

BAB III

ANALISIS DAN PERANCANGAN SISTEM

Dalam pengembangan aplikasi Pemetaan Pedagang Kaki Lima di Surabaya penulis menerapkan konsep pengembangan *Software Development Life Cycle* (SDLC) secara *agile*. Metode yang penulis gunakan adalah Agile Model Driven Development (AMDD). Adapun langkah-langkah yang penulis tempuh untuk membuat aplikasi tersebut adalah sebagai berikut:

3.1 Analisis Sistem

Pada bagian pendahuluan telah dijelaskan mengenai latar belakang pengembangan aplikasi ini, yaitu belum adanya pemetaan terhadap PKL tersentra dan terbina, yang bertujuan untuk memberikan informasi tempat yang memungkinkan untuk dibangun tempat PKL kepada pimpinan yang dilakukan secara manual. Sedangkan jumlah PKL tersentra dan terbina yang ada di Surabaya sudah begitu banyak. Data yang ada di Dinas Koperasi hanya berbentuk teks sehingga sulit untuk melihat lokasi-lokasi dimana saja PKL Tersentra berada. Dan jika data tersebut hanya berbentuk teks maka sulit untuk mengatur letak jika ingin membangun sentra PKL yang baru. Serta dinas koperasi juga belum memberikan informasi jenis makanan yang di jual di PKL tersentra kepada masyarakat umum, yang seharusnya jika di lakukan akan memberikan dampak positif bagi penjualan di PKL tersentra.

Dinas Koperasi Pemkot Surabaya membutuhkan sebuah aplikasi pemetaan PKL. Pemetaan yang dimaksudkan sesuai dengan sistem informasi geografis, karena dengan menggunakan sistem informasi geografis dapat memberikan solusi

pemetaan yang dibutuhkan oleh dinas koperasi yaitu pemetaan dengan menggunakan peta secara geografis. Dengan menggunakan media peta digital sehingga lokasi PKL dapat terlihat secara visual sesuai dengan data text maupun data lokasi lapangan. Dengan adanya aplikasi ini dapat memberikan informasi untuk analisa lokasi PKL untuk pengembangan lokasi. Aplikasi ini juga memiliki fitur pencarian makanan minum yang tersedia di PKL tersentra, fitur ini diperuntukan untuk masyarakat umum. Aplikasi ini menggunakan *layer Latitude* pada *google map* sebagai *layer* dasar pada peta. Aplikasi ini berbasis web karena kemudahannya yang bisa diakses dimanapun dan kapanpun selama masih terhubung ke *internet*.

3.2 Perancangan Sistem

Setelah mendapat gambaran umum sistem maka langkah selanjutnya dapat dilakukan perancangan. Penulis menggunakan pengembangan Agile Model Driven Development (AMDD), sehingga perancangan dan pemodelan sistem dilakukan secara bertahap tidak dikerjakan keseluruhan diawal. Kemudian dilanjutkan dengan penulisan kode menggunakan TDD. Kedua hal tersebut (pemodelan dan *coding*) dilakukan secara terus menerus hingga aplikasi selesai dibuat. Tentunya iterasi yang dilakukan sesuai dengan *requirements* awal yang digambarkan pada saat *envisioning*. Langkah-langkah dalam pengembangan menggunakan AMDD adalah sebagai berikut:

3.2.1 Envisioning

Pada tahap ini diidentifikasi gambaran umum dari sistem. Langkah-langkah pada tahap envisioning adalah sebagai berikut :

A. Usage Model

Tahap ini menggambarkan bagaimana pengguna berinteraksi dengan sistem. Penulis menggunakan *user stories* dan *use case* untuk menggambarkan interaksi tersebut.

a. User Stories

Pada aplikasi yang dibuat 4 aktor utama yang berinteraksi dengan sistem adalah administrator, pimpinan, petugas PKL, dan masyarakat. *User stories* yang berhubungan dengan aktor ditunjukkan pada tabel 3.1

Tabel 3.1 *User Stories Administrator*

Id	Story
A01	Halaman admin hanya bisa di akses oleh admin.
A02	Sebagai admin, saya harus login terlebih dahulu sebagai administrator.
A03	Sebagai admin, saya bisa memasukan, mengedit, dan menghapus data PKL.
A04	Sebagai admin, saya bisa memasukan, mengedit, dan menghapus data pedagang.
A05	Sebagai admin, saya bisa memasukan, mengedit, dan menghapus data menu makanan.
A06	Sebagai admin, saya bisa memasukan, mengedit, dan menghapus data menu minuman.
A07	Sebagai admin, saya bisa memasukan, mengedit, dan menghapus user.
A08	Sebagai admin, saya bisa mengakses peta PKL.
A09	Sebagai admin, saya bisa melakukan <i>view radius</i> .
A10	Sebagai admin, saya bisa melihat PKL berdasarkan omset nya.

Objek-objek yang dapat diidentifikasi dari *user stories* administrator adalah administrator, PKL, pedagang, menu makanan, menu minuman, user, dan omset.

User stories yang berhubungan dengan pimpinan ditunjukkan pada tabel 3.2

Tabel 3.2 *User Stories Pimpinan*

Id	Story
P01	Sebagai pimpinan, saya harus login sebagai pimpinan terlebih dahulu.
P02	Sebagai pimpinan, saya dapat melakukan <i>view radius</i> dan menyimpan lokasi rencana PKL tersentra yang baru.
P03	Sebagai pimpinan, saya dapat melihat grafik omset dari PKL.

Objek-objek yang dapat diidentifikasi dari *user stories* pimpinan adalah pimpinan.

User stories yang berhubungan dengan petugas PKL ditunjukkan pada tabel 3.3.

Tabel 3.3 *User Stories* Petugas PKL

Id	Story
PT01	Sebagai petugas PKL, saya harus login sebagai petugas PKL terlebih dahulu
PT02	Sebagai petugas PKL, saya bisa menambah, mengedit, dan menghapus data omset.

Objek yang dapat diidentifikasi dari *user stories* petugas PKL adalah petugas PKL dan omset.

User stories yang berhubungan dengan petugas PKL ditunjukkan pada tabel 3.4.

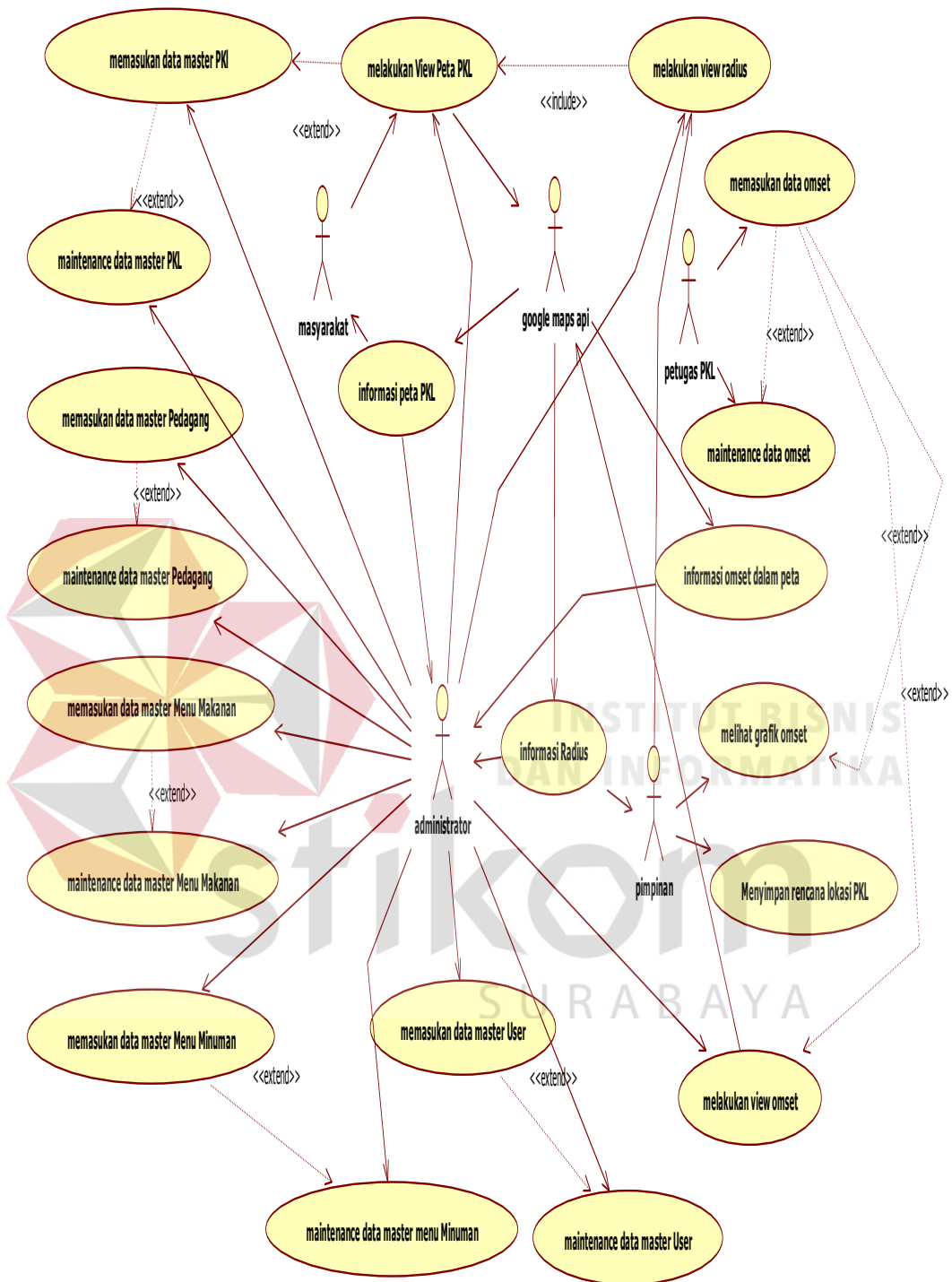
Tabel 3.4 *User Stories* Masyarakat

Id	Story
M01	Sebagai masyarakat, saya bisa melihat peta PKL yang ada di Surabaya.
M02	Sebagai masyarakat, saya bisa mengetahui lokasi PKL dengan mencari makanan atau minuman yang dijual.

Objek-objek yang dapat diidentifikasi dari *user stories* masyarakat adalah masyarakat.

b. Use Case Aplikasi Pemetaan Pedagang Kaki Lima

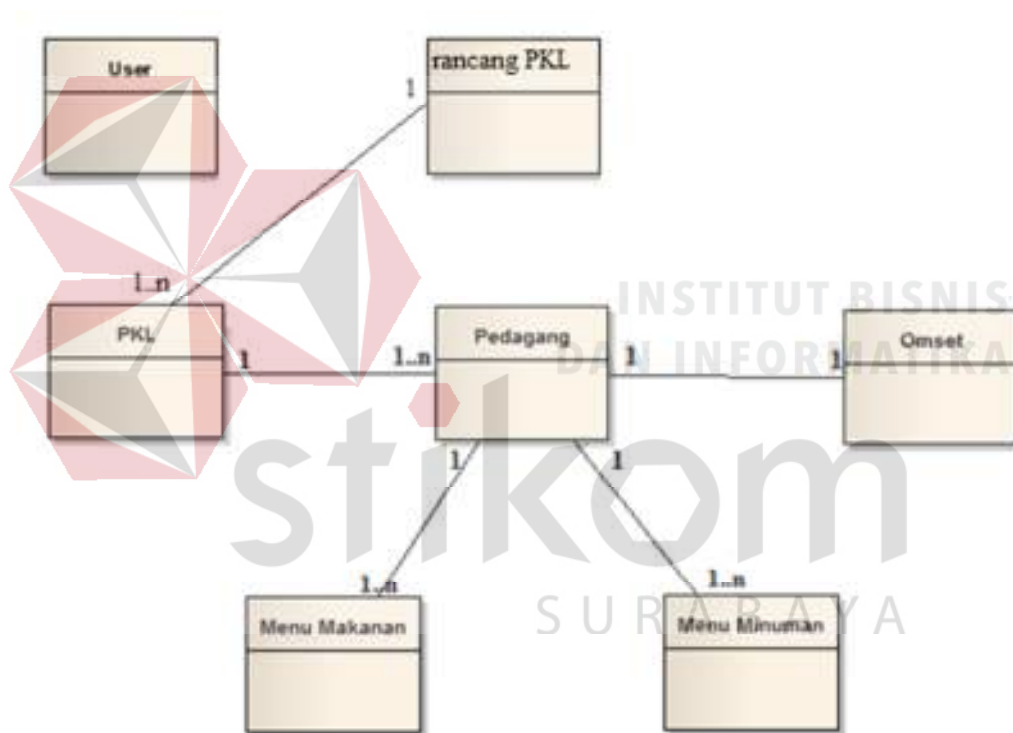
Use case Aplikasi Pemetaan Pedagang Kaki Lima di Surabaya di tunjukkan pada gambar 3.1. 4 aktor yang ada pada *use case* merupakan representasi dari apa yang ada pada *user stories*.



