**PEMBUATAN MODUL PETA DAN BELANJA PADA APLIKASI RUMAH WARGA DI TOMATECH MOBILE DEV**

**NUGROHO TRI PAMBUDI**



**PROGRAM KEAHLIAN MANAJEMEN INFORMATIKA**

**PROGRAM DIPLOMA**

**INSTITUT PERTANIAN BOGOR**

**BOGOR**

**2016**

**PERNYATAAN MENGENAI LAPORAN AKHIR DAN SUMBER INFORMASI SERTA PELIMPAHAN HAK CIPTA**

Dengan ini saya menyatakan laporan akhir Pembuatan Modul Peta dan Belanja pada Aplikasi Rumah Warga di Tomatech Mobile Dev adalah benar karya saya dengan arahan komisi pembimbing dan belum diajukan dalam bentuk apa pun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam deks dan dicantumkan dalam Daftar Pustaka di bagian akhir laporan ini.

Dengan ini saya melimpahkan hak cipta dari karya tulis saya kepada Institut Pertanian Bogor.

Bogor, Juni 2016

Nugroho Tri Pambudi

NIM J3C21312

# ABSTRACT

NUGROHO TRI PAMBUDI. Development map and shop module in Rumah Warga applications at Tomatech Mobile Dev. Supervised by MUHAMMAD SYAMSUN.

Tomatech Mobile Dev is an information technology company that specializes in web and mobile applications. Tomatech Mobile Dev have an idea to create an application called Rumah Warga. The application is created for be the solution of the existing problems in a residential which is divided into several clusters. Usually in a residential area, the communication among citizens rarely happens because of the activities of citizens is busy. It even made a citizen does not know the people who live next to their home. Rumah Warga applications are created using laravel framework with the programming language PHP and uses PostgreSQL as a Database Management System (DBMS). Rumah Warga applications using prototype methodology which have five stages. Five stages consist of communication, quick plan, modeling quick design, construction of prototype, and deployment delivery and feedback.

Keyword: Clusters, Prototype methodology, Rumah Warga applications, Tomatech Mobile Dev, *Web*

# 

# RINGKASAN

NUGROHO TRI PAMBUDI. Pembuatan Modul Peta dan belanja pada Aplikasi Rumah Warga di Tomatech Mobile Dev. Dibimbing oleh MUHAMMAD SYAMSUN.

Tomatech Mobile Dev merupakan perusahaan yang bergerak dibidang teknologi informasi yang mengkhususkan diri dalam aplikasi *web* dan juga *mobile*. Tomatech Mobile Dev menawarkan ide-ide kreatif untuk memberikan solusi dari suatu permasalahan.

Pada lingkungan perumahan perumahan yang terbagi menjadi beberapa klaster, komunikasi antar warga merupakan aktivitas yang jarang terjadi. Hal tersebut disebabkan karena padatnya kegiatan setiap warga. Hal ini dapat membuat seorang warga tidak mengenal dan mengetahui profil dari tetangganya sendiri. Berdasarkan permasalahan tersebut, Tomatech Mobile Dev menciptakan gagasan untuk membuat sebuah aplikasi bernama Rumah Warga.

Di dalam aplikasi Rumah Warga terdapat modul peta yang menampilkan peta klaster dan data warga yang dikelola oleh Penanggung Jawab (PJ) lokasi yang diberikan wewenang langsung oleh Tomatech Mobile Dev. Selain itu terdapat pula modul belanja yang menyediakan fasilitas jual beli sederhana dalam klaster.

Modul peta dan belanja pada aplikasi Rumah Warga dibuat menggunkan metode prototipe. Metode prototipe memiliki lima tahapan, yaitu komunikasi, perencanaan cepat, pemodelan perancangan cepat, pembentukan prototipe, dan penyerahan sistem ke pengguna & umpan balik. Aplikasi Rumah Warga dibuat menggunakan *framework* Laravel dengan bahasa pemrograman PHP dan menggunakan PostgreSQL sebagai *Database Management System* (DBMS).

Kata kunci: Aplikasi Rumah Warga, Klaster, Metode prototipe, Tomatech Mobile Dev, *Web*

**PEMBUATAN MODUL PETA DAN BELANJA PADA APLIKASI RUMAH WARGA DI TOMATECH MOBILE DEV**

**NUGROHO TRI PAMBUDI**

Laporan Akhir

sebagai salah satu syarat untuk memperoleh gelar

Ahli Madya

pada

Program Keahlian Manajemen Informatika

**PROGRAM KEAHLIAN MANAJEMEN INFORMATIKA**

**PROGRAM DIPLOMA**

**INSTITUT PERTANIAN BOGOR**

**BOGOR**

**2016**

|  |  |  |
| --- | --- | --- |
| Judul Laporan Akhir | : | Pembuatan Modul Peta dan Belanja pada Aplikasi Rumah Warga di Tomatech Mobile Dev |
| Nama | : | Nugroho Tri Pambudi |
| NIM | : | J3C213126 |

Disetujui oleh

Dr Ir Muhammad Syamsyun, MSc

Pembimbing

Diketahui oleh

Dr Ir Bagus P. Purwanto, MAgr Dr Ir Muhammad Syamsyun, MSc

Direktur Koordinator Program Keahlian

Tanggal lulus:

**PRAKATA**

Puji dan syukur kehadirat Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya serta nikmat sehat dan segala kemudahan yang telah diberikan sehigga tugas akhir ini dapat terselesaikan. Tugas akhir ini merupakan hasil dari kegiatan Praktik Kerja Lapangan (PKL) di Tomatech Mobile Dev dengan bahan kajian Pembuatan Modul Peta dan Belanja pada Aplikasi Rumah Warga di Tomatech Mobile Dev. Kegiatan PKL dilaksanakan mulai tanggal 1 Februari sampai 9 April 2016.

Terima kasih kepada Bapak Dr. Ir. Muhammad Syamsun, M.Sc selaku dosen pembimbing, Syahril Hermana selaku pembimbing lapangan dan Bapak Sandy Wibisono dari pihak Tomatech Mobile Dev atas semua bantuan dan bimbingannya juga kepada keluarga untuk semua doa dan dukungannya. Serta terima kasih kepada teman-teman dari Manajemen Informatika angkatan 50 Program Diploma Institut Pertanian Bogor dan pihak yang telah berkontribusi dan mendukung penulis dalam pengerjaan tugas akhir ini.

Penulis menyadari bahwa penyusunan tugas akhir ini masih jauh dari kesempurnaan, oleh karena itu penulis mengharapkan kritik dan saran yang bersifat membangun dari semua pihak. Semoga tugas akhir ini bermanfaat bagi penulis serta pembaca.

Bogor, Juni 2016

Nugroho Tri Pambudi

# DAFTAR ISI

[DAFTAR ISI xiii](#_Toc454545734)

[DAFTAR TABEL xv](#_Toc454545735)

[DAFTAR GAMBAR xv](#_Toc454545736)

[DAFTAR LAMPIRAN xvi](#_Toc454545737)

[1 PENDAHULUAN 1](#_Toc454545738)

[1.1 Latar Belakang 1](#_Toc454545739)

[1.2 Tujuan 1](#_Toc454545740)

[1.3 Ruang Lingkup 1](#_Toc454545741)

[2 METODE KAJIAN 2](#_Toc454545742)

[2. 1 Tempat dan Waktu PKL 2](#_Toc454545743)

[2. 2 Metode Bidang Kajian 2](#_Toc454545744)

[2.2.1 Komunikasi 2](#_Toc454545745)

[2.2.2 Perencanaan Cepat 3](#_Toc454545746)

[2.2.3 Pemodelan Perancangan Cepat 3](#_Toc454545747)

[2.2.4 Pembentukan Prototipe 3](#_Toc454545748)

[2.2.5 Penyerahan Sistem ke Pengguna & Umpan Balik 3](#_Toc454545749)

[2. 3 Tinjauan Pustaka 3](#_Toc454545750)

[2.3.1 *Unified Modeling System* 3](#_Toc454545751)

[2.3.2 *Database* 5](#_Toc454545752)

[2.3.3 *Database Management System* 5](#_Toc454545753)

[2.3.4 PostgreSQL 5](#_Toc454545754)

[2.3.5 *Hyper Text Markup Language* 5](#_Toc454545755)

[2.3.6 *Cascading Style Sheet* 6](#_Toc454545756)

[2.3.7 PHP *Hypertext Preprocessor* 6](#_Toc454545757)

[2.3.8 *Framework* 6](#_Toc454545758)

[2.3.9 Laravel 6](#_Toc454545759)

[2.3.10 Bootstrap 6](#_Toc454545760)

[3 KEADAAN UMUM TOMATECH MOBILE DEV 6](#_Toc454545761)

[3.1 Sejarah 6](#_Toc454545762)

[3.2 Kegiatan 7](#_Toc454545763)

[3.3 Struktur Organisasi 7](#_Toc454545764)

[3.4 Visi dan Misi 8](#_Toc454545765)

[4 PEMBUATAN MODUL PETA DAN BELANJA PADA APLIKASI RUMAH WARGA DI TOMATECH MOBILE DEV 8](#_Toc454545766)

[4.1 Prototipe Pertama 8](#_Toc454545767)

[4.1.1 Komunikasi 8](#_Toc454545768)

[4.1.2 Perencanaan Cepat 8](#_Toc454545769)

[4.1.3 Pemodelan Perancangan Cepat 10](#_Toc454545770)

[4.1.4 Pembentukan Prototipe 19](#_Toc454545771)

[4.1.5 Penyerahan Sistem ke Pengguna & Umpan Balik 22](#_Toc454545772)

[4.2 Prototipe Kedua 23](#_Toc454545773)

[4.2.1 Komunikasi 23](#_Toc454545774)

[4.2.2 Perencanaan Cepat 23](#_Toc454545775)

[4.2.3 Pemodelan Perancangan Cepat 24](#_Toc454545776)

[4.2.4 Pembentukan Prototipe 25](#_Toc454545777)

[4.2.5 Penyerahan Sistem ke Pengguna & Umpan Balik 26](#_Toc454545778)

[4.3 Evaluasi Seminar 26](#_Toc454545779)

[5 SIMPULAN 27](#_Toc454545780)

[5.1 Simpulan 27](#_Toc454545781)

[5.2 Saran 27](#_Toc454545782)

[DAFTAR PUSTAKA 27](#_Toc454545783)

[LAMPIRAN 29](#_Toc454545784)

# DAFTAR TABEL

1. [Simbol-simbol *use case diagram* (Rosa dan Shalahuddin 2013) 3](#_Toc454258154)
2. [Simbol-simbol *activity diagram* (Rosa dan Shalahuddin 2013) 4](#_Toc454258155)
3. [Simbol-simbol *class diagram* (Rosa dan Shalahuddin 2013) 5](#_Toc454258156)
4. [Deskripsi Kebutuhan fungsional prototipe pertama 9](#_Toc454258157)
5. [*Activity diagram* prototipe pertama 10](#_Toc454258158)
6. [Deskripsi tabel negara 13](#_Toc454258159)
7. [Deskripsi tabel provinsi 13](#_Toc454258160)
8. [Deskripsi tabel kota 13](#_Toc454258161)
9. [Deskripsi tabel kecamatan 13](#_Toc454258162)
10. [Deskripsi tabel kelurahan 14](#_Toc454258163)
11. [Deskripsi tabel klaster 14](#_Toc454258164)
12. [Deskripsi tabel peta 14](#_Toc454258165)
13. [Deskripsi tabel detail\_peta 15](#_Toc454258166)
14. [Deskripsi tabel users 15](#_Toc454258167)
15. [Deskripsi tabel transaksi 16](#_Toc454258168)
16. [Deskripsi tabel detail\_transaksi 16](#_Toc454258169)
17. [Deskripsi tabel detail\_barang 16](#_Toc454258170)
18. [Deskripsi tabel barang 17](#_Toc454258171)
19. [Perancangan masukan prototipe pertama 18](#_Toc454258172)
20. [Perancangan keluaran prototipe pertama 18](#_Toc454258173)
21. [Implementasi masukan prototipe pertama 20](#_Toc454258174)
22. [Implementasi keluaran prototipe pertama 21](#_Toc454258175)
23. [Pengujian *blackbox* prototipe pertama 22](#_Toc454258176)
24. [Deskripsi kebutuhan fungsional prototipe kedua 24](#_Toc454258177)
25. [*Activity diagram* prototipe kedua 24](#_Toc454258178)
26. [Daftar saran dalam forum 26](#_Toc454258179)

# DAFTAR GAMBAR

1. [Metode prototipe (Pressman 2012) 2](file:///D:\Documents\Tugas%20Akhir\TA%20COOOOOY(BUDI).docx#_Toc454258180)
2. [Struktur organisasi Tomatech Mobile Dev 7](file:///D:\Documents\Tugas%20Akhir\TA%20COOOOOY(BUDI).docx#_Toc454258181)
3. [*Use case diagram* prototipe pertama 11](file:///D:\Documents\Tugas%20Akhir\TA%20COOOOOY(BUDI).docx#_Toc454258182)
4. [*Class diagram* prototipe pertama 12](file:///D:\Documents\Tugas%20Akhir\TA%20COOOOOY(BUDI).docx#_Toc454258183)
5. [Implementasi *database* prototipe pertama 19](#_Toc454258184)
6. [Implementasi halaman *login* 20](file:///D:\Documents\Tugas%20Akhir\TA%20COOOOOY(BUDI).docx#_Toc454258185)
7. [Implementasi halaman utama 20](#_Toc454258186)
8. [Implementasi proses prototipe pertama 21](#_Toc454258187)
9. [*Use case diagram* prototipe kedua 24](file:///D:\Documents\Tugas%20Akhir\TA%20COOOOOY(BUDI).docx#_Toc454258188)
10. [Perancangan antarmuka halaman keranjang 25](#_Toc454258189)
11. [Kode program pada fungsi keranjang 25](#_Toc454258190)
12. [Implementasi halaman keranjang 26](#_Toc454258191)

# DAFTAR LAMPIRAN

1. [Alur proses *login* 31](#_Toc454258352)
2. [Alur proses menambah peta 31](#_Toc454258353)
3. [Alur proses melihat peta 32](#_Toc454258354)
4. [Alur proses mengubah peta 32](#_Toc454258355)
5. [Alur proses menghapus peta 33](#_Toc454258356)
6. [Alur proses menambah detail peta 33](#_Toc454258357)
7. [Alur proses melihat detail peta 34](#_Toc454258358)
8. [Alur proses mengubah detail peta 34](#_Toc454258359)
9. [Alur proses menghapus detail peta 35](#_Toc454258360)
10. [Alur proses menambah produk baru 35](#_Toc454258361)
11. [Alur proses menambah produk 36](#_Toc454258362)
12. [Alur proses melihat produk 36](#_Toc454258363)
13. [Alur proses memperbarui produk 37](#_Toc454258364)
14. [Alur proses menghapus produk 37](#_Toc454258365)
15. [Alur proses melihat katalog 38](#_Toc454258366)
16. [Alur proses membeli produk 38](#_Toc454258367)
17. [Alur proses konfirmasi pesanan 39](#_Toc454258368)
18. [Alur proses konfirmasi transaksi 39](#_Toc454258369)
19. [Alur proses membatalkan pesanan 40](#_Toc454258370)
20. [Alur proses membatalkan transaksi 40](#_Toc454258371)
21. [Perancangan masukan *login* 41](#_Toc454258372)
22. [Perancangan masukan konversi peta bagian pertama 41](#_Toc454258373)
23. [Perancangan masukan konversi peta bagian kedua 41](#_Toc454258374)
24. [Perancangan masukan ubah peta 42](#_Toc454258375)
25. [Perancangan masukan tambah detail peta 42](#_Toc454258376)
26. [Perancangan masukan ubah detail peta 42](#_Toc454258377)
27. [Perancangan masukan tambah produk baru 43](#_Toc454258378)
28. [Perancangan masukan tambah produk 43](#_Toc454258379)
29. [Perancangan masukan perbarui produk 43](#_Toc454258380)
30. [Perancangan masukan masukan kuantitas pesanan 44](#_Toc454258381)
31. [Perancangan keluaran informasi peta 44](#_Toc454258382)
32. [Perancangan keluaran informasi detail peta 44](#_Toc454258383)
33. [Perancangan keluaran informasi produk 45](#_Toc454258384)
34. [Perancangan keluaran informasi peta klaster 45](#_Toc454258385)
35. [Perancangan keluaran informasi detail produk 45](#_Toc454258386)
36. [Implementasi masukan *login* 46](#_Toc454258387)
37. [Implementasi masukan konversi peta bagian pertama 46](#_Toc454258388)
38. [Implementasi masukan konversi peta bagian kedua 46](#_Toc454258389)
39. [Implementasi masukan ubah peta 47](#_Toc454258390)
40. [Implementasi masukan tambah detail peta 47](#_Toc454258391)
41. [Implementasi masukan ubah detail peta 47](#_Toc454258392)
42. [Implementasi masukan tambah produk baru 48](#_Toc454258393)
43. [Implementasi masukan tambah produk 48](#_Toc454258394)
44. [Implementasi masukan perbarui produk 48](#_Toc454258395)
45. [Implementasi masukan masukan kuantitas pesanan 49](#_Toc454258396)
46. [Implementasi keluaran informasi peta 49](#_Toc454258397)
47. [Implementasi keluaran informasi detail peta 49](#_Toc454258398)
48. [Implementasi keluaran informasi produk 50](#_Toc454258399)
49. [Implementasi keluaran informasi peta klaster 50](#_Toc454258400)
50. [Implementasi keluaran informasi detail produk 50](#_Toc454258401)
51. [Alur proses menambah produk ke keranjang 51](#_Toc454258402)
52. [Alur proses malihat keranjang 51](#_Toc454258403)
53. [Alur proses menghapus produk di keranjang 51](#_Toc454258404)

# PENDAHULUAN

## Latar Belakang

Pada era modern seperti sekarang ini, masyarakat yang mengutamakan fasilitas dan keamanan memilih perumahan yang terbagi menjadi beberapa klaster untuk menjadi pilihan mereka dalam memenuhi kebutuhan papan. Pada lingkungan perumahan komunikasi antar warga merupakan aktivitas yang jarang terjadi. Hal tersebut disebabkan karena padatnya kegiatan setiap warga. Hal ini dapat membuat seorang warga tidak mengenal dan mengetahui profil dari tetangganya sendiri. Bahkan, untuk membeli kebutuhan sehari-hari akan sulit dilakukan karena sedikitnya waktu luang. Berdasarkan permasalahan tersebut, Tomatech Mobile Dev menciptakan gagasan untuk membuat sebuah aplikasi bernama Rumah Warga.

Tomatech Mobile Dev merupakan perusahaan yang bergerak dibidang teknologi informasi yang mengkhususkan diri dalam aplikasi *web* dan juga *mobile*. Tomatech Mobile Dev menawarkan ide-ide kreatif untuk memberikan solusi dari suatu permasalahan. Begitu pun dengan aplikasi Rumah Warga yang dibuat untuk menjadi solusi permasalahan yang ada di sebuah klaster.

Aplikasi Rumah Warga menyediakan fasilitas pengelolaan data warga yang berada pada sebuah klaster. Data sebuah klaster dikelola oleh Penanggung Jawab (PJ) lokasi yang diberikan wewenang langsung oleh Tomatech Mobile Dev. Apikasi Rumah Warga memiliki tujuh modul yaitu modul peta, modul belanja, modul kalender, modul pesan, modul tagihan, modul surat, dan modul nomor penting. Semua Pembahasan dalam kajian laporan ini dikhususkan pada modul peta dan modul belanja.

## Tujuan

Tujuan pembuatan modul peta dan modul belanja pada aplikasi Rumah Warga antara lain:

1. Menyediakan fungsi manipulasi data lokasi warga.
2. Menyediakan antarmuka katalog produk untuk pengguna.
3. Menyediakan fungsi keranjang belanja.
4. Menyediakan fungsi beli produk.

## Ruang Lingkup

Ruang lingkup dari modul peta dan modul belanja pada aplikasi Rumah Warga ini memiliki batasan pengembangan sistem yang dibuat yaitu:

1. PJ lokasi diberi wewenang langsung oleh Tomatech Mobile Dev.
2. Peta dikelola oleh admin sedangkan detail peta dikelola oleh PJ lokasi.
3. Penjual merupakan warga klaster dan didaftarkan oleh PJ lokasi.
4. Metode pembayaran terbatas hanya *Cash on Delivery* (CoD).

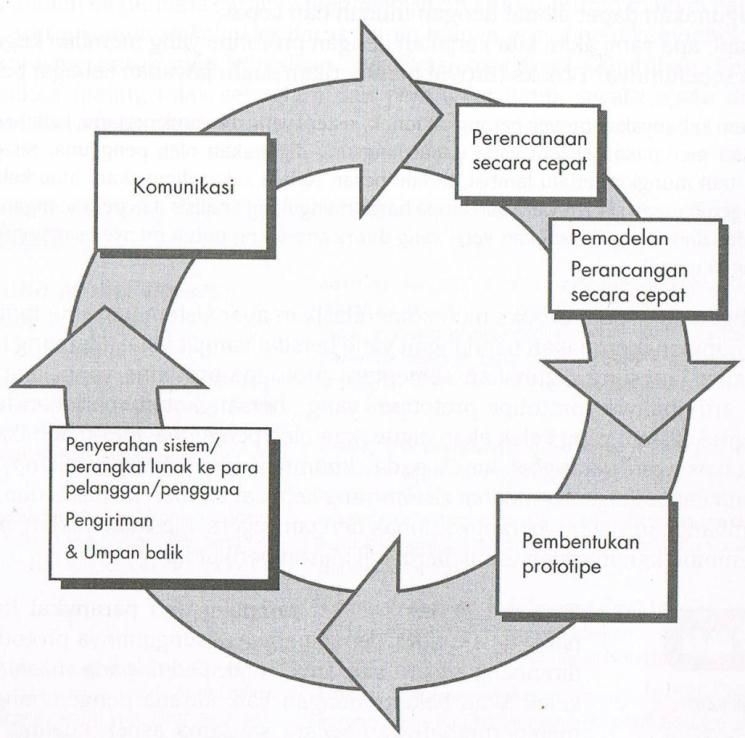
# METODE KAJIAN

## Tempat dan Waktu PKL

Kegiatan Praktik Kerja Lapangan (PKL) dilaksanakan di Tomatech Mobile Dev yang beralamat di Rumah Bali Lantai 1, Ruang Gianyar, Jl. Pandawa Raya No. 9-10 Bogor, 16153. Kegiatan dilakukan mulai hari Senin hingga Jum’at dalam satu minggu dari pukul 09.00 - 17.00 WIB. Setiap minggu pada hari Jum’at dilakukan presentasi kemajuankerja.

## Metode Bidang Kajian

Metode yang digunakan dalam pembuatan modul peta dan belanja pada aplikasi Rumah Warga adalah metode prototipe. Alasan penggunaan metode prototipe yaitu kebutuhan aplikasi terdefinisikan secara umum, namun tidak terperinci. Metode prototipe memiliki lima tahapan yaitu komunikasi, perencanaan secara cepat, pemodelan rancangan cepat, pembentukan prototipe, dan penyerahan sistem ke pengguna & umpan balik (Pressman 2012). Tahapan-tahapan dari metode prototipe dapat dilihat pada Gambar 1.



Gambar 1 Metode prototipe (Pressman 2012)

### Komunikasi

Tahap komunikasi dilakukan oleh pihak pengembang dengan klien untuk mendefinisikan tujuan aplikasi, mengidentifikasi kebutuhan aplikasi, dan menggambarkan kebutuhan untuk pengembangan aplikasi.

### Perencanaan Cepat

Perencanaan pada metode prototipe dilakukan secara cepat. Perencanaan dibuat berdasarkan kebutuhan-kebutuhan yang telah diidentifikasi pada tahap komunikasi.

### Pemodelan Perancangan Cepat

Pemodelan rancangan cepat berfokus pada aspek aplikasi yang akan terlihat oleh pengguna. Pemodelan perancangan ini dilakukan secara cepat.

### Pembentukan Prototipe

Tahap ini merupakan tahap konstruksi pembuatan prototipe aplikasi yang dilakukan oleh pengembang. Aplikasi dibuat berdasarkan perencanaan dan pemodelan yang telah dilakukan pada tahap sebelumnya.

### Penyerahan Sistem ke Pengguna & Umpan Balik

Prototipe yang telah dibuat diserahkan ke pengguna. Kemudian pengguna mengevaluasi dan memberikan umpan balik kepada pengembang mengenai prototipe aplikasi yang telah dibuat. Umpan balik akan digunakan sebagai tolak ukur untuk menlanjutkan iterasi jika prototipe belum memenuhi kebutuhan dari pengguna.

