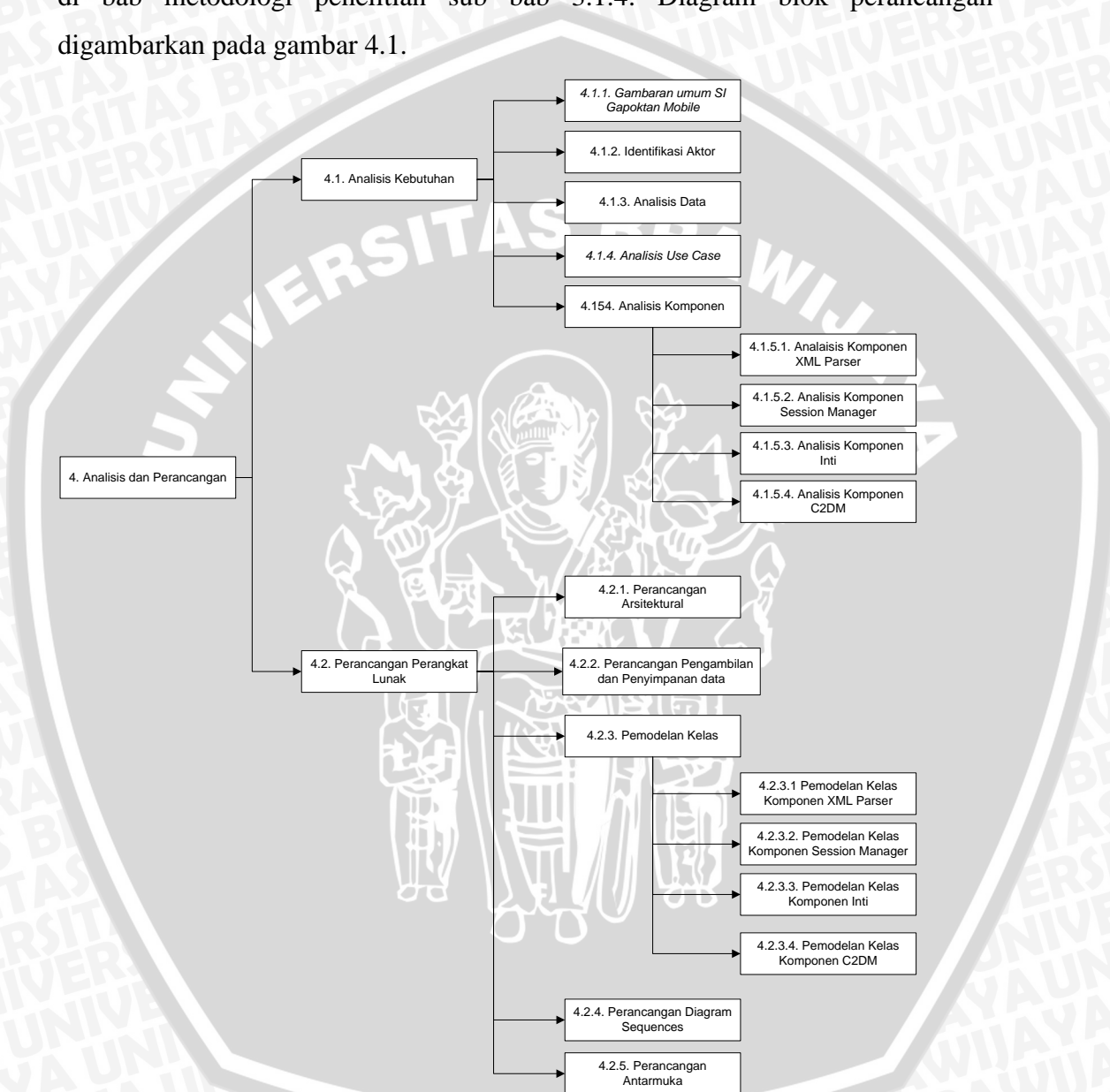


BAB IV ANALISIS DAN PERANCANGAN

Bab ini membahas mengenai perancangan perangkat lunak SI Gapoktan *Mobile*. Perancangan dilakukan dengan model CBSE seperti yang telah dijelaskan di bab metodologi penelitian sub bab 3.1.4. Diagram blok perancangan digambarkan pada gambar 4.1.



Gambar 4.1. Diagram blok perancangan
Sumber : [Perancangan]

Tahap analisis kebutuhan terdiri atas lima langkah tahap. Tahap pertama melakukan penjabaran tentang gambaran umum perangkat lunak SI Gapoktan *Mobile*. Pada tahap kedua melakukan proses identifikasi aktor yang terlibat dalam sistem perangkat lunak. Tahap ketiga melakukan proses analisis data yang



diperlukan. Tahap keempat melakukan analisis *use case* berdasarkan penjabaran gambaran umum perangkat lunak. Tahap kelima melakukan pembagian sistem perangkat lunak menjadi empat komponen yaitu komponen XML Parser, komponen session manager, komponen inti, dan komponen GCDM.

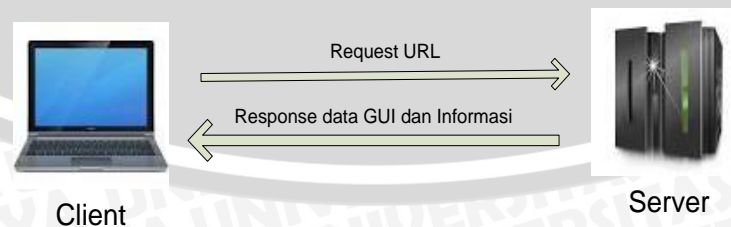
Desain sistem perangkat lunak memiliki empat tahap, yaitu perancangan arsitektural, pemodelan kelas untuk menggambarkan perancangan struktur *class-class* yang menyusun perangkat lunak SI Gapoktan *Mobile*, perancangan *sequences diagram* dan perancangan antarmuka pengguna dari perangkat lunak SI Gapoktan *Mobile*.

4.1 Analisis Kebutuhan

Proses analisis kebutuhan mengacu pada gambaran umum perangkat lunak SI Gapoktan *Mobile* dan hasil pengumpulan, pemahaman dan penetapan kebutuhan – kebutuhan yang ingin didapatkan oleh pengguna. Proses analisis kebutuhan ini diawali dengan penjabaran gambaran umum perangkat lunak SI Gapoktan *Mobile*, identifikasi aktor yang terlibat, analisis data yang akan disimpan, penjabaran tentang daftar kebutuhan dan kemudian memodelkannya ke dalam diagram *use case*. Analisis kebutuhan ini bertujuan untuk menggambarkan kebutuhan – kebutuhan yang harus disediakan oleh sistem agar dapat memenuhi kebutuhan pengguna.

4.1.1 Gambaran Umum Perangkat Lunak SI Gapoktan *Mobile*

Perangkat lunak ini adalah perangkat lunak yang berfungsi membantu gapoktan dalam memasarkan produk hasil pertanian. Sistem yang ada saat ini berupa *website*. Arsitektur sistem awal antrian sebelum aplikasi diterapkan akan ditunjukkan pada Gambar 4.2.



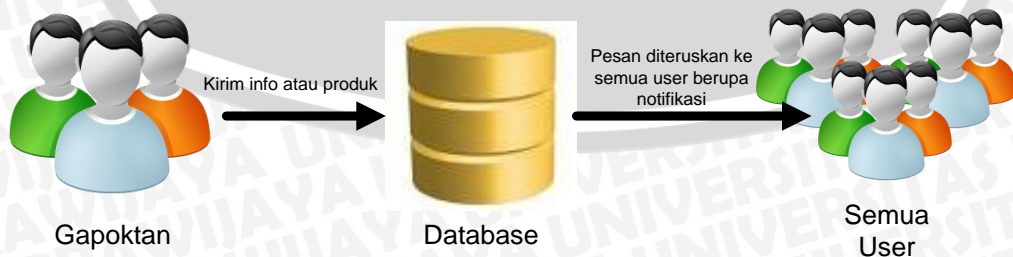
Gambar 4.2 Diagram blok sistem informasi gapoktan berbasis website
Sumber : [Perancangan]

Skenario sistem pemasaran saat ini adalah menggunakan *website*. Gapoktan didaftarkan ke dalam sistem informasi gapoktan berbasis *web* oleh admin *website*. Gapoktan yang telah memiliki akun dapat mengunggah foto serta informasi produk pertanian, informasi terkini tentang gapoktan, dan mengirim pesan rahasia ke gapoktan lain. Pada sistem informasi gapoktan berbasis *website* informasi terkini dan produk pertanian terbaru akan di *broadcast* ke semua gapoktan, sedangkan pesan rahasia merupakan pesan komunikasi antara gapoktan pengirim dan gapoktan penerima pesan. Diagram blok alur pendaftaran gapoktan dan penyebaran informasi di sistem informasi gapoktan berbasis *website* dijelaskan pada gambar 4.3.

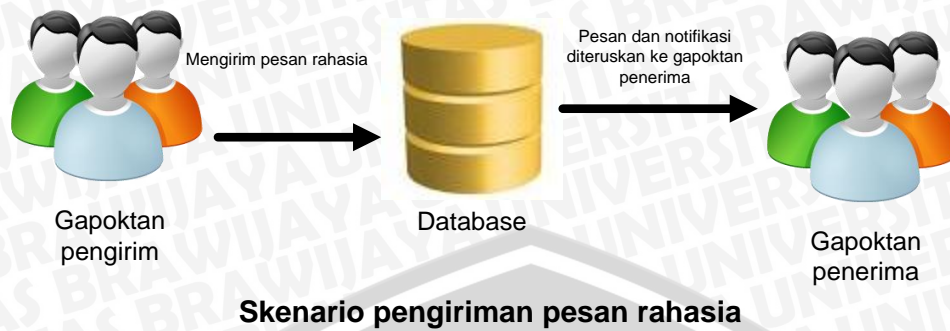


Gambar 4.3 Diagram blok pendaftaran gapoktan pada sistem informasi gapoktan
Sumber : [Perancangan]

SI Gapoktan *Mobile* merupakan pengembangan dari sistem informasi gabungan berbasis *website*. Sistem ini mengimplementasi seluruh skenario dari sistem informasi gapoktan berbasis *website*. SI Gapoktan *Mobile* merupakan aplikasi berbasis android yang memanfaatkan fitur *web services* dari si gapoktan berbasis *website*. Diagram blok skenario penyebaran informasi dijelaskan pada gambar 4.4.



Skenario penyebaran informasi gapoktan dan produk hasil pertanian ke semua user sistem informasi



Skenario pengiriman pesan rahasia

Gambar 4.4 skenario penyebaran informasi sistem informasi gapoktan
Sumber : [Perancangan]

SI Gapoktan *Mobile* akan ditambahkan fitur GCDM. Fitur ini dapat memberikan notifikasi secara *realtime* di aplikasi berplatform android. Fitur ini mengirimkan notifikasi yang tercatat oleh *server*, kemudian mengirimkan ke *web services google push notifikasi*. *Web services* push notification kemudian mengirimkan *alert* ke aplikasi *client* menggunakan akun *Email google*. *Alert* tersebut kemudian ditampilkan di layar SI Gapoktan *Mobile*. Berikut adalah gambar alur pengiriman informasi GCDM.



