Gambar 3.7.



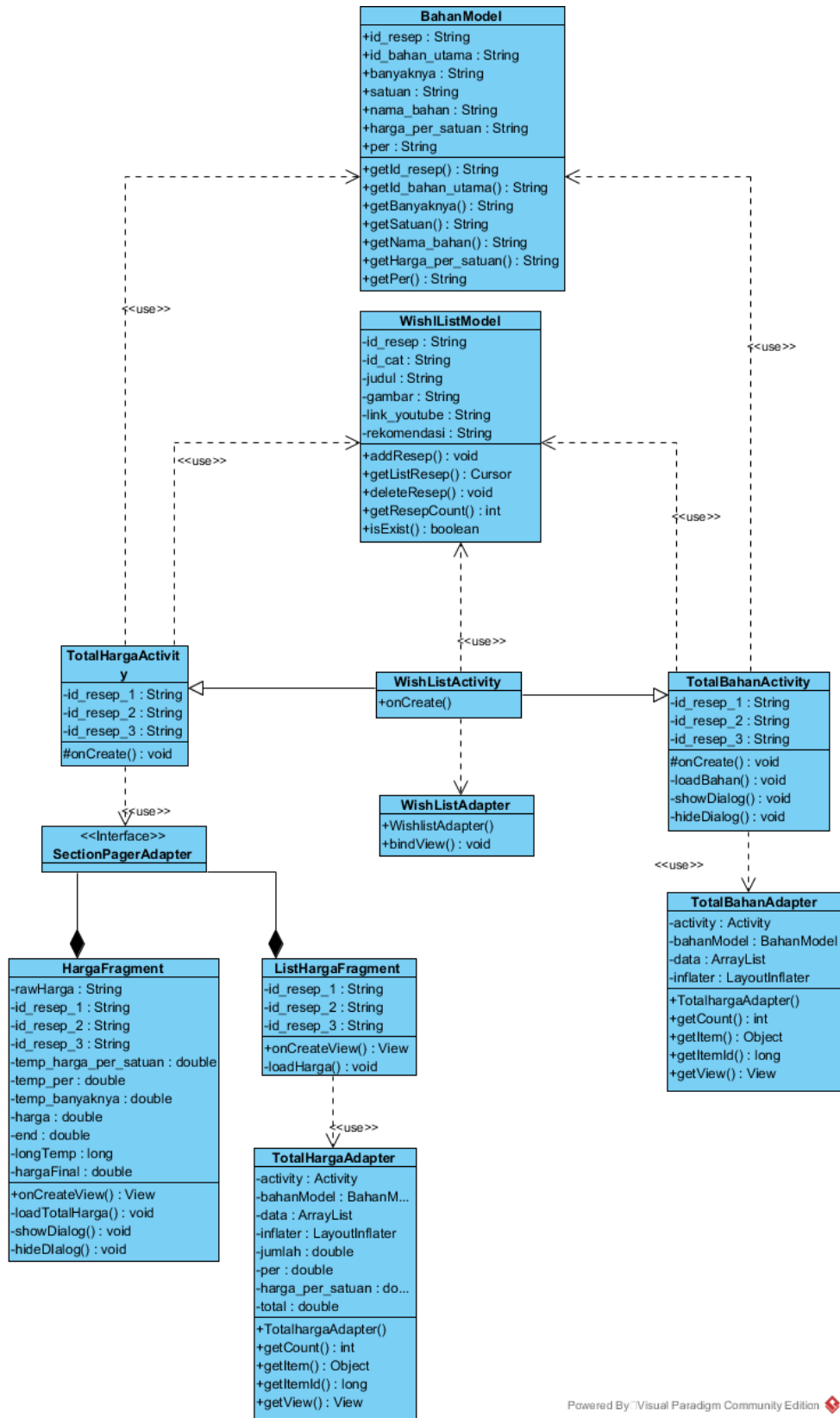
**Gambar 3.7:** Class Diagram Menu Utama

*Class diagram* Detail Resep, yang terdapat pada Gambar 3.8, memerlukan dua buah model yaitu Bahan Model dan Cara Model. Kedua model tersebut diperlukan untuk menyediakan data bahan yang dibutuhkan serta cara memasak dalam sebuah resep masakan. Data tersebut kemudian diolah dalam *controller* BahanResepFragment dan CaraResepFragment untuk kemudian dialirkan kepada masing-masing *view*



**Gambar 3.8:** *Class Diagram* Detail Resep

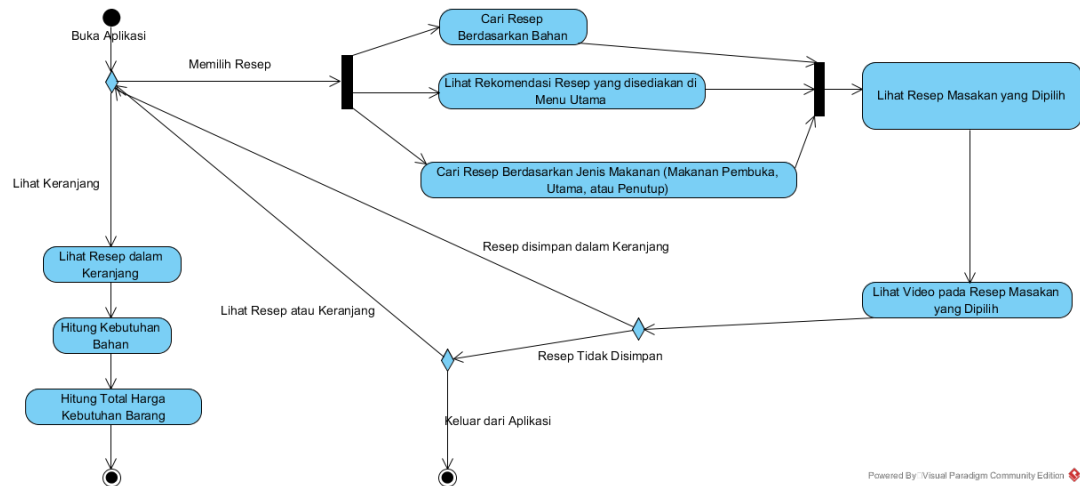
Wishlist merupakan *class diagram* terakhir yang dibuat oleh penulis. Terdapat dua model yang digunakan dalam *class diagram* ini yakni Bahan Model dan WishList Model. WishList Model digunakan untuk mengalirkan data resep yang telah disimpan oleh pengguna sebelumnya untuk kemudian dilakukan pencocokan data dengan basis data bahan (Bahan Model). Setelah dicocokkan, pengguna atau *user* dapat melakukan perhitungan total bahan yang dibutuhkan serta melakukan perhitungan total harga bahan yang dibutuhkan oleh pengguna beserta dengan daftar harga tiap bahan yang dibutuhkan. Gambar *class diagram* wishlist dapat dilihat pada Gambar 3.9.