## Tinjauan Pustaka

### *Unified Modeling System*

Unified Modeling System (UML) adalah salah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement,* membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Rosa dan Shalahuddin 2013).

1. *Use Case Diagram*

*Use case diagram* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat (Rosa dan Shalahuddin 2013). Simbol-simbol diagram *use case* dapat dilihat pada Tabel 1.

Tabel 1 Simbol-simbol *use case diagram* (Rosa dan Shalahuddin 2013)

| **Nama** | **Simbol** | **Deskripsi** |
| --- | --- | --- |
| Aktor | actor.pngnama actor | Orang yang berinteraksi dengan sistem |
| *Use Case* |  | Menggambarkan fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau actor |
| Asosiasi |  | Komunikasi antara aktor dan *use case* yang berpartisipasi pada *use case* atau *use case* memiliki interaksi dengan actor |
| *Include* | <<include>> | Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan tersebut |

1. *Activity Diagram*

*Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak (Rosa dan Shalahuddin 2013). Simbol-simbol *activity diagram* dapat dilihat pada Tabel 2.

Tabel 2 Simbol-simbol *activity diagram* (Rosa dan Shalahuddin 2013)

| **Nama** | **Simbol** | **Deskripsi** |
| --- | --- | --- |
| Status awal |  | Status awal aktivitas sistem |
| Aktivitas |  | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja |
| Panah |  | Mewakili urutan antara kegiatan |
| Percabangan |  | Asosisasi percabangan dimana jika ada pilihan aktivitas lebih dari satu |
| Status akhir |  | Status akhir yang dilakukan sistem |
| *Swimlane* | swimlane.png | Batasan pelaku proses. |

1. *Class Diagram*

*Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem (Rosa dan Shalahuddin 2013). Simbol-simbol *class diagram* dapat dilihat pada Tabel 3.

Tabel 3 Simbol-simbol *class diagram* (Rosa dan Shalahuddin 2013)

|  |  |  |
| --- | --- | --- |
| **Nama** | **Simbol** | **Keterangan** |
| Kelas | nama\_kelas  +atribut  +operasi() | Kelas pada stuktur sistem |
| Asosiasi |  | Relasi antar kelas dengan makna umum. |
| Asosiasi berarah |  | Relasi antar kelas yang menunjukan hubungan langsung. |
| Generalisasi |  | Relasi antar kelas yang menunjukan hubungan kelas turunan dan kelas induk. |
| Agregasi |  | Relasi antar kelas dengan makna semua-bagian (*whole-part*) |

### *Database*

Sekumpulan data yang berelasi secara logis yang digunakan bersama, dan sebuah deskripsi dari data tersebut, yang dirancang untuk memenuhi kebutuhan informasi dari sebuah organisasi (Connolly dan Begg 2005).

### *Database Management System*

Merupakan sebuah sistem perangkat lunak yang memungkinkan penggunanya untuk mendefinisikan, membuat, memelihara, dan mengontrol akses suatu *database* (Connolly dan Begg 2005).

### PostgreSQL

PostgreSQL merupakan salah satu objek-relasional sistem *database* yang bersifat *open source*. PostgreSQL, awalnya disebut Postgres, dibuat di University of California at Berkeley (UCB) oleh seorang profesor ilmu komputer bernama Michael Stonebraker. Stonebraker mulai Postgres pada tahun 1986 sebagai sebuah proyek ikutan pendahulunya, Ingres, kini dimiliki oleh Computer Associates.

### *Hyper Text Markup Language*

*HyperText Markup Language* (HTML) adalah sebuah sistem untuk menandai, atau penandaan, sebuah dokumen sehingga dapat dipublikasikan di *web*. HTML mendefinisikan apa yang umumnya dikirimkan antara *node* dalam jaringan. (Flanagan 2011).

### *Cascading Style Sheet*

Ada banyak variabel dalam tampilan visual dari sebuah dokumen HTML: *font*, warna, spasi, dan sebagainya. Standar *Cascading Style Sheet* (CSS) menyebutkan variabel tersebut dan menamakannya sebagai *style property*. CSS mendefinisikan sifat yang menentukan *font*, warna, *margin*, *border*, *background*, gambar, perataan teks, ukuran elemen, dan posisi elemen. Untuk menentukan visual penampilan elemen HTML, kita tentukan nilai properti CSS (Flanagan 2011).

### PHP *Hypertext Preprocessor*

PHP *Hypertext Preprocessor* (PHP) adalah bahasa HTML-*scripting* yang tertanam dengan lisensi gratis dan bahasa pemrograman yang populer. PHP mendukung banyak *web-server* termasuk Apache HTTP Server dan Microsoft’s Internet Information Server. PHP ditujukan untuk memungkinkan pengembang Web untuk membuat halaman yang dinamis dengan cepat (Connolly & Begg 2005).

### *Framework*

*Framework* adalah komponen pemrograman yang siap *re-use* (bias digunakan ulang) kapan saja, sehingga *programmer* tidak harus membuat skrip yang sama untuk tugas yang sama. Kelebihan *framework* adalah kerangka kerja dari *framework* tersebut dalam menyelesaikan modul-modul yang dikembangkan, sehingga mengeluarkan metode pekerjaan yang lebih efisien, lebih rapi, lebih bersifat umum, dan lebih homogen (Purbadian 2015).

### Laravel

Laravel merupakan sebuah *framework* PHP yang dibuat oleh Taylor Otwell pada April tahun 2011. Salah satu kelebihan *framework* Laravel adalah *expressif*, yang artinya ketika melihat suatu *syntax* Laravel, seorang *programmer* “diharapkan” akan langsung mengetahui kegunaan dari *syntax* tersebut meskipun belum pernah mempelajari ataupun memnggunakannya (Purbadian 2015).

### Bootstrap

Awalnya dibuat oleh seorang desainer dan pengembang di Twitter, Bootstrap telah menjadi salah satu *frameworks* *front-end* dan proyek *open source* yang paling populer di dunia. Bootstrap diciptakan di Twitter pada pertengahan 2010 oleh Mark Otto dan Jacob. Sebelum menjadi *open-sourced framework*, Bootstrap dikenal sebagai Twitter Blueprint.

# KEADAAN UMUM TOMATECH MOBILE DEV

## Sejarah

Tomatech Mobile Dev didirikan pada Agustus 2012 dengan legalitas PT Tomat Ekno. Dimulai dengan sebuah tim kecil yang telah secara individual berpengalaman, Tomatech Mobile Dev telah mengambil setiap proyek *Mobile* *Apps* dan *Web Apps*, melayani via *offline* atau dengan *web* *server*. Sekarang, Tomatech telah bekerja sama dengan perusahaan dari beberapa negara seperti Indonesia, Jepang, Australia, dan Singapura yang membutuhkan layanan untuk mendirikan *Mobile Apps* dan *Web Apps* profesional.

Klien Tomatech umumnya berasal dari berbagai negara termasuk Indonesia, Jepang, Australia, dan negara-negara lain dari seluruh dunia. Tomatech menyediakan layanan untuk membuat konsep desain dasar dari antarmuka pengguna, membuat hanya aplikasi (lokal tanpa koneksi internet), sampai dengan pembuatan program aplikasi *mobile* yang kompleks (dengan *web* *server* / API) dan juga game *mobile*.

## Kegiatan

Sebagai perusahaan yang bekerja di bidang teknologi *mobile* dan *website*, Tomatech Mobile Dev memiliki beberapa produk dan jasa yang ditawarkan kepada klien. Produk dan jasa tersebut meliputi pengembangan aplikasi berbasis *mobile* (iOS dan Android), pengembangan sistem informasi dan pengembangan *website*.

## Struktur Organisasi

Struktur organisasi Tomatech Mobile Dev dikepalai oleh komisaris. Dibawah komisaris terdapat direktur yang membawahi manajer keuangan, program analis dan manajer pemasaran. Sedangkan *programmer* dipimpin langsung oleh program analis. Struktur organisasi Tomatech Mobile Dev dapat dilihat pada Gambar 2.

Bambang Hengky Rainanto

Komisaris

Achmad Cahya Aditya

Programmer

Sandy Wibisono

Direktur

Muhammad Zaki Alvan

Program Analis

Denny Dharmawan

Manajer Pemasaran

Muhammad Fahri

Manajer Pemasaran

Syahril Hermana

Programmer

Dita Oktavianita

Manajer Keuangan

Gambar 2 Struktur organisasi Tomatech Mobile Dev

## Visi dan Misi

Visi Tomatech Mobile Dev yaitu menjadi menjadi perusahaan pengembang aplikasi *mobile* dan *website* terbaik di bidang pengembangan profesionalisme kreatif dan inovatif, memberikan layanan dan hasil yang maksimal seperti yang dibutuhkan klien. Adapun misi dari Tomatech Mobile Dev yaitu:

1. Menciptakan ide-ide kreatif yang tak terbatas untuk setiap produk
2. Menjaga hubungan yang baik dengan klien
3. Memberikan desain antarmuka pengguna yang terbaik untuk setiap proyek
4. Mengambil perhatian untuk memenuhi kebutuhan klien

# PEMBUATAN MODUL PETA DAN BELANJA PADA APLIKASI RUMAH WARGA DI TOMATECH MOBILE DEV

## Prototipe Pertama

Prototipe pertama dibuat untuk memenuhi kebutuhan aplikasi yang diinginkan oleh klien. Dengan pembuatan prototipe pertama ini, klien dapat memberikan umpan balik yang dapat digunakan sebagai acuan pada iterasi prototipe selanjutnya bila aplikasi belum memenuhi permintaan klien.

### Komunikasi

Tahap komunikasi merupakan tahapan awal untuk memulai sebuah prototipe. Pada tahap ini, komunikasi dilakukan pihak pengembang perangkat lunak dengan pihak Tomatech Mobile Dev untuk mengidentifikasi kebutuhan aplikasi yang akan dibuat. Berdasarkan komunikasi yang telah dilakukan, pihak Tomatech Mobile Dev menginginkan modul peta dibuat untuk mengonversi peta dan mengelola detail peta. Sedangkan modul belanja dibuat menggunakan model *Customer to Customer* (C2C).

### Perencanaan Cepat

Pada tahap perencanaan cepat ini, dilakukan pemilihan perangkat yang akan digunakan dalam pembuatan aplikasi. Pada tahap ini pula dijabarkan kebutuhan aplikasi menjadi fungsionalitas-fungsionalitas.

1. **Perangkat Keras**

Perangkat keras yang digunakan dalam pembuatan aplikasi ini yaitu *notebook* Acer Aspire E1-470 dengan spesifikasi sebagai berikut:

1. Intel(R) Core(TM) i3-3217U CPU @ 1.80GHz (4 CPUs).
2. RAM 2GB.
3. Kapasitas hard disk 500GB.
4. Intel(R) HD Graphics 4000.
5. Ukuran layar 14”.
6. Resolusi 1366 x 768 *pixels.*
7. **Perangkat Lunak**

Perangkat lunak yang digunakan dalam pembuatan aplikasi ini adalah:

1. *Text editor*: JetBrains PhpStrom 8.0.3 dan Notepad ++ v6.2.2.
2. *Web server*: XAMPP v3.2.2.
3. *Database Management System* (DBMS): PostgreSQL.
4. *Database tools*: pgAdmin III versi 1.22 dan Navicat Premium versi 11.0.6.
5. *Web browser*: Google Chrome versi 50.0 dan Mozilla Firefox versi 46.0.1.
6. Desain Antarmuka: CorelDraw X7.
7. **Analisis Kebutuhan Sistem**

Analisis kebutuhan sistem dilakukan untuk menentukan fungsi dan fitur dari aplikasi yang akan dibuat. Kebutuhan awal pada modul peta dijabarkan secara terperinci. Sedangkan pada modul belanja kebutuhan dijabarkan secara umum terkait dengan *e-commerce* dengan jenis *Customer to Customer* (C2C).

1. **Analisis Kebutuhan Informasi**

Analisis kebutuhan informasi dilakukan untuk menjabarkan kebutuhan aplikasi yang diinginkan klien. Kebutuhan informasi yang akan dibuat pada prototipe pertama yaitu:

1. Menyediakan fasilitas konversi peta dari file berekstensi .svg.
2. Menyediakan fasilitas kelola peta
3. Menyediakan fasilitas kelola detail peta.
4. Menyediakan fasilitas untuk penjual menambahkan produk.
5. Menyediakan fasilitas untuk pelanggan memesan produk.
6. **Analisis Kebutuhan Fungsional**

Berdasarkan analisis kebutuhan informasi yang telah dilakukan, maka dapat diketahui kebutuhan fungsional modul peta dan modul belanja pada aplikasi Rumah Warga. Deskripsi kebutuhan fungsional pada prototipe pertama dapat dilihat pada Tabel 4.

Tabel 4 Deskripsi Kebutuhan fungsional prototipe pertama

| **Nama Fungsi** | **Deskripsi** |
| --- | --- |
| *Login* | Proses validasi akun pengguna |
| Menambah peta | Proses menambah data peta dengan mengkonversi file berekstensi .svg oleh admin |
| Melihat peta | Melihat data dan *preview* peta oleh admin. |
| Mengubah peta | Proses mengubah data peta yang ada oleh admin |
| Menghapus peta | Proses menghapus data peta oleh admin |
| Menambah detail peta | Proses menambah data detail peta oleh PJ lokasi |
| Melihat detail peta | Melihat data detail peta oleh PJ lokasi atau warga |
| Mengubah detail peta | Proses mengubah data detail peta oleh PJ lokasi |
| Menghapus detail peta | Proses menghapus data detail peta oleh PJ lokasi |
| Menambah produk baru | Proses menambahkan produk baru oleh penjual |
| Menambah produk | Proses menambah data produk oleh penjual |
| Melihat produk | Melihat data produk oleh penjual |
| Memperbarui produk | Proses memperbarui data produk oleh penjual |
| Menghapus produk | Proses menghapus data produk oleh penjual |
| Melihat katalog | Menampilkan katalog belanja per produk |
| Membeli produk | Melakukan pembelian produk tertentu |
| Konfirmasi pesanan | Mengkonfirmasi pesanan telah sampai oleh warga |
| Konfirmasi transaski | Mengkonfirmasi transaksi pesanan oleh penjual |
| Membatalkan pesanan | Membatalkan pesanan oleh warga |
| Membatalkan transaksi | Membatalkan transaksi pesanan oleh penjual |

### Pemodelan Perancangan Cepat

Tahap pemodelan perancangan cepat merupakan tahap pembuatan rancangan sistem. Rancangan sistem dibuat berdasarkan hasil analisis kebutuhan yang telah dilakukan pada tahap-tahap sebelumnya. Para tahap ini dibuat beberapa perancangan, yaitu perancangan *use case diagram*, perancangan *activity diagram*, perancangan *class diagram*, perancangan *database*, perancangan antarmuka, perancangan masukan, perancangan proses, dan perancangan keluaran.

1. **Perancangan *Use Case Diagram***

*Use case diagram* dibuat untuk menggambarkan aktivitas yang dapat dilakukan oleh pengguna dalam sistem. *Use case diagram* pada prototipe pertama dapat dilihat pada Gambar 3.

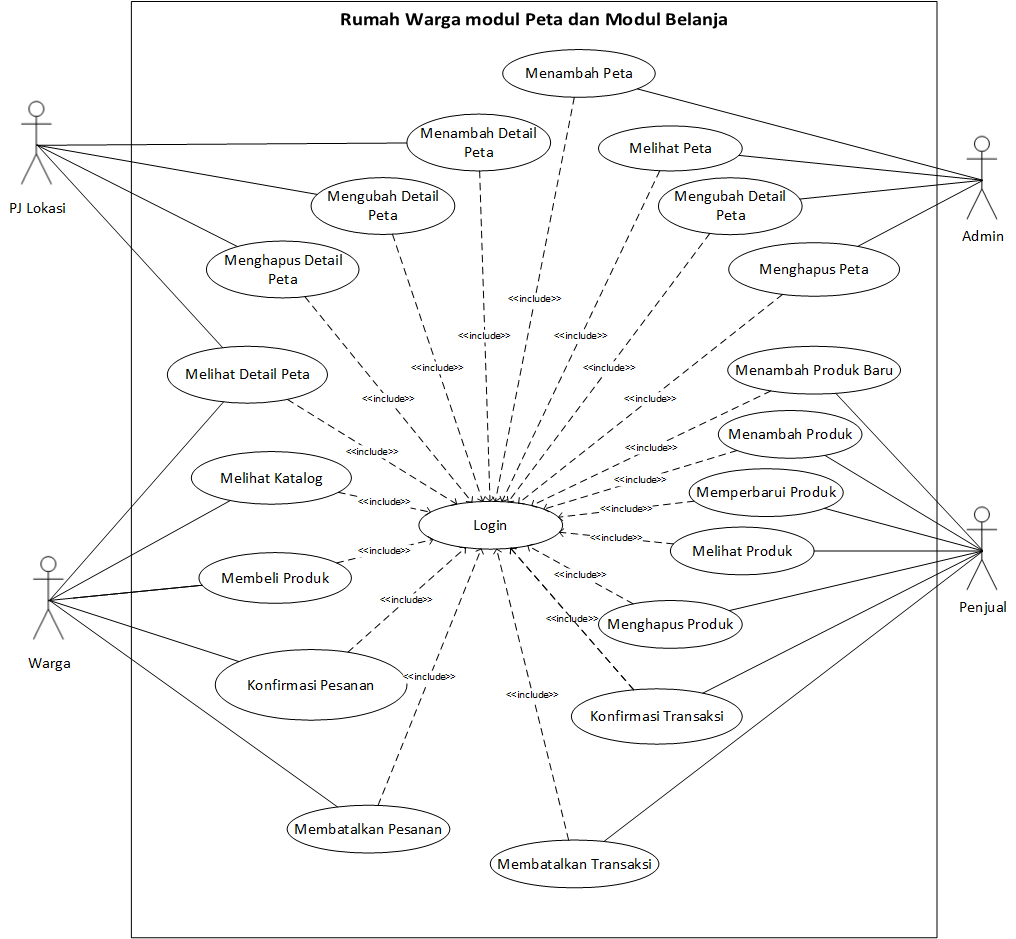
1. **Perancangan *Activity Diagram***

*Activity diagram* bertujuan untuk menggambarkan alur proses dari setiap fungsi yang dibuat dalam aplikasi*. Activity diagram* pada prototipe pertama dapat dilihat pada Tabel 5.

Tabel 5 *Activity diagram* prototipe pertama

| **Nama Proses** | **Tampilan** |
| --- | --- |
| *Login* | Lampiran 1 |
| Menambah peta | Lampiran 2 |
| Melihat peta | Lampiran 3 |
| Mengubah peta | Lampiran 4 |
| Menghapus peta | Lampiran 5 |
| Menambah detail peta | Lampiran 6 |
| Melihat detail peta | Lampiran 7 |
| Mengubah detail peta | Lampiran 8 |
| Menghapus detail peta | Lampiran 9 |
| Menambah produk baru | Lampiran 10 |
| Menambah produk | Lampiran 11 |
| Melihat produk | Lampiran 12 |
| Memperbarui produk | Lampiran 13 |
| Menghapus produk | Lampiran 14 |
| Melihat katalog | Lampiran 15 |
| Membeli produk | Lampiran 16 |
| Konfirmasi pesanan | Lampiran 17 |
| Konfirmasi transaksi | Lampiran 18 |
| Membatalkan pesanan | Lampiran 19 |
| Membatalkan transaksi | Lampiran 20 |

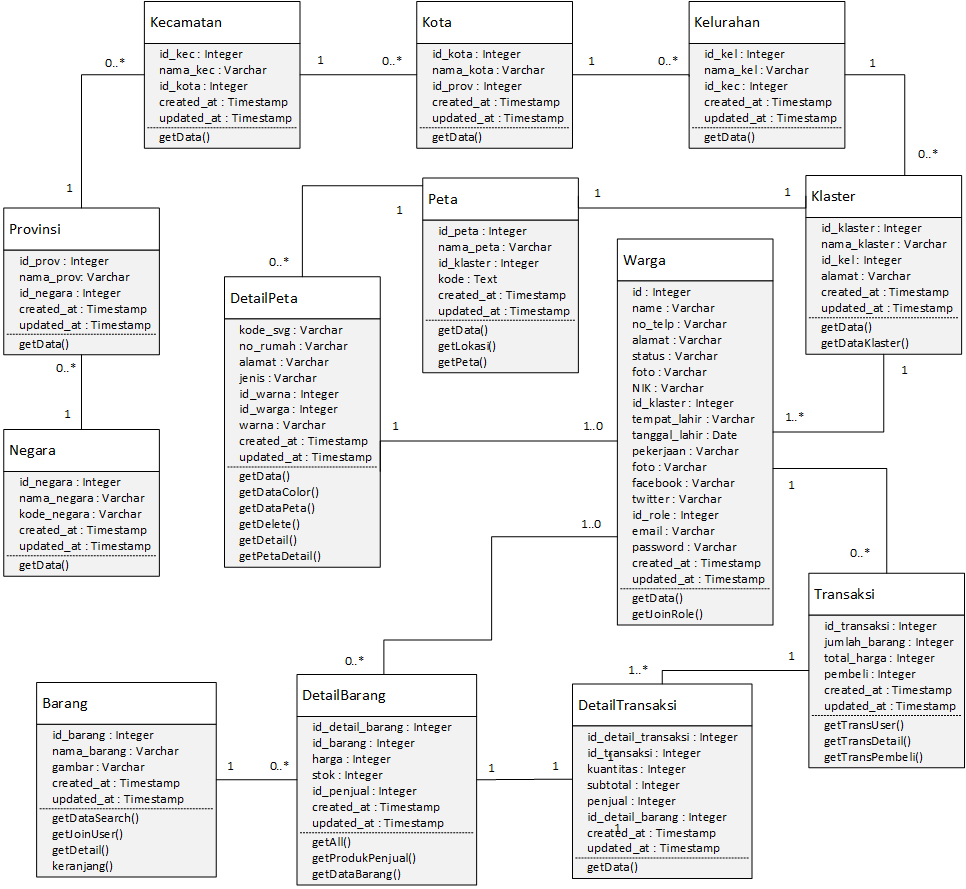
Gambar 3 *Use case diagram* prototipe pertama



1. **Perancangan *Class Diagram***

*Class diagram* dibuat untuk menggambarkan atribut dan operasi yang digunakan dalam pembuatan aplikasi. *Class diagram* dirancang berdasarkan model yang digunakan dalam sistem. *Class diagram* pada prototipe pertama dapat dilihat pada Gambar 4.

Gambar 4 *Class diagram* prototipe pertama

**

1. **Perancangan *Database***

Perancangan *database* merupakan rancangan yang mengambarkan struktur *database* yang akan digunakan dalam pembuatan aplikasi Rumah Warga. *Database* aplikasi Rumah Warga ini adalah ‘db\_neighborhood’. Tabel yang digunakan pada modul peta dan modul belanja berjumlah 13 tabel.

1. Tabel negara

Tabel negara digunakan untuk menyimpan data negara yang terdaftar di sistem. Deskripsi tabel negara dapat dilihat pada Tabel 6.

1. Tabel provinsi

Tabel dsetail\_peta digunakan untuk menyimpan data provinsi yang terdaftar di sistem. Deskripsi tabe detail\_peta dapat dilihat pada Tabel 7.

Tabel 6 Deskripsi tabel negara

| **Atribut** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| id\_negara | Integer | Primary Key |
| nama\_negara | Varchar(50) |  |
| kode\_negara | Varchar(3) |  |
| created\_at | Timestamp |  |
| created\_by | Integer |  |
| updated\_at | Timestamp |  |
| updated\_by | Integer |  |

Tabel 7 Deskripsi tabel provinsi

|  |  |  |
| --- | --- | --- |
| **Atribut** | **Tipe Data** | **Keterangan** |
| id\_prov | Integer | Primary Key |
| nama\_prov  id\_negara  created\_at  created\_by  updated\_at  updated\_by | Varchar(50)  Integer  Timestamp  Integer  Timestamp  Integer | Foreign Key |

1. Tabel kota

Tabel kota digunakan untuk menyimpan data kota yang terdaftar di sistem. Deskripsi tabel kota dapat dilihat pada Tabel 8.

Tabel 8 Deskripsi tabel kota

|  |  |  |
| --- | --- | --- |
| **Atribut** | **Tipe Data** | **Keterangan** |
| id\_kota | Integer | Primary Key |
| nama\_kota  id\_prov  created\_at  created\_by  updated\_at  updated\_by | Varchar(50)  Integer  Timestamp  Integer  Timestamp  Integer | Foreign Key |

1. Tabel kecamatan

Tabel kecamatan digunakan untuk menyimpan data kecamatan yang terdaftar di sistem. Deskripsi tabel kecamatan dapat dilihat pada Tabel 9.