Gambar 4.5. Diagram blok pengiriman informasi GCDM.
Sumber : [Perancangan]

Aplikasi ini mengimplementasi fitur dari sistem informasi gapoktan yang telah dijelaskan pada bab 3 sub bab 3.1.3 tentang analisis kebutuhan dengan beberapa pengembangan. Aplikasi ini memiliki 6 fungsi utama yaitu: notifikasi, produk, profil, info gapoktan, pesan rahasia, dan daftar list gapoktan.

1. Notifikasi

Notifikasi merupakan fitur yang memberikan pesan *real time* kepada user jika ada informasi tertentu. Notifikasi dibagi menjadi dua yaitu notifikasi umum dan khusus.



a. Notifikasi Umum

Notifikasi umum adalah pesan yang di kirimkan ke semua user. Notifikasi ini akan aktif jika user menambah hasil produk pertanian, dan jika user memberikan informasi gapoktan terbaru.

b. Notifikasi Khusus

Notifikasi khusus adalah pesan yang dikirimkan ke user yang berkaitan dengan informasi. Notifikasi ini akan memberikan pesan kepada penerima pesan rahasia, penerima komentar pada info, dan pemberi informasi tentang gapoktan jika terdapat komentar.

Notifikasi ini dilengkapi dengan fitur GCDM yang memberikan informasi secara real time kepada user.

2. Produk Gapoktan

Produk gapoktan berisi produk terbaru dari gapoktan. Produk ini dibagi menjadi dua yaitu produk primer dan produk sekunder. Produk primer merupakan produk hasil panen gapoktan. Produk sekunder merupakan produk hasil olahan dari gapoktan seperti saos, tepung terigu, dll.

3. Info Gapoktan

Fitur ini merupakan turunan dari fitur forum pada sistem informasi gapoktan berbasis web. Info gapoktan merupakan informasi terkini tentang gapoktan. Fitur ini bertujuan untuk memberikan informasi kegiatan yang dilakukan oleh gapoktan kepada semua user. Fitur ini dapat ditanggapi oleh user lain dengan cara memberikan komentar.

4. Profil

Profil berisi data informasi gapoktan. Data informasi gapoktan berupa nama gapoktan, id gapoktan, alamat gapoktan, foto gapoktan, *Email* gapoktan, dan nomor telepon.

5. Pesan Rahasia

Pesan rahasia (*private message*) bertujuan untuk memberikan komunikasi secara privat antar 2 gapoktan. Data yang diinputkan berupa teks yang akan dikirimkan kepada gapoktan yang lain.

6. Daftar List Gapoktan

Berisi list seluruh gapoktan yang terdaftar di SI Gpaoktan *Mobile*. Fitur ini bertujuan mempermudah user mencari gapoktan berdasarkan nama.

4.1.2 Identifikasi Aktor

Tahap ini adalah tahap untuk melakukan identifikasi terhadap aktor yang akan berinteraksi dengan perangkat lunak SI Gapoktan *Mobile*. Tabel 4.1 memperlihatkan aktor – aktor yang terlibat beserta penjelasannya masing-masing yang merupakan hasil dari proses identifikasi aktor.

Tabel 4.1 Identifikasi aktor

Aktor	Deskripsi
Gapoktan	Gapoktan adalah pengguna utama SI Gapoktan <i>Mobile</i> . Gapoktan dapat menggunakan 6 fungsi utama yaitu notifikasi, produk, profil, info gapoktan, pesan rahasia, dan daftar gapoktan.
Administrator	<i>Administrator</i> adalah pengguna aplikasi <i>administrator</i> perangkat lunak sistem informasi gabungan kelompok tani. <i>Administrator</i> memiliki hak untuk mengolah perangkat lunak lebih lanjut. <i>Administrator</i> hanya dapat melakukan aktivitasnya di sistem informasi gabungan kelompok tani berbasis <i>website</i> .
Penyuluh	Penyuluh adalah pengguna aplikasi SI Gapoktan <i>Mobile</i> pertama kali. Penyuluh dapat melakukan semua fungsi yang dilakukan oleh gapoktan. Peranan penyuluh dalam aplikasi ini adalah sebagai aktor gapoktan. Hal ini berfungsi agar penyuluh dapat melatih gapoktan dalam menggunakan aplikasi ini.

Sumber : [Perancangan]

4.1.3 Analisis Use Case

Analisis *use case* dibagi mejadi dua, yaitu daftar kebutuhan dan diagram *use case*. Pada daftar kebutuhan akan dibagi menjadi dua yaitu spesifikasi kebutuhan fungsional Gapoktan dankebutuhan fungsional *administrator*. Spesifikasi kebutuhan fungsional gapoktan ditunjukkan pada tabel 4.2.

Tabel 4.2 Spesifikasi kebutuhan fungsional gapoktan

Nomor SRS	Kebutuhan	Use Case
SRS_001_01	Perangkat lunak harus bisa melakukan <i>login</i> user	<i>Login</i>
SRS_001_02	Perangkat lunak harus mampu melakukan <i>share</i> informasi tentang gapoktan ke semua user	Posting Info Gapoktan
SRS_001_03	Perangkat lunak harus mampu melakukan <i>share</i> produk yang diisikan oleh gapoktan	Posting Produk
SRS_001_04	Perangkat lunak harus mampu mengirimkan pesan rahasia antar gapoktan	Kirim Pesan Rahasia
SRS_001_05	Perangkat lunak harus mampu membaca memberikan notifikasi secara <i>real time</i> kepada gapoktan	Menerima Notifikasi
SRS_001_06	Perangkat lunak harus bisa melakukan edit profil gapoktan	Mengedit Profil
SRS_001_07	Perangkat lunak harus bisa lihat keseluruhan gapoktan yang terdaftar	Lihat semua gapoktan
SRS_001_08	Perangkat lunak harus bisa melakukan <i>logout</i> , agar aplikasi bisa digunakan oleh user lain	Logout

Sumber : [Perancangan]

Spesifikasi kebutuhan fungsional *administrator* ditunjukkan pada Tabel 4.3.

Tabel 4.3 Spesifikasi kebutuhan fungsional *administrator*

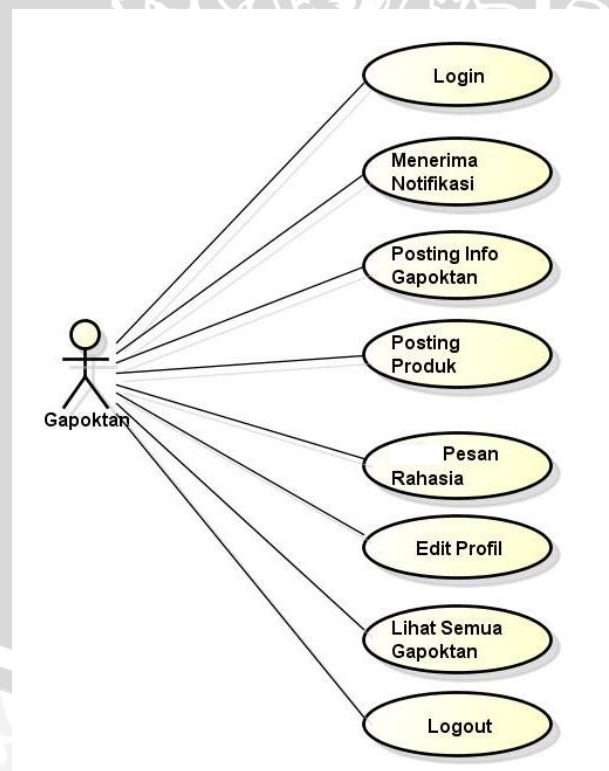
Nomor SRS	Kebutuhan	Use Case
SRS_002_01	Perangkat lunak harus menyediakan fasilitas untuk <i>login</i> sehingga hanya <i>administrator</i> saja yang dapat mengelola sistem.	<i>Login</i>
SRS_002_02	Perangkat lunak harus menyediakan fasilitas agar <i>administrator</i> dapat menambah gapoktan baru.	Tambah Gapoktan
SRS_002_03	Perangkat lunak harus menyediakan fasilitas agar <i>administrator</i> dapat menghapus info gapoktan.	Hapus Info Gapoktan
SRS_002_04	Perangkat lunak harus menyediakan fasilitas agar <i>administrator</i> dapat menghapus produk gapoktan.	Hapus Produk Gapoktan
SRS_002_05	Perangkat lunak harus menyediakan fasilitas agar <i>administrator</i> dapat menghapus gapoktan.	Hapus Gapoktan
SRS_002_06	Perangkat lunak harus	Logout

	menyediakan fasilitas adar administrator dapat keluar dari sistem.
--	--

Diagram *use case* adalah salah satu diagram untuk memodelkan aspek perilaku sistem. Diagram *use case* menunjukkan sekumpulan *use case*, aktor, dan hubungannya. *Use case* merupakan fungsionalitas dari sistem yang diinisialisasi oleh aktor. *Use case* SI Gapoktan *Mobile* dibagi menjadi dua yaitu diagram *use case* untuk user gapoktan dan user *administrator*.

4.1.3.1 Diagram Use Case Gapoktan

Diagram *use case* ini melibatkan gapoktan sebagai aktor dan satu buah *use case*. Satu buah *use case* ini termasuk dalam bagian *software* SI Gapoktan *Mobile*. Satu buah *use case* ini juga akan disertai dengan skenario *use case* untuk menjelaskan rangkaian aktifitas yang terjadi di *use case* tersebut. Diagram *use case* gapoktan ditunjukkan pada Gambar 4.6 berikut ini :



Gambar 4.6 Use Case Gapoktan SI Gapoktan Mobile
Sumber : [Perancangan]

1. Skenario Use Case Login

Tabel 4.4 Use case Login

Skenario Kasus Pada Sistem	
Nomor <i>Use Case</i>	SRS_001_01
Nama	<i>Login</i>
Tujuan	Masuk ke dalam sistem SI Gapoktan <i>Mobile</i> .
Deskripsi	<i>Use Case</i> ini menjelaskan bagaimana Gapoktan masuk ke dalam sistem dengan menginputkan <i>username</i> dan <i>password</i>
Aktor	Gapoktan
Skenario Utama	
Kondisi Awal	Perangkat lunak SI Gapoktan <i>Mobile</i> sudah terpasang di android.
Aksi Aktor	Reaksi Sistem
Memasukan <i>username</i> ke kolom <i>username</i> dan <i>password</i> ke kolom <i>password</i> .	<i>Username</i> dan <i>password</i> diterima oleh SI Gapoktan <i>Mobile</i> . <i>Username</i> dan <i>Password</i> kemudian diteruskan ke <i>web services login</i> sistem informasi gapoktan dengan method HTTP Post. Kemudian <i>username</i> dan <i>password</i> di cek ke dalam <i>database</i> . Jika ditemukan <i>username</i> dan <i>password</i> yang sama, maka <i>login</i> berhasil dilakukan dan SI Gapoktan <i>Mobile</i> memberikan notifikasi kepada gapoktan serta masuk kedalam halaman home sistem. Jika <i>username</i> dan <i>password</i> tidak ditemukan maka <i>login</i> gagal dilakukan dan SI Gapoktan <i>Mobile</i> Memberi notifikasi kepada gapoktan.
Kondisi Akhir	Gapoktan masuk ke dalam halaman home dan mendapatkan pemberitahuan “selamat datang [nama gapoktan]..!!”