Gambar 3.9: Class Diagram Wishlist

### 3.2.4 Activity Diagram

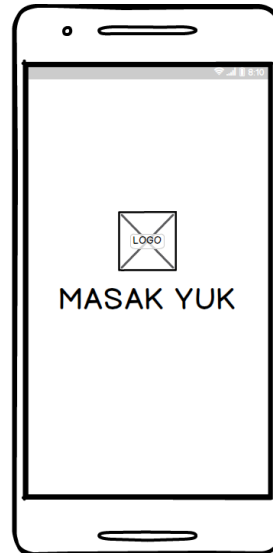
Alur kerja aplikasi yang dikembangkan oleh penulis atau peneliti dapat dijelaskan melalui *activity diagram* yang terdapat pada Gambar 3.10



**Gambar 3.10:** Activity Diagram dari Aplikasi yang Dibuat Penulis

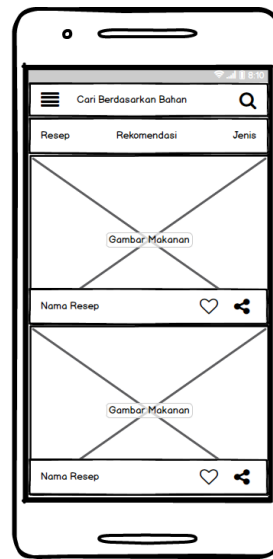
Pada awal membuka aplikasi, pengguna aplikasi dapat memilih resep atau ingin melihat keranjang atau *wishlist* apabila sudah pernah memasukkan resep ke dalam keranjang sebelumnya. Apabila pengguna ingin melihat resep, maka pengguna dapat memilih resep dengan cara melihat daftar resep, mencari resep berdasarkan bahan, melihat rekomendasi resep yang disediakan, serta mencari resep berdasarkan jenis makanan (makanan pembuka, makanan utama, dan makanan penutup). Setelah itu, pengguna melihat resep yang telah dipilih serta dapat melihat video pembuatannya juga. Apabila resep disimpan ke dalam keranjang, maka pengguna dapat kembali memilih resep atau melihat keranjang langsung untuk menghitung kebutuhan bahan serta total harga kebutuhan bahan. Sedangkan apabila resep masakan yang telah dilihat tidak dimasukkan ke dalam keranjang, maka pengguna dapat memilih resep masakan kembali, melihat keranjang yang sudah diisi sebelumnya, atau dapat keluar dari aplikasi.]

### 3.2.5 Desain *Wireframe* atau Kerangka Desain



**Gambar 3.11:** Halaman Awal

Desain halaman selamat datang ketika aplikasi pertama kali dimulai ditunjukkan pada Gambar 3.11. Terdapat logo serta gambar latar belakang yang dibuat oleh penulis. Setelah tiga detik, pengguna akan diarahkan kepada laman utama aplikasi yang terdapat pada Gambar 3.12



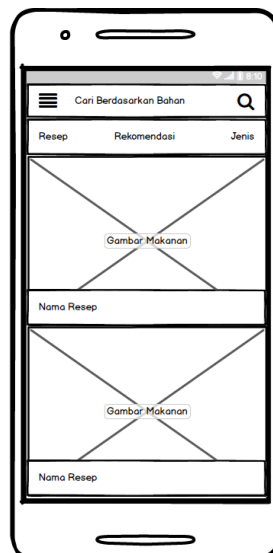
**Gambar 3.12:** Menu Utama

Terdapat banyak resep masakan yang dapat dipilih pada laman ini. Pada laman menu utama, apabila pengguna mengusap layar dari arah kiri ke kanan, maka akan muncul menu navigasi seperti pada Gambar 3.13. Pada menu navigasi, pengguna dapat memilih untuk melihat *wishlist*, memberi masukan untuk aplikasi, membagikan aplikasi ke media sosial, dan keluar dari aplikasi



**Gambar 3.13:** Navigasi

Kembali lagi pada menu utama, apabila pengguna mengusap layar ke kanan atau menekan tab rekomendasi, maka akan muncul resep masakan yang direkomendasikan oleh aplikasi.



**Gambar 3.14:** Menu Utama *Tab* Rekomendasi

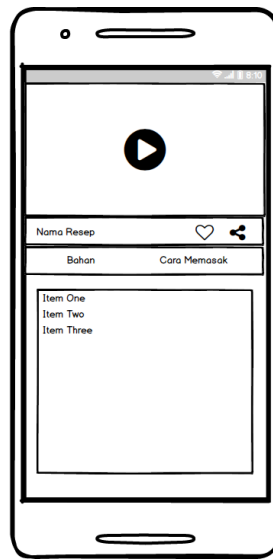


Pada tab rekomendasi, apabila pengguna mengusap layar ke arah kanan, maka pengguna akan menemukan pilihan penampilan resep berdasarkan jenis makanannya yaitu makanan pembuka, makanan utama, dan makanan penutup.



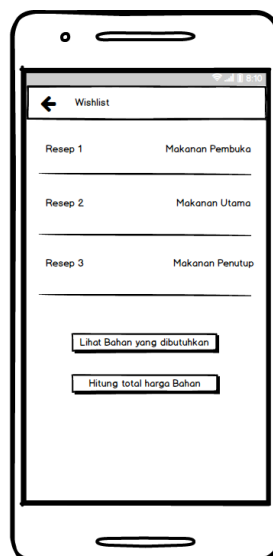
**Gambar 3.15:** Menu Utama *Tab* Jenis

Apabila pengguna memilih satu dari resep yang berada pada list yang disediakan dari menu-menu yang telah disebutkan sebelumnya, maka akan muncul laman detail resep yang terdiri dari video, bilah bahan resep serta bilah cara memasak dari resep tersebut



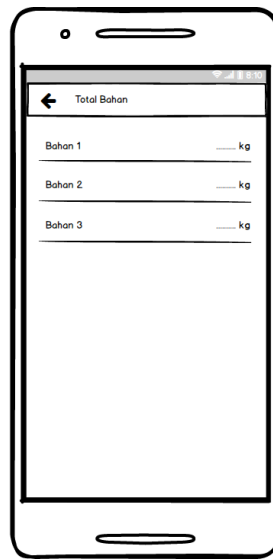
**Gambar 3.16:** Detail Resep

Pengguna juga dapat melihat *wishlist* dengan memanggil menu navigasi seperti pada Gambar 3.13 dan memilih menu *wishlist*. Laman wishlist akan muncul seperti pada Gambar 3.17