Tabel 9 Deskripsi tabel kecamatan

| **Atribut** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| id\_kec | Integer | Primary Key |
| nama\_kec | Varchar(50) |  |
| id\_kota  created\_at | Integer  Timestamp | Foreign Key |
| created\_by | Integer |  |
| updated\_at | Timestamp |  |
| updated\_by | Integer |  |

1. Tabel kelurahan

Tabel kelurahan digunakan untuk menyimpan data kelurahan yang terdaftar di sistem. Deskripsi tabel kelurahan dapat dilihat pada Tabel 10.

Tabel 10 Deskripsi tabel kelurahan

|  |  |  |
| --- | --- | --- |
| **Atribut** | **Tipe Data** | **Keterangan** |
| id\_kel | Integer | Primary Key |
| nama\_kel  id\_kec  created\_at  created\_by  updated\_at  updated\_by | Varchar(50)  Integer  Timestamp  Integer  Timestamp  Integer | Foreign Key |

1. Tabel klaster

Tabel klaster digunakan untuk menyimpan data klaster yang terdaftar di sistem. Deskripsi klaster dapat dilihat pada Tabel 11.

Tabel 11 Deskripsi tabel klaster

|  |  |  |
| --- | --- | --- |
| **Atribut** | **Tipe Data** | **Keterangan** |
| id\_klaster | Integer | Primary Key |
| nama\_klaster  id\_kel  alamat  created\_at  created\_by  updated\_at  updated\_by | Varchar(50)  Integer  Varchar(100)  Timestamp  Integer  Timestamp  Integer | Foreign Key |

1. Tabel peta

Tabel peta digunakan untuk menyimpan data peta yang terdaftar untuk sebuah klaster. Deskripsi peta dapat dilihat pada Tabel 12.

Tabel 12 Deskripsi tabel peta

|  |  |  |
| --- | --- | --- |
| **Atribut** | **Tipe Data** | **Keterangan** |
| id\_peta | Integer | Primary Key |
| nama\_peta  id\_klaster  kode  created\_at  created\_by  updated\_at  updated\_by | Varchar(50)  Integer  Text  Timestamp  Integer  Timestamp  Integer | Foreign Key |

1. Tabel detail\_peta

Tabel detail\_peta digunakan untuk menyimpan data detail peta pada peta. Deskripsi tabel peta dapat dilihat pada Tabel 13.

Tabel 13 Deskripsi tabel detail\_peta

|  |  |  |
| --- | --- | --- |
| **Atribut** | **Tipe Data** | **Keterangan** |
| kode\_svg | Varchar(10) | Primary Key |
| jenis  id\_peta  id\_warga  alamat  no\_rumah  warna  id\_warna  created\_at  created\_by  updated\_at  updated\_by | Varchar(50)  Integer  Integer  Varchar(100)  Varchar(5)  Varchar(20)  Integer  Timestamp  Integer  Timestamp  Integer | Foreign Key  Foreign Key |

1. Tabel users

Tabel users digunakan untuk menyimpan data users yang terdaftar di sistem. Deskripsi tabel users dapat dilihat pada Tabel 14.

Tabel 14 Deskripsi tabel users

| **Atribut** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| id | Integer | Primary Key |
| name  email  password  id\_klaster  status  no\_telp  alamat  parent  tempat\_lahir  tanggal\_lahir  facebook  twitter  pekerjaan  foto  NIK  remember\_token  id\_role  created\_at  created\_by  updated\_at  updated\_by | Varchar(50)  Varchar(50)  Varchar(20)  Integer  Varchar(50)  Varchar(12)  Varchar(100)  Integer  Varchar(50)  Date  Varchar(50)  Varchar(50)  Varchar(50)  Text  Varchar(16)  Varchar(100)  Integer  Timestamp  Integer  Timestamp  Integer | Unique  Foreign Key  Foreign Key |

1. Tabel transaksi

Tabel transaksi digunakan untuk menyimpan data pembelian barang oleh pengguna. Deskripsi tabel transaksi dapat dilihat pada Tabel 15.

Tabel 15 Deskripsi tabel transaksi

|  |  |  |
| --- | --- | --- |
| **Atribut** | **Tipe Data** | **Keterangan** |
| id\_transaksi | Integer | Primary Key |
| jumlah\_barang  total\_harga  pembeli  created\_at  created\_by  updated\_at  updated\_by | Integer  Integer  Integer  Timestamp  Integer  Timestamp  Integer | Foreign Key |

1. Tabel detail\_transaksi

Tabel detail\_transaksi digunakan untuk menyimpan detail dari setiap transaksi yang ada. Deskripsi tabel detail\_transaksi dapat dilihat pada Tabel 16.

Tabel 16 Deskripsi tabel detail\_transaksi

|  |  |  |
| --- | --- | --- |
| **Atribut** | **Tipe Data** | **Keterangan** |
| id\_detail\_transaksi  id\_transaksi  id\_detail\_barang | Integer  Integer  Integer | Primary Key  Foreign Key  Foreign Key |
| kuantitas  subtotal  penjual  status  created\_at  created\_by  updated\_at  updated\_by | Integer  Integer  Integer  Varchar(20)  Timestamp  Integer  Timestamp  Integer |  |

1. Tabel detail\_barang

Tabel barang digunakan untuk menyimpan data detail barang yang dijual oleh penjual. Deskripsi tabel detail barang dapat dilihat pada Tabel 17.

Tabel 17 Deskripsi tabel detail\_barang

|  |  |  |
| --- | --- | --- |
| **Atribut** | **Tipe Data** | **Keterangan** |
| id\_detail\_barang  id\_penjual  id\_barang | Integer  Integer  Integer | Primary Key  Foreign Key  Foreign Key |
| stok  harga  created\_at  created\_by  updated\_at  updated\_by | Integer  Integer  Timestamp  Integer  Timestamp  Integer |  |

1. Tabel barang

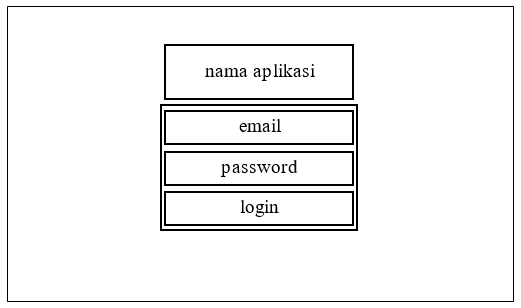
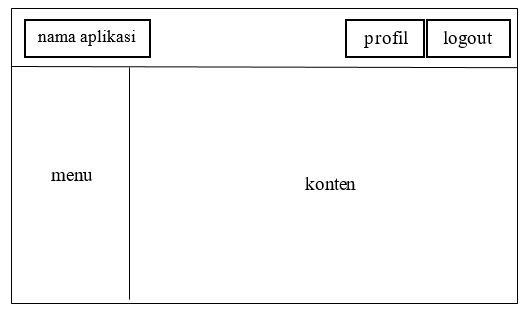
Tabel barang digunakan untuk menyimpan data barang yang terdaftar di sistem. Deskripsi tabe barang dapat dilihat pada Tabel 18.

Tabel 18 Deskripsi tabel barang

|  |  |  |
| --- | --- | --- |
| **Atribut** | **Tipe Data** | **Keterangan** |
| id\_barang  nama\_barang  gambar | Integer  Varchar(50)  Text | Primary Key |
| created\_at  created\_by  updated\_at  updated\_by | Timestamp  Integer  Timestamp  Integer |  |

1. **Perancangan Antarmuka**

Perancangan antarmuka dilakukan untuk menggambarkan antarmuka yang akan dibuat dalam aplikasi Rumah Warga ini. Pada perancangan antarmuka ini, antarmuka untuk halaman *login* yang merupakan halaman awal dari aplikasi dibuat sederhana. Perancangan antarmuka halaman *login* dapat dilihat pada Gambar 5.

****Perancangan antarmuka halaman utama dibuat seragam untuk tiap-tiap pengguna yang ada di dalam sistem. Antarmuka juga dibuat *responsive* agar memudahkan pengguna untuk mengakses aplikasi pada berbagai *platform*. Perancangan antarmuka untuk halaman utama dapat diihat pada Gambar 6.

Gambar 5 Perancangan antarmuka halaman *login*

Gambar 6 Perancangan antarmuka halaman utama

1. **Perancangan Masukan**

Perancangan masukan dibuat untuk menentukan masukan data yang dilakukan oleh pengguna sistem. Perancangan ini dilakukan untuk mengurangi kesalahan dalam proses masukan data. Perancangan masukan pada prototipe pertama dapat dilihat pada Tabel 19.

Tabel 19 Perancangan masukan prototipe pertama

|  |  |  |
| --- | --- | --- |
| **Perancangan Masukan** | **Deskripsi** | **Tampilan** |
| *Login* | Memasukan alamat email dan kata sandi pengguna | Lampiran 21 |
| Konversi peta | Memasukan kode dan nama peta oleh admin | Lampiran 22, Lampiran 23 |
| Ubah peta | Memasukan kode dan nama peta oleh admin | Lampiran 24,  Lampiran 23 |
| Tambah detail peta | Memasukan data warga atau keterangan peta oleh PJ lokasi | Lampiran 25 |
| Ubah detail peta | Memasukan data warga atau keterangan peta PJ lokasi | Lampiran 26 |
| Tambah produk baru | Memasukan data produk baru oleh penjual | Lampiran 27 |
| Tambah produk | Memasukan data produk oleh penjual | Lampiran 28 |
| Perbarui produk | Memasukan data produk oleh penjual | Lampiran 29 |
| Masukan kuantitas pesanan | Memasukan kuantitas produk yang dipesan oleh warga | Lampiran 30 |

1. **Perancangan Proses**

Perancangan proses dibuat untuk menggambarkan alur data yang berjalan pada sistem. Perancangan proses digambarkan dalam bentuk *activity diagram*. Sebuah *activity diagram* merepresentasikan sebuah *use case*. Perancangan proses dalam bentuk activity diagram ini dapat dilihat pada Tabel 5.

1. **Perancangan Keluaran**

Perancangan keluaran dibuat untuk menampilkan keluaran dari masukan yang dilakukan oleh pengguna. Deskripsi perancangan keluaran pada prototipe pertama dapat dilihat pada Tabel 20.

Tabel 20 Perancangan keluaran prototipe pertama

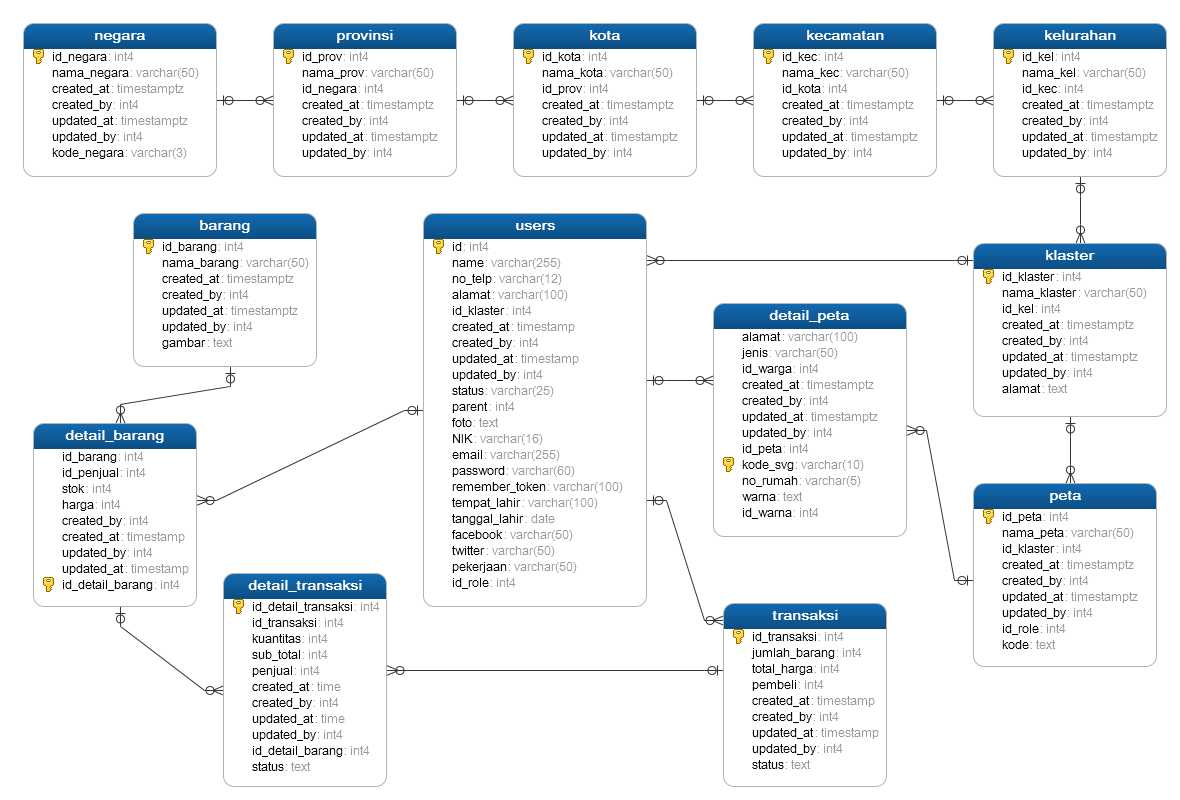
| **Perancangan Keluaran** | **Deskripsi** | **Tampilan** |
| --- | --- | --- |
| Informasi peta | Halaman keluaran dari proses konversi dan ubah peta | Lampiran 31 |
| Informasi detail peta | Halaman keluaran dari proses tambah dan ubah detail peta | Lampiran 32 |
| Informasi produk | Halaman keluaran dari proses tambah dan perbarui produk | Lampiran 33 |
| Informasi peta klaster | Halaman keluaran untuk menampilkan detail peta | Lampiran 34 |
| Informasi detail produk | Halaman keluaran untuk menampilkan detail produk | Lampiran 35 |

### Pembentukan Prototipe

Pembentukan prototipe merupakan implementasi dari tahap pemodelan perancangan cepat yang telah dilakukan sebelumnya. Pada tahap ini, dilakukan implementasi perancangan *database*, perancangan antarmuka, perancangan masukan, perancangan proses, dan perancangan keluaran.

1. **Implementasi *Database***

Perancangan *database* diimplementasikan menggunakan PostgresSQL sebagai DBMS dan diberi nama ‘db\_neighborhood’. Pada prototipe pertama ini, *database* untuk modul peta dan modul belanja berjumlah 13 tabel. Implementasi database pada prototipe pertama dapat dilihat pada Gambar 5.



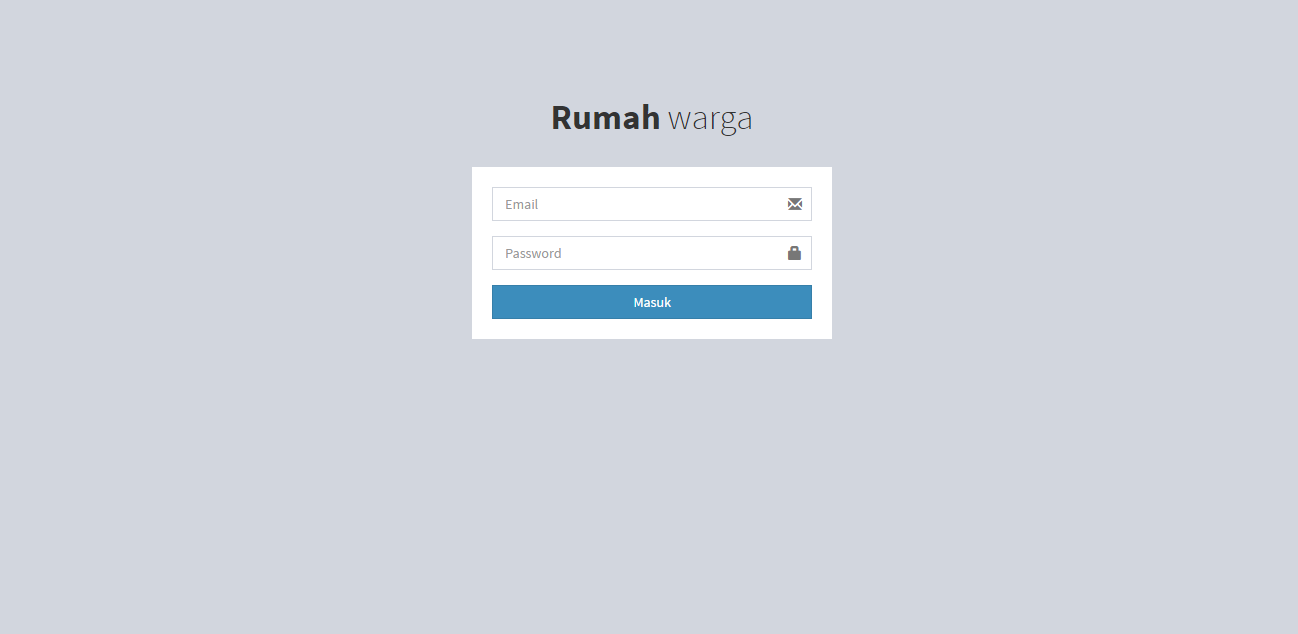
Gambar 5 Implementasi *database* prototipe pertama

1. **Implementasi Antarmuka**

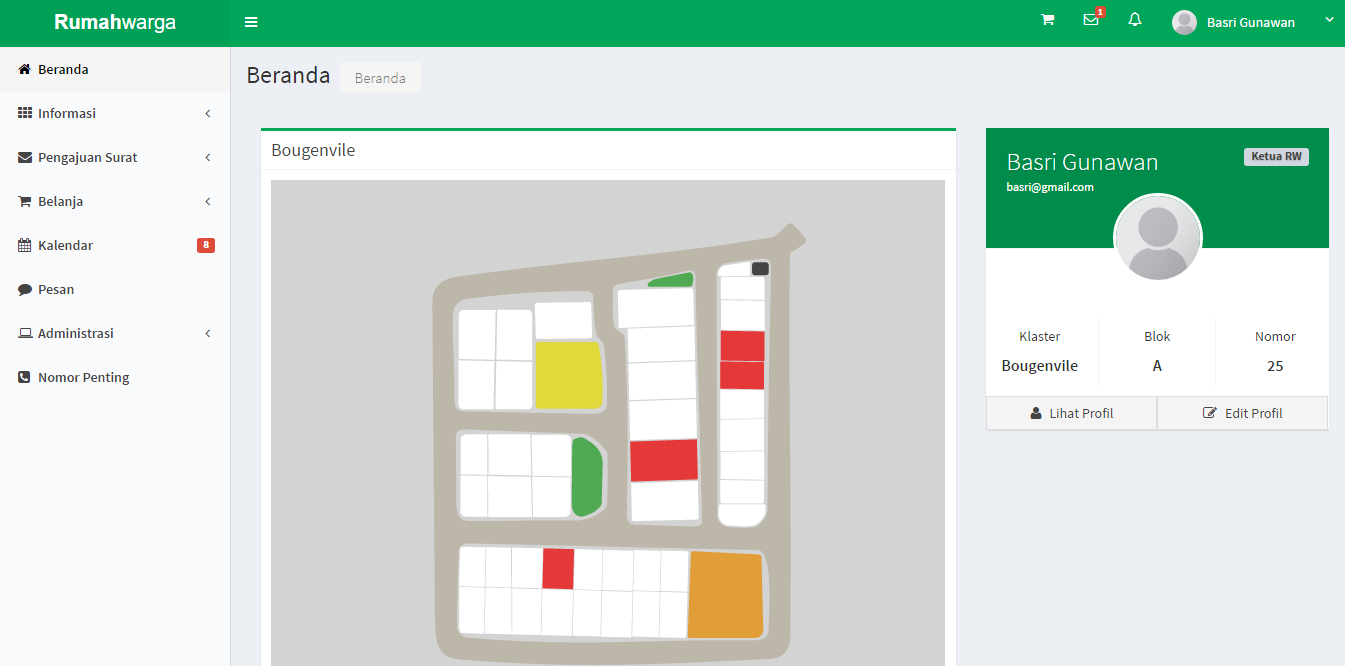
Implementasi antamuka merupakan penerapan dari tahap perancangan antarmuka yang telah dibuat sebelumnya. Implementasi antarmuka menggunakan *template* CSS dari Bootstrap. Implementasi rancangan antarmuka halaman *login* dan halaman utama dapat dilihat pada Gambar 6 dan Gambar 7.

1. **Impemantasi Masukan**

Pada tahap ini, perancangan masukan yang dibuat pada tahap sebelumnya akan diimplementasikan dengan menjadi tampilan masukan. Impementasi masukan pada prototipe pertama dapat dilihat pada Tabel 21.



Gambar 6 Implementasi halaman *login*



Gambar 7 Implementasi halaman utama

Tabel 21 Implementasi masukan prototipe pertama

| **Perancangan Masukan** | **Deskripsi** | **Tampilan** |
| --- | --- | --- |
| *Login* | Memasukan alamat email dan kata sandi pengguna | Lampiran 36 |
| Konversi peta | Memasukan kode dan nama peta oleh admin | Lampiran 37,  Lampiran 38 |
| Ubah peta | Memasukan kode dan nama peta oleh admin | Lampiran 39,  Lampiran 38 |
| Tambah detail peta | Memasukan data warga atau keterangan peta oleh PJ lokasi | Lampiran 40 |
| Ubah detail peta | Memasukan data warga atau keterangan peta PJ lokasi | Lampiran 41 |
| Tambah produk baru | Memasukan data produk baru oleh penjual | Lampiran 42 |
| Tambah produk | Memasukan data produk oleh penjual | Lampiran 43 |
| Perbarui produk | Memasukan data produk oleh penjual | Lampiran 44 |
| Masukan kuantitas pesanan | Memasukan kuantitas produk yang dipesan oleh warga | Lampiran 45 |

1. **Implementasi Proses**

Implementasi proses merupakan tahapan pembuatan proses-proses yang ada pada sistem dengan membuat kode program. Kode program dibuat menggunakan bahasa PHP dengan *framework* Laravel. Salah satu contoh implementasi proses dapat dilihat pada Gambar 8.

public function postAddMap(Request $request){  
 try {  
 $data = PetaModel::*findOrNew*($request->input("id\_peta"));  
 $data->nama\_peta = $request->input("nama\_peta");  
 $data->kode = $request->input("kode");  
 $data->id\_klaster = $request->input("klaster");  
 $data->created\_by= Auth::*user*()->id;  
 $data->updated\_by= Auth::*user*()->id;  
 $data->save();  
 Session::*flash*('flash\_message','Peta Berhasil ditambahkan.');  
 return redirect("peta");  
  
 }catch (QueryException $e){  
 Log::*error*($e->getMessage());  
 return redirect('/peta');  
 }  
}  
  
public function getMapName($id\_klaster){  
 $klaster = new KlasterModel();  
 $data = $klaster->getDataKlaster($id\_klaster);  
 $html = '<label>Nama Peta <small>(nama klaster)</small>:</label>  
 <input class="form-control" type="text" id="setting-map-name" name="nama\_peta" value="'.$data->nama\_klaster.'"/>';  
 return $html;  
}

Gambar 8 Implementasi proses prototipe pertama

1. **Implementasi Keluaran**

Implementasi keluaran merupakan penerapan dari tahapan perancangan keluaran. Hasil yang ditampilkan pada implementasi keluaran merupakan hasil dari masukan data. Implementasi keluaran pada prototipe pertama dapat dilihat pada Tabel 22.

Tabel 22 Implementasi keluaran prototipe pertama

| **Perancangan Keluaran** | **Deskripsi** | **Tampilan** |
| --- | --- | --- |
| Informasi peta | Halaman keluaran dari proses konversi dan ubah peta | Lampiran 46 |
| Informasi detail peta | Halaman keluaran dari proses tambah dan ubah detail peta | Lampiran 47 |
| Informasi produk | Halaman keluaran dari proses tambah dan perbarui produk | Lampiran 48 |
| Informasi peta klaster | Halaman keluaran untuk menampilkan detail peta | Lampiran 49 |
| Informasi detail produk | Halaman keluaran untuk menampilkan detail produk | Lampiran 50 |

### Penyerahan Sistem ke Pengguna & Umpan Balik

Tahap penyerahan sistem ke pengguna & umpan balik merupakan tahap akhir dari sebuah prototipe. Prototipe yang telah dibuat kemudian diserahkan kepada klien untuk dilakukan pengujian. Pengujian prototipe dilakukan menggunakan metode *blackbox.* Pengujian ini bertujuan untuk menyesuaikan program yang telah dibuat dengan kebutuhan yang diinginkan klien. Cara kerja pengujian *blackbox* yaitu dengan melakukan *input* data untuk setiap proses masukan. Deskripsi pengujian *blackbox* pada prototipe dapat dilihat pada Tabel 23.