Sumber : [Perancangan]

2. Skenario Use Case Menerima Notifikasi

Tabel 4.5 Use case Menerima Notifikasi

Skenario Kasus Pada Sistem	
Nomor <i>Use Case</i>	SRS_001_02
Nama	Menerima Notifikasi
Tujuan	Memberi pemberitahuan (notifikasi) jika ada produk baru, komentar, dan pesan yang masuk.
Deskripsi	<i>Use Case</i> ini menjelaskan gapoktan menerima pemberitahuan jika ada produk baru, komentar, serta pesan rahasia.
Aktor	Gapoktan
Skenario Utama	
Kondisi Awal	Gapoktan telah berhasil <i>login</i> ke dalam sistem dan terdaftar di GCDM (untuk notifikasi secara <i>real time</i>).
Aksi Aktor	Reaksi Sistem
Memposting produk	Jika gapoktan memposting produk maka notifikasi akan di

baru, mengomentari info gapoktan, mengirim pesan rahasia.	<i>share</i> ke semua gapoktan. Pertama, <i>database</i> akan mencatat pemberitahuan dan memberi kode bahwa pemberitahuan akan di <i>share</i> ke semua gapoktan. Jika gapoktan memngirim pesan rahasia atau komentar, maka notifikasi akan di <i>share</i> ke gapoktan penerima pesan atau penulis info gapoktan. Semua pemberitahuan akan dikirimkan ke google push notification <i>web services</i> untuk dikirimkan ke SI Gapoktan <i>Mobile</i> via google account secara real time.
Kondisi Akhir	Jika terdaftar di GCDM maka gapoktan menerima notifikasi yang berkaitan dengan akunya secara realtime. Jika tidak terdaftar maka notifikasi hanya bisa dilihat pada fitur notifikasi.

Sumber : [Perancangan]

3. Skenario Use Case Posting Info Gapoktan

Tabel 4.6 Use case Posting Info Gapoktan

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_03
Nama	Posting Info Gapoktan
Tujuan	Memberi informasi terbaru tentang kegiatan yang dilakukan gapoktan
Deskripsi	Use Case ini menjelaskan gapoktan memposting informasi kegiatan gapoktan.
Aktor	Gapoktan
Skenario Utama	
Kondisi Awal	Gapoktan telah berhasil <i>login</i> ke dalam sistem dan masuk kedalam fitur Info Gapoktan.
Aksi Aktor	Reaksi Sistem
Menuliskan informasi terkini pada kolom Info Gapoktan.	Informasi akan dikirimkan ke <i>web services</i> insert_posting sistem informasi gapoktan. Kemudian sistem menginputkan informasi ke <i>database</i> .
Kondisi Akhir	Info terbaru akan muncul di fitur info gapoktan urutan paling atas.

Sumber : [Perancangan]

4. Skenario Use Case Posting Produk

Tabel 4.7 Use case Posting Produk

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_04
Nama	Posting Produk
Tujuan	Memberi informasi produk terbaru yang dihasilkan gapoktan.
Deskripsi	Use Case ini menjelaskan bagaimana gapoktan memposting produk terbaru.

Aktor	Gapoktan
Skenario Utama	
Kondisi Awal	Gapoktan telah berhasil <i>login</i> ke dalam sistem dan masuk kedalam fitur tambah produk.
Aksi Aktor	Reaksi Sistem
Menuliskan produk terbaru ke dalam sistem informasi gapoktan	Informasi akan dikirimkan ke <i>web services</i> insert_produk sistem informasi gapoktan. Kemudian sistem menginputkan informasi ke <i>database</i> .
Kondisi Akhir	Produk terbaru akan muncul di fitur info gapoktan urutan paling atas.

Sumber : [Perancangan]

5. Skenario Use Case Kirim Pesan Rahasia

Tabel 4.8 Use case Kirim Pesan Rahasia

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_05
Nama	Kirim Pesan Rahasia
Tujuan	Gapoktan dapat berkomunikasi secara rahasia.
Deskripsi	Use Case ini menjelaskan bagaimana gapoktan mengirim dan menerima pesan secara <i>private</i> .
Aktor	Gapoktan
Skenario Utama	
Kondisi Awal	Gapoktan memilih profil gapoktan yang akan dikirimkan pesan.
Aksi Aktor	Reaksi Sistem
Menuliskan Pesan kedalam kolom pesan yang telah disediakan.	Pesan akan dikirimkan ke <i>web services</i> insert_pesan sistem informasi gapoktan. Kemudian sistem menginputkan informasi ke <i>database</i> .
Kondisi Akhir	Pesan akan terkirim ke penerima. Pesan akan muncul di halaman pesan gapoktan.

Sumber : [Perancangan]

6. Skenario Use Case Edit Profil

Tabel 4.9 Use case Edit Profil

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_06
Nama	Edit Profil
Tujuan	Gapoktan dapat mengedit profil data organisasinya.
Deskripsi	Use Case ini menjelaskan bagaimana gapoktan mengedit profil organisasinya.
Aktor	Gapoktan
Skenario Utama	
Kondisi Awal	Gapoktan masuk kedalam fitur lihat profil dan tekan tombol edit profil.

Aksi Aktor	Reaksi Sistem
Menuliskan <i>Email, password, deskripsi, kota, kecamatan, dan no hp</i> ke kolom yang telah disediakan	<i>Email, password, deskripsi, kota, kecamatan, dan no hp</i> akan dikirimkan ke <i>web services</i> <i>edit_profil</i> sistem informasi gapoktan. Kemudian sistem mengubah data lama dengan data baru yang diinputkan
Kondisi Akhir	Profil akan berubah sesuai dengan data yang diinputkan.

Sumber : [Perancangan]

7. Skenario Use Case Lihat Semua Gapoktan

Tabel 4.10 Use case Lihat Semua Gapoktan

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_07
Nama	Lihat Semua Gapoktan
Tujuan	User dapat melihat list semua gapoktan.
Deskripsi	Use Case ini menjelaskan bagaimana sistem menampilkan list gapoktan
Aktor	Gapoktan
Skenario Utama	
Kondisi Awal	Gapoktan masuk kedalam fitur lihat profil dan tekan tombol edit profil.
Aksi Aktor	Reaksi Sistem
Masuk ke dalam fitur list gapoktan	<i>Email, password, deskripsi, kota, kecamatan, dan no hp</i> akan dikirimkan ke <i>web services</i> <i>get_all_gapoktan</i> sistem informasi. Data yang dihasilkan berupa XML yang akan diparsing dan di petakan ke dalam tampilan SI Gapoktan <i>Mobile</i> .
Kondisi Akhir	Pada fitur list gapoktan akan muncul seluruh gapoktan yang terdaftar di SI Gapoktan <i>Mobile</i> .

Sumber : [Perancangan]

8. Skenario Use Case Logout

Tabel 4.11 Use case Logout

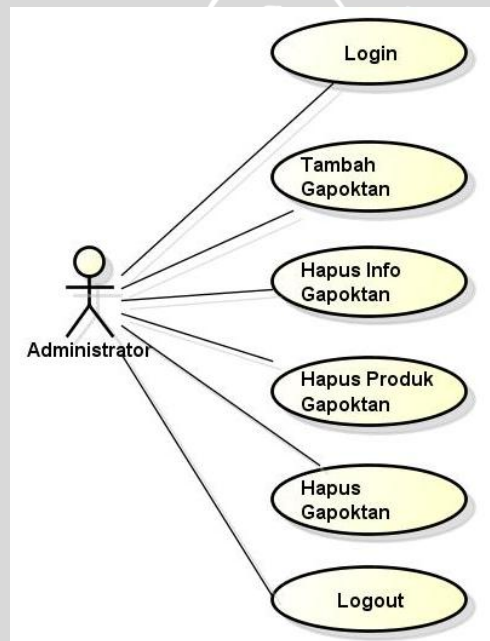
Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_08
Nama	Logout
Tujuan	User keluar dari sistem
Deskripsi	Use Case ini menjelaskan bagaimana user keluar dari sistem
Aktor	Gapoktan
Skenario Utama	
Kondisi Awal	User berhasil <i>login</i> dan masuk ke dalam halaman home.
Aksi Aktor	Reaksi Sistem
Menekan tombol logout	Sistem akan memanggil object <i>session_manager</i> , dan

	mematikan session di SI Gapoktan Mobile.
Kondisi Akhir	User keluar dari halaman home dan masuk ke halaman <i>login</i> .

Sumber : [Perancangan]

4.1.3.2 Diagram Use Case Administrator

Diagram *use case* ini melibatkan *administrator* sebagai aktor dan satu buah *use case*. *Use case ini* merupakan analisis *use case* pada sistem informasi gapoktan berbasis web. Usecase administrator terdiri dari 6 aktivitas yaitu login, tambah gapoktan, hapus info gapoktan, hapus produk gapoktan, hapus gapoktan, dan logout. Satu buah *use case* ini juga akan disertai dengan skenario *use case* untuk menjelaskan rangkaian aktivitas yang terjadi di *use case* tersebut. Diagram *use case* untuk *administrator* ditunjukkan pada Gambar 4.7 berikut ini :



Gambar 4.7 Use Case Administrator SI Gapoktan Mobile
Sumber : [Perancangan]

1. Skenario Use Case Login

Tabel 4.12 Use case Login

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_01
Nama	<i>Login</i>
Tujuan	Masuk ke dalam sistem informasi gapoktan
Deskripsi	<i>Use Case</i> ini menjelaskan bagaimana Gapoktam masuk ke dalam sistem dengan menginputkan <i>username</i> dan

	<i>password</i>
Aktor	<i>Administrator</i>
Skenario Utama	
Kondisi Awal	User masuk ke URL sistem informasi gapoktan berbasis web
Aksi Aktor	Reaksi Sistem
Memasukan <i>username</i> ke kolom <i>username</i> dan <i>password</i> ke kolom <i>password</i> .	<i>Username</i> dan <i>password</i> diterima oleh sistem informasi gapoktan berbasis web, kemudian dicocokkan ke <i>database</i> apakah <i>username</i> dan <i>password</i> ada atau tidak.
Kondisi Akhir	Jika <i>username</i> dan <i>password</i> terdaftar, maka user berhasil masuk sebagai admin. Jika tidak terdaftar, maka akan muncul peringatan bahwa <i>username</i> dan <i>password</i> tidak tersedia.