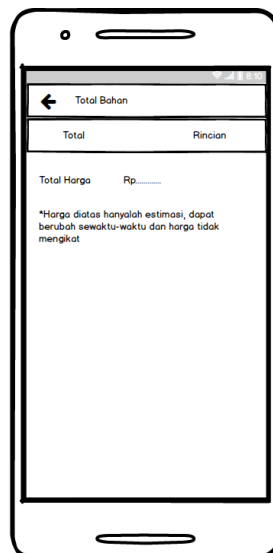
**Gambar 3.17:** Keranjang atau *Wishlist*

Pada laman *wishlist*, pengguna dapat menghitung total bahan yang dibutuhkan untuk memasaknya berdasarkan resep yang telah dipilih dan ada pada keranjang



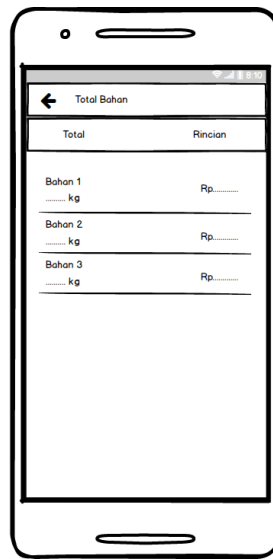
**Gambar 3.18:** Total Bahan

Selain itu, pada laman *wishlist*, pengguna juga dapat menghitung total harga dari bahan-bahan dari resep yang telah masuk dalam *wishlist*.



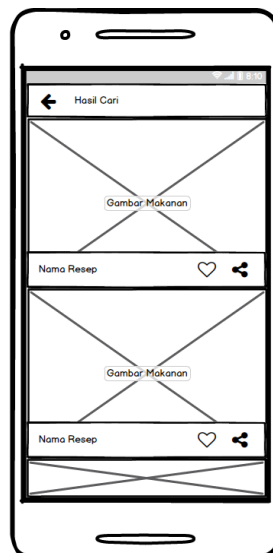
**Gambar 3.19:** Total Harga

Pengguna juga dapat melihat rincian harga dari tiap bahan yang dibutuhkan oleh resep-resep yang terdapat pada *wishlist*.



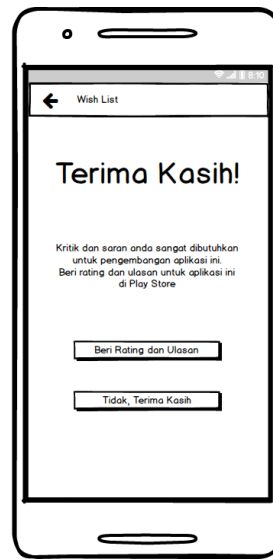
**Gambar 3.20:** Rincian Total Harga

Kembali ke menu utama, pengguna juga dapat melakukan pencarian resep berdasarkan bahan yang dimiliki oleh pengguna.



**Gambar 3.21:** Hasil Pencarian

Dan terakhir, desain *wireframe* dari laman beri masukan yang dapat diakses melalui menu navigasi.



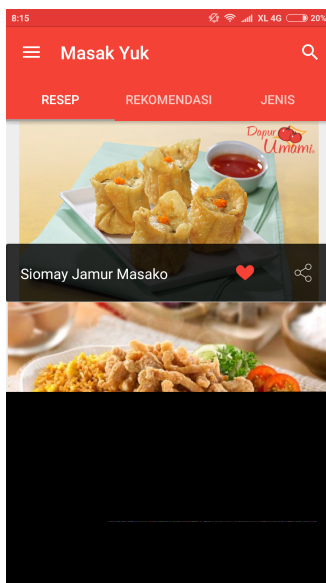
**Gambar 3.22:** Halaman Beri Masukan

### 3.2.6 Desain *Mock-Up* atau *User Interface*

Setelah membuat wireframe, penulis kemudian langsung mengimplementasikan *wireframe* tersebut ke dalam bentuk *mock-up*. Dalam pengembangannya, *mock-up* langsung dibuat dengan menggunakan xml sehingga mempercepat penulis dalam mengembangkan aplikasi resep masakan ini karena desain dari aplikasi tersebut sudah dikembangkan sekaligus dengan *mock-up*. Selanjutnya penulis hanya perlu mengimplementasikan sistem *Back-End* yang bekerja pada aplikasi resep masakan ini. Pengembangan *mock-up* dapat dilihat pada Gambar 3.23 sampai Gambar 3.35. Urutan pengembangan *mock-up* sesuai dengan urutan pengembangan *wireframe*.



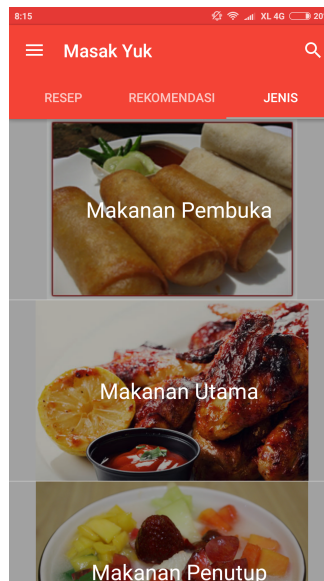
**Gambar 3.23:** Halaman Awal atau Halaman Selamat Datang