Tabel 23 Pengujian *blackbox* prototipe pertama

| **Pengujian** | **Skenario Pengujian** | **Hasil yang diharapkan** | **Hasil Uji** |
| --- | --- | --- | --- |
| *Login* | Memasukan email dan kata sandi yang benar | Masuk kedalam halaman utama | Berhasil |
|  | Memasukan email dan kata sandi yang salah | Muncul peringatan email atau kata sandi salah | Berhasil |
| Konversi peta | Seluruh *input* pada *form* diisi dengan benar | Data berhasil disimpan | Berhasil |
|  | Mengosongkan salah satu *input* pada *form* | Muncul peringatan *input* belum diisi | Berhasil |
| Ubah peta | Seluruh *input* pada *form* diisi dengan benar | Data berhasil disimpan | Berhasil |
|  | Mengosongkan *input* pada *form* | Muncul peringatan *input* belum diisi | Berhasil |
| Tambah detail peta | Seluruh *input* pada *form* diisi dengan benar | Data berhasil disimpan | Berhasil |
|  | Mengosongkan *input* yang bersifat wajib diisi | Muncul peringatan *input* belum diisi | Berhasil |
| Ubah detail peta | Seluruh *input* pada *form* diisi dengan benar | Data berhasil disimpan | Berhasil |
|  | Mengosongkan *input* yang bersifat wajib diisi | Muncul peringatan *input* belum diisi | Berhasil |
| Tambah produk baru | Seluruh *input* pada *form* diisi dengan benar | Data berhasil disimpan | Berhasil |
|  | Menambahkan data yang sudah ada | Muncul peringatan data telah ada | Gagal |
| Tambah produk | Seluruh *input* pada *form* diisi dengan benar | Data berhasil disimpan | Berhasil |
|  | Mengosongkan *input* pada *form* | Muncul peringatan *input* belum diisi | Berhasil |
| Perbarui produk | Seluruh *input* pada *form* diisi dengan benar | Data berhasil disimpan | Berhasil |
|  | Mengosongkan *input* pada *form* | Muncul peringatan *input* belum diisi | Berhasil |
| Masukan kuantitas pesanan | Memasukan nilai normal | Data berhasil disimpan | Berhasil |
|  | Memasukan nilai dibawah nilai stok | Muncul peringatan *input* salah | Berhasil |

Setelah tahap pengujian selesai, dilakikan perbaikan aplikasi pada proses yang pengujiannya masih gagal. Penyerahan aplikasi ini diikuti dengan umpan balik dari klien untuk menyempurnakan sistem pada prototipe selanjutnya.

## Prototipe Kedua

Prototipe kedua dilakukan berdasarkan hasil dari prototipe pertama. Tujuan dari prototipe kedua ini adalah untuk memenuhi kebutuhan yang belum terjabarkan pada prototipe pertama.

### Komunikasi

Tahap komunikasi kembali dilakukan pengembang dengan pihak Tomatech Mobile Dev untuk mengevaluasi kebutuhan fungsional prototipe pertama. Pada tahap ini pihak Tomatech Mobile Dev menjabarkan kebutuhan pada modul belanja untuk menggunakan keranjang pada saat warga memesan produk. Keranjang dibuat menggunakan *session* dan hilang bila pengguna melakukan *logout*.

### Perencanaan Cepat

Tahap perencanaan cepat pada prototipe kedua menjelaskan penambahan kebutuhan saja. Perangkat keras dan perangkat lunak pendukung sama dengan yang digunakan pada prototipe pertama.

1. **Analisis Kebutuhan Informasi**

Berdasarkan tahap komunikasi pada prototipe kedua yang telah dilakukan, terdapat penambahan kebutuhan sistem. Kebutuhan informasi yang ditambahkan pada prototipe kedua adalah menyediakan fasilitas keranjang.

1. **Analisis Kebutuhan Fungsional**

Kebutuhan fungsional pada prototipe kedua terdapat penambahan. Penambahan disesuaikan dengan kebutuhan informasi yang telah di ketahui. Deskripsi kebutuhan fungsional pada prototipe kedua dapat dilihat pada Tabel 24.

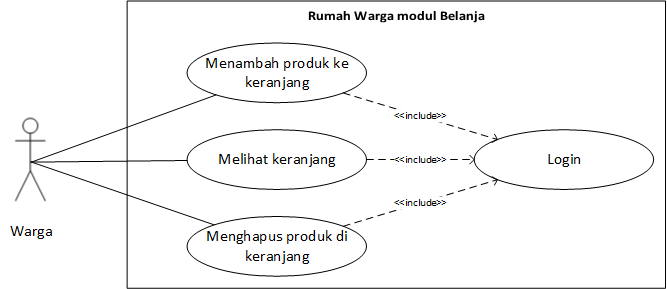
Tabel 24 Deskripsi kebutuhan fungsional prototipe kedua

| **Nama Fungsi** | **Deskripsi** |
| --- | --- |
| Menambah produk ke keranjang | Proses menambah data produk yang dipesan ke dalam keranjang belanja oleh warga |
| Melihat keranjang | Melihat data produk yang ada di dalam keranjang belanja oleh warga |
| Menghapus produk di keranjang | Proses menghapus data produk yang ada di dalam keranjang belanja oleh warga |

### Pemodelan Perancangan Cepat

Tahap pemodelan perancangan cepat prototipe kedua hanya menjelaskan penambahan yang tidak ada di prototipe pertama saja. Berdasarkan hasil dari tahap komunikasi, fungsi keranjang dibuat menggunakan *session* sehingga tidak menambahkan tabel di *database*. Sehingga, penambahan hanya terdapat pada perancangan *use case diagram*, perancangan *activity diagram,* dan perancangan antarmuka.

1. **Perancangan *Use Case Diagram***

Perancangan *use case diagram* pada prototipe kedua terdapat penambahan tiga *use* *case*. *Use case diagram* pada prototipe kedua dapat dilihat pada Gambar 9.

Gambar 9 *Use case diagram* prototipe kedua

1. **Perancangan *Activity Diagram***

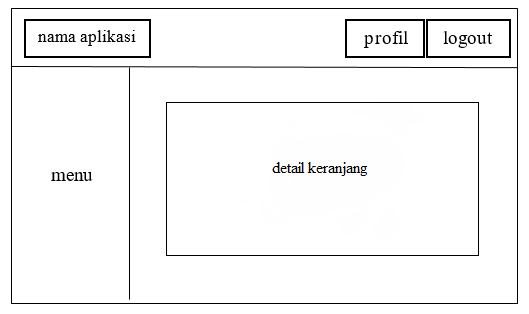
Pada perancangan *activity diagram* prototipe kedua ini, dibuat *activity diagram* tambahan yang tidak terdapat pada prototipe pertama. *Activity diagram* pada prototipe kedua dapat dilihat pada Tabel 25.

Tabel 25 *Activity diagram* prototipe kedua

| **Nama Proses** | **Tampilan** |
| --- | --- |
| Menambah produk ke keranjang | Lampiran 51 |
| Melihat keranjang | Lampiran 52 |
| Menghapus produk di keranjang | Lampiran 53 |

1. **PerancanganAntarmuka**

Terdapat penambahan antarmuka pada prototipe kedua yaitu halaman keranjang. Halaman keranjang digunakan untuk menampilkan produk-produk yang ada di dalam keranjang belanja. Perancangan antarmuka halaman keranjang belanja dapat dilihat pada Gambar 10.



Gambar 10 Perancangan antarmuka halaman keranjang

### Pembentukan Prototipe

Seperti pada prototipe pertama, pembentukan prototipe merupakan implementasi dari tahap pemodelan perancangan cepat yang telah dilakukan. Pada tahap ini akan dibuat fungsi keranjang. Potongan kode program pada fungsi keranjang dapat dilihat pada Gambar 11.

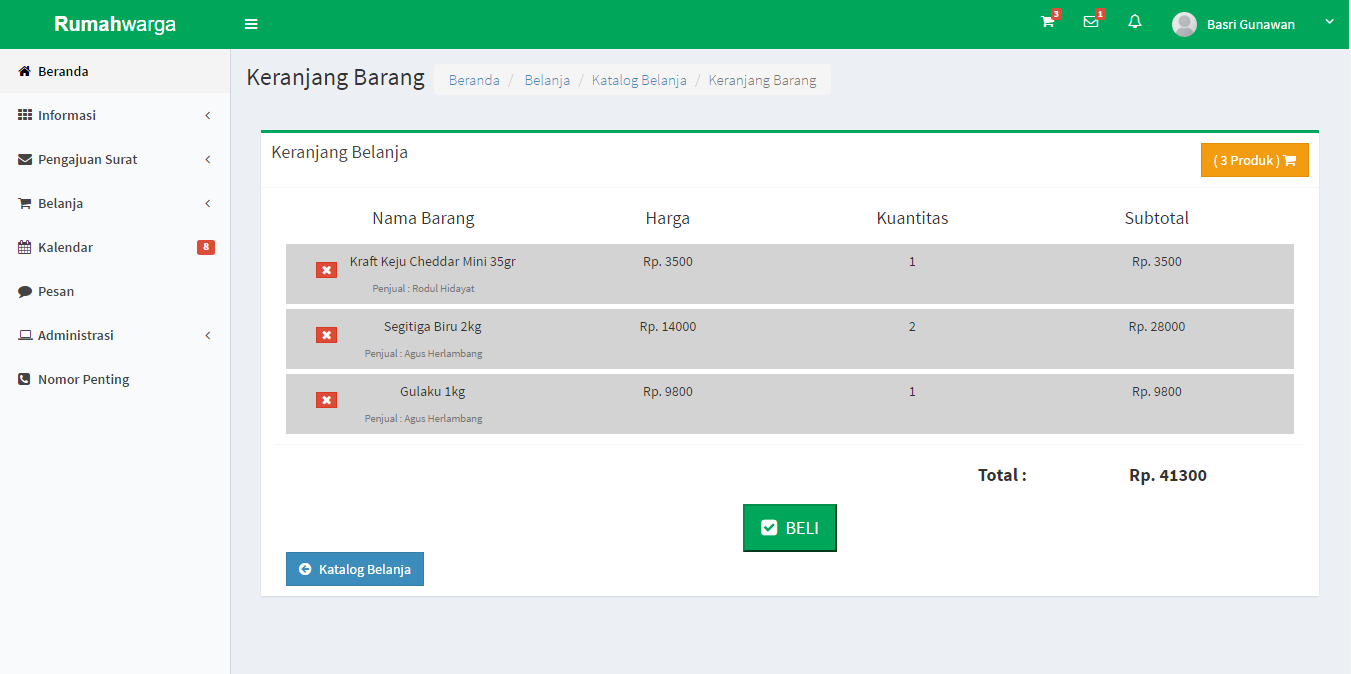
Pada tahap ini implementasi antarmuka dilakukan berdasarkan perancangan antarmuka yang telah dilakukan. Implementasi antarmuka halaman keranjang dapat dilihat pada Gambar 12.

public function postAddCart(Request $request){  
 try {  
 $id\_detail\_barang = $request->input('id\_detail\_barang');  
 $jumlah\_barang = $request->input('jumlah');  
 $nama\_barang = $request->input('nama\_barang');  
 $harga = $request->input('harga');  
 $penjual = $request->input('penjual');  
 $penjualID = $request->input('penjualID');

DB::*table*('detail\_barang')->where('id\_detail\_barang',

'=',$id\_detail\_barang)->decrement('stok',$jumlah\_barang);  
 $data = array(  
 'id' => $id\_detail\_barang,  
 'name' => $nama\_barang,  
 'qty' => $jumlah\_barang,  
 'price' => $harga,  
 'options' => array(  
 'seller' => $penjual,  
 'sellerID' => $penjualID  
 )  
 );  
 Cart::*add*($data);  
 Session::*flash*('flash\_message', 'Barang berhasil ditambahkan ke keranjang belanja.');  
 return redirect('katalog');

Gambar 11 Kode program pada fungsi keranjang



Gambar 12 Implementasi halaman keranjang

### Penyerahan Sistem ke Pengguna & Umpan Balik

Setelah prototipe kedua selesai dibuat, prototipe kemudian diserahkan ke klien untuk dilakukan evaluasi. Pada prototipe kedua ini, tidak dilakukan perancangan dan impementasi masukan maupun keluaran, sehingga tidak diperlukan pengujian kembali. Berdasarkan evaluasi yang dilakukan klien, tidak ada penambahan kebutuhan untuk protorite kedua. Prototipe kedu ini sudah siap untuk digunakan dan dikembangkan oleh pihak Tomatech Mobile Dev.

## Evaluasi Seminard

Selama pemaparan aplikasi dalam forum diskusi, tercatat terdapat saran untuk pertimbangan perbaikan. Daftar saran dapat dilihat pada Tabel 26.

Tabel 26 Daftar saran dalam forum

| **Saran** | **Status** | **Keterangan** |
| --- | --- | --- |
| Menambahkan legenda pada peta klaster | Tidak ditambahkan | Keterangan peta sudah ada saat *mouse* meng-*hover* bagian peta |
| Menambahkan alamat rinci klaster | Ditambahkan | Alamat klaster dimasukan pada saat mendaftarkan klaster |
| Menambahkan riwayat transaksi | Ditambahkan | Riwayat transaksi ditambahkan untuk warga dan penjual |
| Mengurangi stok produk saat dimasukan kedalam keranjang | Ditambahkan | Stok produk otomatis berkurang saat produk dimasukan ke keranjang |
| Menambahkan fitur pesan produk untuk beberapa hari yang akan datang | Tidak ditambahkan | Fitur pesan produk kurang diperlukan pada aplikasi |

# SIMPULAN

## Simpulan

Modul peta dan modul belanja pada aplikasi Rumah Warga memiliki fungsi sebagai berikut:

1. Penanggung Jawab Lokasi dapat menambahkan, melihat, mengubah dan menghapus data lokasi warga pada sebuah klaster.
2. Katalog produk ditampilkan per produk dengan detail produk berisikan data dari para penjual.
3. Pembeli dapat menambahkan produk ke keranjang, melihat keranjang dan menghapus produk dari keranjang.
4. Pembeli dapat membeli produk yang sebelumnya telah dimasukan ke dalam keranjang.

## Saran

Modul peta dan modul belanja pada aplikasi Rumah Warga yang telah dibuat sudah memenuhi kebutuhan klien. Namun, dalam penambahan peta masih terbatas untuk menggunakan file berekstensi .svg saja. Hal tersebut membatasi fitur dan kurang memberikan kemudahan untuk pengguna. Oleh karena itu, diperlukan pengembangan pada konversi peta untuk menambahkan pilihan file berektensi yang lebih umum dan mudah digunakan oleh pengguna.

# DAFTAR PUSTAKA

Bootsrap. <http://getbootstrap.com/>. [27 Mei 2016].

Connolly T, Begg C. 2005*. Database Systems: A Practical Approach to Design, Implementation, and Management. Fourth Edition. New York.*

Flanagan D. 2011. *JavaScript: The Devinitive Guide, Sixth Edition*. Sebastopol: O’Reilly Media Inc.

PostgreSQL. About. <http://www.postgresql.org/about/>. [27 Mei 2016].

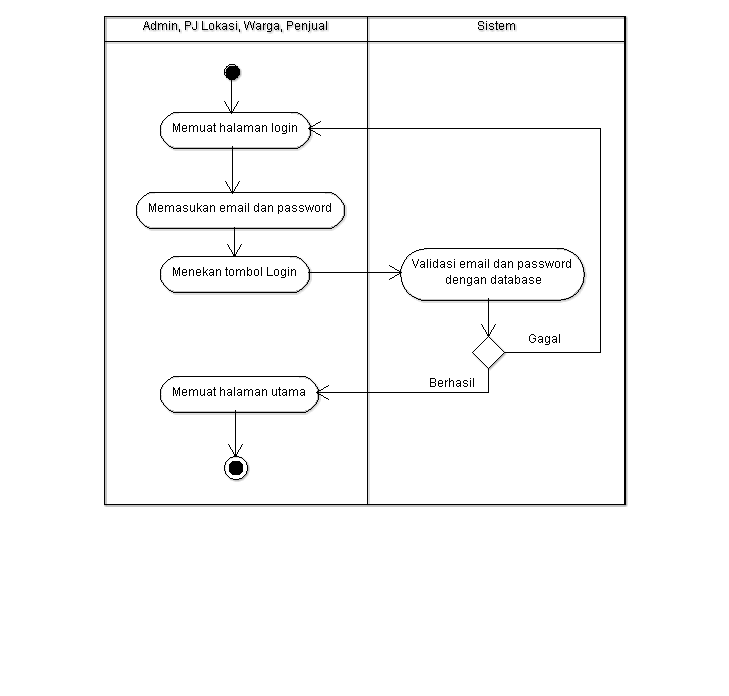
Pressman RS. 2012. *Rekayasa Perangkat Lunak pendekatan praktisi*. Ed ke-7. Yogyakarta: Andi.

Purbadian, Yenda. 2015. *Framework Laravel 5 Panduan Praktis dan Trik Jitu*. Cirebon: CV. Asfa Solution.

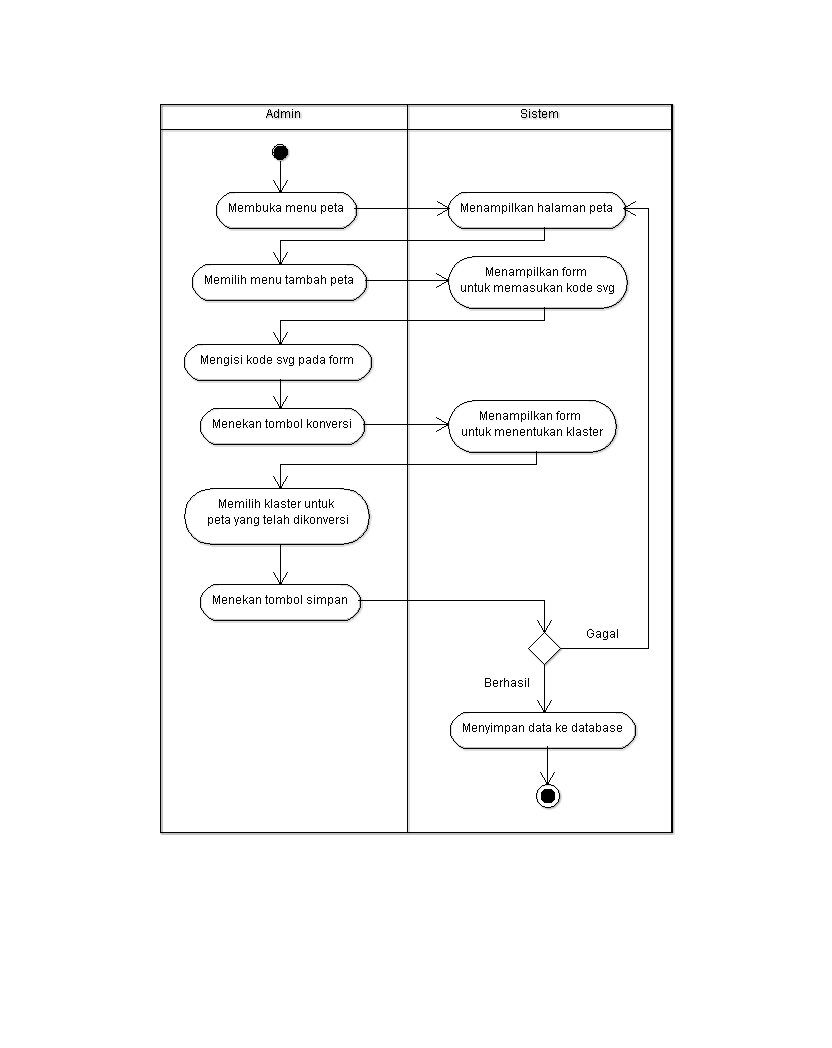
Shalahuddin M, Rosa. 2013. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.