Gambar 3.1 Use Case Diagram Aplikasi Pemetaan Pedagang Kaki Lima di Surabaya

B. Domain Model

Objek-objek yang telah teridentifikasi pada *user stories* administrator, pimpinan, petugas PKL, dan masyarakat jika di gabung akan terlihat pada gambar 3.2. dimana penulis menyatukan objek administrator, pimpinan, petugas PKL, dan masyarakat menjadi satu entitas yaitu *user*. sehingga entitas yang muncul adalah sebagai berikut: *user*, rancang PKL, PKL, pedagang, menu makanan, menu minuman, dan omset.



Gambar 3.2 Domain Model

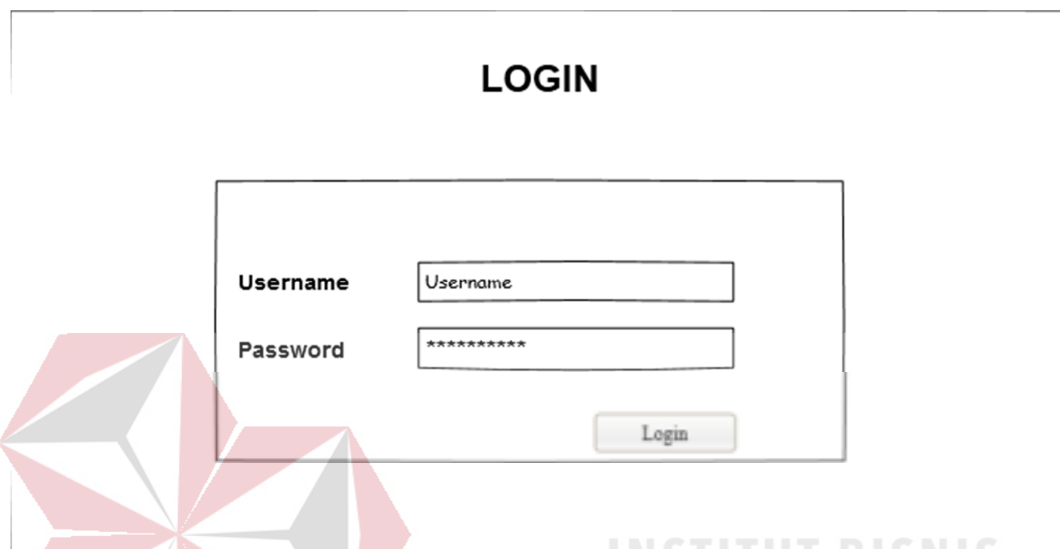
C. User Interface Model (UI)

Pada tahap ini penulis membuat sketsa antar muka dari aplikasi. Sketsa yang dibuat didasarkan pada *user stories* dan *use case* yang telah dibuat. Sketsa yang dibuat diperuntukkan kepada administrator, pimpinan, petugas PKL, dan masyarakat.

a. Sketsa Halaman Login

Halaman login menampilkan sebuah tombol yang bertuliskan “Login”.

Sketsa halaman login ditunjukkan pada gambar 3.3.



The image shows a login form sketch. At the top center, the word "LOGIN" is written in bold. Below it is a rectangular box containing the form fields. On the left side of this box, the labels "Username" and "Password" are listed. To the right of "Username" is a text input field with the placeholder text "Username". To the right of "Password" is a text input field with placeholder text "*****". Below these two input fields is a button labeled "Login".

Gambar 3.3 Sketsa Login

b. Sketsa Halaman Admin

Pada header halaman admin terdapat menu Form Master, Peta PKL, dan Tabel Daftar User. Sketsa halaman admin di tunjukan pada gambar 3.4.



The image shows a sketch of an admin page header. It features a horizontal navigation bar with three menu items: "Form Master", "Peta PKL", and "Tabel Daftar User". On the far right of this bar is a "Log Out" button. Below the navigation bar is a large, empty rectangular area representing the main content of the page.

Gambar 3.4 Sketsa Halaman admin

c. Sketsa Halaman Tabel PKL

Halaman tabel PKL ini terdiri dari 2 bagian, yaitu sebelah kiri terdapat form untuk menginputkan data sedangkan sebelah kanan berupa tabel PKL. Pada form input terdapat 3 *button* yaitu *button* map untuk menampilkan map, Simpan untuk menyimpan data, dan *Reset* untuk mereset form. Isi dari *form input* antara lain id PKL, nama PKL, jenis PKL, *latitude*, dan *longitude*. Pada tabel PKL sebelah kanan, akan berisi kolom id PKL, nama PKL, jenis PKL, *latitude*, *longitude*, dan *action*. *Action* ini berisi tombol untuk *edit* atau hapus. Sketsa halaman tabel PKL di tunjukan pada gambar 3.5.

Form Master

Peta PKL

Tabel Daftar User

Log Out

Tambah Tabel PKL

ID PKL:

Nama PKL:

Jenis PKL:

Latitude:

Longitude:

Map Simpan Reset

Master Tabel PKL

ID PKL	Jenis PKL	Nama PKL	Latitude	Longitude	Action

STIKOM SURABAYA

Gambar 3.5 Sketsa halaman Tabel PKL

d. Sketsa Halaman Tabel Pedagang

Halaman tabel pedagang ini terdiri dari 2 bagian, yaitu sebelah kiri terdapat form untuk menginputkan data sedangkan sebelah kanan berupa tabel pedagang. Pada form input terdapat 2 *button* yaitu *button* Simpan untuk menyimpan data dan *Reset* untuk mereset form. Isi dari *form input* antara lain nama PKL, nama

pedagang, jenis usaha, dan alamat. Pada tabel pedagang sebelah kanan, akan berisi kolom id pedagang, nama PKL, nama pedagang, jenis usaha, alamat, dan *action*. *Action* ini berisi tombol untuk *edit* atau hapus. Sketsa halaman tabel PKL di tunjukkan pada gambar 3.6.

ID Pedagang	Nama PKL	Nama Pedagang	Jenis Usaha	Alamat	Action

Gambar 3.6 Sketsa halaman Tabel Pedagang

e. Sketsa halaman Tabel Menu Makanan

Halaman tabel menu makanan ini terdiri dari 2 bagian, yaitu sebelah kiri terdapat form untuk menginputkan data sedangkan sebelah kanan berupa tabel menu makanan. Pada form input terdapat 2 *button* yaitu *button* Simpan untuk menyimpan data dan *Reset* untuk mereset form. Isi dari *form input* antara lain id menu makanan, nama PKL, nama pedagang, dan nama makanan. Pada tabel menu makanan sebelah kanan, akan berisi kolom id menu makanan, nama pedagang, nama makanan, dan *action*. *Action* ini berisi tombol untuk *edit* atau hapus. Sketsa halaman tabel menu makanan di tunjukkan pada gambar 3.7 pada halaman 43.

The screenshot shows a web application interface with a top navigation bar containing 'Form Master', 'Peta PKL', 'Tabel Daftar User', and 'Log Out'. The main content area is divided into two panels:

- Left Panel: 'Tambah Tabel Menu Makanan'**
 - Form fields: 'ID Menu Makanan' (text), 'Nama PKL' (text), 'Nama Pedagang' (text), 'Nama Makanan' (text).
 - Buttons: 'Simpan' and 'Reset'.
- Right Panel: 'Master Tabel Menu Makanan'**
 - Table with columns: 'ID Menu Makanan', 'Nama Pedagang', 'Nama Makanan', 'Action'.
 - The table is currently empty.

Gambar 3.7 Sketsa Halaman Tabel Menu Makanan

f. Sketsa Halaman Tabel Menu Minuman

Halaman tabel menu minuman ini terdiri dari 2 bagian, yaitu sebelah kiri terdapat form untuk menginputkan data sedangkan sebelah kanan berupa tabel menu minuman. Pada form input terdapat 2 *button* yaitu *button* Simpan untuk menyimpan data dan *Reset* untuk mereset form. Isi dari *form input* antara lain id menu makanan, nama PKL, nama pedagang, dan nama minuman. Pada tabel menu minuman sebelah kanan, akan berisi kolom id menu minuman, nama pedagang, nama minuman, dan *action*. *Action* ini berisi tombol untuk *edit* atau *hapus*. Sketsa halaman tabel menu minuman di tunjukan pada gambar 3.8 pada halaman 44.

The screenshot shows a web application interface with a top navigation bar containing 'Form Master', 'Peta PKL', 'Tabel Daftar User', and 'Log Out'. The main content area is split into two panels. The left panel, titled 'Tambah Tabel Menu Minuman', features four text input fields for 'ID Menu Minuman', 'Nama PKL', 'Nama Pedagang', and 'Nama Minuman', and two buttons labeled 'Simpan' and 'Reset'. The right panel, titled 'Master Tabel Menu Minuman', displays a table with the following structure:

ID Menu Minuman	Nama Pedagang	Nama Minuman	Action

Gambar 3.8 Sketsa Halaman Tabel Menu Minuman

g. Sketsa Halaman Tabel User

Halaman tabel user ini terdiri dari 2 bagian, yaitu sebelah kiri terdapat form untuk menginputkan data sedangkan sebelah kanan berupa tabel user. Pada form input terdapat 2 *button* yaitu *button* Simpan untuk menyimpan data dan *Reset* untuk mereset form. Isi dari *form input* antara lain id user, *username*, *password*, penempatan PKL, dan hak akses. Pada tabel user sebelah kanan, akan berisi kolom id user, *username*, hak akses, dan *action*. *Action* ini berisi tombol untuk *edit* atau hapus. Sketsa halaman tabel User di tunjukan pada gambar 3.9 pada halaman 45.

ID User	Username	Hak Akses	Action

Gambar 3.9 Sketsa Halaman Tabel User

h. Sketsa Halaman Petugas PKL

Pada header halaman Petugas PKL terdapat menu home dan tabel omset. Sketsa halaman petugas PKL ditunjukkan pada gambar 3.10.

Gambar 3.10 Sketsa Halaman Petugas PKL

i. Sketsa Halaman Tabel Omset

Halaman tabel omset ini terdiri dari 2 bagian, yaitu sebelah kiri terdapat form untuk menginputkan data sedangkan sebelah kanan berupa tabel omset. Pada form input terdapat 2 *button* yaitu *button* Simpan untuk menyimpan data dan *Reset* untuk mereset form. Isi dari *form input* antara lain id omset, nama PKL, nama pedagang, tanggal, dan jumlah omset. Pada tabel omset sebelah kanan, akan berisi kolom id omset, nama PKL, nama pedagang, tanggal, jumlah omset, dan *action*. *Action* ini berisi tombol untuk *edit* atau hapus. Sketsa halaman tabel Omset di tunjukkan pada gambar 3.11.

Sketsa halaman Tabel Omset yang menunjukkan form input di sisi kiri dan tabel data di sisi kanan. Form input memiliki field untuk ID Omset, Nama PKL, Nama Pedagang, Tanggal, dan Jumlah Omset, serta tombol Simpan dan Reset. Tabel memiliki header dengan kolom ID Omset, Nama PKL, Nama Pedagang, Tanggal, Jumlah Omset, dan Action.