**Gambar 3.24:** Halaman Utama

**Gambar 3.25:** Menu Navigasi

**Gambar 3.26:** Halaman Utama *Tab* Rekomendasi



**Gambar 3.27:** Halaman Utama *Tab Jenis*

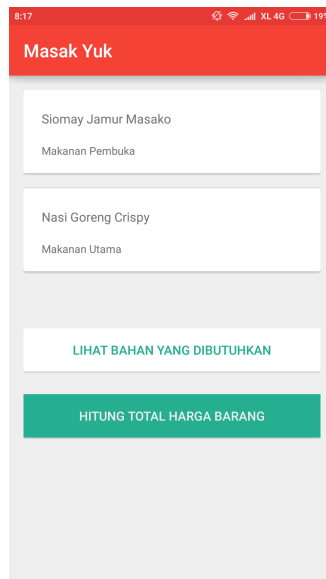


**Gambar 3.28:** Detail Bahan Resep

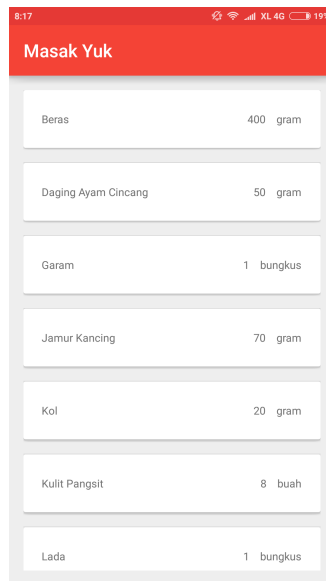




**Gambar 3.29:** Detail Cara Memasak



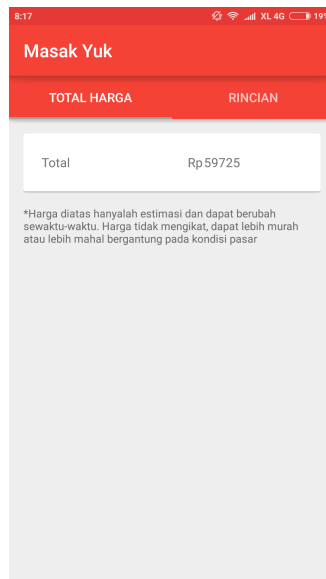
**Gambar 3.30:** *Wishlist*



The screenshot shows a mobile application interface with a red header titled "Masak Yuk". Below the header, there is a list of ingredients, each in a white box with a light gray border. The ingredients and their quantities are:

Ingredient	Quantity
Beras	400 gram
Daging Ayam Cincang	50 gram
Garam	1 bungkus
Jamur Kancing	70 gram
Kol	20 gram
Kulit Pangsit	8 buah
Lada	1 bungkus

**Gambar 3.31: Total Bahan**

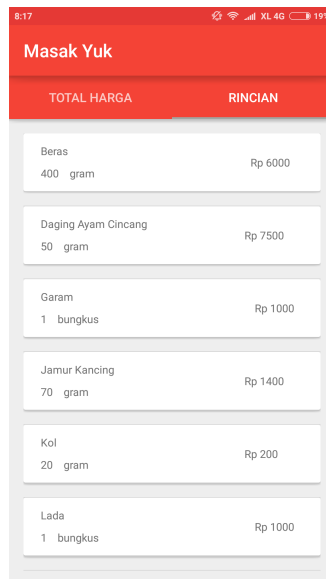


The screenshot shows the same mobile application interface. Below the ingredients list, there is a red header with two tabs: "TOTAL HARGA" (selected) and "RINCIAN". Below the tabs, there is a white box with a light gray border containing the total price:

Category	Value
Total	Rp59725

Below the price box, there is a disclaimer in small text: "\*Harga diatas hanyalah estimasi dan dapat berubah sewaktu-waktu. Harga tidak mengikat, dapat lebih murah atau lebih mahal bergantung pada kondisi pasar"

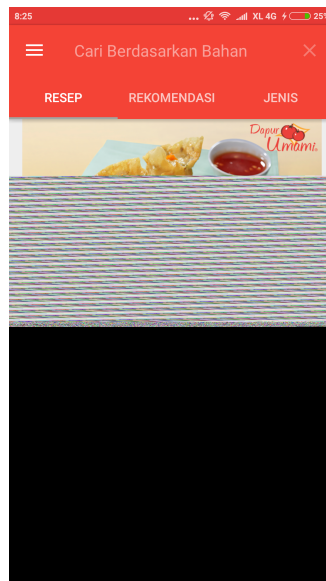
**Gambar 3.32: Total Harga**



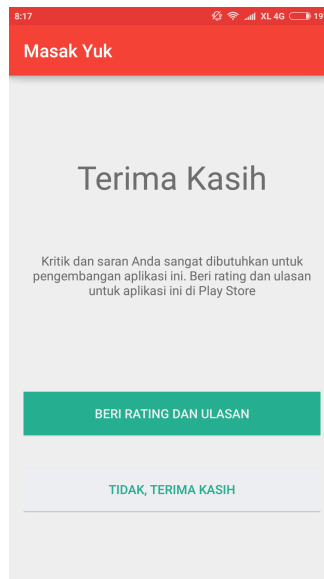
The screenshot shows the 'Masak Yuk' app interface. At the top, there is a red header with the text 'Masak Yuk'. Below the header, there are two tabs: 'TOTAL HARGA' and 'RINCIAN'. The 'RINCIAN' tab is selected, displaying a list of ingredients and their prices. The ingredients are listed in a table format with their names, quantities, and prices.

Item	Price
Beras 400 gram	Rp 6000
Daging Ayam Cincang 50 gram	Rp 7500
Garam 1 bungkus	Rp 1000
Jamur Kancing 70 gram	Rp 1400
Kol 20 gram	Rp 200
Lada 1 bungkus	Rp 1000

**Gambar 3.33:** Rincian Total Harga



**Gambar 3.34:** Tab Pencarian



**Gambar 3.35:** Laman Beri Masukan atau *Feedback*

### 3.3 Konstruksi dan Pembangunan

Tahapan-tahapan dalam proses konstruksi dan pembangunan aplikasi berbasis resep masakan ini adalah sebagai berikut:

1. Membuat Basis Data
2. Membuat API untuk Mengakses Basis Data
3. Melakukan Input Data ke Basis Data
4. Implementasi Desain Aplikasi
5. Implementasi Metode Pertukaran Data antara Aplikasi dengan Basis Data
6. Mengintegrasikan YouTube dengan Aplikasi Android
7. Mengimplementasi Fitur Lainnya

### 3.3.1 Membuat Basis Data

Basis data dibuat dengan menggunakan Basis Data MySQL atau MariaDB. Untuk mempercepat proses pengerjaan, penulis menggunakan aplikasi phpMyAdmin dalam membuat basis data secara keseluruhan, termasuk tabel dan entitas-entitas serta relasi yang terdapat di dalamnya. Basis data dibuat sesuai dengan ERD yang telah dibuat pada tahap desain.