Sumber : [Perancangan]

2. Skenario Use Case Tambah Gapoktan

Tabel 4.13 Use case Tambah Gapoktan

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_02
Nama	Tambah gapoktan
Tujuan	Menambah user baru ke sistem informasi gapoktan berbasis web.
Deskripsi	Use Case ini menjelaskan bagaimana <i>administrator</i> menambah user baru ke sistem.
Aktor	<i>Administrator</i>
Skenario Utama	
Kondisi Awal	User berhasil <i>login</i> ke sistem informasi gapoktan berbasis web.
Aksi Aktor	Reaksi Sistem
Memasukan <i>username</i> , nama gapoktan, dan <i>password</i> .	Data akan diterima oleh sistem informasi gapoktan berbasis web. Kemudian data dimasukkan ke dalam <i>database</i> .
Kondisi Akhir	User gapoktan akan bertambah.

Sumber : [Perancangan]

3. Skenario Use Case Hapus Info Gapoktan

Tabel 4.14 Use case Hapus Info Gapoktan

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_03
Nama	Hapus Info Gapoktan
Tujuan	<i>Administrator</i> mampu menghapus yang dianggap merugikan user lain di sistem informasi gapoktan.
Deskripsi	Use Case ini menjelaskan cara <i>administrator</i> menghapus

	info gapoktan.
Aktor	<i>Administrator</i>
Skenario Utama	
Kondisi Awal	Gapoktan telah berhasil <i>login</i> ke dalam sistem dan masuk kedalam fitur Hapus Info Gapoktan.
Aksi Aktor	Reaksi Sistem
Memilih info yang akan dihapus	User akan diberi peringatan, apakah user yakin akan menghapus info. Jika user yakin maka sistem akan menghapus info gapoktan dari <i>database</i> .
Kondisi Akhir	Info gapoktan akan terhapus dari sistem informasi gapoktan.

Sumber : [Perancangan]

4. Skenario Use Case Hapus Produk

Tabel 4.15 Use case Hapus Produk

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_04
Nama	Hapus Produk
Tujuan	<i>Administrator</i> dapat menghapus produk yang dianggap merugikan bagi user lain pada sistem informasi gapoktan berbasis web.
Deskripsi	<i>Use Case</i> ini menjelaskan bagaimana <i>administrator</i> menghapus produk gapoktan.
Aktor	<i>Administrator</i>
Skenario Utama	
Kondisi Awal	Gapoktan telah berhasil <i>login</i> ke dalam sistem dan masuk kedalam fitur Hapus produk.
Aksi Aktor	Reaksi Sistem
Memilih produk yang akan dihapus.	User akan diberi peringatan, apakah user yakin akan menghapus produk. Jika user yakin maka sistem akan menghapus produk gapoktan dari <i>database</i> .
Kondisi Akhir	Info gapoktan akan terhapus dari sistem informasi gapoktan.

Sumber : [Perancangan]

5. Skenario Use Case Hapus Gapoktan

Tabel 4.16 Use case Hapus Gapoktan

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_05
Nama	Hapus Gapoktan
Tujuan	Menghapus gapoktan yang telah tidak aktif lagi.
Deskripsi	<i>Use Case</i> ini menjelaskan bagaimana <i>administrator</i> menghapus gapoktan yang sudah tidak aktif lagi.
Aktor	<i>Administrator</i>

Skenario Utama	
Kondisi Awal	Gapoktan telah berhasil <i>login</i> ke dalam sistem dan masuk kedalam fitur hapus gapoktan.
Aksi Aktor	Reaksi Sistem
Memilih produk yang akan dihapus.	User akan diberi peringatan, apakah user yakin akan menghapus gapoktan. Jika user yakin maka sistem akan menghapus gapoktan dari <i>database</i> .
Kondisi Akhir	Gapoktan akan terhapus dari sistem informasi gapoktan.

Sumber : [Perancangan]

6. Skenario Use Case Logout

Tabel 4.17 Use case Logout

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_09
Nama	Logout
Tujuan	User keluar dari sistem
Deskripsi	Use Case ini menjelaskan bagaimana user keluar dari sistem
Aktor	Administrator
Skenario Utama	
Kondisi Awal	User telah berhasil <i>login</i> ke sistem.
Aksi Aktor	Reaksi Sistem
Menekan tombol logout	Sistem akan mendestroy session yang ada. Kemudian user akan masuk ke halaman index.
Kondisi Akhir	User keluar dari sistem dan masuk ke halaman index.

Sumber : [Perancangan]

4.1.4. Analisis Komponen

Analisis komponen bertujuan untuk membagi SI Gapoktan *Mobile* menjadi komponen-komponen sehingga mudah untuk dibangun. Berdasarkan diagram blok perancangan sistem, komponen SI Gapoktan *Mobile* dibagi menjadi tiga yaitu : komponen XML Parser, komponen session manager, dan komponen GCDM, dan komponen Inti.

1. Komponen XML Parser

Komponen XML parser merupakan komponen dimana sistem melakukan request data ke *web service* sistem informasi gabungan kelompok tani. Komponen ini akan diletakan ke *package parser*. Komponen ini terdiri dari *method* HTTP Post untuk melakukan request data ke url *web services*, *method* XML Parser untuk mengubah format data XML yang didapat dari

web service ke dalam list, dan *image loader* untuk mengunduh gambar dari sistem informasi gapoktan.

2. Komponen Session Manager

Komponen session manager merupakan komponen untuk menjaga user tetap *login* di SI Gapoktan *Mobile* sebelum menekan tombol logout. Komponen ini akan diletakkan di *package* gapoktan pada kelas session manager. Object dari kelas ini menyimpan data *username* dan *Email* dari user yang *login*.

3. Komponen Inti

Komponen inti merupakan komponen utama dalam SI Gapoktan *Mobile*. Komponen ini terdiri dari kelas-kelas yang memetakan fitur dari SI Gapoktan *Mobile*. Komponen ini juga berfungsi sebagai penghubung antara tampilan sistem dan data model.

4. Komponen GCDM

Komponen GCDM merupakan komponen untuk memberikan notifikasi secara real time kepada user yang sedang *login*. Komponen ini akan diletakkan pada *package* pushnotification. Komponen ini terdiri dari method register untuk mendaftarkan akun ke *web services* google, dan method message handler untuk menerima notifikasi terbaru.

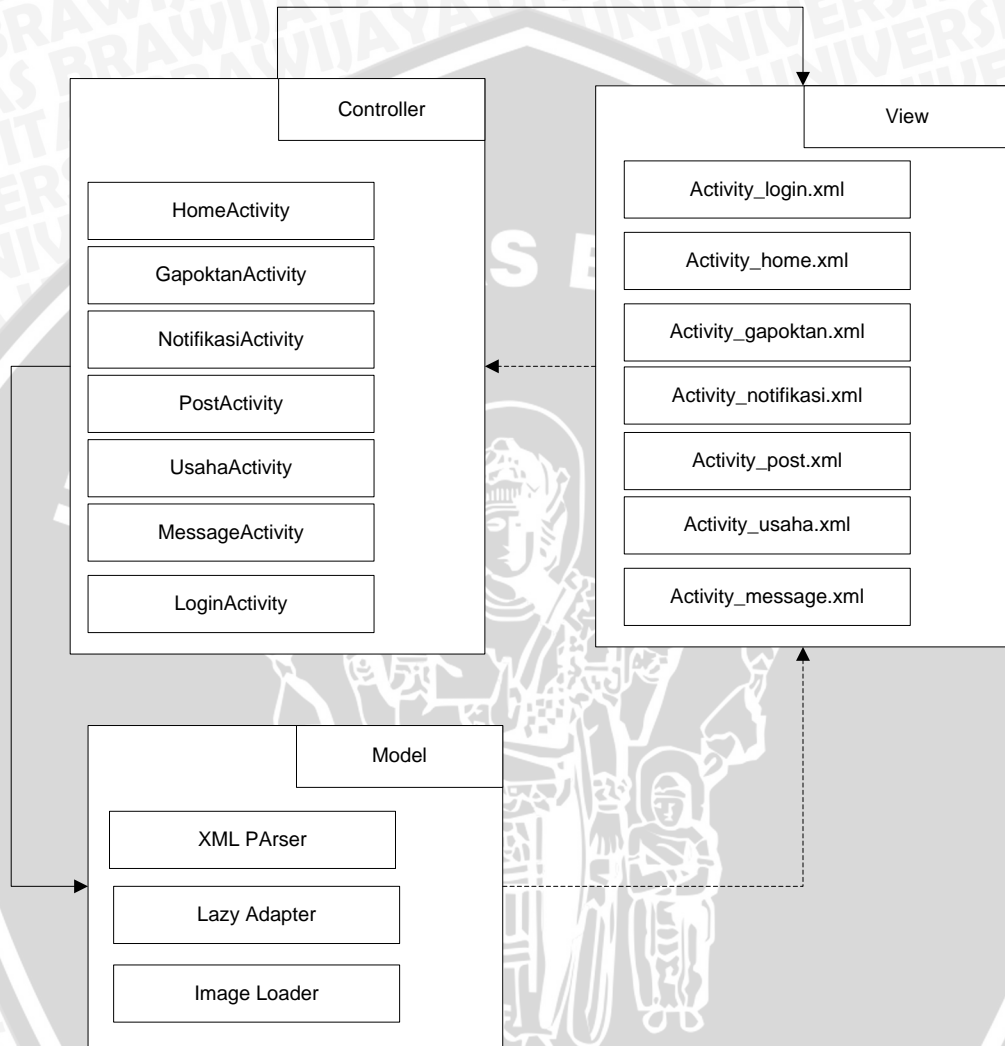
4.2 Perancangan Perangkat Lunak

Perancangan perangkat lunak memiliki empat tahap, yaitu perancangan arsitektural, pemodelan kelas untuk menggambarkan perancangan struktur *class-class* yang menyusun perangkat lunak, perancangan sequences diagram untuk mengetahui proses dan alur program saat dijalankan, dan perancangan antarmuka. Perancangan perangkat lunak pada skripsi ini menggunakan pendekatan desain berorientasi objek yang direpresentasikan dengan menggunakan UML (*Unified Modelling Language*).