# LAMPIRAN

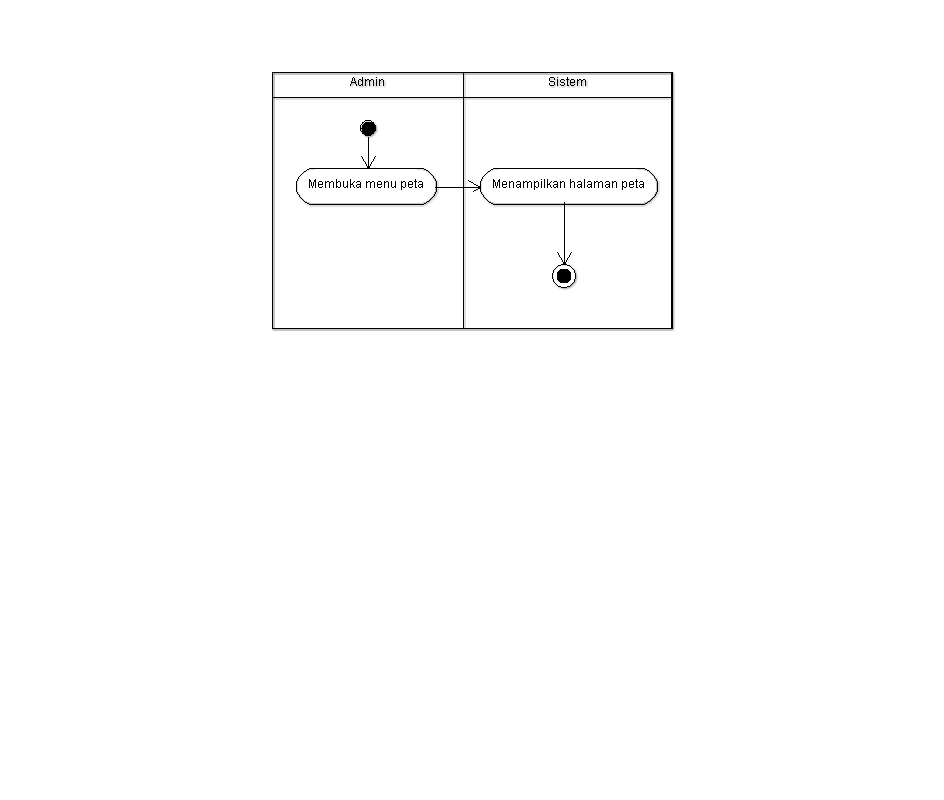
Lampiran 1 Alur proses *login*



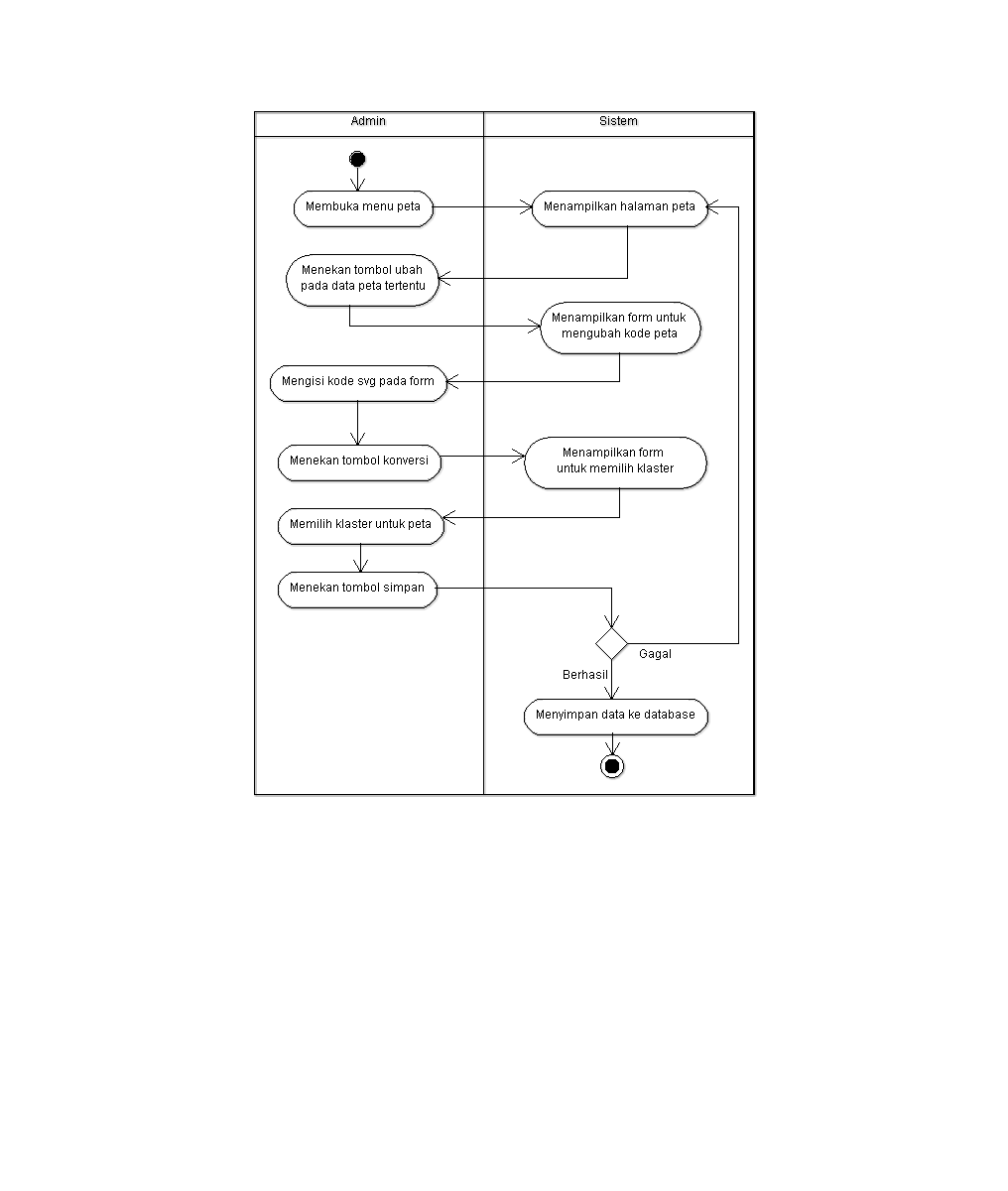
Lampiran 2 Alur proses menambah peta



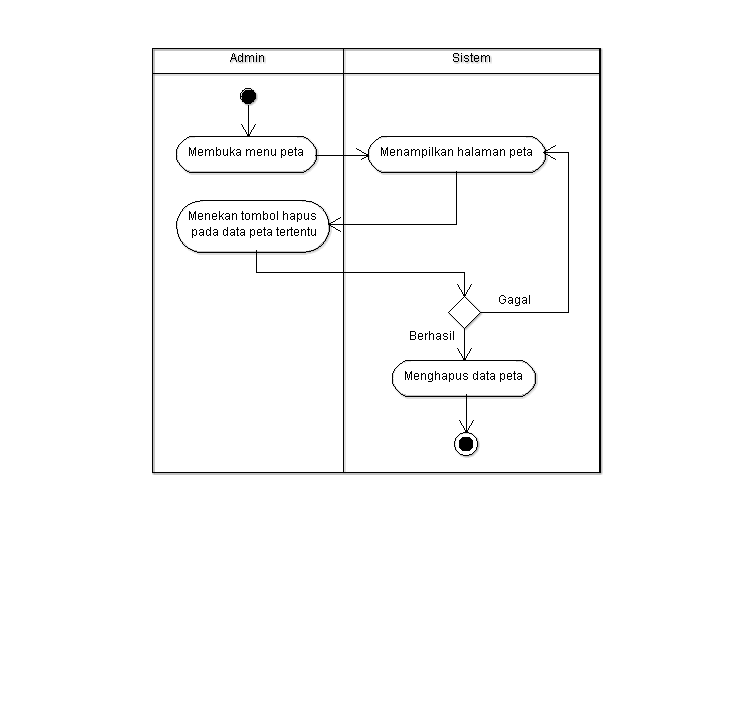
Lampiran 3 Alur proses melihat peta



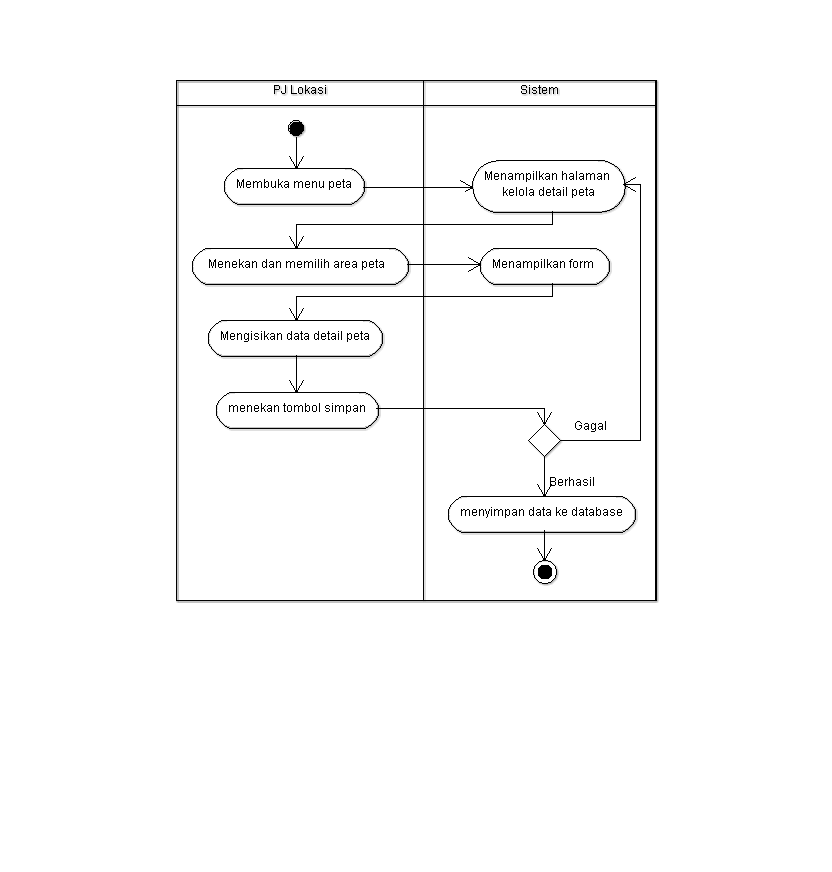
Lampiran 4 Alur proses mengubah peta



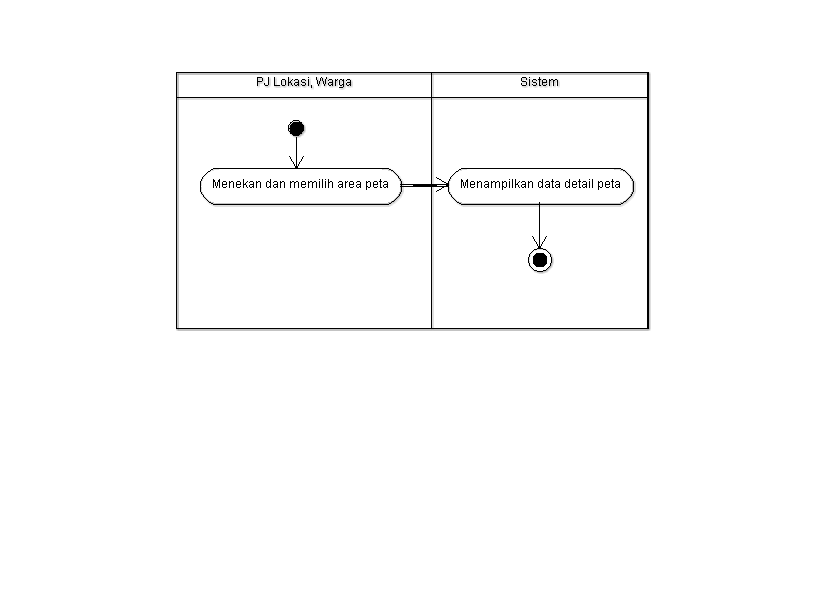
Lampiran 5 Alur proses menghapus peta



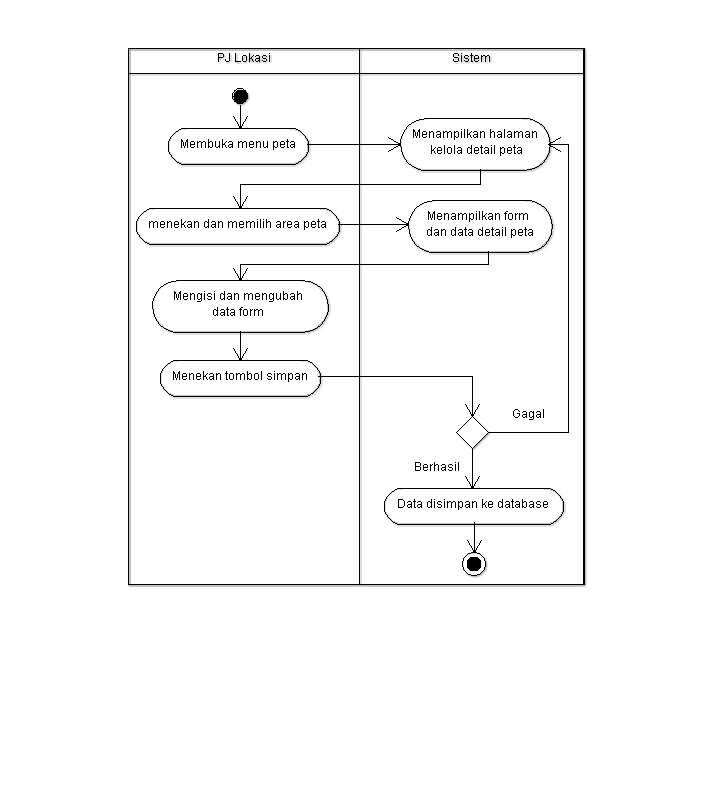
Lampiran 6 Alur proses menambah detail peta



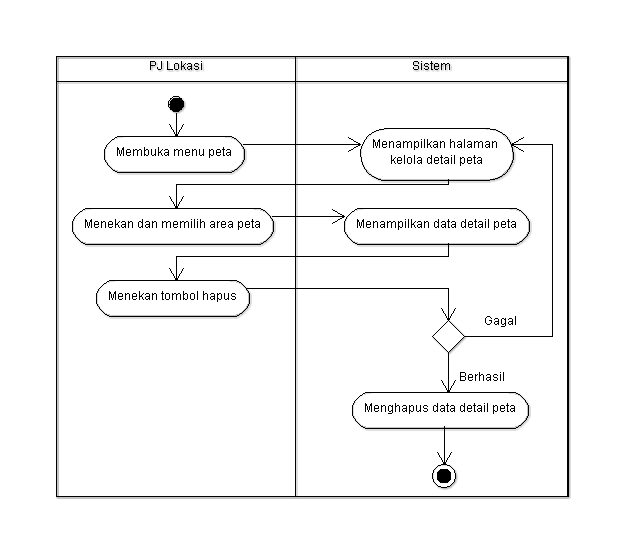
Lampiran 7 Alur proses melihat detail peta



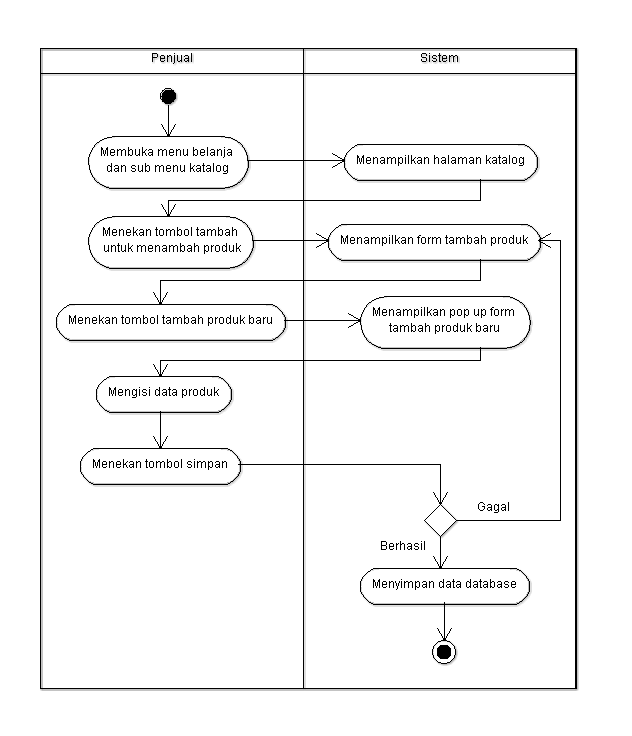
Lampiran 8 Alur proses mengubah detail peta



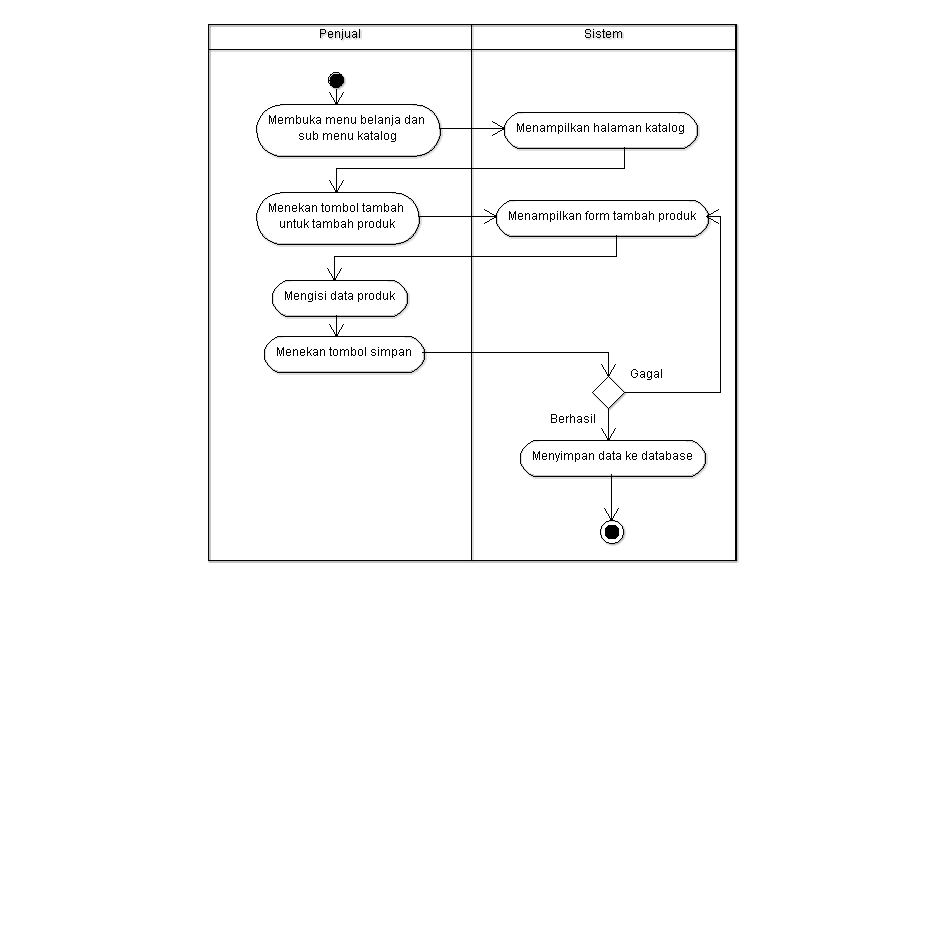
Lampiran 9 Alur proses menghapus detail peta



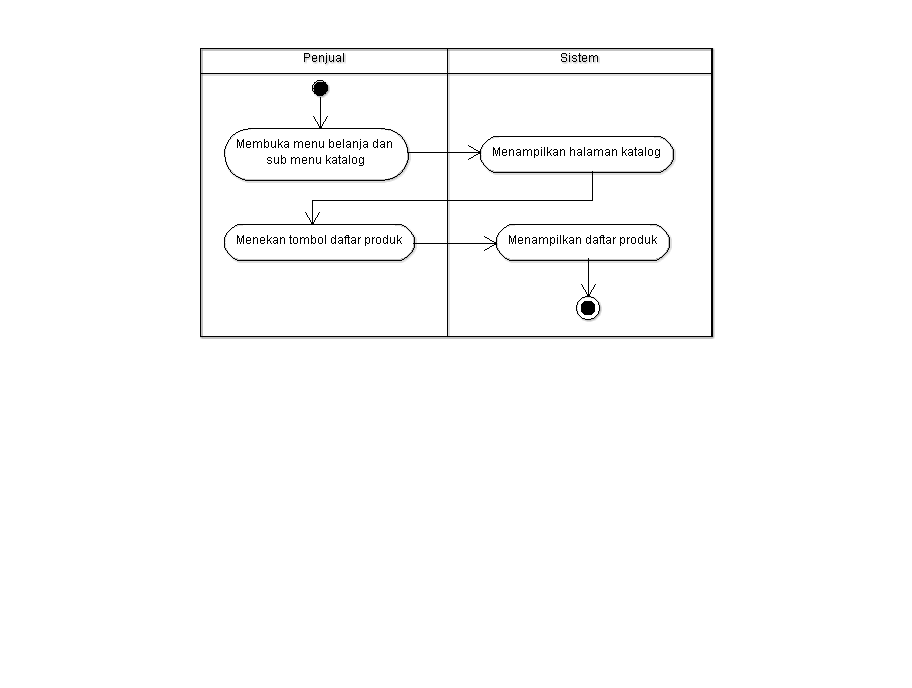
Lampiran 10 Alur proses menambah produk baru



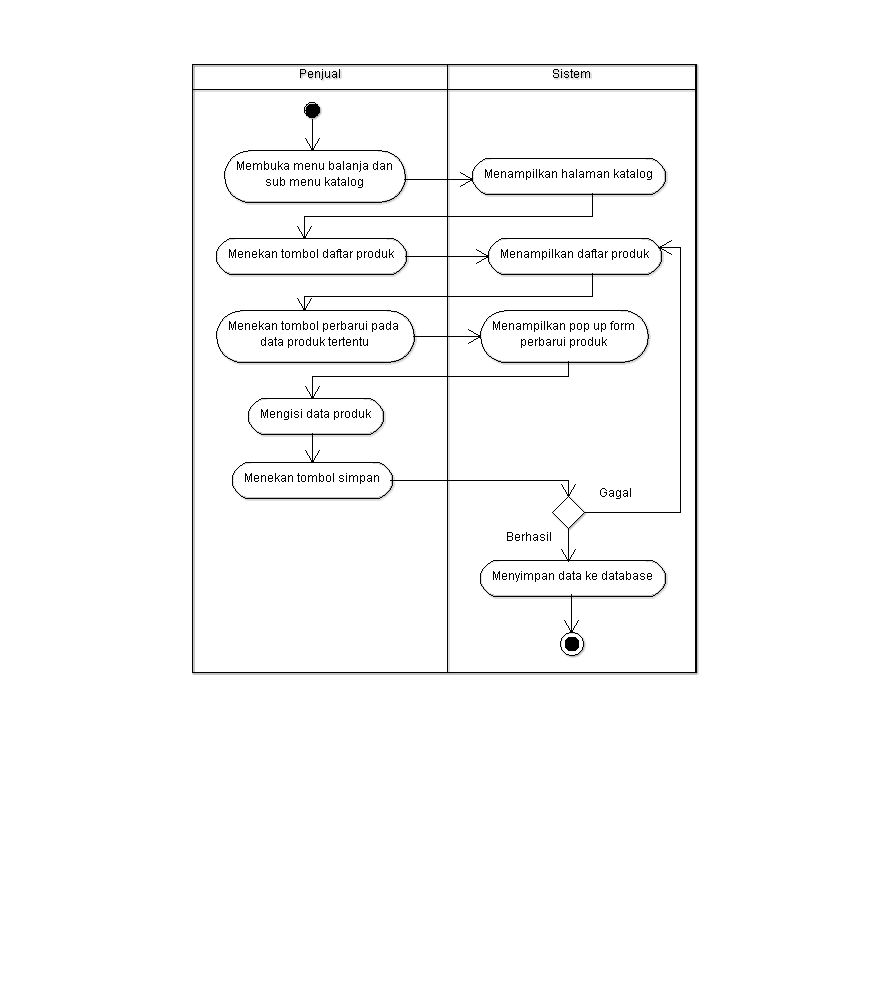
Lampiran 11 Alur proses menambah produk



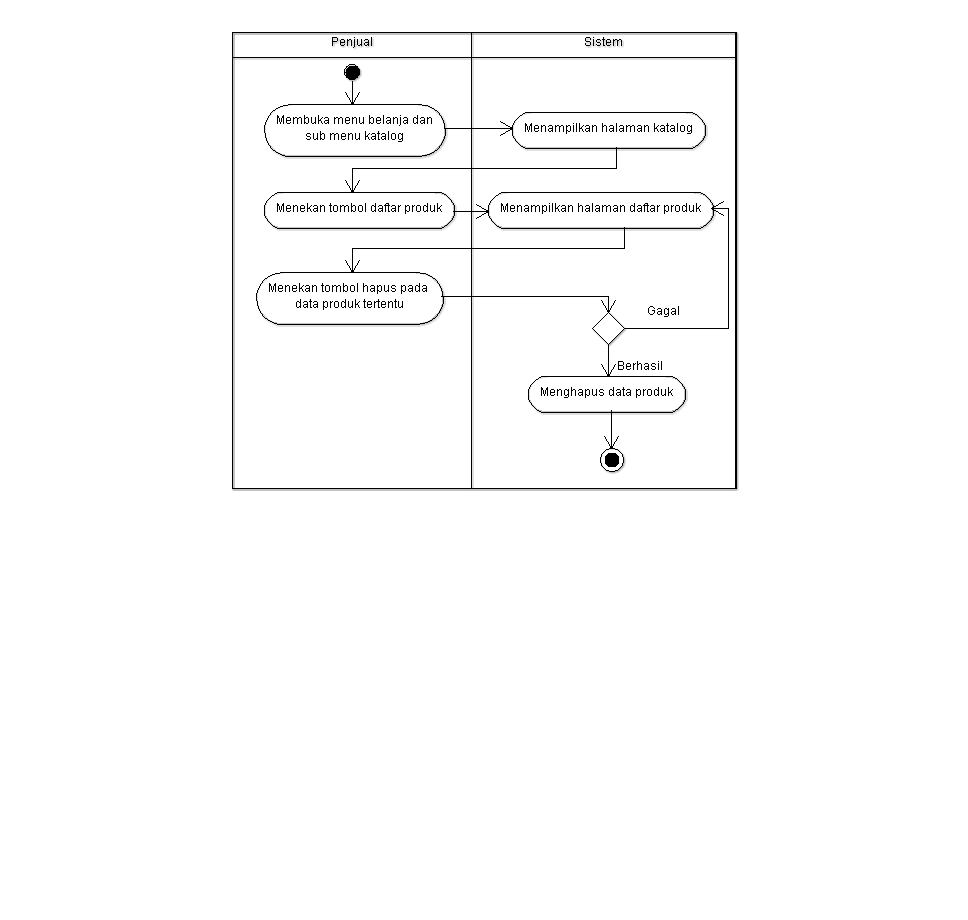
Lampiran 12 Alur proses melihat produk



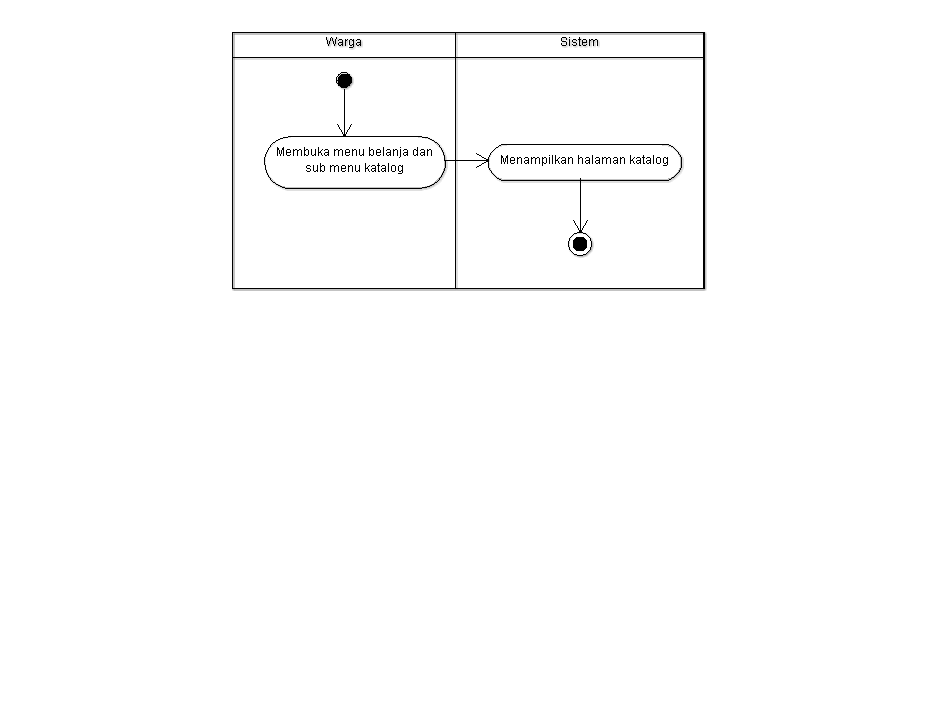
Lampiran 13 Alur proses memperbarui produk