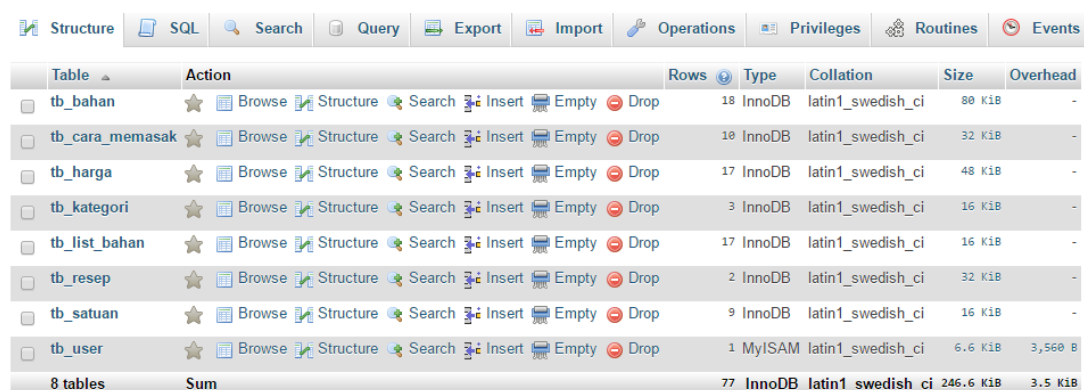
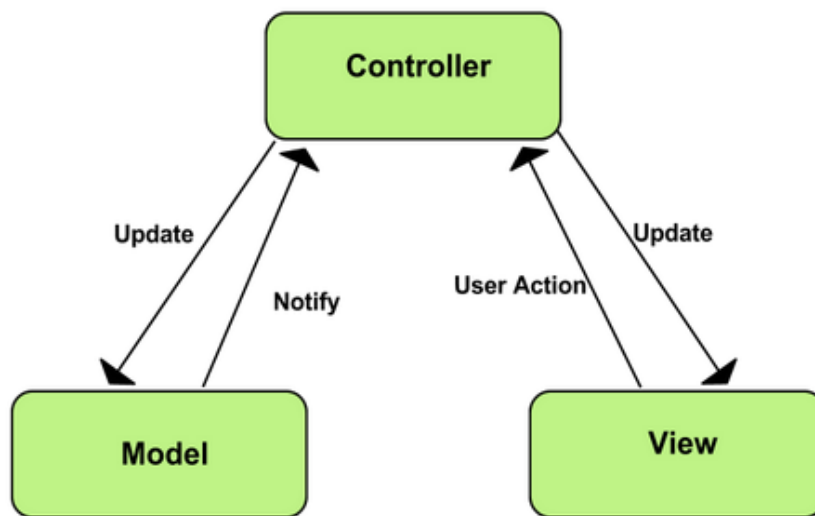


Table	Action	Rows	Type	Collation	Size	Overhead
tb_bahan	Browse Structure Search Insert Empty Drop	18	InnoDB	latin1_swedish_ci	80 K1B	-
tb_cara_memasak	Browse Structure Search Insert Empty Drop	10	InnoDB	latin1_swedish_ci	32 K1B	-
tb_harga	Browse Structure Search Insert Empty Drop	17	InnoDB	latin1_swedish_ci	48 K1B	-
tb_kategori	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 K1B	-
tb_list_bahan	Browse Structure Search Insert Empty Drop	17	InnoDB	latin1_swedish_ci	16 K1B	-
tb_resep	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	32 K1B	-
tb_satuan	Browse Structure Search Insert Empty Drop	9	InnoDB	latin1_swedish_ci	16 K1B	-
tb_user	Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	6.6 K1B	3,560 B
8 tables	Sum	77	InnoDB	latin1_swedish_ci	246.6 K1B	3.5 K1B

Gambar 3.36: Basis Data yang Dibuat dengan phpMyAdmin

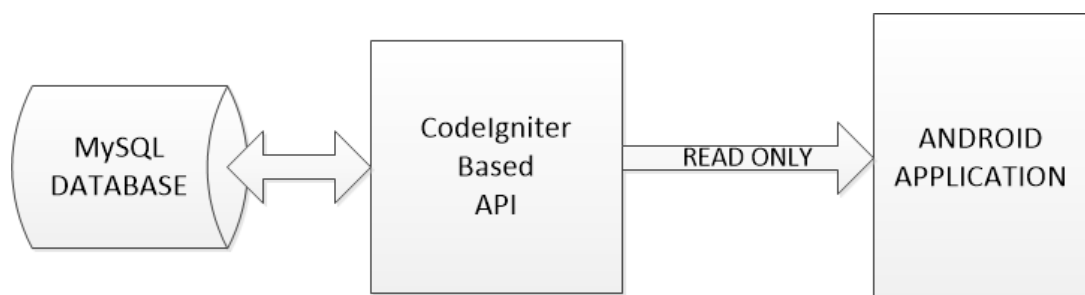
### 3.3.2 Membuat API untuk Mengakses Basis Data

Untuk dapat mengakses data yang terdapat pada basis data, dimana data tersebut tersimpan dalam server dengan jenis basis data MySQL atau MariaDB, maka penulis perlu membuat kumpulan API. Pengguna membuat API dengan menggunakan *php framework* yaitu CodeIgniter. CodeIgniter mengimplementasi arsitektur Model-View-Controller dimana semua perintah menuju database secara langsung disimpan dalam Model dan mengolah data yang didapatkan oleh Model dengan menggunakan Controller. Data yang sudah diolah kemudian ditampilkan melalui View.



**Gambar 3.37:** Bagan Alur Kerja MVC

Namun untuk membuat API, penulis tidak menggunakan View dari CodeIgniter tersebut. Penulis hanya perlu menyimpan *query* basis data pada model dan API tersimpan dalam bagian Controller. Aplikasi Android nantinya akan mengakses Controller tersebut dan dijadikan sebagai API untuk mendapatkan akses kepada basis data. API yang dibuat untuk Aplikasi Android hanya memiliki metode yang bersifat *read only*. Data akan diterima oleh aplikasi dalam bentuk JSON (JavaScript Object Notation).



**Gambar 3.38:** Bagan Alur Kerja CodeIgniter-Based API

Sampel kode dari CodeIgniter-Based API yang telah dibuat oleh Penulis dapat dilihat pada Gambar 3.39 atau pada Lampiran C dan D.

```

Welcome.php
16  function get_resep(){
17      $query = $this->Masakyuk_model->get_resep();
18      $response = array();
19      $cek = $query;
20      if($cek > 0){
21          $response["resep"] =array();
22          foreach ($query as $row) {
23              $data=array();
24              $data["id_resep"]  = $row["id_resep"];
25              $data["id_cat"] = $row["id_cat"];
26              $data["judul"] = $row["judul"];
27              $data["gambar"] = $row["gambar"];
28              $data["link_youtube"] = $row["link_youtube"];
29              $data["rekomendasi"] = $row["rekomendasi"];
30              array_push($response["resep"], $data);
31          }
32          $response["success"] = 1;
33          $response["message"] = "Semua data resep";
34          echo json_encode($response);
35      } else {
36          $response["success"] = 0;
37          $response["message"] = "Data resep tidak ditemukan";
38          echo json_encode($response);
39      }
40  }
41

```

**Gambar 3.39:** Sampel Kode CodeIgniter API-Based yang Dibuat Penulis

### 3.3.3 Melakukan Input Data ke Basis Data

Input data ke dalam *database* dapat dilakukan dengan dua cara, yaitu memasukkan data melalui phpMyAdmin atau memasukkan data melalui aplikasi web khusus untuk memasukkan data ke dalam basis data.