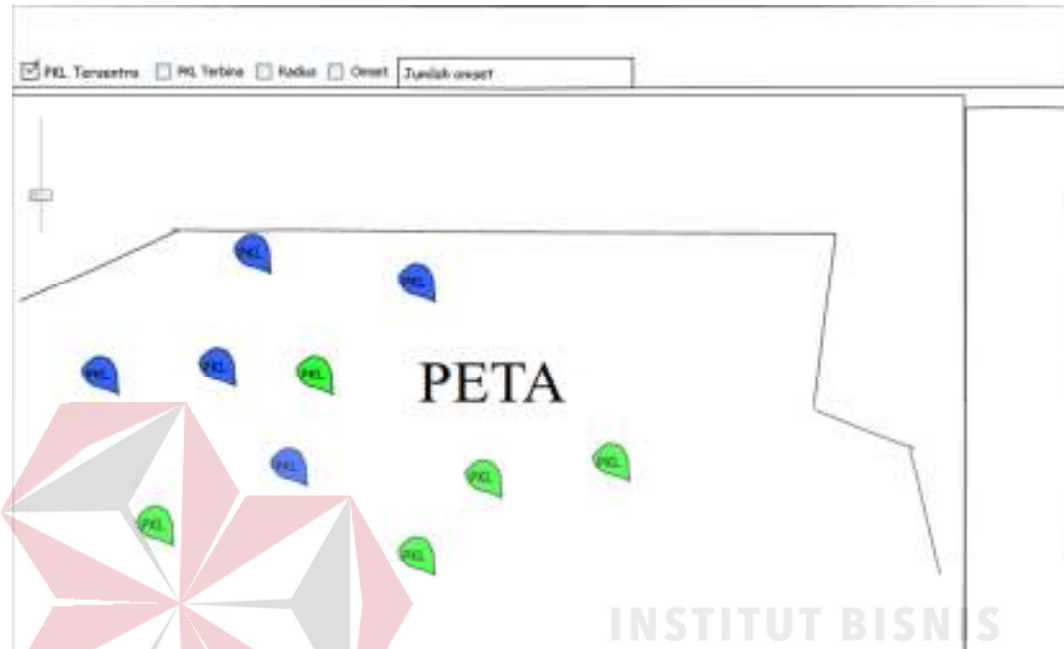
ID Omset	Nama PKL	Nama Pedagang	Tanggal	Jumlah Omset	Action

Gambar 3.11 Sketsa Halaman Tabel Omset

j. Sketsa Halaman Peta PKL

Halaman peta PKL ini terdiri dari 3 bagian, yaitu header menu yang berisi checkbox PKL tersentra, PKL terbina, radius, dan omset. Dan di bawah header terdapat peta PKL yang menunjukkan lokasi PKL, dan sebelah kiri terdapat tempat

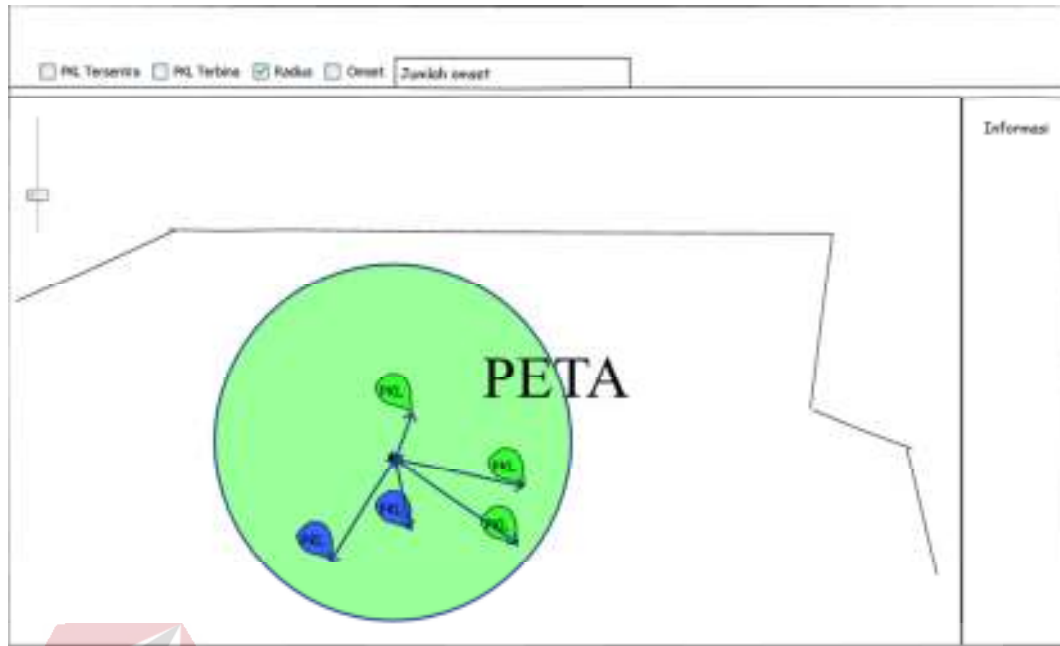
informasi jarak jika mengaktifkan menu radius. Sketsa halaman peta PKL di tunjukan pada gambar 3.12.



Gambar 3.12 Sketsa Halaman Peta PKL

k. Sketsa Halaman Radius

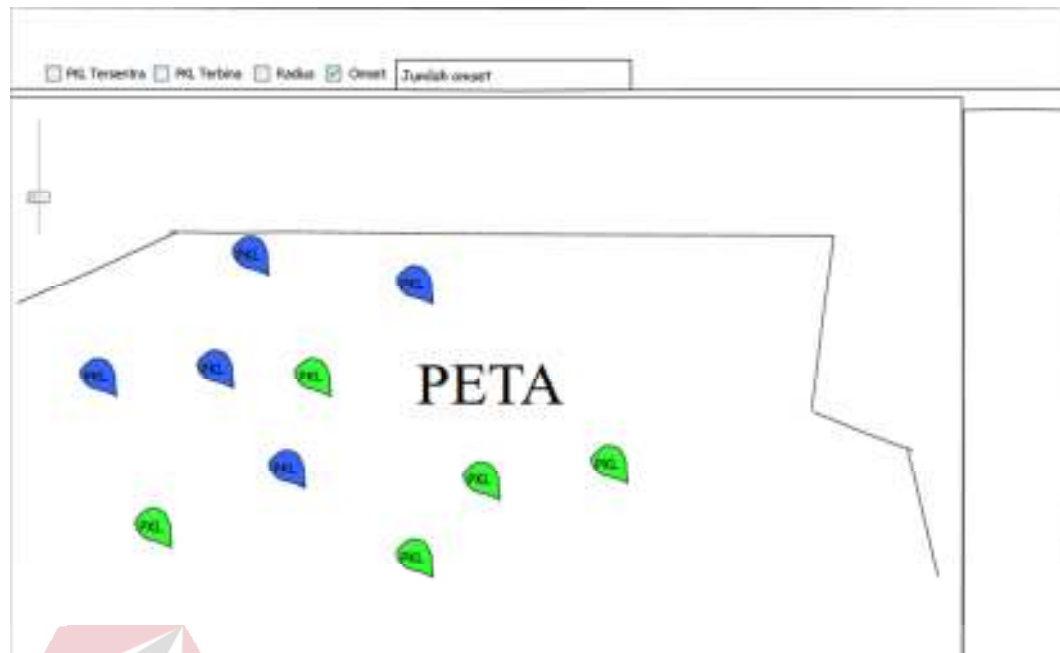
Halaman radius ini terdiri dari 3 bagian, yaitu header menu yang berisi checkbox PKL tersentra, PKL terbina, radius, dan omset. Dan di bawah header terdapat peta PKL yang dapat di diberi radius tapi sebelumnya harus mengaktifkan menu radius terlebih dahulu, dan sebelah kiri terdapat tempat informasi jarak. Sketsa halaman radius di tunjukan pada gambar 3.13 pada halaman 48.



Gambar 3.13 Sketsa Halaman Radius

1. Sketsa Halaman Omset

Halaman peta PKL ini terdiri dari 3 bagian, yaitu header menu yang berisi checkbox PKL tersentra, PKL terbina, radius, dan omset. Dan di bawah header terdapat peta PKL yang menunjukkan lokasi PKL, dan sebelah kiri terdapat tempat informasi jarak jika mengaktifkan menu radius. Fungsi omset ini akan aktif jika mencentang *checkbox* omset dan menentukan *range* omset. Sketsa halaman peta PKL di tunjukan pada gambar 3.14 pada halaman 49.

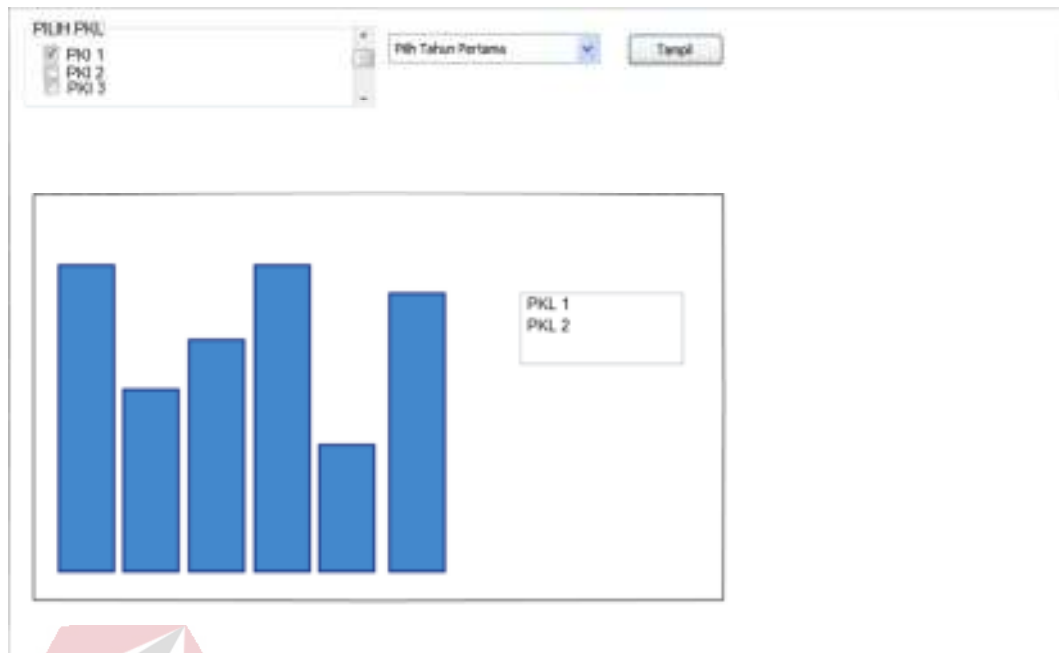


Gambar 3.14 Sketsa Halaman Omset

m. Sketsa halaman Grafik Omset PKL

Halaman grafik omset PKL ini terdiri dari 3 bagian, yaitu menu untuk memilih PKL, kolom untuk memilih tahun, dan tempat untuk menampilkan grafik. Dimana user akan memilih PKL terlebih dahulu yang ingin dilihat omsetnya setelah itu memilih tahun dan aplikasi akan menampilkan grafiknya. Sketsa halaman grafik omsets PKL di tunjukan pada gambar 3.15 pada halaman

50.



Gambar 3.15 Sketsa Halaman Grafik Omset PKL

3.2.2 Iterasi Pemodelan

Pada tahap pemodelan iterasi penulis menyusun jadwal iterasi yang akan dilakukan berdasarkan gambaran yang didapat pada saat tahap *envisioning*.

Jadwal iterasi ditunjukkan pada tabel 3.5

Tabel 3.5 Jadwal Iterasi Dalam Pengembangan

Iterasi	Implementasi	Use Case	Prioritas
1	User Stories A03, A04, A05, A06, A07, dan PT02	Memasukkan data master PKL, maintenance data master PKL, memasukkan data master pedagang, maintenance data master pedagang, memasukkan data master menu makanan, maintenance data master menu makanan, memasukkan master menu minuman, maintenance data master menu minuman, memasukkkan data master user, maintenance data master user, memasukkan data master omset, dan maintenance data master omset.	10
2	A08, A09, A10, P02, M01, M02,	Melakukan view peta PKL, informasi peta PKL, Melakukan view radius, informasi radius, melakukan view omset,	5

Tabel 3.5 Jadwal Iterasi Dalam Pengembangan

Iterasi	Implementasi	Use Case	Prioritas
	dan P03	dan melihat grafik omset PKL.	