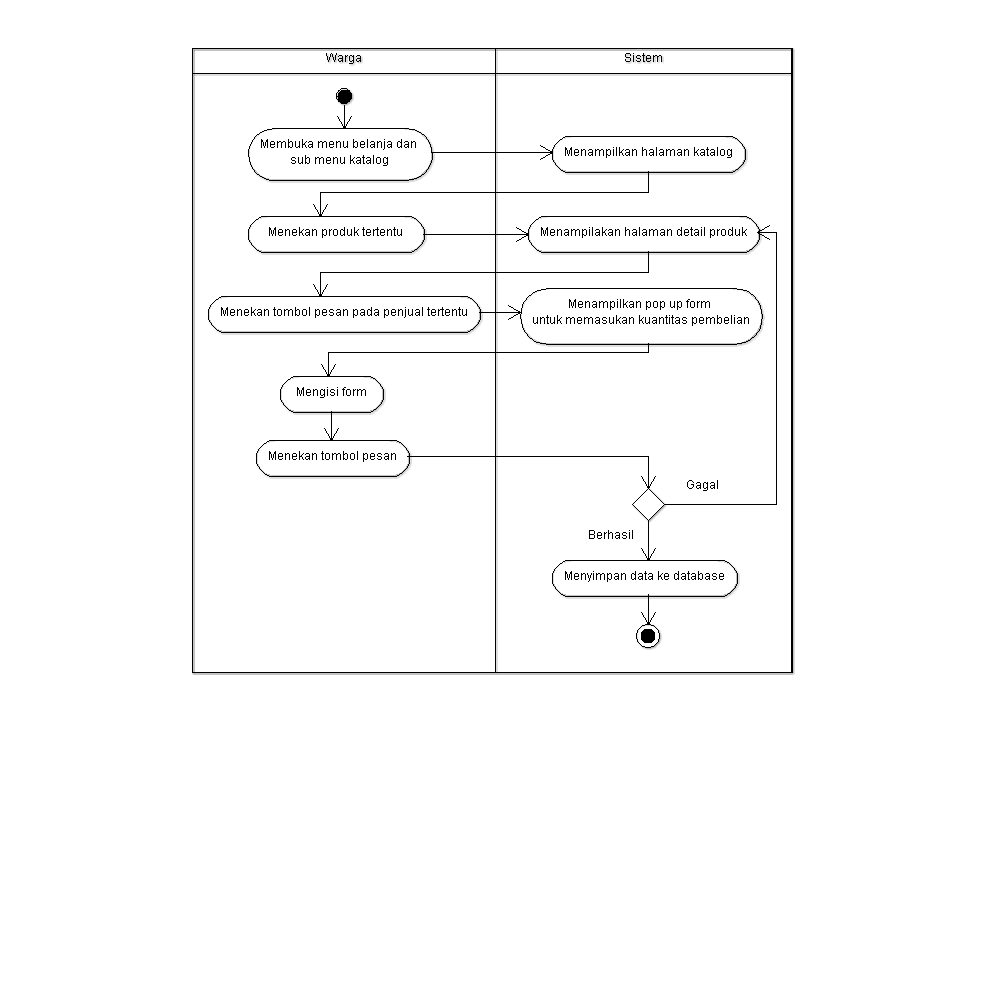
Lampiran 14 Alur proses menghapus produk



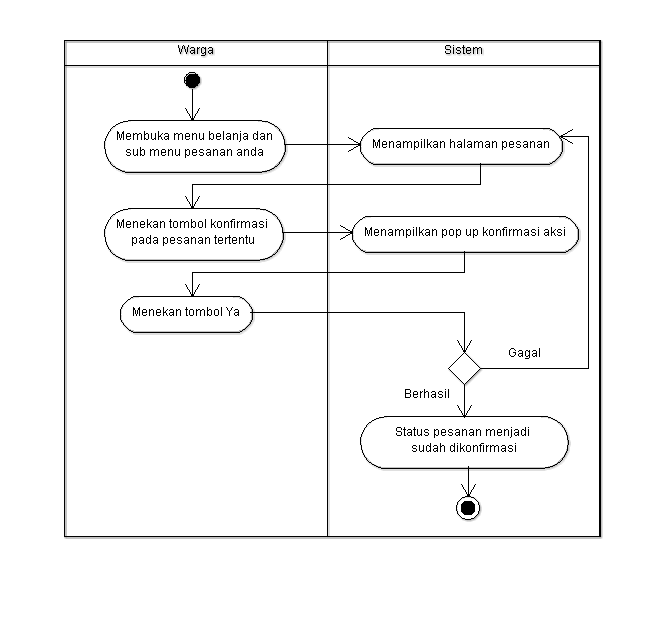
Lampiran 15 Alur proses melihat katalog



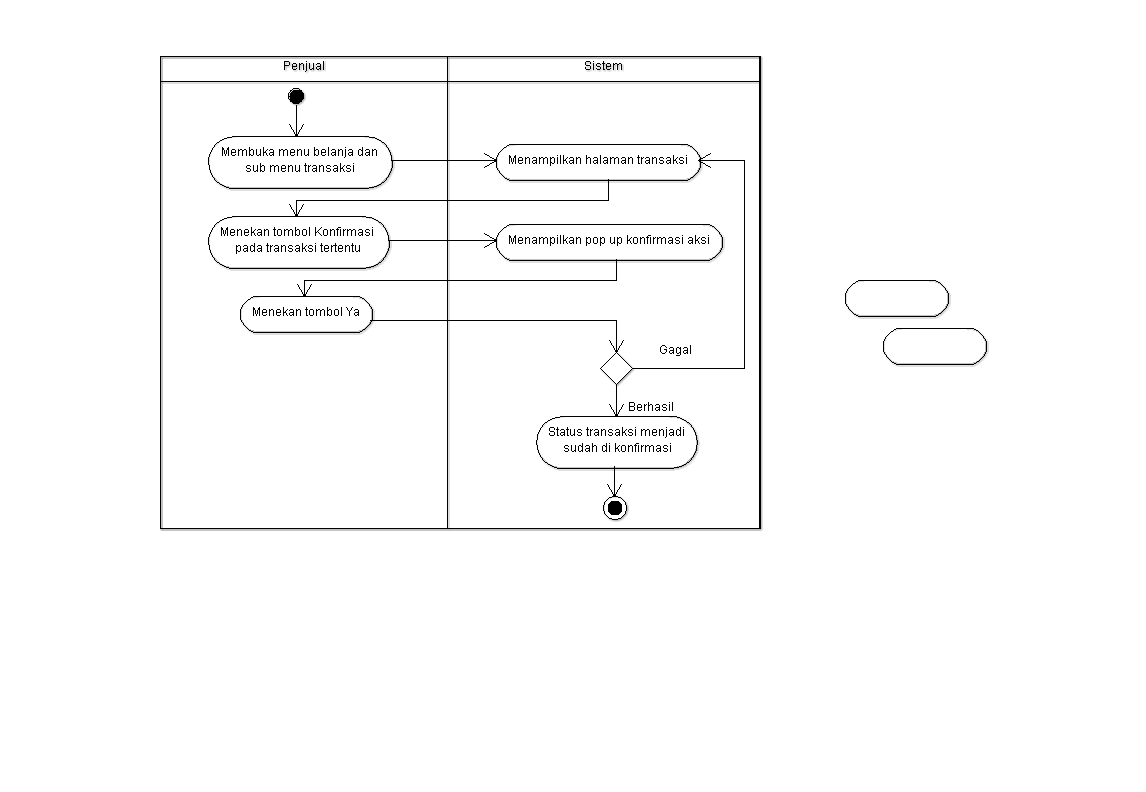
Lampiran 16 Alur proses membeli produk



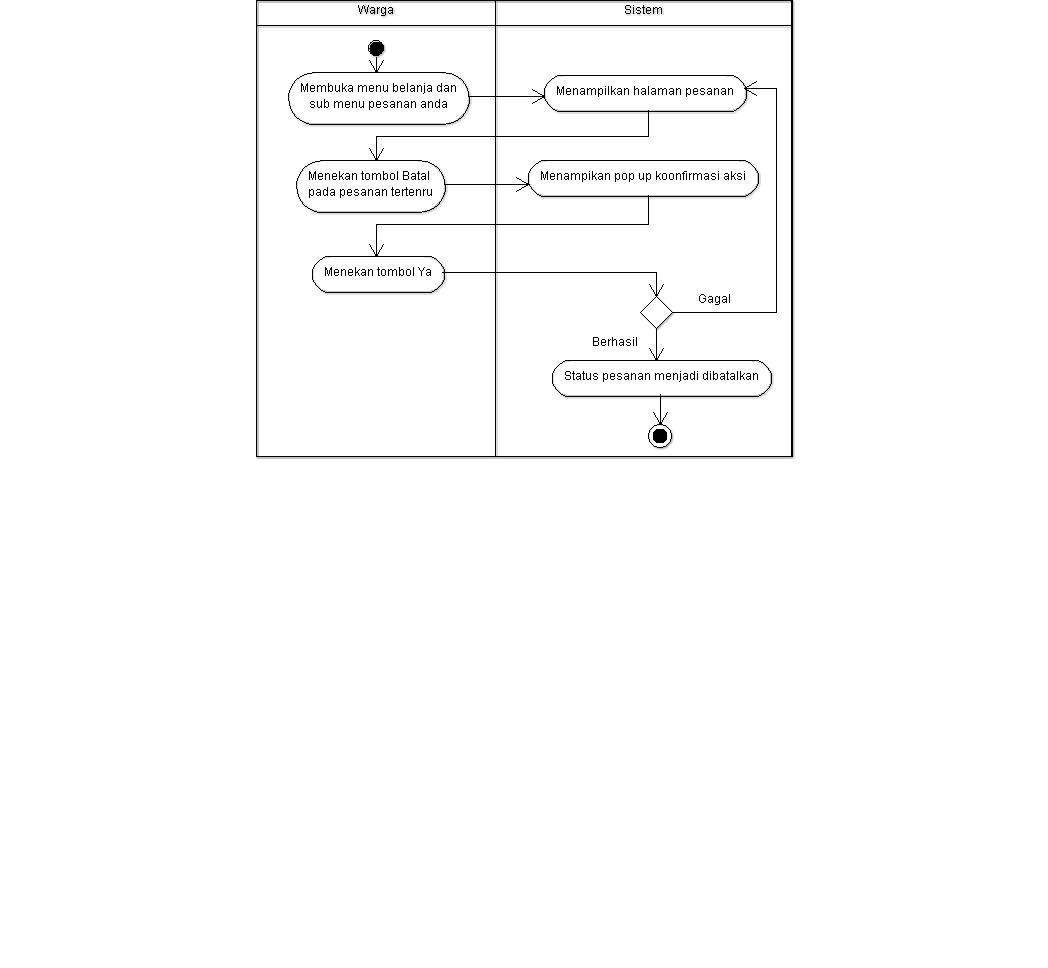
Lampiran 17 Alur proses konfirmasi pesanan



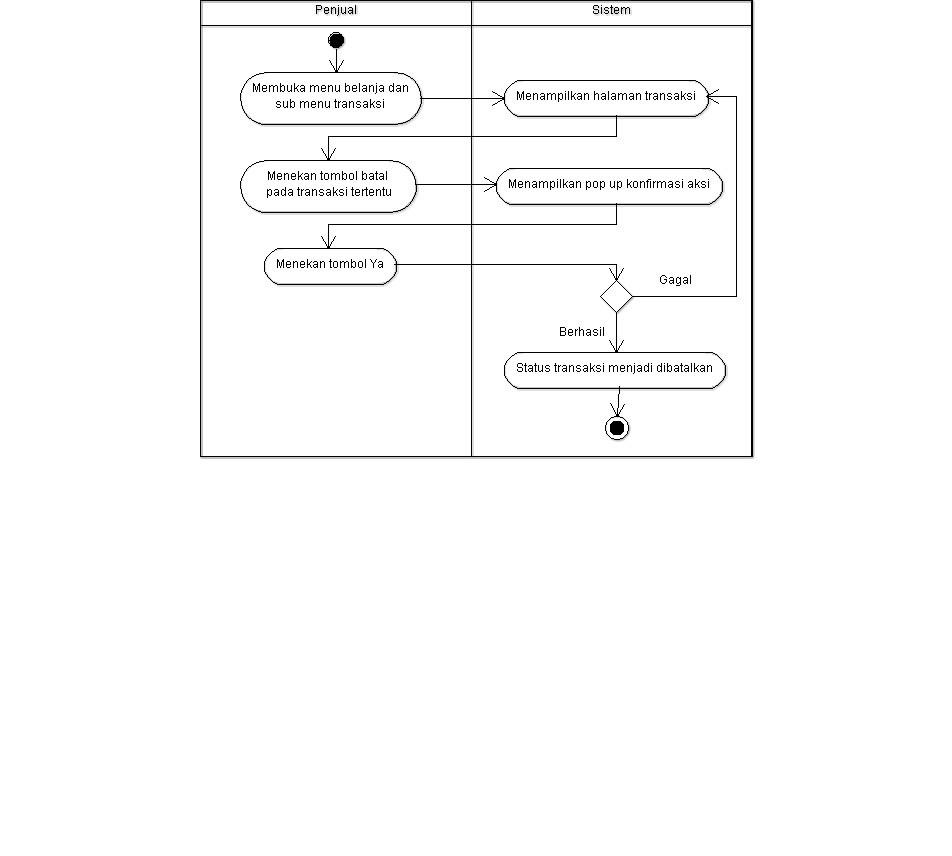
Lampiran 18 Alur proses konfirmasi transaksi



Lampiran 19 Alur proses membatalkan pesanan



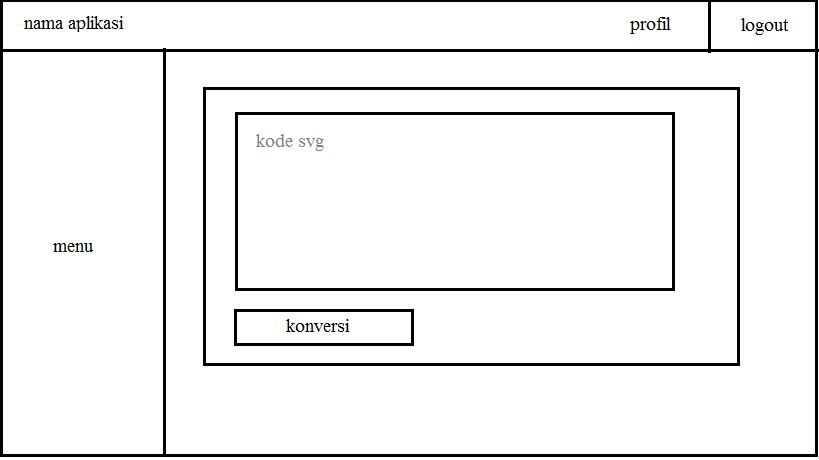
Lampiran 20 Alur proses membatalkan transaksi



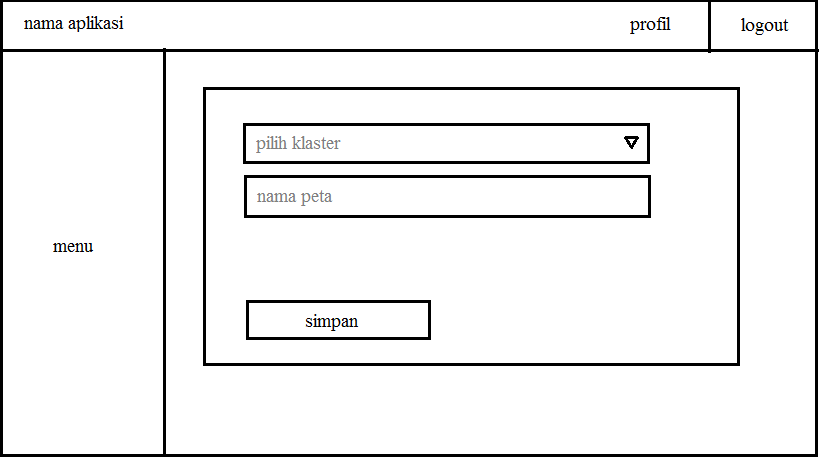
Lampiran 21 Perancangan masukan *login*



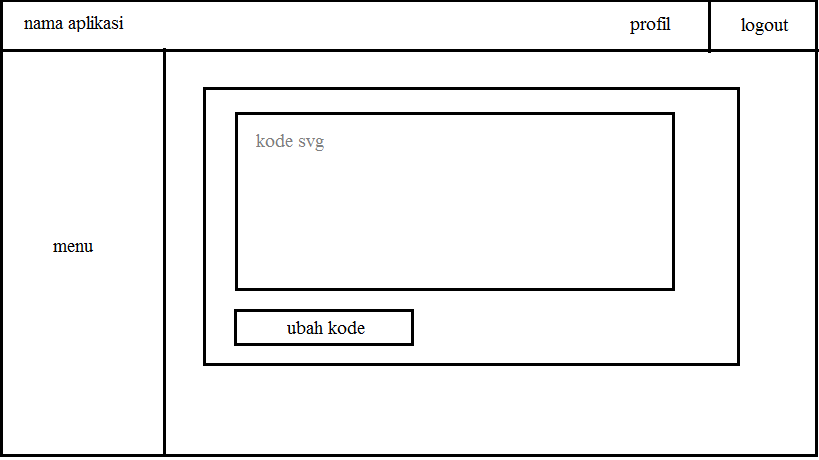
Lampiran 22 Perancangan masukan konversi peta bagian pertama



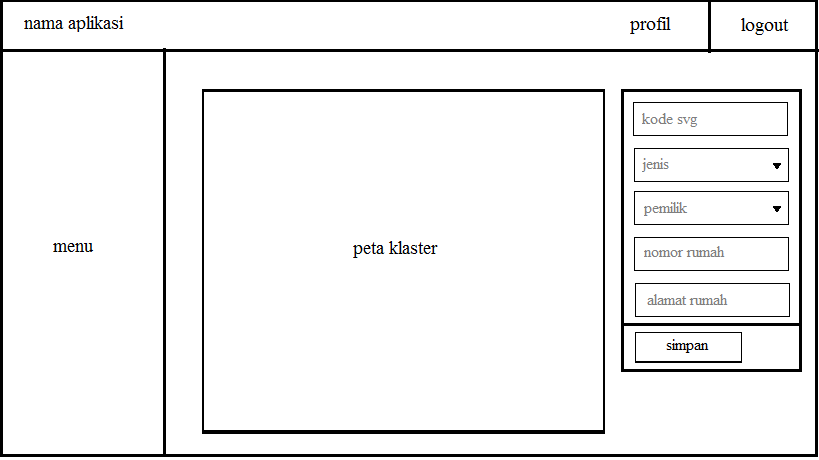
Lampiran 23 Perancangan masukan konversi peta bagian kedua



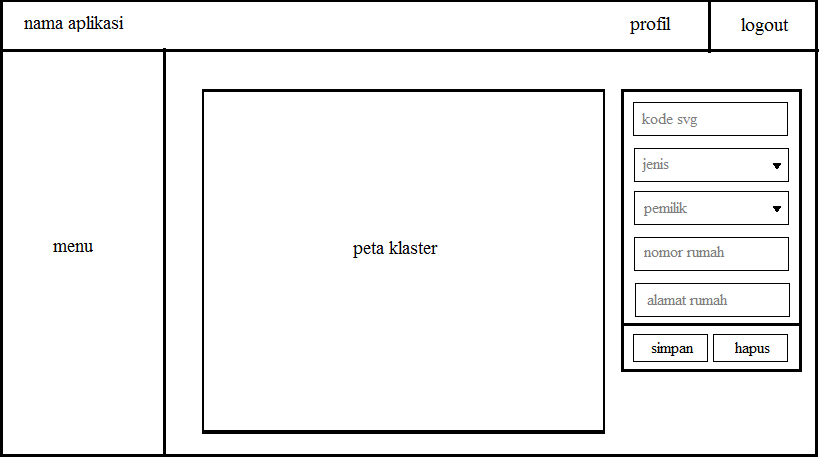
Lampiran 24 Perancangan masukan ubah peta



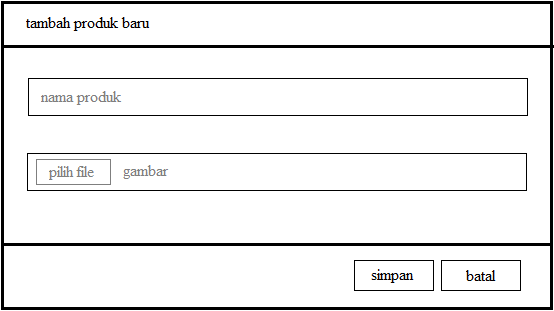
Lampiran 25 Perancangan masukan tambah detail peta



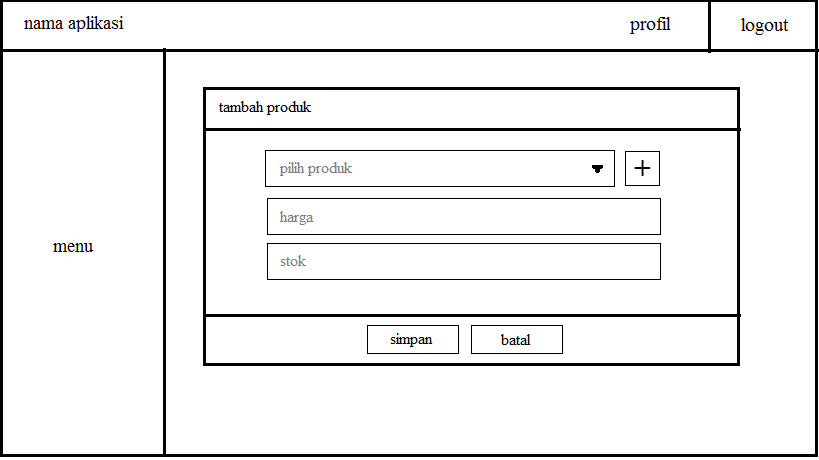
Lampiran 26 Perancangan masukan ubah detail peta



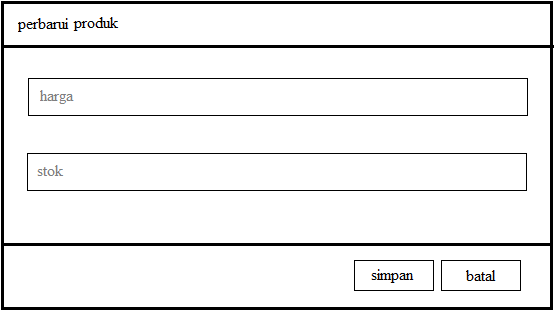
Lampiran 27 Perancangan masukan tambah produk baru



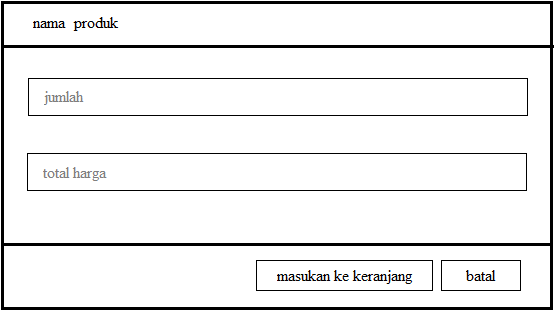
Lampiran 28 Perancangan masukan tambah produk



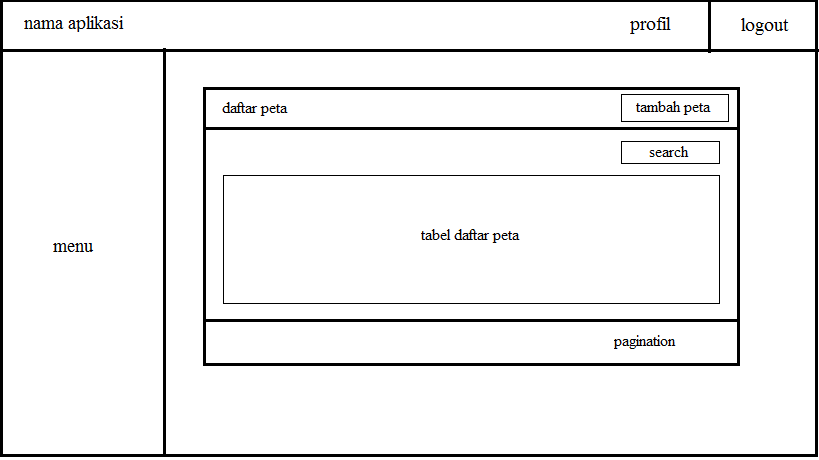
Lampiran 29 Perancangan masukan perbarui produk