4.2.1 Perancangan Arsitektural

Perangkat lunak SI Gapoktan *Mobile* dikembangkan dengan mengadopsi pola arsitektur *Model-View-Controller* (MVC). Berdasarkan pola arsitektur MVC ini, perangkat lunak SI Gapoktan *Mobile* memiliki 3 bagian utama, yaitu *class*

pada *package* parser yang berfungsi sebagai *model*, *class* pada *package* gapoktan yang berfungsi sebagai *controller* dan *class* pada folder *layout* sebagai antar muka. Pola arsitektur MVC pada perangkat lunak SI Gapoktan *Mobile* ditunjukkan pada Gambar 4.7

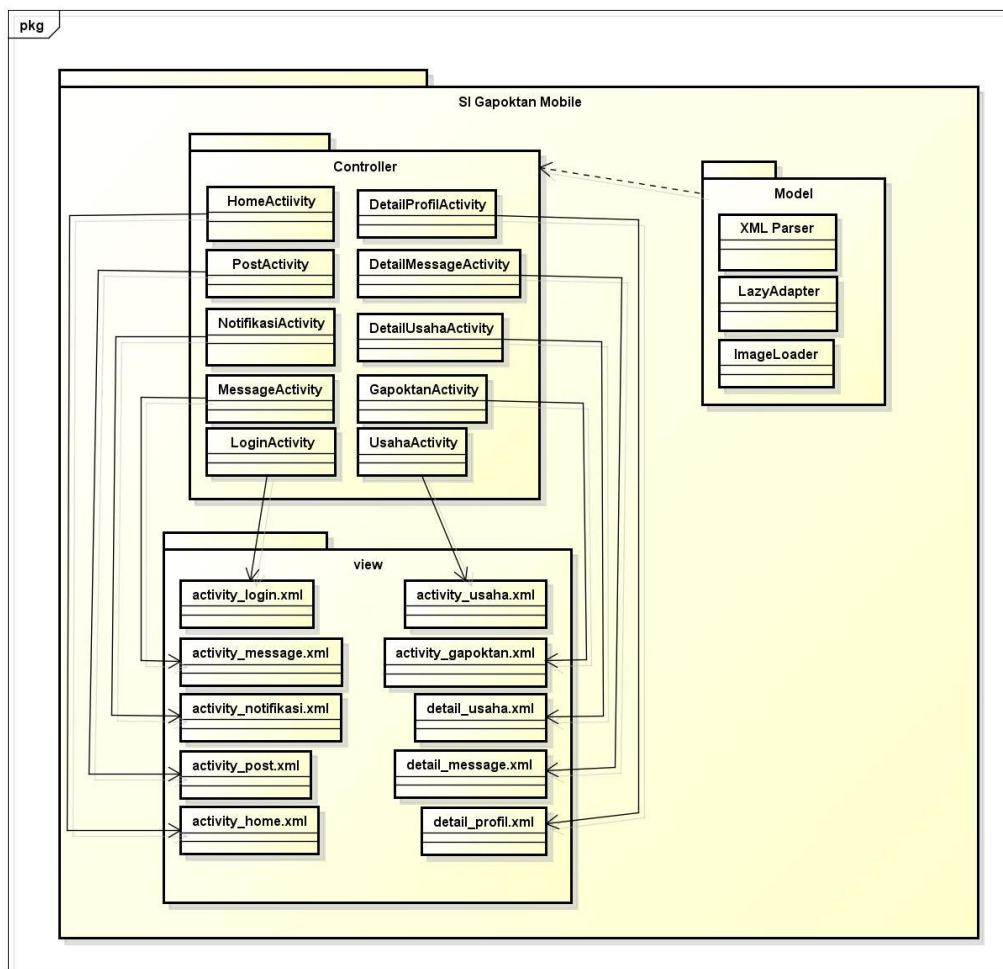


Gambar 4.8 Arsitektur MVC perangkat lunak SI Gapoktan *Mobile*
Sumber : [Perancangan]

Class XMLparser bekerja sebagai *model* dengan cara memodelkan data yang didapat dari *web services* dengan dibantu beberapa *Lazy Adapter* dan *Image Loader*. *Class* pada *package* gapoktan bekerja sebagai *controller* dengan melakukan pengaturan aliran komunikasi antara *class* XML Parser dengan *class-class* antarmuka.

4.2.2 Pemodelan Kelas

Pemodelan kelas memberikan gambaran pemodelan elemen-elemen kelas yang membentuk sebuah perangkat lunak. Kelas bisa didapatkan dengan menganalisis secara detail terhadap *use case* yang dimodelkan. SI Gapoktan *Mobile* memiliki 22 kelas yang dibagi menjadi 4 komponen yaitu komponen XML Parser, komponen *session manager*, komponen inti, dan komponen GCDM. SI Gapoktan mobile juga disusun menggunakan struktur *model-view-controller* (MVC). Gambar 4.15 menunjukkan pemodelan MVC diagram kelas dari perangkat lunak SI Gapoktan *Mobile* yang akan dibuat.



powered by Astah

Gambar 4.10 Pemodelan MVC Kelas Perangkat Lunak SI Gapoktan *Mobile*
Sumber : [Perancangan]

Pada pemodelan kelas di atas digambarkan *interface* dan beberapa kelas utama yang menyusun perangkat lunak SiGapoktan *Mobile*. Bagian *Controller* memiliki delapan kelas yang menangani tiap fitur yang tersedia di SI Gapoktan

Mobile. Bagian *Model* memiliki tiga buah kelas yaitu XML Parser untuk mengambil data dan menerima data dari *web services*, LazyAdapter untuk mengubah format data XML dari *web services* ke dalam bentuk list, dan ImageLoader untuk mengunduh image yang kemudian ditampilkan ke *interface*. Bagian *views* terdiri dari kelas-kelas xml yang berfungsi sebagai *layout* di *platform android*.

4.2.2.1 Pemodelan Kelas Komponen XML Parser

Kelas komponen XML Parser merupakan komponen yang berfungsi untuk mengambil dan mengubah format data dari *web services* sehingga dapat olah di sistem perangkat lunak SI Gapoktan *Mobile*. Kelas komponen XML Parser terdiri dari tiga kelas yaitu kelas XML Parser, Kelas LazyAdapter, dan Kelas Image Loader.

A. Kelas XML Parser

Kelas XML Parser digunakan untuk mengunduh data dari URL *web services*, kemudian menyimpannya di dalam atribut data. Struktur kelas XMLParser digambarkan di gambar 4.11.

XML Parser	
1 +	getXmlLogin(url : String, username : String, password : String) : String
2 +	getXmlAll(url : String) : String
3 +	getload2par(url : String, userid : String, username : String, message : String) : String
4 +	getload1par(url : String, userid : String) : String
5 +	getDomElement(xml : String) : Document
6 +	getElementValue(elem : Node) : String
7 +	getValue(item : Element, str : String) : String

Gambar 4.11 Kelas XML Parser SI Gapoktan *Mobile*
Sumber : [Perancangan]

Berikut adalah penjelasan kelas XML Parser :

1. *Method* getXML login berfungsi untuk mengirimkan parameter username dan password dan mengunduh nilai kembalian dari web service login.

2. *Method* getXMLAll berfungsi untuk mengunduh data web service tanpa parameter.
3. *Method* getLoad2Par berfungsi untuk mengunduh data web service dengan 2 parameter
4. *Method* getLoad1Par berfungsi untuk mengunduh data web service dengan 1 parameter
5. *Method* getDomElement berfungsi untuk mengkonversi data XML ke format document.
6. *Method* get element value dan get value digunakan untuk mengambil data dari node XML.

B. Kelas LazyAdapter

Kelas LazyAdapter digunakan untuk mengubah data yang berformat XML menjadi list dan nantinya ditampilkan di *interface* SI Gapoktan *Mobile*. Struktur kelas LazyAdapter digambarkan di gambar 4.12

LazyAdapter	
1-	activity : Activity
2-	data : ArrayList<HashMap<String,String>>
3-	imageloader : ImageLoader
4+	LazyAdapter(a : Activity, ArrayList<HashMap<String,String>> : int) : void
5+	getCount() : int
6+	getItem(position : int) : Object
7+	getItemId(position : int) : long
8+	getView(position : int, counterView : View, parent : ViewGroup) : View

Gambar 4.12 Kelas LazyAdapter SI Gapoktan *Mobile*
Sumber : [Perancangan]

1. Deklarasi object activity, object data dengan tipe hashmap, object image loader.
2. Baris 4 *Method* LazyAdapter berfungsi untuk mempersiapkan layout dan image loader.
3. Baris 5 *Method* getCount berfungsi untuk mengambil data size.
4. Baris 6 *Method* getItem berfungsi untuk mengambil data dengan format object.
5. Baris 7 *Method* getItemId berfungsi mengambil index dari list.

6. Baris 8 *Method* getView memetakan data ke layout list_row.xml. Method ini juga mengubah data dari hash map ke bentuk list.

C. Kelas ImageLoader

Kelas image loader digunakan untuk mengunduh gambar dari sistem informasi gapoktan untuk kemudian di tampilkan di *interface* SI Gapoktan *Mobile*. Struktur kelas LazyAdapter digambarkan di gambar 4.13.

ImageLoader
1. memoryCache : MemoryCache
2. fileCache : FileCache
3. imageViews : Map<ImageView,String>
4. executorservice : ExecutorService
5+ ImageLoader(context : Context) : void
6+ DisplayImage(url : String, imageView : ImageView) : void
7+ queuePhoto(url : String, imageview : ImageView) : void
8+ getBitmap(url : String) : Bitmap
9+ clearCache() : void

Gambar 4.13 Kelas Image Loader SI Gapoktan Mobile

Sumber : [Perancangan]

1. Baris 1-4 Deklarasi object memoryCache, FileCache, imageViews, dan executorservice.
2. Baris 5 *Constructor* imageLoader berfungsi untuk mempersiapkan object-object yang akan digunakan,
3. Baris 6 *Method* displayImage berfungsi untuk mengunduh gambar dan menampilkan dalam image view.
4. Baris 7 *Method* queue photo berfungsi untuk membuat foto dalam antrian
5. Baris 8 *Method* getBitmap berfungsi mengunduh gambar ber ekstensi bitmap.

4.2.3.2 Pemodelan Kelas Komponen Inti

Komponen Inti merupakan kelas *controller* yang berfungsi sebagai penghubung antara kelas model pada komponen XML Parser ke kelas view. Komponen inti terdiri dari 10 kelas yaitu: kelas HomeActivity, NotifikasiActivity, PostActivity, GapoktanActivity, MessageActivity, LoginActivity, UsahaActivity, DetailProfilActivity, dan DetailUsahaActivity.

A. Kelas HomeActivity

Kelas HomeActivity berfungsi sebagai tampilan utama SI Gapoktan *Mobile*. Gambar 4.19 menjelaskan struktur dari kelas HomeActivity. Pada kelas HomeActivity terdapat navigasi – navigasi fitur SI Gapoktan *Mobile*.