3.2.3 Model Stroming

Pada tahap ini penulis memulai melakukan pemodelan berdasarkan iterasi yang telah ditentukan sebelumnya. Tahap yang penulis lakukan adalah membuat flow-of-event dan ERD dari sistem. Iterasi yang di buat yaitu:

A. Iterasi 1

Pada iterasi ini akan dijelaskan tahap-tahap bagaimana administrator melakukan memasukkan data master PKL, maintenance data master PKL, memasukkan data master pedagang, maintenance data master pedagang, memasukkan data master menu makanan, maintenance data master menu makanan, memasukkan master menu minuman, maintenance data master menu minuman, memasukkkan data master user, maintenance data master user, memasukkan data master omset, dan maintenance data master omset. Literasi 1 akan dijelaskan antara lain:

1. Flow Of Event

a. Flow Of Event Memasukkan Data Master PKL

Flow of events untuk *use case* memasukkan data master PKL dapat dilihat pada tabel 3.6 pada halaman 52.

Tabel 3.6 Flow Of Event Memasukkan Data master PKL

Deskripsi Singkat :	Use case untuk melakukan insert ke master data PKL.			
Kondisi Awal :	Admin mengakses Form Master Tabel PKL.			
Kondisi Akhir :	Admin dapat menambah PKL.			
Aliran Kejadian Utama :		Aksi Pemakai		Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master PKL.	2	Sistem menyiapkan form yang akan digunakan untuk mengisi data PKL.
	3	Admin memasukkan data PKL yaitu id PKL, nama PKL, Jenis PKL, dan lokasi PKL. Untuk memberikan lokasi PKL, admin harus menggunakan fasilitas map yang telah disediakan. Setelah admin memasukan semua data, admin akan menekan tombol simpan.	4	Sistem akan melakukan validasi dari inputan admin, yang bertugas melakukan validasi adalah class control. Control akan melakukan insert ke tabel PKL. Setelah melakukan <i>insert</i> , sistem akan menampilkan konfirmasi sukses.
Aliran Kejadian Alternatif :		Aksi Pemakai		Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master PKL.	2	Sistem menyiapkan form yang akan digunakan untuk mengisi data PKL.

	3	Admin memasukkan data PKL yaitu id PKL, nama PKL, Jenis PKL, dan lokasi PKL. Untuk memberikan lokasi PKL, admin harus menggunakan fasilitas map yang telah disediakan. Setelah admin memasukan semua data, admin akan menekan tombol simpan.	4	Sistem akan melakukan validasi dari inputan admin, yang bertugas melakukan validasi adalah class control. Control akan melakukan <i>insert</i> ke tabel PKL. Jika dalam proses insert di temukan error yang berupa data tersebut sudah ada di database atau data yang di inputkan kurang lengkap, maka sistem tidak akan melakukan penyimpanan dan sistem akan memberi <i>warning error</i> .
--	---	--	---	---

b. Flow of Events Maintenance Data Master PKL

Flow of events untuk *use case maintenance* data master PKL dapat dilihat pada tabel 3.7.

Tabel 3.7 Flow Of Events Maintenance Data Master PKL

Deskripsi Singkat :	Use case untuk melakukan <i>maintenance</i> ke master data PKL.		
Kondisi Awal :	Admin mengakses Form Master Tabel PKL.		
Kondisi Akhir :	Admin dapat <i>maintenance</i> PKL.		
Aliran Kejadian Utama :		Aksi Pemakai	Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master PKL.	2 Sistem menyiapkan form yang akan digunakan untuk <i>maintenance</i> PKL.
	3	Admin akan memilih data PKL yang akan di <i>maintenance</i> dengan cara menekan tombol di kolom <i>Action</i> . jika admin melakukan hapus data maka admin menekan tombol silang merah.	4 Sistem akan menampilkan konfirmasi yang berisi “Apakah anda yakin untuk menghapus?”, warning ini sebagai validasi jika admin setuju untuk menghapus data yang telah di pilih. Jika admin memilih OK maka sistem akan melakukan <i>delete</i> data yang telah di pilih.

	5	Admin akan memilih data PKL yang akan di maintenance dengan cara menekan tombol di kolom <i>Action</i> . jika admin melakukan edit data maka admin menekan tombol pensil. Dan admin akan melakukan edit data. Setelah selesai admin akan menekan tombol Update.	6	Sistem akan memberikan form update kepada admin. Pada form itu admin melakukan pembaruan isi data yang telah dipilih. Setelah admin menekan tombol update, sistem akan melakukan update data yang akan di lakukan oleh class control.
--	---	---	---	---

c. Flow Of Event Memasukkan Data Master Pedagang

Flow of events untuk *use case* memasukkan data master PKL dapat dilihat pada tabel 3.8.

Tabel 3.8 Flow Of Event Memasukkan Data Master Pedagang

Deskripsi Singkat :	Use case untuk melakukan insert ke master data pedagang.		
Kondisi Awal :	Admin mengakses Form Master Tabel pedagang.		
Kondisi Akhir :	Admin dapat menambah pedagang.		
Aliran Kejadian Utama :		Aksi Pemakai	Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master pedagang.	2 Sistem menyiapkan form yang akan digunakan untuk mengisi data pedagang.
	3	Admin akan memilih PKL yang akan di tambah pedagangnya. Admin memasukkan data pedagang yaitu id pedagang, nama pedagang, jenis usaha, dan alamat. Setelah admin memasukan semua data, admin akan menekan tombol simpan.	4 Sistem akan melakukan validasi dari inputan admin, yang bertugas melakukan validasi adalah class control. Control akan melakukan insert ke tabel pedagang. Setelah melakukan <i>insert</i> , sistem akan menampilkan konfirmasi sukses.

Aliran Kejadian Alternatif :	Aksi Pemakai		Respon Sistem	
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master pedagang.	2	Sistem menyiapkan form yang akan digunakan untuk mengisi data pedagang.
3	Admin akan memilih PKL yang akan di tambah pedagangnya. Admin memasukkan data pedagang yaitu id pedagang, nama pedagang, jenis usaha, dan alamat. Setelah admin memasukan semua data, admin akan menekan tombol simpan.	4	Sistem akan melakukan validasi dari inputan admin, yang bertugas melakukan validasi adalah class control. Control akan melakukan <i>insert</i> ke tabel pedagang. Jika dalam proses insert di temukan error yang berupa data tersebut sudah ada di database atau data yang di inputkan kurang lengkap, maka sistem tidak akan melakukan penyimpanan dan sistem akan memberi <i>warning error</i> .	

d. Flow of Events Maintenance Data Master Pedagang

Flow of events untuk *use case maintenance* data master pedagang dapat dilihat pada tabel 3.9

Tabel 3.9 Flow Of Events Maintenance Data Master Pedagang

Deskripsi Singkat :	Use case untuk melakukan <i>maintenance</i> ke master data pedagang.			
Kondisi Awal :	Admin mengakses Form Master Tabel Pedagang.			
Kondisi Akhir :	Admin dapat <i>maintenance</i> pedagang.			
Aliran Kejadian Utama :	Aksi Pemakai		Respon Sistem	
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master pedagang.	2	Sistem menyiapkan form yang akan digunakan untuk <i>maintenance</i> pedagang.
	3	Admin akan memilih data pedagang yang	4	Sistem akan menampilkan konfirmasi yang berisi “Apakah anda yakin untuk

		akan di maintenance dengan cara menekan tombol di kolom <i>Action</i> . jika admin melakukan hapus data maka admin menekan tombol silang merah.		menghapus ?”, warning ini sebagai validasi jika admin setuju untuk menghapus data yang telah di pilih. Jika admin memilih OK maka sistem akan melakukan <i>delete</i> data yang telah di pilih.
	5	Admin akan memilih data pedagang yang akan di maintenance dengan cara menekan tombol di kolom <i>Action</i> . jika admin melakukan edit data maka admin menekan tombol pensil. Dan admin akan melakukan edit data. Setelah selesai admin akan menekan tombol Update.	6	Sistem akan memberikan form update kepada admin. Pada form itu admin melakukan pembaruan isi data yang telah dipilih. Setelah admin menekan tombol update, sistem akan melakukan update data yang akan di lakukan oleh class control.

e. Flow Of Event Memasukkan Data Master Menu Makanan

Flow of events untuk *use case* memasukkan data master menu makanan dapat dilihat pada tabel 3.10 di halaman 57.

Tabel 3.10 Flow Of Events Memasukkan Data Master Menu Makanan

Deskripsi Singkat :	Use case untuk melakukan insert ke master data menu makanan.			
Kondisi Awal :	Admin mengakses Form Master Tabel Menu Makanan.			
Kondisi Akhir :	Admin dapat menambah menu makanan.			
Aliran Kejadian Utama :		Aksi Pemakai		Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master menu makanan.	2	Sistem menyiapkan form yang akan digunakan untuk mengisi data menu makanan.
	3	Admin akan memilih PKL kemudian memilih pedagang yang akan di tambah menu makanan. Admin memasukkan data menu makanan yaitu id menu makanan dan nama makanan. Setelah admin memasukan semua data, admin akan menekan tombol simpan.	4	Sistem akan melakukan validasi dari inputan admin, yang bertugas melakukan validasi adalah class control. Control akan melakukan insert ke tabel menu makanan. Setelah melakukan <i>insert</i> , sistem akan menampilkan konfirmasi sukses.
Aliran Kejadian Alternatif :		Aksi Pemakai		Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master menu makanan.	2	Sistem menyiapkan form yang akan digunakan untuk mengisi data menu makanan.
	3	Admin akan memilih PKL kemudian memilih pedagang yang akan di tambah menu makanan. Admin memasukkan data menu makanan yaitu id menu makanan dan nama makanan. Setelah admin memasukan	4	Sistem akan melakukan validasi dari inputan admin, yang bertugas melakukan validasi adalah class control. Control akan melakukan <i>insert</i> ke tabel menu makanan. Jika dalam proses insert di temukan error yang berupa data tersebut sudah ada di database atau data yang di inputkan kurang

		semua data, admin akan menekan tombol simpan.		lengkap, maka sistem tidak akan melakukan penyimpanan dan sistem akan memberi <i>warning error</i> .
--	--	---	--	--

f. Flow Of Event Maintenance Data Master Menu Makanan

Flow of events untuk *use case maintenance* data master menu makanan

dapat dilihat pada tabel 3.11

Tabel 3.11 Flow Of Events Maintenance Data Master Menu Makanan

Deskripsi Singkat :	Use case untuk melakukan <i>maintenance</i> ke master data menu makanan.		
Kondisi Awal :	Admin mengakses Form Master Tabel Menu Makanan.		
Kondisi Akhir :	Admin dapat <i>maintenance</i> menu makanan.		
Aliran Kejadian Utama :		Aksi Pemakai	Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master menu makanan.	2 Sistem menyiapkan form yang akan digunakan untuk <i>maintenance</i> menu makanan.
	3	Admin akan memilih data menu makanan yang akan di <i>maintenance</i> dengan cara menekan tombol di kolom <i>Action</i> . Jika admin melakukan hapus data maka admin menekan tombol silang merah.	4 Sistem akan menampilkan konfirmasi yang berisi “Apakah anda yakin untuk menghapus ?”, warning ini sebagai validasi jika admin setuju untuk menghapus data yang telah di pilih. Jika admin memilih OK maka sistem akan melakukan <i>delete</i> data yang telah di pilih.
	5	Admin akan memilih data menu makanan yang akan di <i>maintenance</i> dengan cara menekan tombol di kolom <i>Action</i> . jika admin melakukan edit data maka admin menekan tombol pensil. Dan admin akan	6 Sistem akan memberikan form update kepada admin. Pada form itu admin melakukan pembaruan isi data yang telah dipilih. Setelah admin menekan tombol update, sistem akan melakukan update data yang akan di lakukan oleh class control.

		melakukan edit data. Setelah selesai admin akan menekan tombol Update.		
--	--	--	--	--

g. Flow Of Event Memasukkan Data Master Menu Minuman

Flow of events untuk *use case* memasukkan data master menu minuman dapat dilihat pada tabel 3.12.