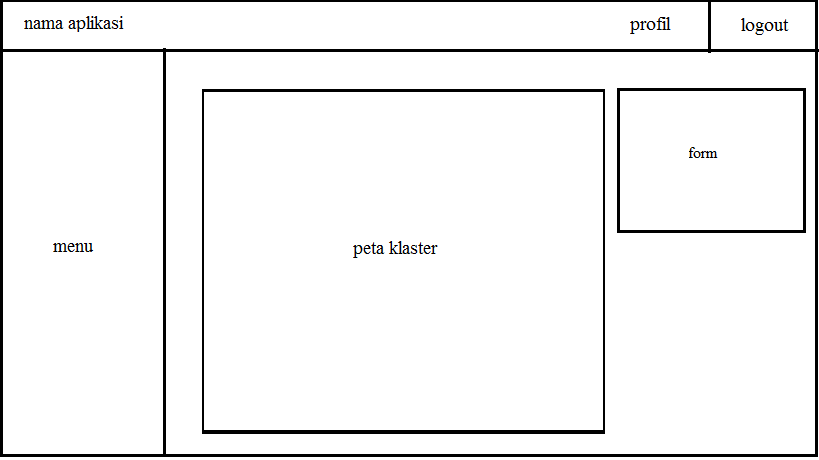
Lampiran 30 Perancangan masukan masukan kuantitas pesanan



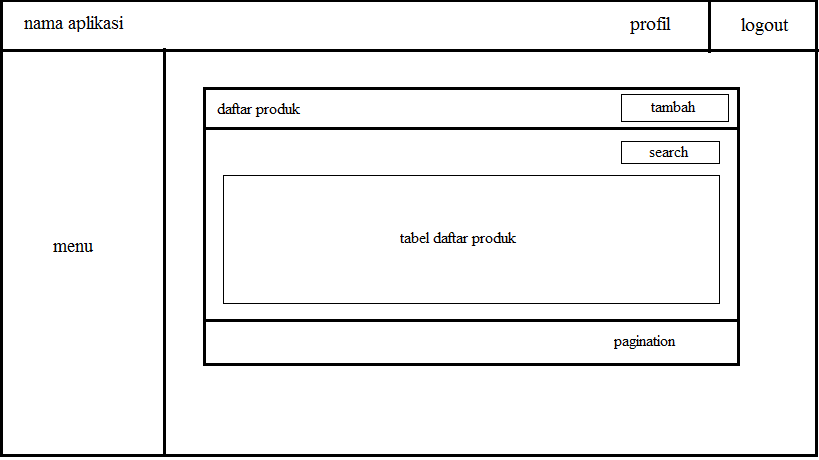
Lampiran 31 Perancangan keluaran informasi peta



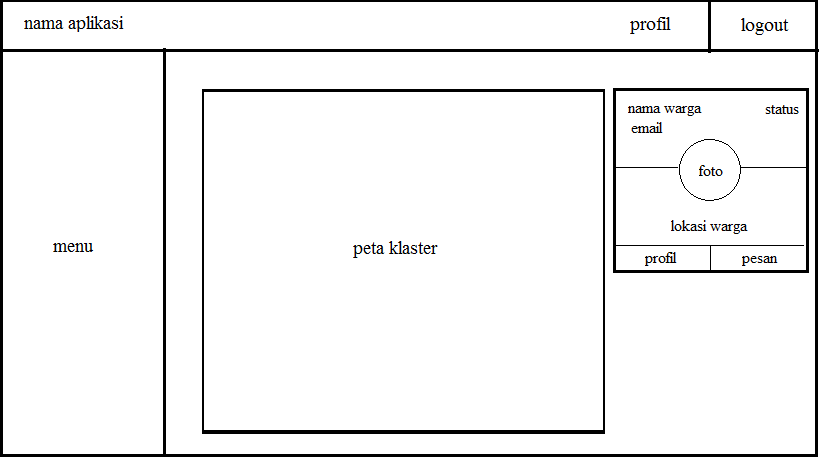
Lampiran 32 Perancangan keluaran informasi detail peta



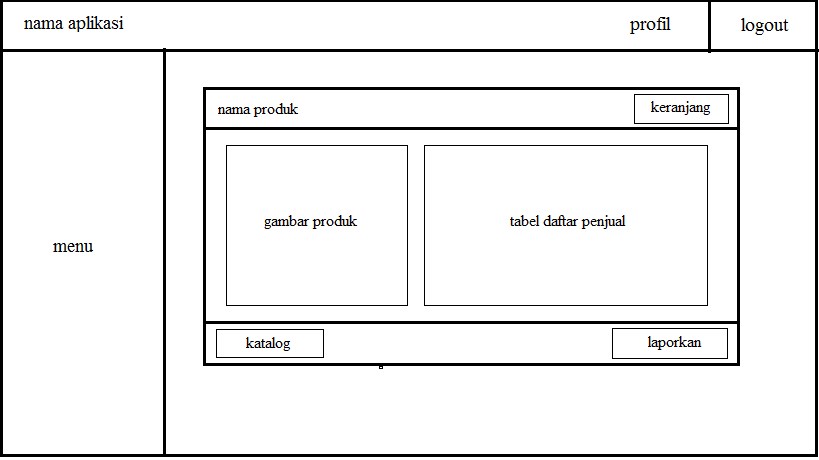
Lampiran 33 Perancangan keluaran informasi produk



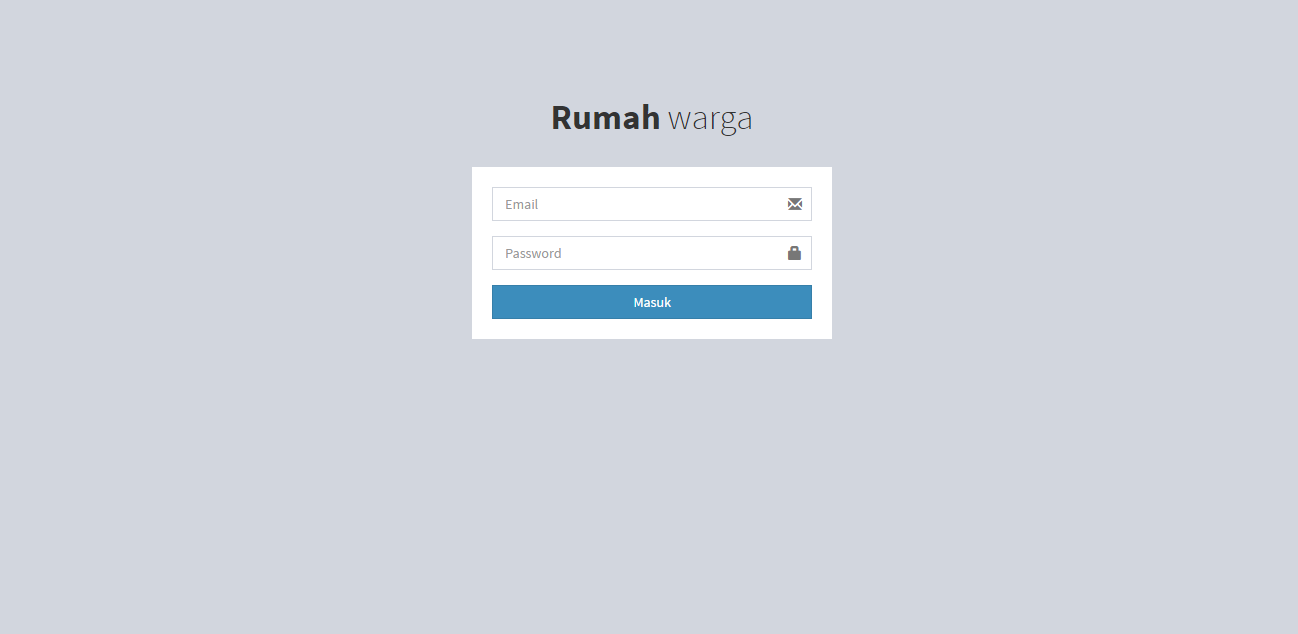
Lampiran 34 Perancangan keluaran informasi peta klaster



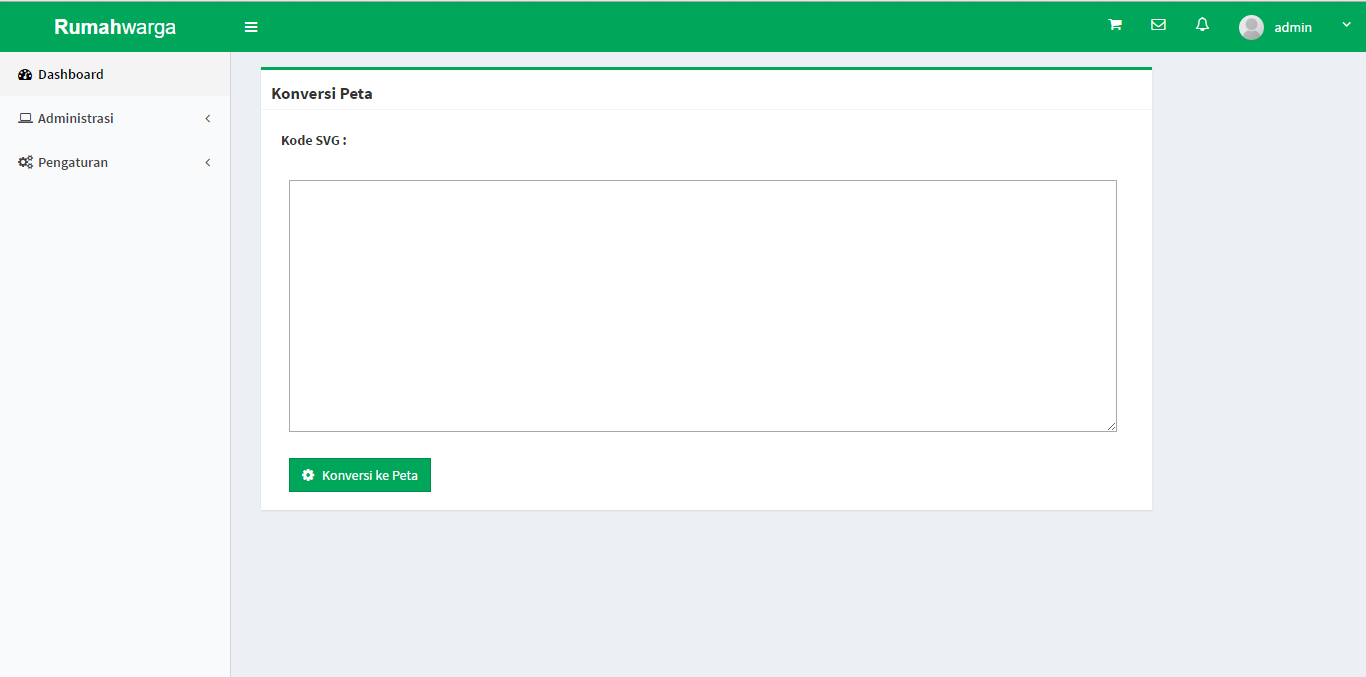
Lampiran 35 Perancangan keluaran informasi detail produk



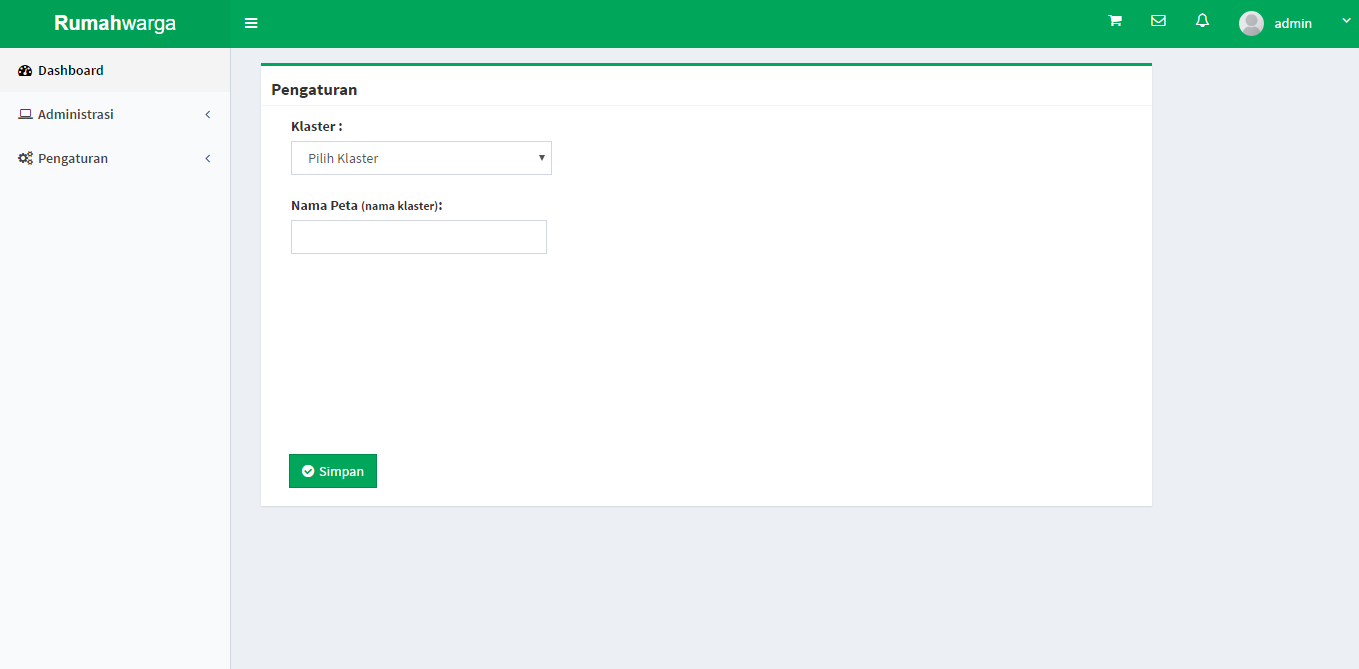
Lampiran 36 Implementasi masukan *login*



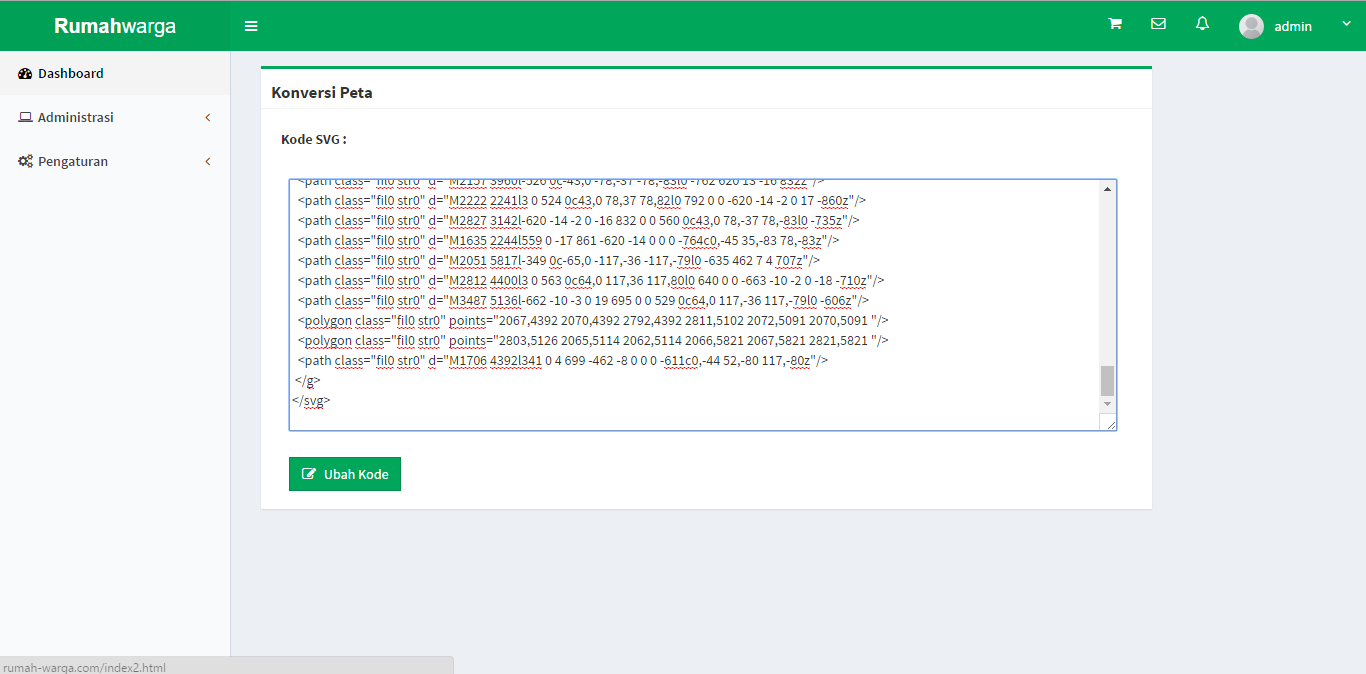
Lampiran 37 Implementasi masukan konversi peta bagian pertama



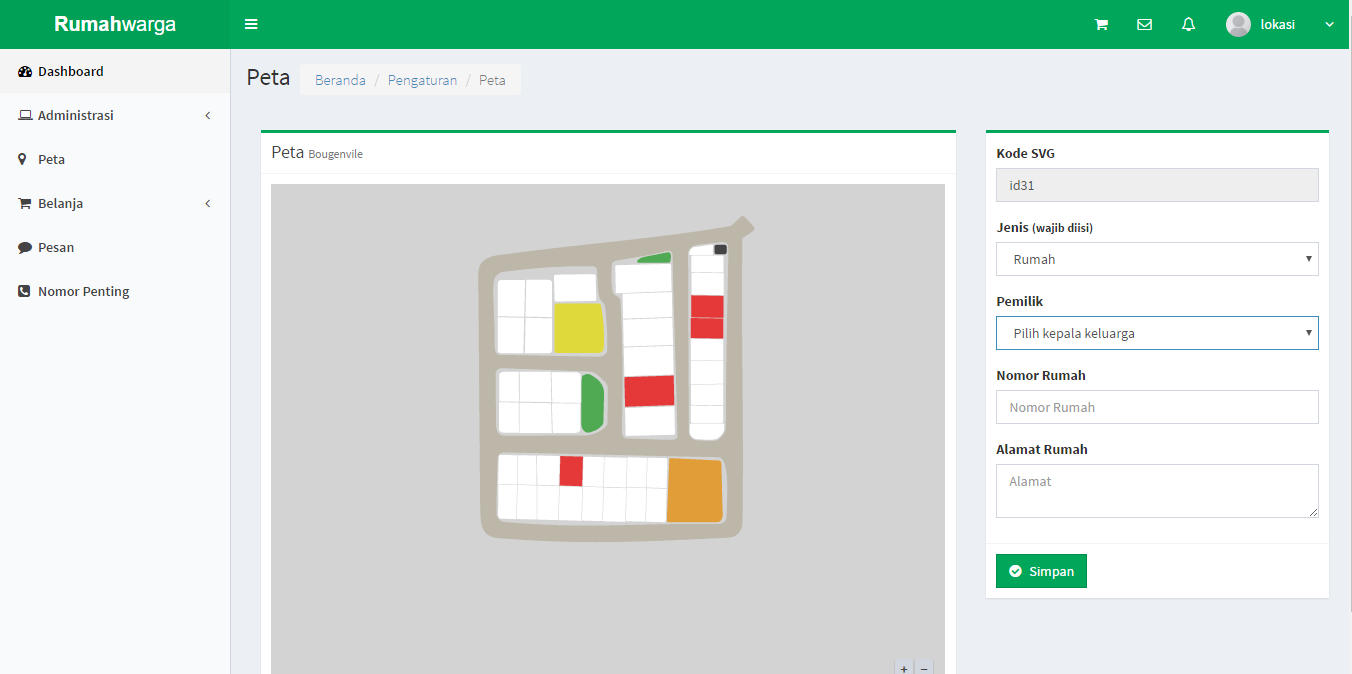
Lampiran 38 Implementasi masukan konversi peta bagian kedua



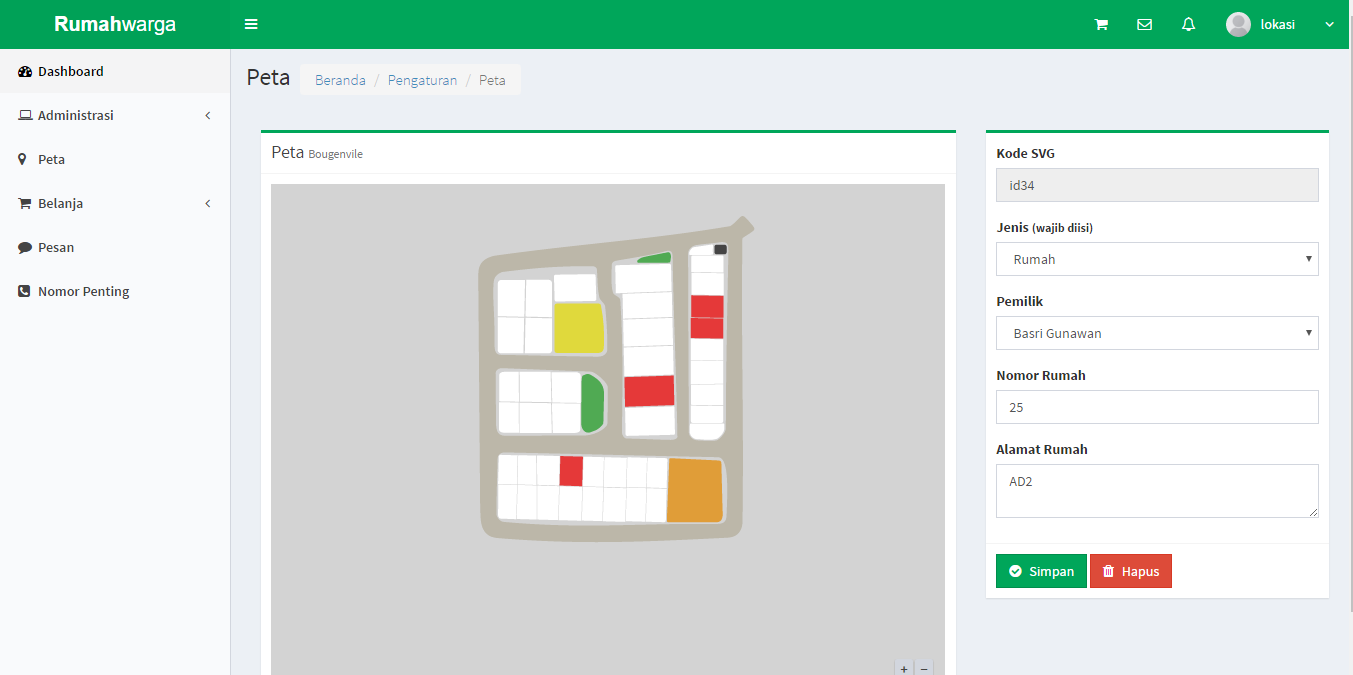
Lampiran 39 Implementasi masukan ubah peta



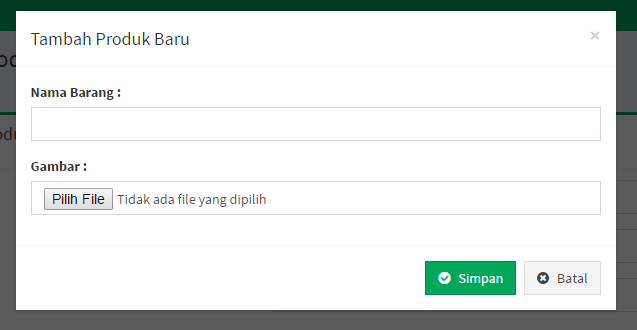
Lampiran 40 Implementasi masukan tambah detail peta