Server: 127.0.0.1 » Database: db\_masakyuk » Table: tb\_resep

Browse Structure SQL Search Insert Export Import Privileges Operation

Column	Type	Function	Null	Value
id_resep	int(11)			
id_cat	int(11)			
judul	text			
gambar	varchar(50)			
link_youtube	varchar(20)			
rekomendasi	tinyint(1)			

Go

**Gambar 3.40:** Input Data dengan phpMyAdmin

Penulis mengembangkan aplikasi berbasis web dengan menggunakan *framework* CodeIgniter yang khusus untuk memasukkan data ke dalam basis data. Adanya aplikasi berbasis web tersebut memudahkan penulis dalam memasukkan data-data ke dalam basis data, mulai dari data resep, bahan, hingga data harga bahan.

Masak Yuk Menu - Administrator

Daftar Resep Tambah

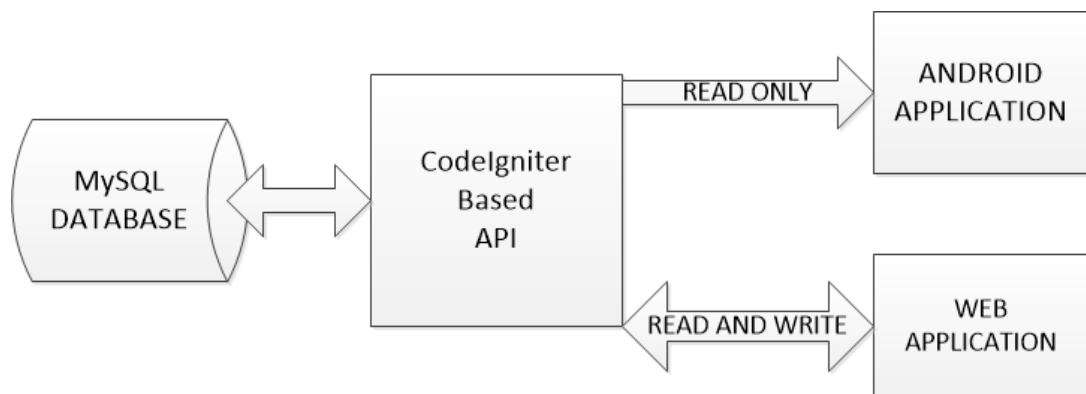
Judul	Kategori	Link YouTube	Rekomendasi	Aksi
Siomay Jamur Masako	Makanan Pembuka	CXBMJMibSWM	Ya	<a href="#">Edit</a> <a href="#">Lihat Bahan</a> <a href="#">Lihat Cara Memasak</a> <a href="#">Hapus</a>
Nasi Goreng Crispy	Makanan Utama	QdhXqHTOPmA	Belum	<a href="#">Edit</a> <a href="#">Lihat Bahan</a> <a href="#">Lihat Cara Memasak</a> <a href="#">Hapus</a>

**Gambar 3.41:** Tampilan List Resep pada Aplikasi Berbasis Web



**Gambar 3.42:** Tampilan Input Resep pada Aplikasi Berbasis Web

Penulis melakukan sedikit modifikasi pada API yang telah dibuat dengan mengubah metode *read only* menjadi *read and write* dengan catatan yaitu write hanya diperbolehkan pada aplikasi berbasis web yang digunakan untuk memasukkan data-data yang dibutuhkan.



**Gambar 3.43:** Bagan Alur Kerja CodeIgniter-Based API Hasil Modifikasi

### 3.3.4 Implementasi Desain Aplikasi

Desain Aplikasi sudah diimplementasikan kedalam bentuk *layout* Android Studio yaitu dengan format xml sehingga penulis dapat melewati tahap implementasi desain ini. Implementasi dapat dilihat pada Bagian 3.2.6.

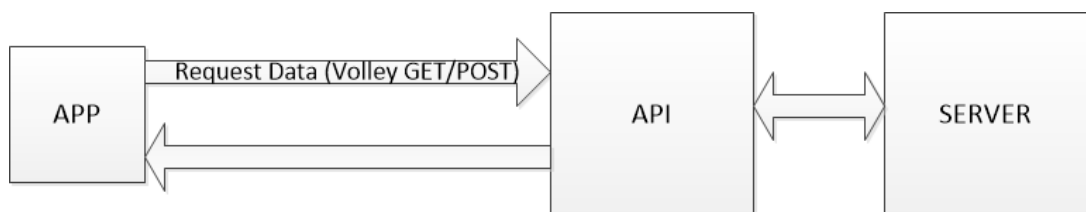
### 3.3.5 Implementasi Metode Pertukaran Data antara Aplikasi dengan Basis Data

Pada tahap ini, penulis mencoba mengimplementasikan API yang telah dibuat ke dalam aplikasi Android. Hal itu dilakukan agar aplikasi dapat melakukan pertukaran data dengan basis data dengan dijumpai oleh API yang telah dibuat sebelumnya. Pada aplikasi berbasis web, pertukaran data dilakukan dengan memanfaatkan Form GET/POST seperti pada Gambar 3.44.



**Gambar 3.44:** Bagan Alur Kerja Form GET/POST

Dalam pengembangan aplikasi Android, salah satu cara untuk dapat memanggil CodeIgniter-Based API yang telah dibuat adalah dengan menggunakan Volley. Android Volley Library adalah sebuah *library* yang disediakan oleh Google untuk dapat memfasilitasi akses *Web-Based* API pada aplikasi Android sebagai media pertukaran data antara aplikasi dengan basis data. Cara Kerja Volley secara umum menyerupai Form GET/POST pada aplikasi berbasis web.