Tabel 3.12 Flow Of Events Memasukkan Data Master Menu Minuman

Deskripsi Singkat :	Use case untuk melakukan insert ke master data menu minuman.			
Kondisi Awal :	Admin mengakses Form Master Tabel Menu Minuman.			
Kondisi Akhir :	Admin dapat menambah menu minuman.			
Aliran Kejadian Utama :		Aksi Pemakai		Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master menu minuman.	2	Sistem menyiapkan form yang akan digunakan untuk mengisi data menu minuman.
	3	Admin akan memilih PKL kemudian memilih pedagang yang akan di tambah menu minuman. Admin memasukkan data menu minuman yaitu id menu minuman dan nama minuman. Setelah admin memasukan semua data, admin akan menekan tombol simpan.	4	Sistem akan melakukan validasi dari inputan admin, yang bertugas melakukan validasi adalah class control. Control akan melakukan insert ke tabel menu minuman. Setelah melakukan <i>insert</i> , sistem akan menampilkan konfirmasi sukses.
Aliran Kejadian Alternatif :		Aksi Pemakai		Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master menu minuman.	2	Sistem menyiapkan form yang akan digunakan untuk mengisi data menu minuman.
	3	Admin akan memilih PKL kemudian memilih	4	Sistem akan melakukan validasi dari inputan admin, yang bertugas

		pedagang yang akan di tambah menu minuman. Admin memasukkan data menu minuman yaitu id menu minuman dan nama minuman. Setelah admin memasukan semua data, admin akan menekan tombol simpan.		melakukan validasi adalah class control. Control akan melakukan <i>insert</i> ke tabel menu minuman. Jika dalam proses insert di temukan error yang berupa data tersebut sudah ada di database atau data yang di inputkan kurang lengkap, maka sistem tidak akan melakukan penyimpanan dan sistem akan memberi <i>warning error</i> .	
--	--	---	--	---	--

h. Flow Of Event Maintenance Data Master Menu Minuman

Flow of events untuk *use case maintenance* data master menu minuman dapat dilihat pada tabel 3.13.

Tabel 3.13 Flow Of Events Maintenance Data Master Menu Minuman

Deskripsi Singkat :	Use case untuk melakukan <i>maintenance</i> ke master data menu minuman.		
Kondisi Awal :	Admin mengakses Form Master Tabel Menu Minuman.		
Kondisi Akhir :	Admin dapat <i>maintenance</i> menu minuman.		
Aliran Kejadian Utama :		Aksi Pemakai	Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master menu minuman.	2 Sistem menyiapkan form yang akan digunakan untuk <i>maintenance</i> menu minuman.
	3	Admin akan memilih data menu minuman yang akan di <i>maintenance</i> dengan cara menekan tombol di kolom <i>Action</i> . Jika admin melakukan hapus data maka admin menekan tombol silang merah.	4 Sistem akan menampilkan konfirmasi yang berisi “Apakah anda yakin untuk menghapus ?”, warning ini sebagai validasi jika admin setuju untuk menghapus data yang telah di pilih. Jika admin memilih OK maka sistem akan melakukan <i>delete</i> data

				yang telah di pilih.
	5	Admin akan memilih data menu minuman yang akan di maintenance dengan cara menekan tombol di kolom <i>Action</i> . jika admin melakukan edit data maka admin menekan tombol pensil. Dan admin akan melakukan edit data. Setelah selesai admin akan menekan tombol Update.	6	Sistem akan memberikan form update kepada admin. Pada form itu admin melakukan pembaruan isi data yang telah dipilih. Setelah admin menekan tombol update, sistem akan melakukan update data yang akan di lakukan oleh class control.

i. Flow Of Event Memasukkan Data Master User

Flow of events untuk *use case* memasukkan data master user dapat dilihat pada tabel 3.14.

Tabel 3.14 Flow Of Events Memasukkan Data Master User

Deskripsi Singkat :	Use case untuk melakukan insert ke master data user.			
Kondisi Awal :	Admin mengakses Form Master Tabel Daftar User.			
Kondisi Akhir :	Admin dapat menambah user.			
Aliran Kejadian Utama :		Aksi Pemakai		Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master daftar user.	2	Sistem menyiapkan form yang akan digunakan untuk mengisi data user.
	3	Admin memasukkan data user yaitu id user dan password. Admin akan memilih penempatan PKL jika yang dimasukan adalah petugas PKL, kemudian admin memilih hak akses. Setelah admin memasukan semua data, admin akan	4	Sistem akan melakukan validasi dari inputan admin, yang bertugas melakukan validasi adalah class control. Control akan melakukan insert ke tabel login. Setelah melakukan <i>insert</i> , sistem akan menampilkan konfirmasi sukses.

		menekan tombol simpan.		
Aliran Kejadian Alternatif :		Aksi Pemakai		Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master daftar user.	2	Sistem menyiapkan form yang akan digunakan untuk mengisi data user.
	3	Admin memasukkan data user yaitu id user dan password. Admin akan memilih penempatan PKL jika yang dimasukan adalah petugas PKL, kemudian admin memilih hak akses. Setelah admin memasukan semua data, admin akan menekan tombol simpan.	4	Sistem akan melakukan validasi dari inputan admin, yang bertugas melakukan validasi adalah class control. Control akan melakukan <i>insert</i> ke tabel login. Jika dalam proses insert di temukan error yang berupa data tersebut sudah ada di database atau data yang di inputkan kurang lengkap, maka sistem tidak akan melakukan penyimpanan dan sistem akan memberi <i>warning error</i> .

j. Flow Of Event Maintenance Data Master User

Flow of events untuk *use case maintenance* data master user dapat dilihat pada tabel 3.15.

Tabel 3.15 Flow Of Events Maintenance Data Master User

Deskripsi Singkat :	Use case untuk melakukan <i>maintenance</i> ke master data user.			
Kondisi Awal :	Admin mengakses Form Master Tabel Daftar User.			
Kondisi Akhir :	Admin dapat <i>maintenance</i> user.			
Aliran Kejadian Utama :		Aksi Pemakai		Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form	2	Sistem menyiapkan form yang akan digunakan untuk <i>maintenance</i> user.

		master daftar user.		
	3	Admin akan memilih data user yang akan di maintenance dengan cara menekan tombol di kolom <i>Action</i> . Jika admin melakukan hapus data maka admin menekan tombol silang merah.	4	Sistem akan menampilkan konfirmasi yang berisi “Apakah anda yakin untuk menghapus ?”, warning ini sebagai validasi jika admin setuju untuk menghapus data yang telah di pilih. Jika admin memilih OK maka sistem akan melakukan <i>delete</i> data yang telah di pilih.
	5	Admin akan memilih data user yang akan di maintenance dengan cara menekan tombol di kolom <i>Action</i> . jika admin melakukan edit data maka admin menekan tombol pensil. Dan admin akan melakukan edit data. Setelah selesai admin akan menekan tombol Update.	6	Sistem akan memberikan form update kepada admin. Pada form itu admin melakukan pembaruan isi data yang telah dipilih. Setelah admin menekan tombol update, sistem akan melakukan update data yang akan di lakukan oleh class control.

k. Flow Of Event Memasukkan Data Master Omset

Flow of events untuk *use case* memasukkan data master omset dapat dilihat pada tabel 3.16.

Tabel 3.16 Flow Of Events Memasukkan Data Master Omset

Deskripsi Singkat :	Use case untuk melakukan insert ke master data omset.
Kondisi Awal :	Petugas PKL mengakses Form Master Tabel omset.
Kondisi Akhir :	Petugas PKL dapat menambah omset.

Aliran Kejadian Utama :		Aksi Pemakai		Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master tabel omset.	2	Sistem menyiapkan form yang akan digunakan untuk mengisi data omset.
	3	Petugas akan memilih pedagang yang akan di tambah omset. Petugas PKL memasukkan data omset yaitu id omset, tanggal dan jumlah omset. Setelah petugas PKL memasukan semua data, Petugas PKL akan menekan tombol simpan.	4	Sistem akan melakukan validasi dari inputan petugas PKL, yang bertugas melakukan validasi adalah class control. Control akan melakukan insert ke tabel omset. Setelah melakukan <i>insert</i> , sistem akan menampilkan konfirmasi sukses.
Aliran Kejadian Alternatif :		Aksi Pemakai		Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master tabel omset.	2	Sistem menyiapkan form yang akan digunakan untuk mengisi data omset.
	3	Petugas akan memilih pedagang yang akan di tambah omset. Petugas PKL memasukkan data omset yaitu id omset, tanggal dan jumlah omset. Setelah petugas PKL memasukan semua data, Petugas PKL akan menekan tombol simpan.	4	Sistem akan melakukan validasi dari inputan admin, yang bertugas melakukan validasi adalah class control. Control akan melakukan <i>insert</i> ke tabel omset. Jika dalam proses insert di temukan error yang berupa data tersebut sudah ada di database atau data yang di inputkan kurang lengkap, maka sistem tidak akan melakukan penyimpanan dan sistem akan memberi <i>warning error</i> .

1. Flow Of Event Maintenance Data Master Omset

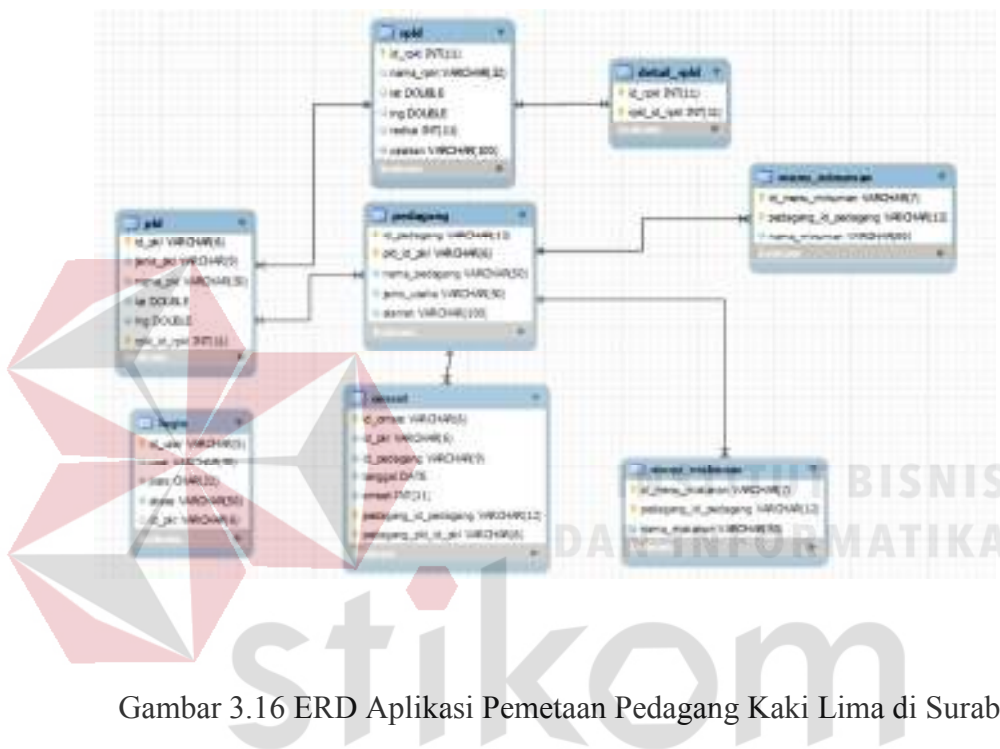
Flow of events untuk *use case maintenance* data master omset dapat dilihat pada tabel 3.17.

Tabel 3.17 Flow Of Events Maintenance Data Master Omset

Deskripsi Singkat :	Use case untuk melakukan <i>maintenance</i> ke master data omset.		
Kondisi Awal :	Petugas PKL mengakses Form Master Tabel Omset.		
Kondisi Akhir :	Petugas PKL dapat <i>maintenance</i> omset.		
Aliran Kejadian Utama :		Aksi Pemakai	Respon Sistem
	1	<i>Use case</i> dimulai ketika pengguna memilih menu form master tabel omset.	2 Sistem menyiapkan form yang akan digunakan untuk <i>maintenance</i> omset.
	3	Petugas PKL akan memilih data omset yang akan di <i>maintenance</i> dengan cara menekan tombol di kolom <i>Action</i> . Jika petugas PKL melakukan hapus data maka petugas PKL menekan tombol silang merah.	4 Sistem akan menampilkan konfirmasi yang berisi “Apakah anda yakin untuk menghapus ?”, warning ini sebagai validasi jika petugas PKL setuju untuk menghapus data yang telah di pilih. Jika petugas PKL memilih OK maka sistem akan melakukan <i>delete</i> data yang telah di pilih.
	5	Petugas PKL akan memilih data omset yang akan di <i>maintenance</i> dengan cara menekan tombol di kolom <i>Action</i> . jika petugas PKL melakukan edit data maka admin menekan tombol pensil. Dan petugas PKL akan melakukan edit data. Setelah selesai admin akan menekan tombol Update.	6 Sistem akan memberikan form update kepada petugas PKL. Pada form itu petugas PKL melakukan pembaruan isi data yang telah dipilih. Setelah admin menekan tombol update, sistem akan melakukan update data yang akan di lakukan oleh class control.