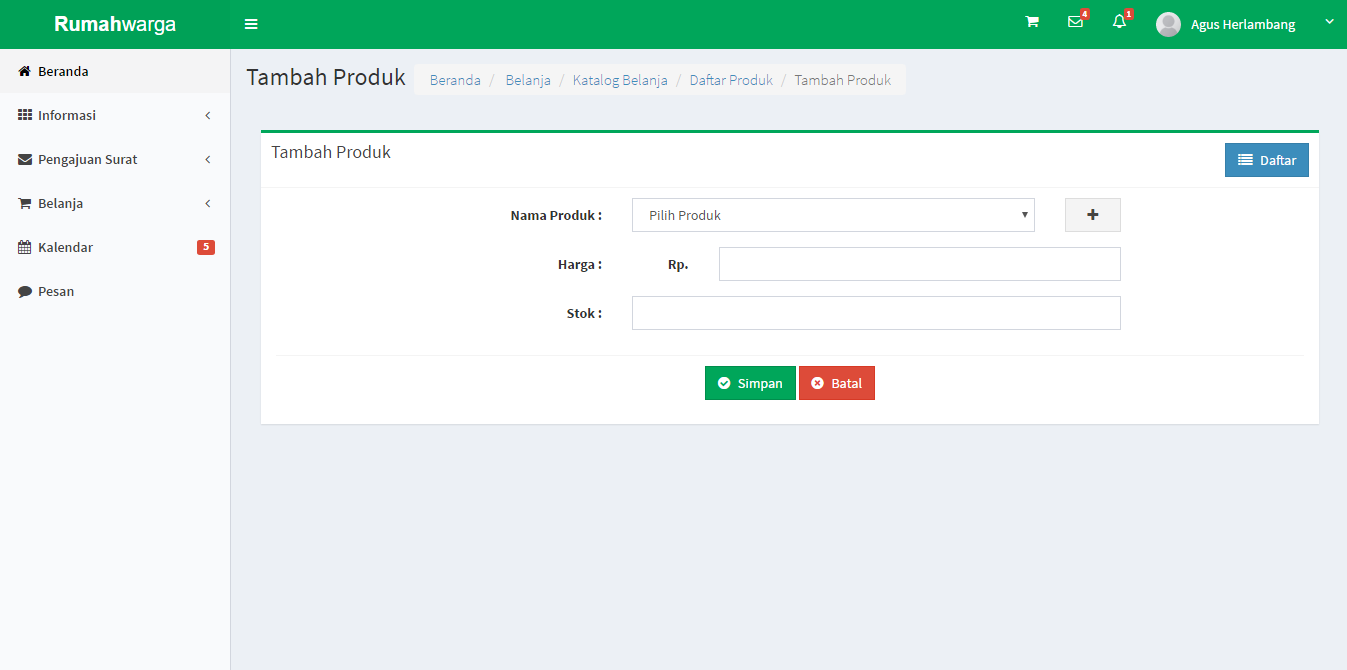
Lampiran 41 Implementasi masukan ubah detail peta



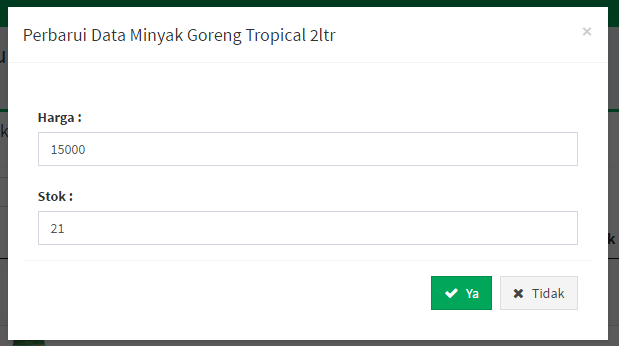
Lampiran 42 Implementasi masukan tambah produk baru



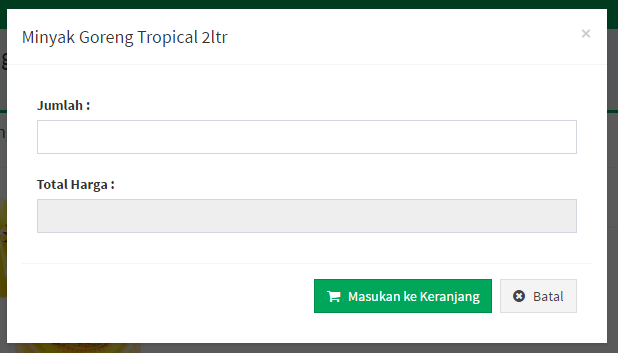
Lampiran 43 Implementasi masukan tambah produk



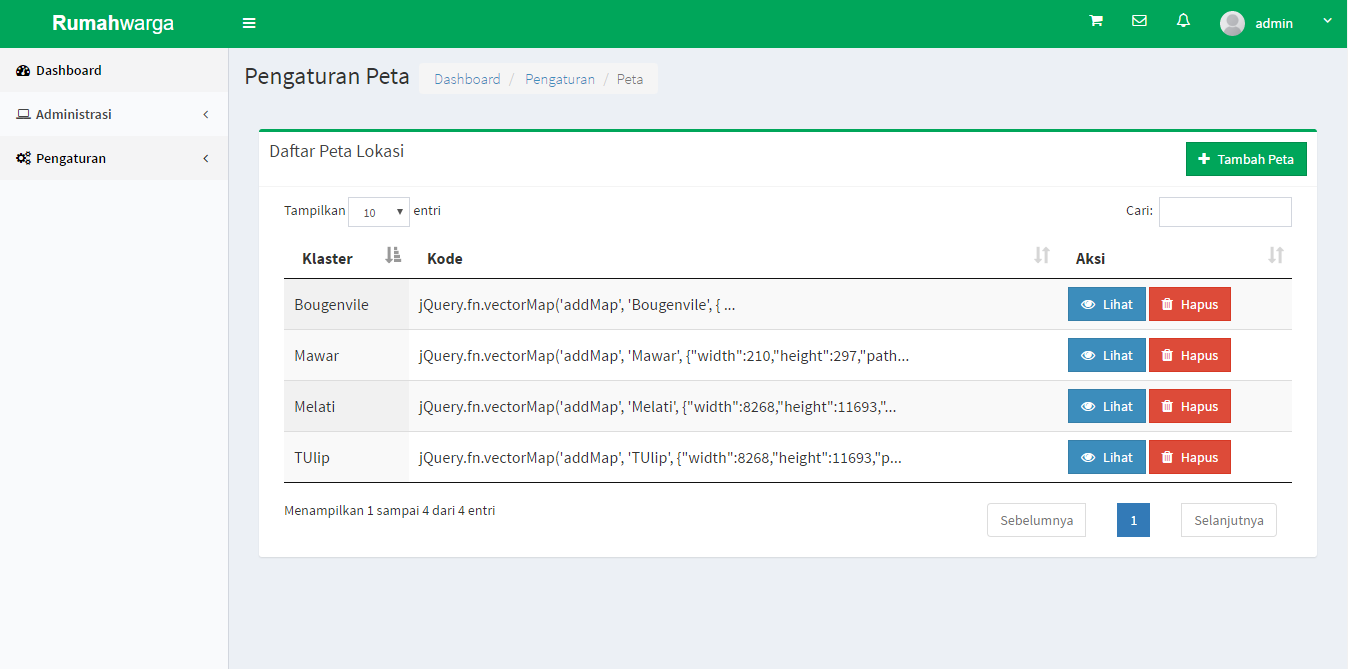
Lampiran 44 Implementasi masukan perbarui produk



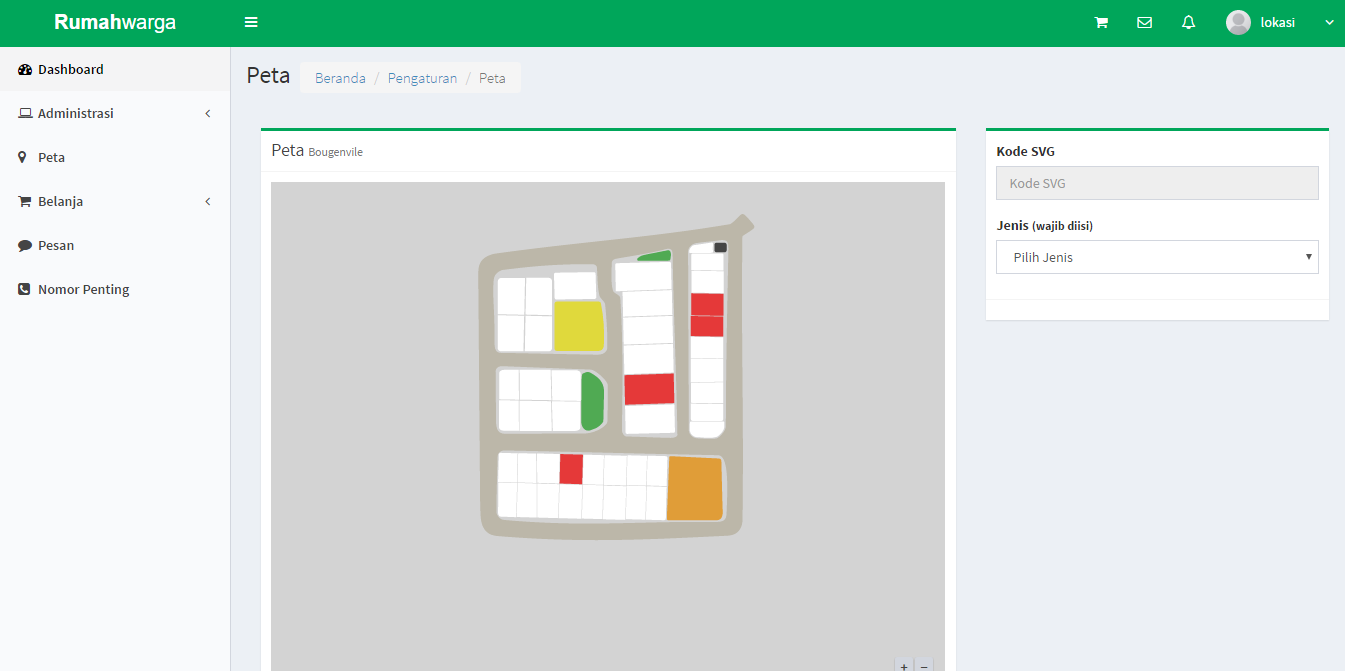
Lampiran 45 Implementasi masukan masukan kuantitas pesanan



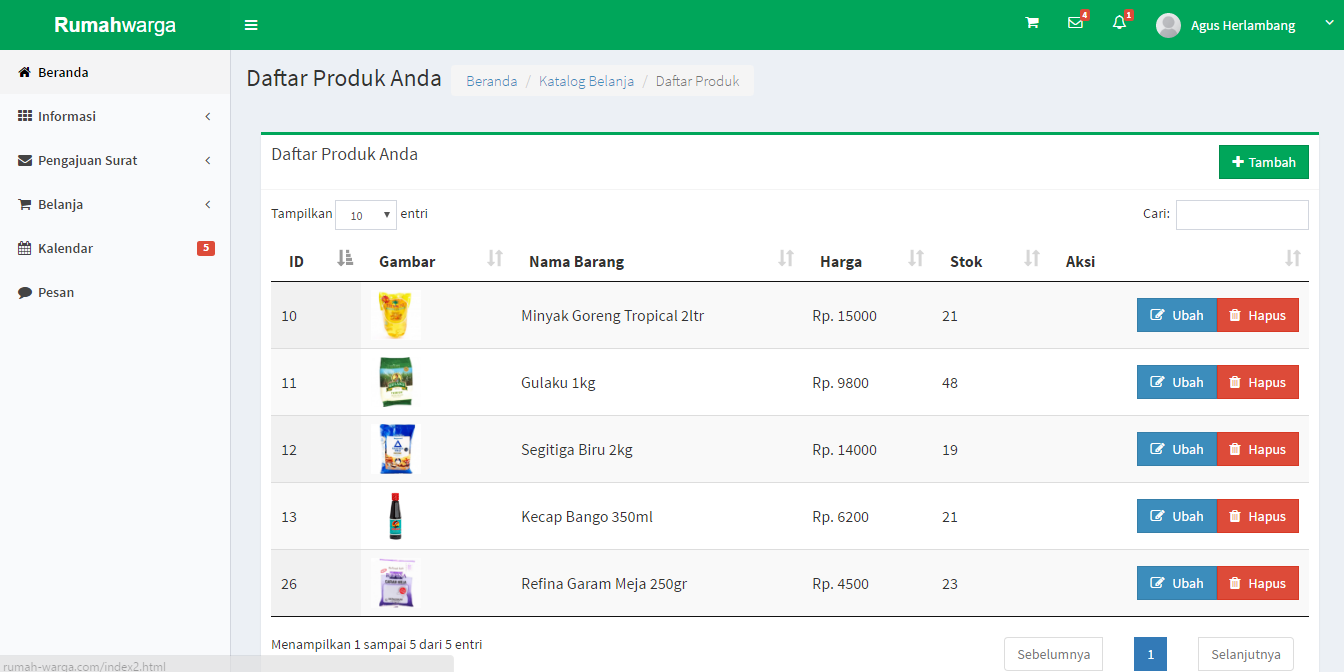
Lampiran 46 Implementasi keluaran informasi peta



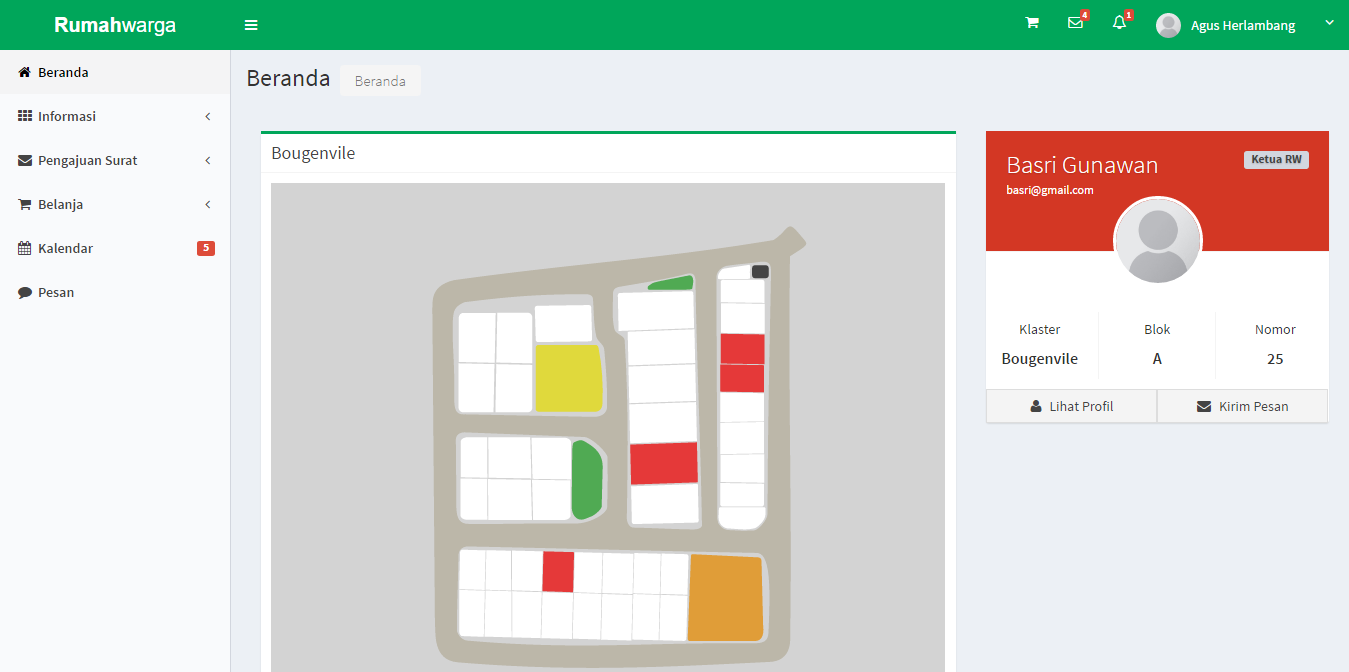
Lampiran 47 Implementasi keluaran informasi detail peta



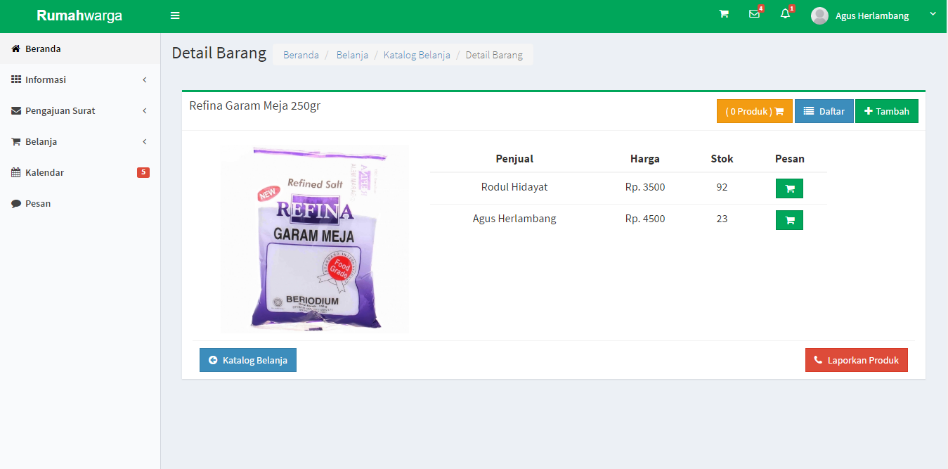
Lampiran 48 Implementasi keluaran informasi produk



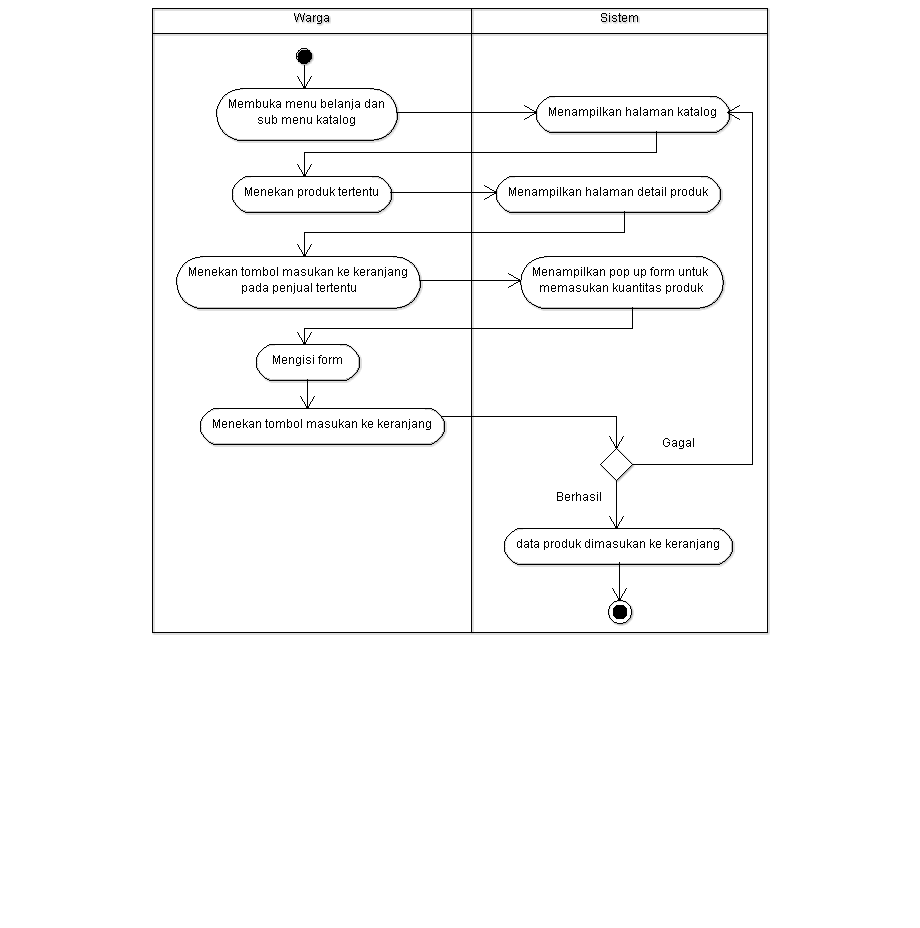
Lampiran 49 Implementasi keluaran informasi peta klaster



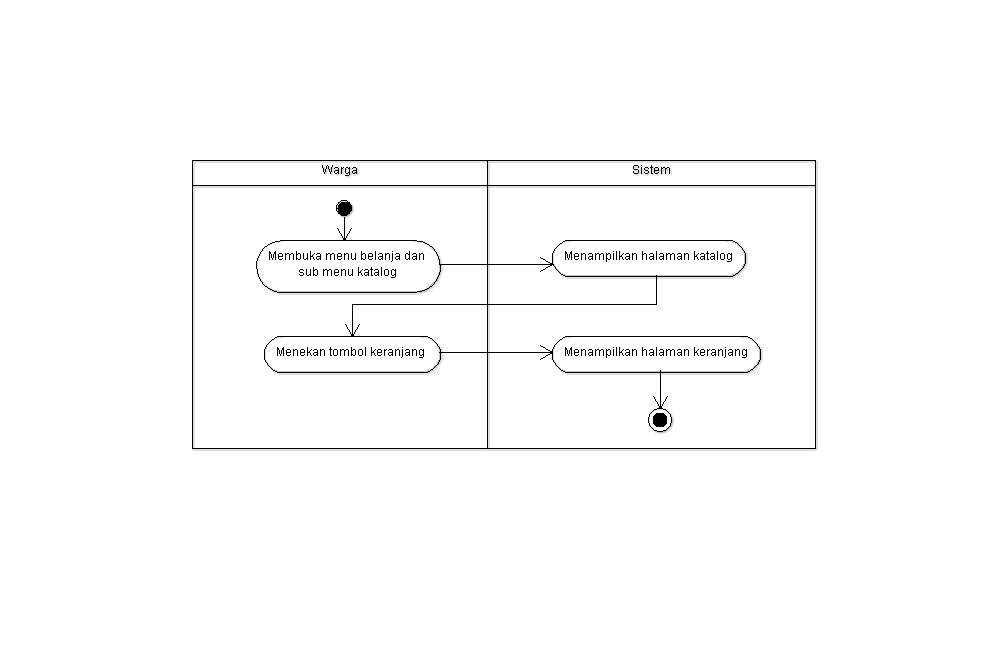
Lampiran 50 Implementasi keluaran informasi detail produk



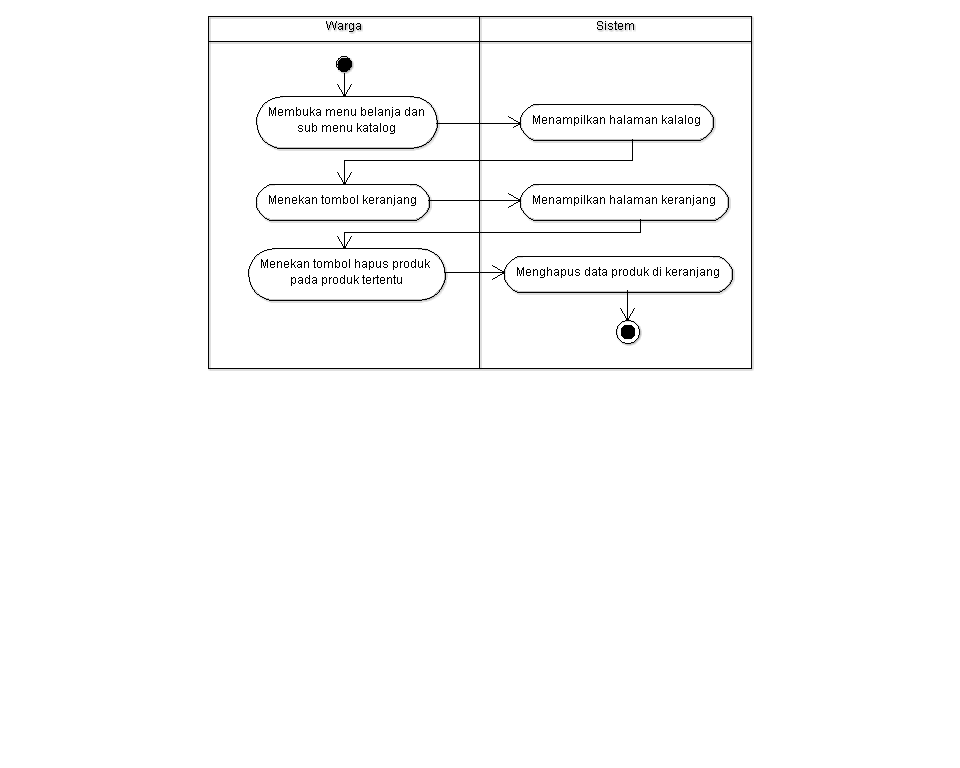
Lampiran 51 Alur proses menambah produk ke keranjang



Lampiran 52 Alur proses malihat keranjang



Lampiran 53 Alur proses menghapus produk di keranjang



**RIWAYAT HIDUP**

Penulis bernama lengkap Nugroho Tri Pambudi, lahir di ogor pada tanggal 18 September 1995. Anak ketiga dari tiga bersaudara pasangan Wardi dan Wasikem.

Penulis memulai pendidikan pada tahun 2001 di Sekolah Dasar (SD) Negeri Laladon 3 Bogor dan lulus pada Tahun 2007. Kemudian melanjutkan pendidikan di Sekolah Menengah Pertama (SMP) Pembangunan 1 Bogor dan lulus pada tahun 2010 Setelah itu melanjutkan pendidikan di Sekolah Mengah Atas (SMA) Negeri 10 Bogor dan lulus pada tahun 2013. Tahun 2013 penulis diterima sebagai mahasiswa Program Diploma Institut Pertanian Bogor melalui jalur Reguler pada Program Keahlian Manajemen Informatika.