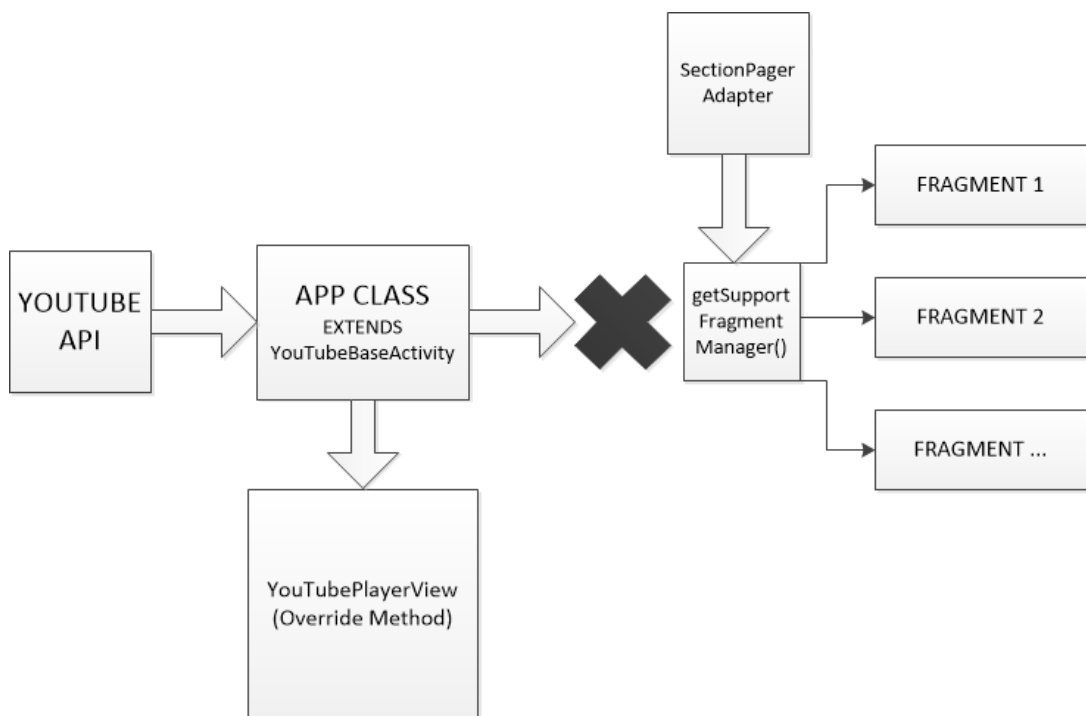
**Gambar 3.45:** Bagan Alur Kerja Volley

Untuk dapat menggunakan Volley, penulis harus terlebih dahulu menambahkan compile 'com.android.volley:volley:1.0.0' pada berkas build.gradle yang terdapat

di dalam direktori app.

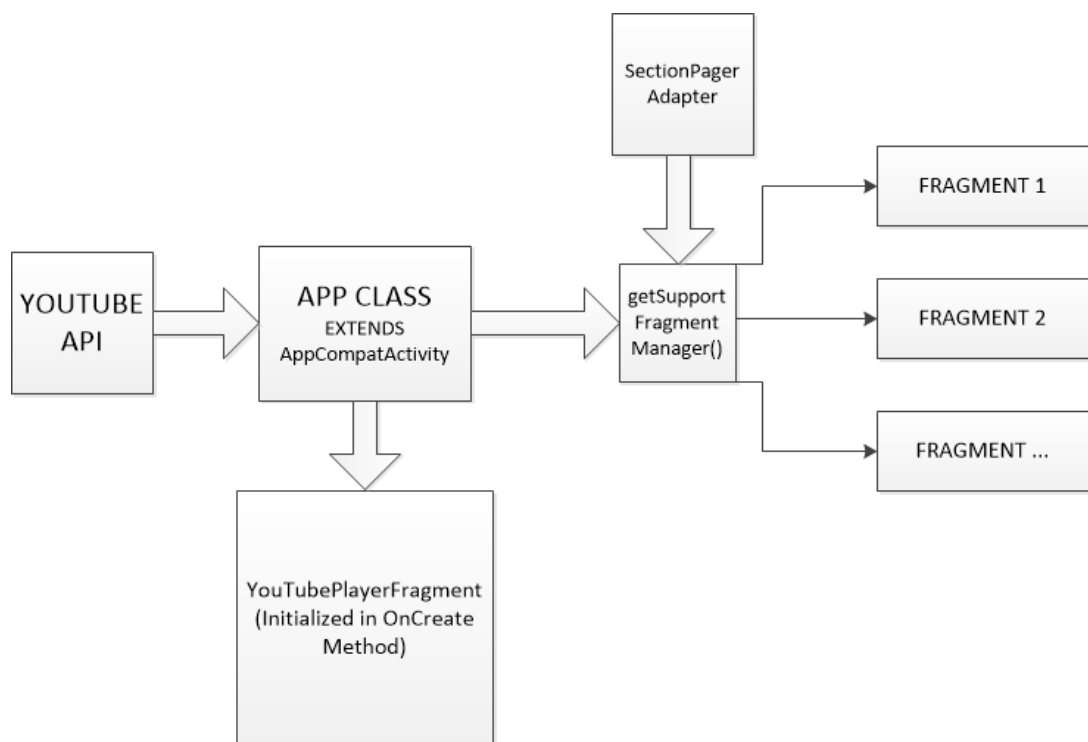
### 3.3.6 Mengintegrasikan YouTube dengan Aplikasi Android

Untuk dapat memainkan video YouTube pada aplikasi Android, maka dibutuhkan YouTube Android Player API sebagai perantara aplikasi dengan basis data YouTube. Penulis melakukan sedikit modifikasi pada implementasi YouTube Android Player API karena penggunaan Fragment pada Resep Activity. Ketika suatu *class* pada Java Android Studio menggunakan Fragment, maka diperlukan *class* yang melakukan ekstensi (*extends*) dengan AppCompatActivity untuk mendukung pengelolaan beberapa Fragment yang akan ditampilkan pada sebuah Activity. Sedangkan YouTube Android Player API mewajibkan penggunaanya untuk melakukan ekstensi terhadap *class* YouTubeBaseActivity



**Gambar 3.46:** Bagan Alur Kerja YouTubePlayerView pada sebuah Activity

Apabila Activity tetap mengekstensi YouTubeBaseActivity, maka Activity tersebut tidak dapat mengelola Fragment yang terdapat didalamnya karena kehilangan metode `getSupportFragmentManager()` yang berasal dari ekstensi *class* terhadap `AppCompatActivity`. Maka dari itu, penulis melakukan modifikasi dengan tidak menggunakan `YouTubePlayerView`, melainkan menggunakan `YouTubePlayerFragment`. Keputusan tersebut diambil karena `YouTubePlayerFragment` "lebih ramah" terhadap Activity yang memiliki Fragment dengan tidak perlu mengekstensi *class* `YouTubeBaseActivity` sehingga seluruh Fragment yang ada pada sebuah Activity dapat dikelola dengan baik. Perbedaannya adalah seluruh metode yang diberikan oleh YouTube Android Player API tidak dilakukan *override*, melainkan langsung diinisialisasikan pada metode `onCreate` sebuah Activity.