2. ERD

ERD Aplikasi Pemetaan Pedagang Kaki Lima di Surabaya ini di tunjukkan pada gambar 3.14. Table yang diidentifikasi antara lain table PKL, pedagang, menu makanan, menu minuman, login, rancang pkl, detail rancang pkl dan omset. Untuk lebih jelasnya relasi antar tabel dapat dilihat pada gambar 3.16.



Gambar 3.16 ERD Aplikasi Pemetaan Pedagang Kaki Lima di Surabaya

B. Iterasi 2

Pada iterasi ini akan dijelaskan tahap-tahap bagaimana administrator atau masyarakat melakukan melakukan view peta PKL serta mendapatkan informasi peta PKL, administrator atau pimpinan melakukan view radius serta mendapatkan informasi radius, dan administrator melakukan view omset.. Literasi 2 akan dijelaskan antara lain:

1. Flow Of Event Melakukan View Peta PKL

Flow of events untuk *use case* melakukan view peta PKL dapat dilihat pada tabel 3.18.

Tabel 3.18. Flow Of Events Melakukan View Peta PKL

Deskripsi Singkat :	Use case untuk melakukan view peta PKL.		
Kondisi Awal :	Admin atau masyarakat melakukan view peta PKL.		
Kondisi Akhir :	Admin atau masyarakat menerima informasi peta PKL.		
Aliran Kejadian Utama :		Aksi Pemakai	Respon Sistem
	1	<i>Use case</i> dimulai ketika admin atau masyarakat memilih menu peta PKL.	2 Sistem menyiapkan halaman serta map yang akan di <i>load</i> sesuai dengan inisialisasi awal.
	3	Admin atau masyarakat akan memilih PKL mana saja yang akan di lihat dengan cara mencentang <i>checkbox</i> PKL tersentra jika ingin melihat PKL tersentra atau mencentang <i>checkbox</i> PKL Terbina jika ingin melihat PKL terbina.	4 Sistem akan <i>request</i> ke <i>google maps</i> api sesuai inputan <i>checkbox</i> dari user. Yang dimana <i>google maps</i> api akan merespon <i>request</i> dan memberikan <i>output</i> map yang berisi <i>marker</i> posisi PKL yang dipilih user.
Aliran Kejadian Alternatif :		Aksi Pemakai	Respon Sistem
	1	<i>Use case</i> dimulai ketika admin atau masyarakat memilih menu peta PKL.	2 Sistem menyiapkan halaman serta map yang akan di <i>load</i> sesuai dengan inisialisasi awal.
	3	Admin atau masyarakat akan memilih PKL mana saja yang akan di lihat dengan cara mencentang <i>checkbox</i> PKL tersentra jika ingin melihat PKL tersentra atau mencentang	4 Sistem akan <i>request</i> ke <i>google maps</i> api sesuai inputan <i>checkbox</i> dari user. Yang dimana <i>google maps</i> api akan merespon <i>request</i> dan memberikan <i>output</i> map yang berisi <i>marker</i> posisi PKL yang dipilih user. Jika dalam <i>Google Maps</i> api tidak memberikan <i>output marker maps</i> yang sesuai itu

		<i>checkbox</i> PKL Terbina jika ingin melihat PKL terbina.		terjadi karena koneksi internet dari user itu terputus atau mengalami gangguan. Maka tampilan maps akan kosong karena sistem tidak menerima hasil dari google <i>maps</i> api.
--	--	---	--	--

2. Flow Of Event Melakukan View Radius

Flow of events untuk *use case* melakukan view Radius dapat dilihat pada tabel 3.19.

Tabel 3.19 Flow Of Events Melakukan View Radius

Deskripsi Singkat :	Use case untuk melakukan view radius.			
Kondisi Awal :	Admin atau pimpinan melakukan view radius.			
Kondisi Akhir :	Admin atau pimpinan menerima informasi radius.			
Aliran Kejadian Utama :		Aksi Pemakai		Respon Sistem
	1	<i>Use case</i> dimulai ketika admin atau pimpinan memilih menu peta PKL dan mencentang menu radius pada menu di <i>header</i> .	2	Sistem menyiapkan halaman serta map yang akan di <i>load</i> sesuai dengan inisialisasi awal.
	3	Admin atau pimpinan akan menentukan diameter radius. Setelah menentukan diameter radius, admin atau pimpinan akan memilih tempat pada peta dengan cara mengklik area di dalam peta kota Surabaya. Admin atau pimpinan juga harus mengatur minimum omset untuk PKL. Dan jika lokasi tersebut di	4	Sistem akan <i>request</i> ke google <i>maps</i> api sesuai dengan inputan. Sistem akan memberikan parameter diameter radius dan posisi <i>latitude longitude</i> titik tengah radius ke google <i>maps</i> api. Google <i>maps</i> api akan merespon <i>request</i> dari sistem. Google maps api akan menampilkan radius yang telah ditentukan serta menampilkan PKL tersentra dan terbina yang tercakup di radius serta jika PKL yang berada di dalam radius tersebut memiliki omset

		<p>anggap sesuai maka lokasi tersebut akan di simpan sebagai rancangan lokasi PKL.</p>	<p>dibawah minimum maka <i>marker</i> akan berwarna merah. Google maps api juga akan menghitung jarak titik tengah radius menuju <i>marker</i> PKL yang masuk di dalam area radius. Jika jarak titik tengah radius ke PKL tersentra melebihi jarak minimum yang telah ditentukan maka <i>marker</i> titik tengah radius akan berwarna hijau, sedangkan jika jaraknya masih di bawah minimum yang telah di tentukan maka akan berwarna merah. Sistem akan menampilkan jarak dari google <i>maps</i> api dan menampilkan di bagian informasi.</p>
<p>Aliran Kejadian Alternatif :</p>		<p>Aksi Pemakai</p>	<p>Respon Sistem</p>
	<p>1</p>	<p><i>Use case</i> dimulai ketika admin atau pimpinan memilih menu peta PKL dan mencentang menu radius pada menu di <i>header</i>.</p>	<p>2</p> <p>Sistem menyiapkan halaman serta map yang akan di <i>load</i> sesuai dengan inisialisasi awal.</p>
	<p>3</p>	<p>Admin atau pimpinan akan menentukan diameter radius. Setelah menentukan diameter radius, admin atau pimpinan akan memilih tempat pada peta dengan cara mengklik area di dalam peta kota Surabaya. Admin atau pimpinan juga harus mengatur</p>	<p>4</p> <p>Sistem akan <i>request</i> ke google <i>maps</i> api sesuai dengan inputan. Sistem akan memberikan parameter diameter radius dan posisi <i>latitude longitude</i> titik tengah radius ke google <i>maps</i> api. Google <i>maps</i> api akan merespon <i>request</i> dari sistem. Google <i>maps</i> api akan menampilkan radius yang telah ditentukan serta menampilkan PKL tersentra dan terbina yang tercakup di</p>

		<p>minimum omset untuk PKL. Serta admin atau pimpinan juga akan mengatur jarak minimum. Dan jika lokasi tersebut di anggap sesuai maka lokasi tersebut akan di simpan sebagai rancangan lokasi PKL.</p>	<p>radius serta jika PKL yang berada di dalam radius tersebut memiliki omset dibawah minimum maka <i>marker</i> akan berwarna merah. <i>Google maps</i> api juga akan menghitung jarak titik tengah radius menuju marker PKL yang masuk di dalam area radius. Jika jarak titik tengah radius ke PKL tersentra melebihi jarak minimum yang telah ditentukan maka <i>marker</i> titik tengah radius akan berwarna hijau, sedangkan jika jaraknya masih di bawah minimum yang telah di tentukan maka akan berwarna merah. Sistem akan menampilkan jarak dari <i>google maps</i> api dan menampilkan di bagian informasi. Jika admin atau Pimpinan mengklik posisi radius pada peta dengan cepat, maka ada beberapa <i>marker</i> PKL yang seharusnya muncul di area radius tetapi tidak muncul. Itu dikarena kan <i>request</i> dari <i>google maps</i> api berjalan lambat. Dan bukan kesalahan dari Sistem.</p>
--	--	---	--

3. Flow Of Event Melakukan View Omset

Flow of events untuk *use case* melakukan view omset dapat dilihat pada tabel 3.20 pada halaman 71.

Tabel 3.20 Flow Of Events Melakukan View Omset

Deskripsi Singkat :	Use case untuk melakukan view omset.		
Kondisi Awal :	Admin melakukan view omset.		
Kondisi Akhir :	Admin menerima informasi omset.		
Aliran Kejadian Utama :		Aksi Pemakai	Respon Sistem
	1	<i>Use case</i> dimulai ketika admin memilih menu peta PKL dan mencentang menu omset pada menu di <i>header</i> .	2 Sistem menyiapkan halaman serta map yang akan di <i>load</i> sesuai dengan inisialisasi awal.
	3	Admin akan menentukan <i>range</i> omset PKL.	4 Sistem akan <i>request</i> ke <i>google maps</i> api sesuai dengan inputan. Sistem akan memberikan parameter latitude longitude mana saja yang akan harus di tampilkan, yang sebelumnya telah di proses oleh sistem yaitu inputan <i>range</i> omset. <i>Google maps</i> api akan merespon <i>request</i> dari sistem. <i>Google maps</i> api akan menampilkan <i>markers</i> sesuai dengan parameter dari sistem.
Aliran Kejadian Alternatif :		Aksi Pemakai	Respon Sistem
	1	<i>Use case</i> dimulai ketika admin memilih menu peta PKL dan mencentang menu omset pada menu di <i>header</i> .	2 Sistem menyiapkan halaman serta map yang akan di <i>load</i> sesuai dengan inisialisasi awal.
	3	Admin akan menentukan <i>range</i> omset PKL.	4 Sistem akan <i>request</i> ke <i>google maps</i> api sesuai dengan inputan. Sistem akan memberikan parameter latitude longitude mana saja yang akan harus di

			tampilkan, yang sebelumnya telah di proses oleh sistem yaitu inputan range omset. Google maps api akan merespon <i>request</i> dari sistem. Google maps api akan menampilkan <i>markers</i> sesuai dengan parameter dari sistem. Jika terdapat <i>markers</i> yang seharusnya muncul tetapi tidak muncul itu karena koneksi internet untuk <i>request</i> ke google maps api terjadi gangguan atau terputus.
--	--	--	--

4. Flow Of Event Melakukan View Grafik Omset PKL

Flow of events untuk *use case* melakukan view grafik omset dapat dilihat pada tabel 3.21.

Tabel 3.21 Flow Of Events Melakukan View Grafik Omset PKL

Deskripsi Singkat :	Use case untuk melakukan view grafik omset PKL.		
Kondisi Awal :	Pimpinan melakukan view grafik omset.		
Kondisi Akhir :	Admin menerima informasi grafik omset.		
Aliran Kejadian Utama :		Aksi Pemakai	Respon Sistem
	1	<i>Use case</i> dimulai ketika pimpinan login sebagai pimpinan.	2 Sistem menyiapkan halaman grafik omset garis untuk di tampilkan pertama kali.
	3	Pimpinan memilih PKL yang akan dilihat grafik nya, setelah itu pimpinan akan memilih tahun yang akan dilihat omset nya. Setelah itu pimpinan menekan tombol	4 Sistem akan memproses inputan dari user yang berupa PKL yang dipilih, serta tahun yang pilih. Sistem akan melakukan <i>SELECT</i> data pada database sesuai dengan inputan user. Setelah itu sistem akan menampilkan dalam bentuk

		tampil.		grafik. Untuk grafik kolom sistem akan menampilkan tabel yang berisi PKL mana yang dibawah rata-rata dan masuk dalam pelatihan.
Aliran Kejadian Alternatif :		Aksi Pemakai		Respon Sistem
	1	<i>Use case</i> dimulai ketika pimpinan login sebagai pimpinan.	2	Sistem menyiapkan halaman grafik omset garis untuk di tampilkan pertama kali.
	3	Pimpinan memilih PKL yang akan dilihat grafik nya, setelah itu pimpinan akan memilih tahun yang akan dilihat omset nya. Setelah itu pimpinan menekan tombol tampil.	4	Sistem akan memproses inputan dari user yang berupa PKL yang dipilih, serta tahun yang pilih. Sistem akan melakukan <i>SELECT</i> data pada database sesuai dengan inputan user. Setelah itu sistem akan menampilkan dalam bentuk grafik. Untuk grafik kolom sistem akan menampilkan tabel yang berisi PKL mana yang dibawah rata-rata dan masuk dalam pelatihan. Jika terjadi kesalahan dalam <i>SELECT</i> data, maka PKL yang di pilih tidak akan tampil. Atau jika PKL yang dipilih tidak tampil maka PKL tersebut belum di <i>inputkan</i> omset nya.