HomeActivity
1 session : SessionManager 2 btnLogout : Button 3 id : String
4+ onCreate(savedInstanceState : Bundle) : void 5+ onCreateOptionsMenu(menu : Menu) : boolean 6+ PostClick(v : View) : void 7+ listGapoktanClick(v : View) : void 8+ usahaClick(v : View) : void 9+ messageClick(v : View) : void 10+ notifikasiClick(v : View) : void 11+ profilClick(v : View) : void 12+ settingClick(v : View) : void 13+ about_usClick(v : View) : void

Gambar 4.14 Kelas Home Activity SI Gapoktan *Mobile*
Sumber : [Perancangan]

1. Baris 1-3 berfungsi untuk mendeklarasikan variabel yang akan dipakai.
2. Baris 5 adalah *method* untuk menampilkan menu utama.
3. Baris 6 adalah *method* menampilkan data info gapoktan .
4. Baris 7 adalah *method* untuk melihat semua gapoktan yang terdaftar.
5. Baris 8 adalah *method* untuk melihat semua produk gapoktan.
6. Baris 9 adalah *method* untuk melihat semua pesan yang telah dikirimkan kepada user lain..
7. Baris 10 adalah *method* untuk melihat data notifikasi.
8. Baris 11 adalah *method* untuk melihat semua gapoktan yang terdaftar.

B. Kelas LoginActivity

LoginActivity
1- nama : String 2- pass : String 3- URL : String 4- KEY_DATA : String 5- KEY_NAMA : String 6- KEY_STATUS : String 7- KEY_ID : String 8- KEY_EMAIL : String 9- alert : AlertDialogManager 10
11+ onCreate(savedInstanceState : Bundle) : void 12+ onCreateOptionsMenu(menu : Menu) : boolean 13+ buttonClick(v : View) : void

Gambar 4.15 Kelas LoginActivity SI Gapoktan *Mobile*
Sumber : [Perancangan]

Gambar 4.15 menjelaskan struktur dari kelas LoginActivity. Kelas LoginActivity berfungsi sebagai jembatan *interface login* dan data dari *web services* di SI Gapoktan *Mobile*. Berikut adalah penjelasan dari *method-method* pada kelas login

1. Baris 1-2 adalah deklarasi variabel nama dan email yang nantinya diambil dari session.
2. Baris 3 adalah deklarasi variabel URL web services login.
3. Baris 4-9 adalah deklarasi Key node XML yang di kirimkan oleh web services login.
4. Baris 13 merupakan method yang dieksekusi jika tombol login ditekan. Pada method ini terdapat logika-logika verifikasi username dan password user oleh web service.
5. Baris 12 merupakan method yang digunakan untuk membuat menu help, about dan logout.

C. Kelas GapoktanActivity

Kelas GapoktanActivity berfungsi sebagai jembatan *interface gapoktan* dan data gapoktan, serta inisialisasi URL *web services* data gapoktan. Gambar 4.16 menjelaskan struktur dari kelas GapoktanActivity.

GapoktanActivity	
1-	KEY_DATA : String
2-	KEY_NAMA : String
3-	KEY_ALAMAT : String
4-	KEY_TELEPON : String
5-	KEY_NO : String
6-	list : ListView
7-	adapter : LazyAdapter
8-	parser : XMLParser
9-	postlist : ArrayList<HashMap<String,String>>
10-	URL : String
11+	onCreate(savedInstanceState : Bundle) : void
12+	onCreateOptionsMenu(menu : Menu) : boolean

Gambar 4.16 Kelas Gapoktan Activity SI Gapoktan Mobile
Sumber : [Perancangan]

1. Baris 1-6 merupakan deklarasi variabel Key node XML yang di kirimkan oleh variabel web service gapoktan. data yang dikirimkan merupakan data nama gapoktan, alamat, telepon, dan nomor.

2. Baris 7 merupakan deklarasi variabel listview yang digunakan untuk menampilkan data list ke interface.
3. Baris 9 merupakan deklarasi variabel yang digunakan untuk menampung data XML dari web service data gapoktan.
4. Baris 11 merupakan deklarasi yang menyimpan variabel URL web services data gapoktan.
5. Baris 12 merupakan methon onCreate yang diakses pertama kali ketika kelas GapoktanActivity dijalankan. Pada method ini berisi logika-logika pengambilan data dari web services dan bagaimana menampilkan data ke interface.
6. Baris 12 merupakan method yang digunakan untuk membuat menu help, about dan logout.

D. Kelas MessageActivity

Kelas MessageActivity berfungsi sebagai jembatan antara data pesan rahasia dan *interface* pesan rahasia. Gambar 4.17 menjelaskan struktur dari kelas MessageActivity.

MessageActivity	
1-	KEY_DATA : String
2-	KEY_NAMA : String
3-	KEY_POST : String
4-	KEY_COMMENT : String
5-	list : ListView
6-	adapter : LazyAdapter
7-	URL : String
8+	onCreate(savedInstanceState : Bundle) : void
9+	onCreateOptionsMenu(menu : Menu) : boolean

Gambar 4.17 Kelas MessageActivity SI Gapoktan Mobile
Sumber : [Perancangan]

1. Baris 1-4 merupakan deklarasi variabel Key node XML yang di kirimkan oleh variabel web service pesan. data yang dikirimkan merupakan data nama, tanggal, dan pesan.
2. Baris 5 merupakan deklarasi variabel listview yang digunakan untuk menampilkan data list ke interface.
3. Baris 7 merupakan deklarasi yang menyimpan variabel URL web services data gapoktan.

4. Baris 8 merupakan method onCreate yang diakses pertama kali ketika kelas MessageActivity dijalankan. Pada method ini berisi logika-logika pengambilan data dari web services message dan bagaimana menampilkan data ke interface.
 5. Baris 9 merupakan method yang digunakan untuk membuat menu help, about dan logout.
- E. Kelas NotifikasiActivity

Kelas NotifikasiActivity berfungsi sebagai jembatan antara data notifikasi dan *interface* notifikasi. Gambar 4.18 menjelaskan struktur dari kelas NotifikasiActivity.

NotifikasiActivity	
1-	list : ListView
2-	adapter : LazyAdapter
3-	KEY_DATA : String
4-	KEY_TGL : String
5-	KEY_KET : String
6-	KEY_LINK : String
7-	session : SessionManager
8-	URL : String
9+	onCreate(savedInstanceState : Bundle) : void
10+	onCreateOptionsMenu(menu : Menu) : boolean

Gambar 4.18 Kelas NotifikasiActivity SI Gapoktan Mobile
Sumber : [Perancangan]

1. Baris 1 merupakan deklarasi variabel listview yang digunakan untuk menampilkan data list ke interface.
2. Baris 3-6 merupakan deklarasi variabel Key node XML yang di kirimkan oleh variabel web service pesan. data yang dikirimkan merupakan data tanggal, keterangan, dan link.
3. Baris 8 merupakan deklarasi yang menyimpan variabel URL web services data gapoktan.
4. Baris 9 merupakan method onCreate yang diakses pertama kali ketika kelas MessageActivity dijalankan. Pada method ini berisi logika-logika pengambilan data dari web services message dan bagaimana menampilkan data ke interface.
5. Baris 10 merupakan method yang digunakan untuk membuat menu help, about dan logout.

F. PostActivity

Kelas PostActivity berfungsi sebagai jembatan antara data info gapoktan dan *interface* info gapoktan. Gambar 4.19 menjelaskan struktur dari kelas PostActivity.

PostActivity	
1	KEY_DATA : String
2	KEY_NAMA : String
3	KEY_POST : String
4	KEY_COMMENT : String
5	list : ListView
6	adapter : LazyAdapter
7	session : SessionManager
8	URL : String
9+	onCreate(savedInstanceState : Bundle) : void
10+	onOptionsItemSelected(menu : Menu) : boolean

Gambar 4.19 Kelas PostActivity SI Gapoktan Mobile
Sumber : [Perancangan]

1. Baris 1-4 merupakan deklarasi variabel Key node XML yang di kirimkan oleh variabel web service info gapoktan. Data yang dikirimkan merupakan data nama, post, dan komentar.
2. Baris 5 merupakan deklarasi variabel listview yang digunakan untuk menampilkan data list ke interface.
3. Baris 8 merupakan deklarasi yang menyimpan variabel URL web services data gapoktan.
4. Baris 9 merupakan methon onCreate yang diakses pertama kali ketika lekas PostActivity dijalankan. Pada method ini berisi logika-logika pengambilan data dari web services info gapoktan dan bagaimana menampilkan data ke interface.
5. Baris 10 merupakan method yang digunakan untuk membuat menu help, about dan logout.

G. UsahaActivity

Kelas UsahaActivity berfungsi sebagai jembatan antara data produk gapoktan dan *interface* produk gapoktan. Gambar 4.20 menjelaskan struktur dari kelas UsahaActivity.

UsahaActivity
1- URL : String
2- KEY_DATA : String
3- KEY_NAMA : String
4- KEY_DES : String
5- KEY_HARGA : String
6- KEY_NO : String
7- adapter : LazyAdapter
8- session : SessionManager
9- postlist : ArrayList<HashMap<String,String>>
10- parser : XMLParser
11+ onCreate(savedInstanceState : Bundle) : void
12+ onCreateOptionsMenu(menu : Menu) : boolean

Gambar 4.20 Kelas UsahaActivity SI Gapoktan Mobile
Sumber : [Perancangan]

1. Baris 1 merupakan deklarasi yang menyimpan variabel URL web services data gapoktan.
2. Baris 2-6 merupakan deklarasi variabel Key node XML yang di kirimkan oleh variabel web service produk gapoktan. Data yang dikirimkan merupakan data nama, deskripsi, dan harga, nomor.
3. Baris 10 digunakan untuk deklarasi variabel parser yang digunakan untuk menerima data XML dari web services produk gapoktan.
4. Baris 11 merupakan methon onCreate yang diakses pertama kali ketika lekas UsahaActivity dijalankan. Pada method ini berisi logika-logika pengambilan data dari web services produk gapoktan dan bagaimana menampilkan data ke interface.
5. Baris 12 merupakan method yang digunakan untuk membuat menu help, about dan logout.

H. DetailUsahaActivity

Kelas DetailUsahaActivity berfungsi sebagai jembatan antara data detail usaha gapoktan dan *interface* detail usaha gapoktan. Gambar 4.21 menjelaskan struktur dari kelas DetailUsahaActivity. Pada kelas ini akan ditampilkan detail produk yang telah dipilih oleh user. Pada kelas ini juga akan dilengkapi foto produk yang dipilih.