**Gambar 3.47:** Bagan Alur Kerja YouTubePlayerFragment pada sebuah Activity

Sampel kode implementasi YouTubePlayerFragment terdapat pada Gambar 3.48.

```

YouTubePlayerFragment youtubeFragment = (YouTubePlayerFragment)
    getFragmentManager().findFragmentById(R.id.youtube_view);
youtubeFragment.initialize(YOUTUBE_API_KEY,
    new YouTubePlayer.OnInitializedListener() {
        @Override
        public void onInitializationSuccess(YouTubePlayer.Provider provider,
            YouTubePlayer youtubePlayer, boolean b) {
            if (!b) {
                youtubePlayer.cueVideo(link_youtube);
            }
        }
        @Override
        public void onInitializationFailure(YouTubePlayer.Provider provider,
            YouTubeInitializationResult errorReason) {
            if (errorReason.isUserRecoverableError()) {
                errorReason.getErrorDialog(getParent(), RECOVERY_REQUEST).show();
            } else {
                String error = String.format("Error initializing YouTube player: %1$s", errorReason.toString());
                Toast.makeText(getApplicationContext(), error, Toast.LENGTH_LONG).show();
            }
        }
    });

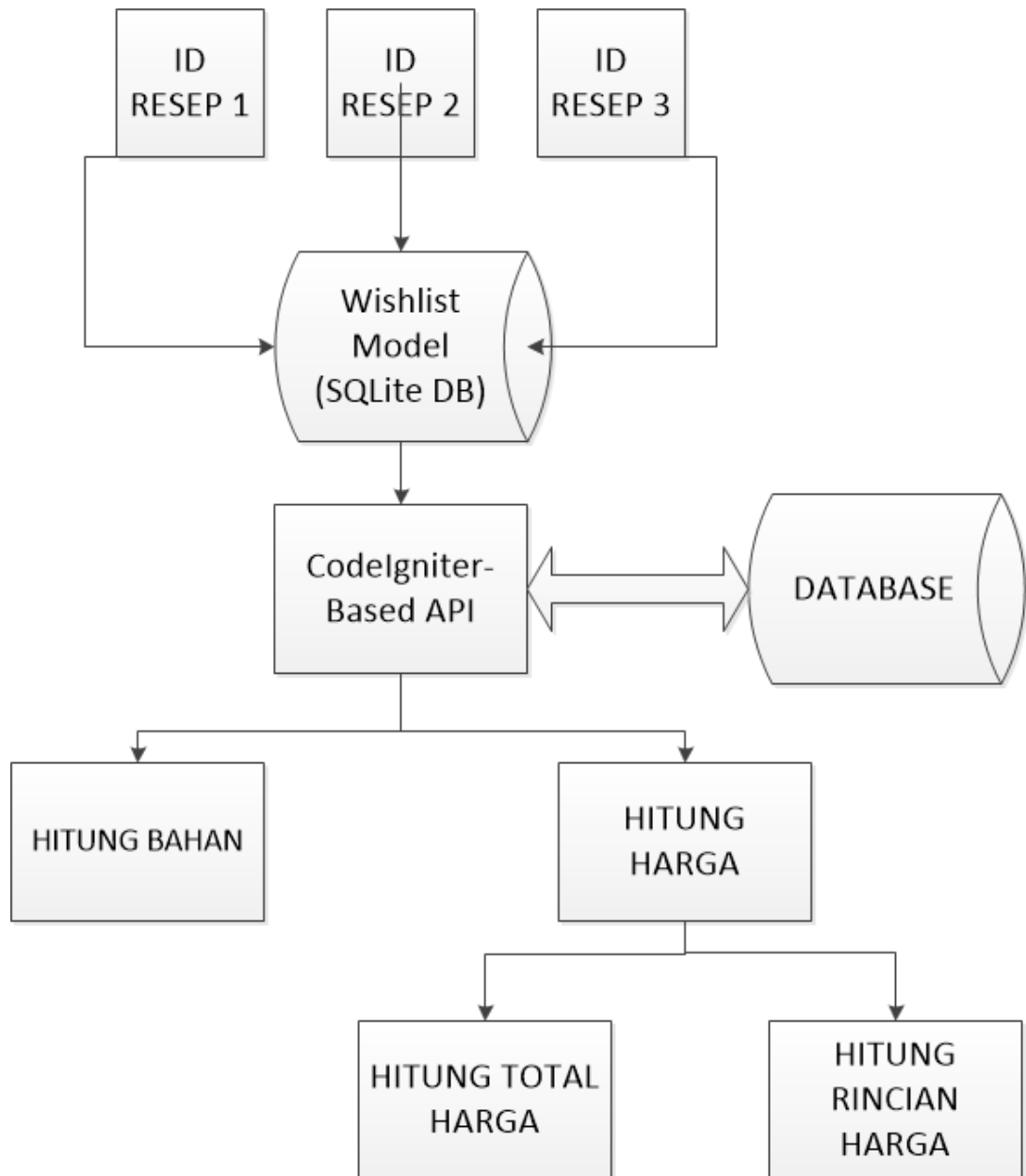
```

**Gambar 3.48:** Sampel Kode Implementasi YouTubePlayerFragment pada Metode onCreate sebuah Activity

### 3.3.7 Mengimplementasi Fitur Lainnya

Fitur lain yang diimplementasikan oleh penulis adalah fitur penyimpanan *wishlist*, menghitung total bahan pada resep yang terdapat dalam *wishlist*, menghitung total harga bahan yang terdapat pada *wishlist* dan menampilkan rincian harga per bahan. *Wishlist* sendiri dibuat dengan memanfaatkan basis data SQLite dengan menggunakan bahasa query yang sama yakni SQL. Basis data SQLite bersifat lokal dan terdapat didalam aplikasi itu sendiri. Jadi, apabila pengguna aplikasi menghapus data aplikasi pada ponselnya, maka data *wishlist* pun juga akan hilang.

Wishlist mampu menampung maksimal tiga resep masakan. Untuk dapat menghitung total bahan dan total harga bahan yang terdapat pada resep-resep tersebut, maka data yang terdapat di dalam SQLite dikirimkan dan diolah oleh API yang telah dibuat untuk kemudian disalurkan kembali hasil perhitungannya pada aplikasi. Alur dari proses penyimpanan pada *wishlist* hingga perhitungan total bahan dan harga bahan dijelaskan pada Gambar 3.49



**Gambar 3.49:** Alur Penyimpanan pada SQLite dan Pengolahan Total Bahan dan Harga Bahan