4.3 Desain Uji Coba

Desain uji coba bertujuan untuk memastikan bahwa aplikasi telah dibuat dengan benar sesuai dengan kebutuhan atau tujuan yang diharapkan. Kekurangan atau kelemahan sistem pada tahap ini akan dievaluasi sebelum diimplementasikan secara nyata. Uji coba yang akan dilakukan antara lain:

1. Uji coba fungsi aplikasi
2. Uji coba aplikasi kepada pengguna

3.3.1 Uji Coba Fungsi Aplikasi

Pengujian ini penulis membagi uji coba (*testing*) dalam dua bagian yaitu *white box testing* dan *black box testing*. Untuk pengujian *white box testing* sudah dilakukan pada pembuatan program dengan model pengembangan TDD. Pada *white box testing* difokuskan pada pengetesan unit-unit terkecil dari bagian kode berupa fungsi dengan menggunakan *unit testing* SimpleTest.

Untuk *black box testing* difokuskan pada alur aplikasi apakah telah berjalan dengan baik atau tidak. Adapun desain uji yang akan di ujikan adalah:

1. Desain Uji Coba White Box

A. Desain Uji Coba TDD pada Fungsi Data Master PKL

Pada desain uji coba ini, bertujuan untuk melakukan unit testing pada fungsi di master PKL. Adapun unit testing yang di tes di tunjukan pada tabel 3.22 antara lain:

Tabel 3.22 Desain Uji Coba Fungsi Data Master PKL

No	Tes	Status
1	test_insert_jika_id_pkl_kosong()	
2	test_insert_jika_jenis_pkl_kosong()	
3	test_insert_jika_nama_pkl_kosong()	
4	test_insert_jika_lat_kosong()	
5	test_insert_jika_lng_kosong()	
6	test_insert_sebuah_record_ke_tabel_sukses()	
7	test_update_jika_id_pkl_kosong()	
8	test_update_jika_jenis_pkl_kosong()	
9	test_update_jika_nama_pkl_kosong()	
10	test_update_jika_lat_kosong()	
11	test_update_jika_lng_kosong()	
12	test_update_sebuah_record_ke_tabel_sukses()	
13	test_delete_sebuah_record_ke_tabel_sukses()	

B. Desain Uji Coba TDD pada Fungsi Data Master Pedagang

Pada desain uji coba ini, bertujuan untuk melakukan unit testing pada fungsi di master pedagang. Adapun unit testing yang di tes di tunjukan pada tabel 3.23 antara lain:

Tabel 3.23 Desain Uji Coba Fungsi Data Master pedagang

No	Tes	Status
1	test_insert_jika_id_pedagang_kosong()	
2	test_insert_jika_id_pkl_kosong()	
3	test_insert_jika_nama_pedagang_kosong()	
4	test_insert_jika_jenis_usaha_kosong()	
5	test_insert_jika_alamat_kosong()	
6	test_insert_sebuah_record_ke_tabel_sukses()	
7	test_update_jika_id_pedagang_kosong()	
8	test_update_jika_id_pkl_kosong()	
9	test_update_jika_nama_pedagang_kosong()	
10	test_update_jika_jenis_usaha_kosong()	
11	test_update_jika_alamat_kosong()	
12	test_update_sebuah_record_ke_tabel_sukses()	
13	test_delete_sebuah_record_ke_tabel_sukses()	

C. Desain Uji Coba TDD pada Fungsi Data Master Menu Makanan

Pada desain uji coba ini, bertujuan untuk melakukan unit testing pada fungsi di master menu makanan. Adapun unit testing yang di tes di tunjukan pada tabel 3.24 antara lain:

Tabel 3.24 Desain Uji Coba Fungsi Data Master Menu Makanan

No	Tes	Status
1	test_insert_jika_id_menu_makanan_kosong()	
2	test_insert_jika_id_pedagang_kosong()	
3	test_insert_jika_nama_makanan_kosong()	
4	test_insert_sebuah_record_ke_tabel_sukses()	
5	test_update_jika_nama_makanan_kosong()	
6	test_update_sebuah_record_ke_tabel_sukses()	
7	test_delete_sebuah_record_ke_tabel_sukses()	

D. Desain Uji Coba TDD pada Fungsi Data Master Menu Minuman

Pada desain uji coba ini, bertujuan untuk melakukan unit testing pada fungsi di master menu minuman. Adapun unit testing yang di tes di tunjukan pada tabel 3.25 antara lain:

Tabel 3.25 Desain Uji Coba Fungsi Data Master Menu Minuman

No	Tes	Status
1	test_insert_jika_id_menu_minuman_kosong()	
2	test_insert_jika_id_pedagang_kosong()	
3	test_insert_jika_nama_minuman_kosong()	
4	test_insert_sebuah_record_ke_tabel_sukses()	
5	test_update_jika_nama_minuman_kosong()	
6	test_update_sebuah_record_ke_tabel_sukses()	
7	test_delete_sebuah_record_ke_tabel_sukses()	

E. Desain Uji Coba TDD pada Fungsi Data Master User

Pada desain uji coba ini, bertujuan untuk melakukan unit testing pada fungsi di master user. Adapun unit testing yang di tes di tunjukan pada tabel 3.26 antara lain:

Tabel 3.26 Desain Uji Coba Fungsi Data Master User

No	Tes	Status
1	test_insert_jika_id_user_kosong()	
2	test_insert_jika_user_kosong()	
3	test_insert_jika_password_kosong()	
4	test_insert_jika_id_pkl_kosong()	
5	test_insert_jika_akses_kosong()	
6	test_insert_sebuah_record_ke_tabel_sukses()	
7	test_update_jika_id_user_kosong()	
8	test_update_jika_user_kosong()	
9	test_update_jika_password_kosong()	
10	test_update_jika_id_pkl_kosong()	
11	test_update_jika_akses_kosong()	
12	test_update_sebuah_record_ke_tabel_sukses()	
13	test_delete_sebuah_record_ke_tabel_sukses()	

F. Desain Uji Coba TDD pada Fungsi Data Master Omset

Pada desain uji coba ini, bertujuan untuk melakukan unit testing pada fungsi di master omset. Adapun unit testing yang di tes di tunjukan pada tabel 3.27 antara lain:

Tabel 3.27 Desain Uji Coba Fungsi Data Master Omset

No	Tes	Status
1	test insert jika id omset kosong()	
2	test insert jika id pkl kosong()	
3	test insert jika id pedagang kosong()	
4	test insert jika tanggal kosong()	
5	test insert jika omset kosong()	
6	test insert sebuah record ke tabel sukses()	
7	test update jika id omset kosong()	
8	test update jika id pkl kosong()	
9	test update jika id pedagang kosong()	
10	test update jika tanggal kosong()	
11	test update jika omset kosong()	
12	test update sebuah record ke tabel sukses()	
13	test delete sebuah record ke tabel sukses()	

2. Desain Uji Coba Black Box

A. Desain Uji Coba Login

Desain uji coba *login* bertujuan untuk menguji apakah fungsi melakukan *login* dapat berjalan sesuai yang diharapkan. Desain uji coba *login* dapat dilihat pada tabel 3.28.

Tabel 3.28 Desain Uji Coba Login

Test Case Id	Tujuan	Input	Output yang di harapkan	status
1	Melakukan login	<i>Username, password</i>	Login Berhasil dan masuk ke halaman utama sistem	
2	Melakukan login (gagal)	<i>Username, password</i>	Muncul pesan “User dan Password anda salah”	

B. Desain Uji Coba Fungsi Tambah Data

Pada desain uji coba ini, bertujuan untuk mengetahui apakah fungsi tambah data dapat menambah data baru kedalam *database*. Desain uji coba dapat dilihat pada tabel 3.29.

Tabel 3.29 Desain Uji Coba Fungsi Tambah Data

Test Case Id	Tujuan	Input	Output yang di harapkan	Status
3	Menambah data PKL (berhasil)	Memasukkan nama PKL = "Jemursari", jenis PKL = "tersentra", memasukkan latitude dan longtitude dengan mengklik tombol "Map". Setelah itu menekan tombol "Simpan"	Muncul pesan "Data Berhasil masuk", kemudian data bertambah di tabel sebelah kanan.	
4	Menambah data PKL (gagal)	Memasukkan nama PKL dengan nama yang sama dengan PKL yang lain, jenis PKL = "tersentra", memasukkan latitude dan longtitude dengan mengklik tombol "Map". Setelah itu menekan tombol "Simpan"	Tidak bisa di klik simpan, karena <i>textbox</i> Nama PKL akan menjadi Merah	
5	Menambah data Pedagang (berhasil)	Memilih nama PKL = "Jemursari", lalu memasukkan nama pedagang = "Karwo", lalu memilih jenis usaha "makanan", kemudian mengisi alamat = "jalan manyar no 31", lalu tekan tombol "Simpan"	Muncul pesan "Data Berhasil masuk", kemudian data bertambah di tabel sebelah kanan	
6	Menambah data Pedagang (gagal)	Memilih nama PKL = "Jemursari", lalu memasukkan nama pedagang = "Karwo", lalu memilih jenis usaha "makanan", kemudian mengisi alamat = "(kosong)", lalu tekan tombol "Simpan"	Tidak bisa di klik simpan, karena <i>textbox</i> alamat akan menjadi Merah	
7	Menambah data menu	Memilih nama PKL="Jemursari", lalu	Muncul pesan "Data Berhasil"	

Tabel 3.29 Desain Uji Coba Fungsi Tambah Data

Test Case Id	Tujuan	Input	Output yang di harapkan	Status
	makanan (berhasil)	memilih nama pedagang="Karwo", kemudian mengisikan nama makanan="Bakso", lalu tekan tombol "Simpan"	masuk", kemudian data bertambah di tabel sebelah kanan	
8	Menambah data menu makanan (gagal)	Memilih nama PKL="Jemursari", lalu memilih nama pedagang="Karwo", kemudian mengisikan nama makanan="(kosong)", lalu tekan tombol "Simpan"	Tidak bisa di klik simpan, karena <i>textbox</i> alamat akan menjadi Merah	
9	Menambah data menu minuman (berhasil)	Memilih nama PKL="Jemursari", lalu memilih nama pedagang="Jarwo", kemudian mengisikan nama minuman="Es Oyen", lalu tekan tombol "Simpan"	Muncul pesan "Data Berhasil masuk", kemudian data bertambah di tabel sebelah kanan	
10	Menambah data menu minuman (gagal)	Memilih nama PKL="Jemursari", lalu memilih nama pedagang="Jarwo", kemudian mengisikan nama minuman="(kosong)", lalu tekan tombol "Simpan"	Tidak bisa di klik simpan, karena <i>textbox</i> alamat akan menjadi Merah	
11	Menambah data user (berhasil)	Mengisikan username = "jemursari", lalu mengisi password = "pkl", kemudian memilih penempatan PKL = "jemursari", lalu memilih hak akses="Petugas PKL", kemudian tekan tombol "Simpan"	Muncul pesan "Data Berhasil masuk", kemudian data bertambah di tabel sebelah kanan	
12	Menambah data user (gagal)	Mengisikan username = "jemursari", lalu mengisi password = "(kosong)", kemudian memilih penempatan PKL = "jemursari", lalu memilih hak akses="Petugas PKL",	Tidak bisa di klik simpan, karena <i>textbox</i> password akan menjadi Merah	

Tabel 3.29 Desain Uji Coba Fungsi Tambah Data

Test Case Id	Tujuan	Input	Output yang di harapkan	Status
		kemudian tekan tombol “Simpan”		
13	Menambah data omset (berhasil)	Memilih nama pedagang = “Karwo”, kemudian memilih tanggal=”12 Februari 2013”, lalu mengisi jumlah omset=”2300000”, kemudian tekan tombol “Simpan”	Muncul pesan “Data Berhasil masuk” , kemudian data bertambah di tabel sebelah kanan	
14	Menambah data omset (gagal)	Memilih nama pedagang = “Karwo”, kemudian memilih tanggal=”13 Maret 2013”, lalu mengisi jumlah omset=”(kosong)”, kemudian tekan tombol “Simpan”	Tidak bisa di klik simpan, karena <i>textbox</i> Jumlah Omset akan menjadi Merah, serta Tanggal hanya bisa di inputkan bulan sekarang atau bulan sebelumnya.	