DetailUsahaActivity	
1-	KEY_DATA : String
2-	KEY_NAMA : String
3-	KEY_HARGA : String
4-	KEY_TANGGAL : String
5-	KEY_GAPOKTAN : String
6-	KEY_STOK : String
7-	KEY_KET : String
8-	
9-	adapter : LazyAdapter
10-	session : SessionManager
11-	parser : XMLParser
12-	KEY_NO : String
13-	URL : String
14-	e : Element
15-	id : String
16+	onCreate(savedInstanceState : Bundle) : void
17+	onCreateOptionsMenu(menu : Menu) : boolean
18+	profilClick(v : View) : void

Gambar 4.21 Kelas DetailUsahaActivity SI Gapoktan *Mobile*

Sumber : [Perancangan]

1. Baris 1-7 merupakan deklarasi variabel Key node XML yang di kirimkan oleh variabel web service detail produk gapoktan. Data yang dikirimkan merupakan data nama produk, harga, tanggal, nama gapoktan, jumlah stok dan keterangan.
2. Baris 11 merupakan deklarasi variabel parser yang digunakan untuk menyimpan data XML dari web service detail produk gapoktan
3. Baris 13 merupakan deklarasi yang menyimpan variabel URL web services data gapoktan.
4. Baris 16 merupakan methon onCreate yang diakses pertama kali ketika lekas DetailUsahaActivity dijalankan. Pada method ini berisi logika-logika pengambilan data dari web services detail produk gapoktan dan bagaimana menampilkan data ke interface.
5. Baris 17 merupakan method yang digunakan untuk membuat menu help, about dan logout.
6. Baris 18 merupakan prthod profilClick. Method ini bekerja jika ada penekanan pada button contact. Method ini berfungsi untuk mengakses kelas profil gapoktan pemilik produk.

I. DetailProfilActivity

Kelas DetailProfilActivity berfungsi sebagai jembatan antara data detail profil gapoktan dan *interface* detail profil gapoktan. Gambar 4.22 menjelaskan struktur dari kelas DetailProfilActivity.

DetailProfilActivity	
1	KEY_DATA : String
2	KEY_NAMA : String
3	KEY_ALAMAT : String
4	KEY_TELEPON : String
5	KEY_EMAIL : String
6	KEY_KETUA : String
7	KEY_KEC : String
8	KEY_KOTA : String
9	KEY_PROP : String
10	list : ListView
11	adapter : LazyAdapter
12	postlist : ArrayList<HashMap<String,String>>
13	URL : String
14+	onCreate(savedInstanceState : Bundle) : void
15+	onCreateOptionsMenu(menu : Menu) : boolean

Gambar 4.22 Kelas DetailProfilActivity SI Gapoktan Mobile
Sumber : [Perancangan]

1. Baris 1-9 merupakan deklarasi variabel Key node XML yang di kirimkan oleh variabel web service detail profil gapoktan. Data yang dikirimkan merupakan data nama, alamat, telepon, email, ketua, kecamatan, kota, dan propinsi.
2. Baris 10 merupakan deklarasi variabel listview yang digunakan untuk menampilkan data list ke interface.
3. Baris 13 merupakan deklarasi yang menyimpan variabel URL web services data gapoktan.
4. Baris 14 merupakan methon onCreate yang diakses pertama kali ketika lekas DetailProfilActivity dijalankan. Pada method ini berisi logika-logika pengambilan data dari web services detail profil gapoktan dan bagaimana menampilkan data ke interface.
5. Baris 15 merupakan method yang digunakan untuk membuat menu help, about dan logout.

J. ProfilActivity

Kelas ProfilActivity berfungsi sebagai jembatan antara data profil user yang *login* dan *interface* profil gapoktan. Gambar 4.23 menjelaskan struktur dari kelas ProfilActivity.

ProfilActivity	
1-	KEY_DATA : String
2-	KEY_NAMA : String
3-	KEY_ALAMAT : String
4-	KEY_TELEPON : String
5-	KEY_EMAIL : String
6-	KEY_KETUA : String
7-	KEY_KEC : String
8-	KEY_KOTA : String
9-	KEY_PROP : String
10-	list : ListView
11-	adapter : LazyAdapter
12-	postlist : ArrayList<HashMap<String,String>>
13-	URL : String
14+	onCreate(savedInstanceState : Bundle) : void
15+	onCreateOptionsMenu(menu : Menu) : boolean

Gambar 4.23 Kelas ProfilActivity SI Gapoktan Mobile
Sumber : [Perancangan]

1. Baris 1-9 merupakan deklarasi variabel Key node XML yang di kirimkan oleh variabel web service profil gapoktan. Data yang dikirimkan merupakan data nama, alamat, telepon, email, ketua, kecamatan, kota, dan propinsi.
2. Baris 10 merupakan deklarasi variabel listview yang digunakan untuk menampilkan data list ke interface.
3. Baris 13 merupakan deklarasi yang menyimpan variabel URL web services data gapoktan.
4. Baris 14 merupakan methon onCreate yang diakses pertama kali ketika lekas ProfilActivity dijalankan. Pada method ini berisi logika-logika pengambilan data dari web services profil gapoktan dan bagaimana menampilkan data ke interface.
5. Baris 15 merupakan method yang digunakan untuk membuat menu help, about dan logout.

K. LoginActivity

Kelas LoginActivity berfungsi sebagai halaman pertama yang diakses user ketika pertama kali masuk SI Gapoktan *Mobile*. Gambar 4.24 menjelaskan struktur kelas LoginActivity.

LoginActivity	
1-	nama : String
2-	pass : String
3-	URL : String
4-	KEY_DATA : String
5-	KEY_NAMA : String
6-	KEY_STATUS : String
7-	KEY_ID : String
8-	KEY_EMAIL : String
10-	alert : AlertDialogManager
11+	onCreate(savedInstanceState : Bundle) : void
12+	onCreateOptionsMenu(menu : Menu) : boolean
13+	buttonClick(v : View) : void

Gambar 4.24 Kelas LoginActivity SI Gapoktan *Mobile*
Sumber : [Perancangan]

1. Baris 1-2 merupakan deklarasi variabel yang digunakan untuk menampung nama dan password.
2. Baris 3 merupakan deklarasi yang menyimpan variabel URL web services data gapoktan.
3. Baris 4-9 merupakan deklarasi variabel Key node XML yang di kirimkan oleh variabel web service profil gapoktan. Data yang dikirimkan merupakan data nama, status login, id, dan email.
4. Baris 10 digunakan untuk mempersiapkan alert dialog manager.
5. Baris 11 merupakan methon onCreate yang diakses pertama kali ketika lekas LoginActivity dijalankan. Pada method ini berisi logika menampilkan kolom username dan password dan bagaimana mengambil data tersebut untuk dikirimkan ke web services login.
6. Baris 15 merupakan method yang digunakan untuk membuat menu help, about dan logout.

4.2.3.3 Pemodelan Kelas Komponen Session Manager

Komponen session manager merupakan umpulan kelas yang berfungsi menjaga user tetap *login* hingga user menekan tombol logout. Komponen session manager terdiri dari dua kelas yaitu SessionManager dan AlertDialogManager.

1. Kelas SessionManager

Kelas session manager berfungsi untuk menyimpan *username* dan *Email* di dalam session. Gambar 4.24 menjelaskan struktur kelas SessionManager

SessionManager
1- pref : SharedPreferences
2- editor : Editor
3- context : Context
4- PRIVATE_MODE : int
5- PREF_NAME : String
6- KEY_NAME : String
7- KEY_EMAIL : String
8- KEY_ID : String
9+ SessionManager(context : Context) : void
10+ createLoginSession(id : String, name : String, email : String) : void
11+ checkLogin() : void
12+ getUserDetail() : HashMap<String,String>
13+ logoutUser() : void
14+ isLoggedIn() : boolean

Gambar 4.25 Kelas SessionManager SI Gapoktan Mobile
Sumber : [Perancangan]

1. Baris 6-8 merupakan deklarasi variabel key nama, email dan id user.
 2. Baris 10 merupakan method createLoginSession. Method ini digunakan untuk membuat session baru.
 3. Baris 11 merupakan method checkLogin yang digunakan untuk pengecekan apakah user sedang login atau tidak.
 4. Baris 12 merupakan method getUserDetail digunakan untuk mengetahui data user yang sedang login.
 5. Baris 13 merupakan method yang digunakan untuk logout user dengan cara menghapus session.
- ##### 2. Kelas AlertDialogManager

Kelas AlertDialogManager berfungsi untuk menampilkan peringatan berupa pop up window di SI Gapoktan *Mobile*. Gambar 4.25 adalah gambar yang menjelaskan struktur dari kelas AlertDialogManager.

AlertDialogManager
1+ showAlartDialog(context : Context, title : String, message : String) : void

Gambar 4.26 Kelas AlertDialogManager SI Gapoktan *Mobile*

Sumber : [Perancangan]

Pada kelas AlertDialogManager terdapat method showAlertDialog yang berfungsi menampilkan alert pop up window.

4.2.3.4 Pemodelan Kelas Komponen GCDM

Komponen GCDM merupakan kumpulan kelas yang berfungsi untuk menerima notifikasi yang dikirimkan oleh sistem informasi gapoktan melalui *google web service*. Komponen ini juga memiliki peran mengirim *Email* dan *username* untuk di registrasikan ke *google web service push notification*. Komponen ini terdiri dari kelas CommonUtilities, ConnectionDetetctor, GCMIntentService, MainActivity, ServerUtilities, dan WakeLocker.

1. Kelas CommonUtilities

Kelas CommonUtilities berfungsi untuk menampilkan notifikasi ke *interface* SI Gapoktan *Mobile*. Gambar 4.27 menjelaskan kelas CommonUtilities.

CommonUtilities
1- SERVER_URL : String
2- SENDER_ID : String
3- TAG : String
4- DISPLAY_MESSAGE_ACTION : String
5- EXTRA_MESSAGE : String
6+ displayMessage(context : Context, message : String) : void

Gambar 4.27 Kelas CommonUtilities SI Gapoktan *Mobile*

Sumber : [Perancangan]

1. Baris 1 mendeklarasikan variabel SERVER_URL yang berfungsi untuk menyimpan URL web service register sistem informasi gapoktan.
2. Baris 2 mendeklarasikan variabel SENDER_ID berisi API_KEY dari web services *web service* google push notification.
3. Baris 6 merupakan method displayMessage berfungsi untuk menampilkan pesan ke *interface*.

2. Kelas ConnectionDetector

Kelas ConnectionDetector berfungsi untuk mengetahui apakah SI Gapoktan Mobile terhubung dengan koneksi internet atau tidak sebelum mendaftarkan Email ke web services google. Gambar 4.28 menjelaskan struktur dari kelas ConnectionDetector.

ConnectionDetector
1- context : Context
2+ ConnectionDetector(context : Context) : void
3+ isConnectingToInternet() : boolean

Gambar 4.28 Kelas ConnectionDetector SI Gapoktan Mobile
Sumber : [Perancangan]

1. Baris 1 merupakan deklarasi variabel context.
 2. Baris 2 merupakan method konstruktor yang pertama kali diakses ketika user mengakses kelas ConnectionDetector.
 3. Baris 3 merupakan method isConnectionto Internet yang berfungsi untuk mengetahui apakah device terhubung dengan internet atau tidak.
3. Kelas GCMInternetService

Kelas GCMInternetService berisi logika penyusunan data notifikasi, pengecekan registrasi dengan google push notification, pendaftaran google push notification, dan pengecekan bila ada error di notifikasi. Gambar 4.29 menjelaskan struktur kelas CGMInternetServices.