C. Desain Uji Coba Fungsi Ubah Data

Pada desain uji coba ini, bertujuan untuk mengetahui apakah fungsi ubah data dapat mengubah data pada database. Desain uji coba dapat dilihat tabel 3.30.

Tabel 3.30 Desain Uji Coba Fungsi Ubah Data

Test Case ID	Tujuan	Input	Output yang diharapkan	Status
15	Mengubah data PKL (berhasil)	Memasukkan nama PKL=”Jemursari”, memasukkan latitude longitude baru dengan menekan tombol “Map”, kemudian menekan tombol “Update”	Muncul Pesan “Data Berhasil ubah”	
16	Mengubah data PKL (gagal)	Memasukkan nama PKL yang sama dengan PKL lain, memasukkan latitude longitude	Tidak bisa di klik “Update”,	

Tabel 3.30 Desain Uji Coba Fungsi Ubah Data

Test Case ID	Tujuan	Input	Output yang diharapkan	Status
		baru dengan menekan tombol “Map”, kemudian menekan tombol “Update”	karena <i>textbox</i> Nama PKL akan menjadi Merah	
17	Mengubah data pedagang (berhasil)	Memasukkan nama pedagang = “Karwo”, lalu memilih jenis Usaha=”Makanan”, lalu memasukkan alamat=”nginden”, kemudian menekan tombol “Update”	Muncul Pesan “Data Berhasil ubah”	
18	Mengubah data pedagang (gagal)	memasukkan nama pedagang = “Karwo”, lalu memilih jenis usaha “makanan”, kemudian mengisi alamat = “(kosong)”, lalu tekan tombol “Simpan”	Tidak bisa di klik “Update”, karena <i>textbox</i> Alamat akan menjadi Merah	
19	Mengubah data menu makanan (berhasil)	Memasukkan nama makanan = “Sego Goreng Jawa” lalu tekan tombol “Update”	Muncul Pesan “Data Berhasil ubah”	
20	Mengubah data menu makanan (gagal)	Memasukkan nama makanan = “(kosong)” lalu tekan tombol “Update”	Tidak bisa di klik “Update”, karena <i>textbox</i> nama makanan akan menjadi Merah	
21	Mengubah data menu minuman (berhasil)	Memasukkan nama minuman = “Es Degan” lalu tekan tombol “Update”	Muncul Pesan “Data Berhasil ubah”	
22	Mengubah data menu minuman (gagal)	Memasukkan nama minuman = “(kosong)” lalu tekan tombol “Update”	Tidak bisa di klik “Update”, karena <i>textbox</i> nama minuman akan menjadi Merah	
23	Mengubah	Mengisikan username =	Muncul	

Tabel 3.30 Desain Uji Coba Fungsi Ubah Data

Test Case ID	Tujuan	Input	Output yang diharapkan	Status
	data user (berhasil)	“jemursari”, lalu mengisi password = “pkl”, kemudian memilih penempatan PKL = “jemursari”, lalu memilih hak akses=“Petugas PKL”, kemudian tekan tombol “Update”	Pesan “Data Berhasil ubah”	
24	Mengubah data user (gagal)	Mengisikan <i>username</i> dengan <i>username</i> yang telah ada, lalu mengisi <i>password</i> = “(kosong)”, kemudian memilih penempatan PKL = “jemursari”, lalu memilih hak akses=“Petugas PKL”, kemudian tekan tombol “Update”	Tidak bisa di klik “Update”, karena <i>textbox username</i> dan <i>password</i> akan menjadi Merah	
25	Mengubah data omset (berhasil)	Memilih nama pedagang = “Karwo”, kemudian memilih tanggal=“12 Februari 2013”, lalu mengisi jumlah omset=“2300000”, kemudian tekan tombol “Update”	Muncul Pesan “Data Berhasil ubah”	
26	Mengubah data omset (gagal)	Memilih nama pedagang = “Karwo”, kemudian memilih tanggal=“13 Maret 2013”, lalu mengisi jumlah omset=“(kosong)”, kemudian tekan tombol “Update”	Tidak bisa di klik “Update”, karena <i>textbox Jumlah Omset</i> akan menjadi Merah, serta Tanggal hanya bisa di inputkan bulan sekarang atau bulan sebelumnya.	

D. Desain Uji Coba Fungsi Hapus Data

Pada desain uji coba ini, bertujuan untuk mengetahui apakah fungsi hapus data dapat menghapus data pada database. Desain uji coba dapat dilihat pada tabel 3.31.

Tabel 3.31 Desain Uji Coba Fungsi Hapus Data

Test Case id	Tujuan	Input	Output	Status
27	Menghapus data PKL (berhasi)	Memilih data PKL pada data tabel, lalu menekan tombol silang merah atau hapus dan mengkonfirmasi pesan hapus	Akan menampilkan pesan “Data berhasil Dihapus”, dan pada data tabel tidak akan menampilkan data yang terhapus.	
28	Menghapus data PKL (gagal)	Memilih data PKL pada data tabel, lalu menekan tombol silang merah atau hapus dan mengkonfirmasi pesan hapus	Muncul peringatan “Data Tidak Berhasil Dihapus”, pada data tabel masih menampilkan data terhapus.	
29	Menghapus data pedagang (berhasi)	Memilih data pedagang pada data tabel, lalu menekan tombol silang merah atau hapus dan mengkonfirmasi pesan hapus	Akan menampilkan pesan “Data berhasil Dihapus”, dan pada data tabel tidak akan menampilkan data yang terhapus.	
30	Menghapus data pedagang (gagal)	Memilih data pedagang pada data tabel, lalu menekan tombol silang merah atau hapus dan mengkonfirmasi pesan hapus	Muncul peringatan “Data Tidak Berhasil Dihapus”,	

Tabel 3.31 Desain Uji Coba Fungsi Hapus Data

Test Case id	Tujuan	Input	Output	Status
			pada data tabel masih menampilkan data terhapus.	
31	Menghapus data menu makanan (berhasil)	Memilih data menu makanan pada data tabel, lalu menekan tombol silang merah atau hapus dan mengkonfirmasi pesan hapus	Akan menampilkan pesan “Data berhasil Dihapus” , dan pada data tabel tidak akan menampilkan data yang terhapus.	
32	Menghapus data menu makanan (berhasil)	Memilih data menu makanan pada data tabel, lalu menekan tombol silang merah atau hapus dan mengkonfirmasi pesan hapus	Muncul peringatan “Data Tidak Berhasil Dihapus” , pada data tabel masih menampilkan data terhapus.	
33	Menghapus data menu minuman (berhasil)	Memilih data menu minuman pada data tabel, lalu menekan tombol silang merah atau hapus dan mengkonfirmasi pesan hapus	Akan menampilkan pesan “Data berhasil Dihapus” , dan pada data tabel tidak akan menampilkan data yang terhapus.	
34	Menghapus data menu minuman (gagal)	Memilih data menu minuman pada data tabel, lalu menekan tombol silang merah atau hapus dan mengkonfirmasi pesan hapus	Muncul peringatan “Data Tidak Berhasil Dihapus” , pada data tabel masih menampilkan data terhapus.	
35	Menghapus	Memilih data user pada data	Akan	

Tabel 3.31 Desain Uji Coba Fungsi Hapus Data

Test Case id	Tujuan	Input	Output	Status
	data user (berhasil)	tabel, lalu menekan tombol silang merah atau hapus dan mengkonfirmasi pesan hapus	menampilkan pesan “Data berhasil Dihapus”, dan pada data tabel tidak akan menampilkan data yang terhapus.	
36	Menghapus data user (gagal)	Memilih data user pada data tabel, lalu menekan tombol silang merah atau hapus dan mengkonfirmasi pesan hapus	Muncul peringatan “Data Tidak Berhasil Dihapus”, pada data tabel masih menampilkan data terhapus.	
37	Menghapus data omset (berhasil)	Memilih data omset pada data tabel, lalu menekan tombol silang merah atau hapus dan mengkonfirmasi pesan hapus	Akan menampilkan pesan “Data berhasil Dihapus”, dan pada data tabel tidak akan menampilkan data yang terhapus.	
38	Menghapus data omset (gagal)	Memilih data omset pada data tabel, lalu menekan tombol silang merah atau hapus dan mengkonfirmasi pesan hapus	Muncul peringatan “Data Tidak Berhasil Dihapus”, pada data tabel masih menampilkan data terhapus.	

E. Desain Uji Coba Fungsi View Peta PKL

Pada desain uji coba ini, bertujuan untuk mengetahui apakah fungsi view peta PKL sesuai yang diharapkan. Desain uji coba dapat dilihat pada tabel 3.32.

Tabel 3.32 Desain Uji Coba Fungsi View Peta PKL

Test Case Id	Tujuan	Input	Output	Status
39	Melihat peta PKL tersentra (berhasil)	Memilih Menu PKL tersentra	Peta memunculkan <i>marker</i> PKL tersentra	
41	Melihat peta PKL terbina (berhasil)	Memilih Menu PKL terbina	Peta memunculkan <i>marker</i> PKL terbina	
43	Melihat radius (berhasil)	Menentukan diameter dan posisi radius	Marker PKL yang tercakup di radius muncul.	
45	Melihat omset (berhasil)	Menentukan range omset	Memunculkan marker PKL yang sesuai dengan range omset	

3.3.2 Uji Coba Aplikasi Kepada Pengguna

Prosedur untuk melakukan pengujian aplikasi kepada pengguna adalah sebagai berikut:

1. Uji coba kepada Dinas Koperasi Surabaya

Uji coba ini dilakukan kepada pegawai dinas koperasi, pimpinan dinas koperasi, dan petugas PKL. Dimana hasil uji coba ini menjadi ukuran dalam menjawab tujuan aplikasi.

2. Uji coba kepada Masyarakat

Uji coba ini dilakukan kepada 30 orang masyarakat dari beberapa daerah di Surabaya. Dimana hasil dari uji coba ini digunakan untuk revisi dan evaluasi aplikasi.

3. Instrumen Uji Coba

Instrumen uji coba yang digunakan dalam pembuatan aplikasi adalah sebagai berikut:

a. Kuesioner

Kuesioner diberikan kepada pengguna untuk mendapatkan informasi mengenai cara penggunaan aplikasi dan alur program sehingga dapat memperoleh tujuan dari pembuatan aplikasi ini. Adapun desain kuesioner uji coba dapat dilihat pada lampiran.

b. Teknik Analisis Data

Untuk menganalisis apakah aplikasi tersebut layak atau tidak dari serangkaian uji coba tersebut diatas. Maka akan digunakan analisis rasional dan ditentukan sendiri oleh individu yang merupakan justifikasi yang bersifat subjektif.

Pada proses uji coba ini, pengguna diharuskan mencoba aplikasi ini sehingga menghasilkan total nilai akhir. Uji coba ini dilakukan dengan cara melakukan survey langsung kepada 30 orang pengguna yaitu masyarakat dengan cara mengisi kuesioner. Desain tabel karakteristik pengguna dapat dilihat pada tabel 3.33.

Tabel 3.33 Desain Karakteristik Pengguna

No	Nama Pengguna	Umur (tahun)	Pekerjaan	Total Nilai Akhir

Kuesioner yang telah diisi akan dimasukkan kedalam rekapitulasin kuesioner, dimana desain tabel kuesioner yang telah diisi dapat dilihat pada tabel 3.34, yang berfungsi untuk mengetahui hasil keseluruhan dari pengguna terhadap aplikasi ini.

Tabel 3.34 Desain Rekapitulasi Kuesioner

Pertanyaan No.	Penilaian					Σ	Nilai Akhir
	1	2	3	4	5		
Tampilan							
A	1						
	2						
	3						
Navigasi							
B	1						
	2						
Manfaat							
C	1						
	2						