GCMIntentService
1- TAG : String
2+ GCMIntentService() : void
3+ onRegistered(context : Context, registrationId : String) : void
4+ onUnregistered(context : Context, registrationId : String) : void
5+ onMessage(context : Context, intent : Intent) : void
6+ onDeleteMessage(context : Context, total : int) : void
7+ onError(context : Context, errorId : String) : void
8+ onRecoverlabError(context : Context, errorId : String) : boolean
9+ generateNotificaton(context : Context, message : String) : void

Gambar 4.29 Kelas GCMInternetServices SI Gapoktan Mobile
Sumber : [Perancangan]

1. Baris 2 merupakan konstruktor yang pertama diakses ketika masuk ke kelas GCM Intent Service.

2. Baris 3 method ini akan dipanggil jika device telah terdaftar dalam *Google Cloud to Device Messaging* (GCDM). Method ini menampilkan notifikasi kepada user.
 3. Baris 4 method ini akan dipanggil jika device belum terdaftar dalam *Google Cloud to Device Messaging* (GCDM). Method ini menampilkan notifikasi kepada user dan mengirimkan `registrationId` kepada C2DM.
 4. Baris 5 method `onMessage` diterima jika ada pesan baru yang diterima oleh device.
 5. Baris 7 diterima jika ada error dalam pengiriman pesan.
 6. Baris 9 merupakan method `generateNotification` yang digunakan untuk membentuk pesan notifikasi. Pada method ini di load icon, pengesetan getar, dan memutar suara nada dering.
4. Kelas `MainActivity`

Kelas `MainActivity` berfungsi sebagai controller dari GCDM. Pada kelas `MainActivity` terdapat parameter intent `username` dan `Email` yang akan di daftarkan ke `google push notification web services`. Gambar 4.30 menjelaskan struktur dari kelas `MainActivity`.

MainActivity
1- lblMessage : TextView
2- mRegisterTask : AsyncTask<void,void,void>
3- cd : ConnectionDetector
4- name : String
5- email : String
6+ onCreate(savedInstanceState : Bundle) : void
7+ onDestroy() : void

Gambar 4.30 Kelas `MainActivity` SI Gapoktan *Mobile*
Sumber : [Perancangan]

1. Baris 3 digunakan sebagai variabel penyimpanan hasil pengecekan deteksi apakah device terkoneksi dengan internet atau tidak.
2. Baris 4 – 5 merupakan deklarasi variabel yang digunakan untuk menampung data nama dan email.
3. Baris 6 diakses ketika kelas `MainActivity` di akses. Pada kelas ini berisi logika registrasi device dan pengecekan device.

5. Kelas ServerUtilities

Kelas ServerUtilities berfungsi sebagai penghubung ke server. Gambar 4.31 menjelaskan struktur kelas ServerUtilities.

ServerUtilities
1- MAX_ATTEMPS : int 2- BACKOFF_MILLI_SECOND : int 3- random : Random
4+ register(context : Context, name : String, email : String, regId : String) : void 5+ unRegister(context : Context, regId : String) : void 6+ post(endpoint : String, params : Map<String,String>) : void

Gambar 4.31 Kelas ServerUtilities SI Gapoktan Mobile
Sumber : [Perancangan]

1. Baris 4 merupakan method register yang berfungsi mengirimkan username, email, dan registrationId ke server.
2. Baris 5 merupakan method unRegister yang berfungsi memutus layanan GCDM.

kelas ServerUtilities terdapat method register yang berfungsi untuk mendaftarkan akun dan membuat koneksi ke *web service* google push notification. Method unRegister berfungsi untuk memutus koneksi akun dengan server. Sedangkan method post berfungsi untuk mengirim data parameter ke server.

6. Kelas WakeLocker

Kelas Wakelocker berfungsi agar CPU tetap melakukan pengecekan jika sewaktu-waktu ada notifikasi masuk. Gambar 4.32 menjelaskan struktur dari kelas WakeLocker.

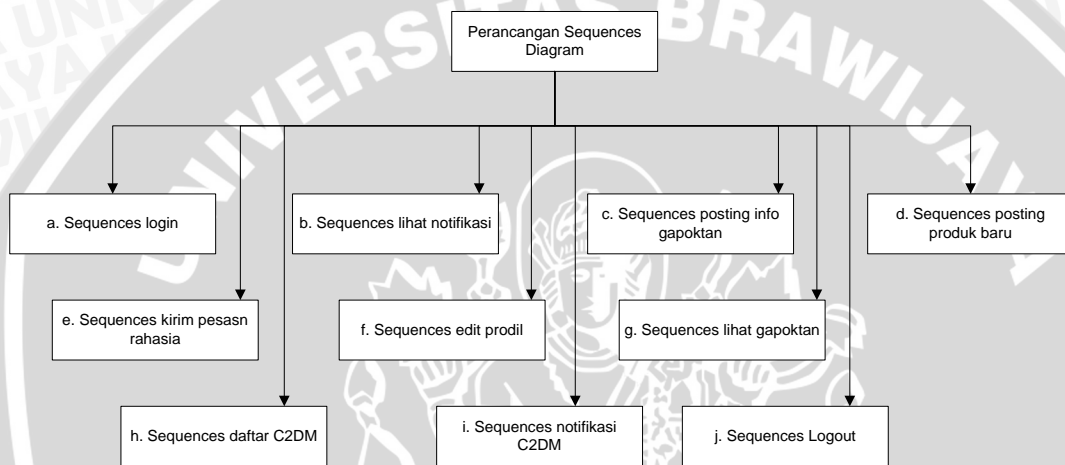
WakeLocker
1- wakelock : PowerManager.WakeLock
2 + acquire(context : Context) : void 3 + release() : void

Gambar 4.32 Kelas WakeLocker SI Gapoktan Mobile
Sumber : [Perancangan]

4.2.3 Perancangan Sequences Diagram

Sequence diagram menunjukkan pemodelan aliran jalannya proses interaksi antar objek atau *class* yang disusun berdasarkan urutan waktu pada SI

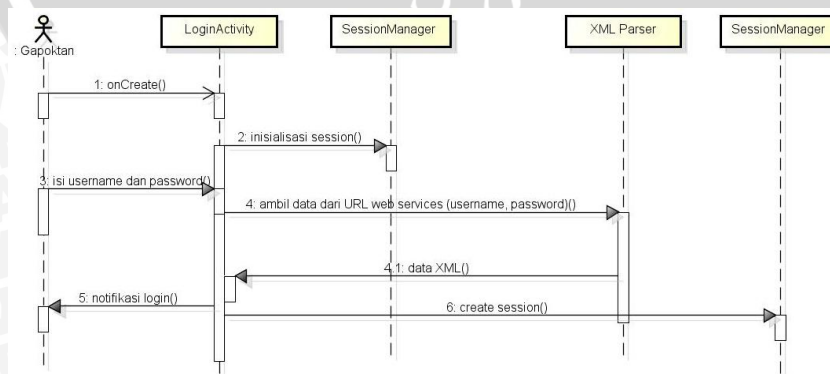
Gapoktan *Mobile*. *Sequence diagram* digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Diagram sekuensial (*sequence diagram*) disusun dengan mengambil acuan pada *use case* dan *class -class* yang membentuk fungsionalitas yang digambarkan pada *use case* tersebut. Pada SI Gapoktan *Mobile sequence diagram* terdiri dari *sequence diagram login*, lihat notifikasi, posting info gapoktan, posting produk baru, kirim pesan rahasia, edit profil, lihat list gapoktan, kirim komentar, notifikasi GCDM, dan logout. Gambar 4.33 menjelaskan diagram blok perancangan *sequences diagram*.



Gambar 4.33 Diagram blok perancangan *sequences diagram*
Sumber : [Perancangan]

A. *Sequence diagram login*

Sequences diagram login menggambarkan proses ketika user akan masuk ke dalam SI Gapoktan *Mobile*. Gambar 4.34 menjelaskan *sequences diagram login*.

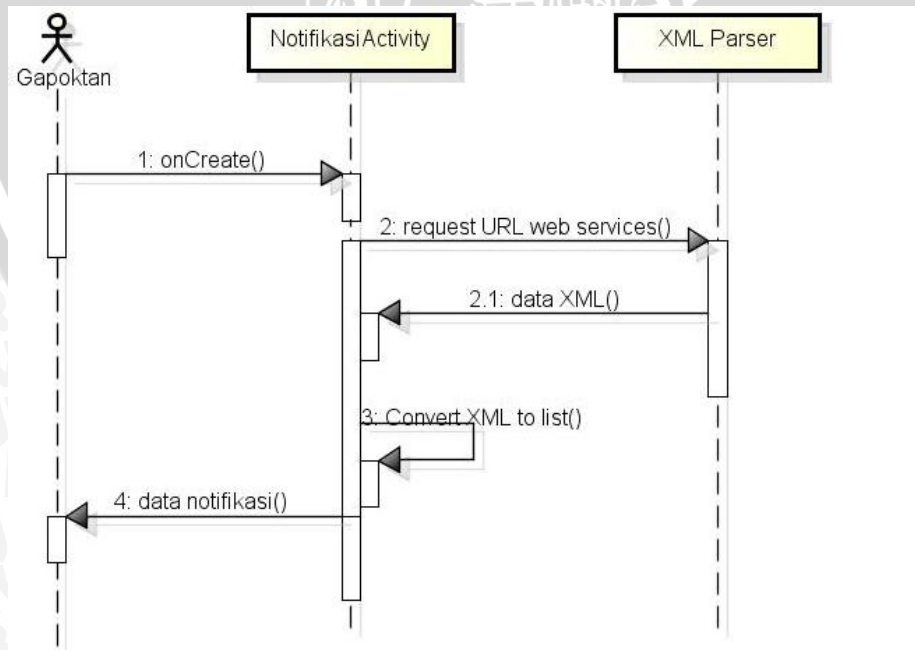


Gambar 4.34 *Sequence diagram login* SI Gapoktan *Mobile*
Sumber : [Perancangan]

1. User akan mengakses kelas *LoginActivity* dan menampilkan *interface login*.
2. Melakukan inisialisasi session pada kelas *SessionManager*.
3. User memasukkan *username* dan *password* ke dalam kolom yang telah disediakan. Setelah *username* dan *password* diisi, maka user menekan tombol *login* untuk masuk ke *SI Gapoktan Mobile*.
4. Kemudian sistem akan memanggil kelas *XML Parser* untuk melakukan request pada URL *web service*. (4.1) *Web service* kemudian memberikan data dengan format *XML* yang akan di parsing kedalam element list.
5. Jika *username* dan *password* ada di *database*, maka nilai kembalian akan berupa “*login sukses*” dan user berhasil masuk ke sistem.
6. Proses terakhir adalah penyimpanan session user yang sedang *login*.

B. *Sequences diagram* lihat notifikasi

Sequences diagram lihat notifikasi menggambarkan proses user melihat notifikasi pada *SI Gapoktan Mobile*. Gambar 4.35 menjelaskan *sequences diagram* *lihat notifikasi*.



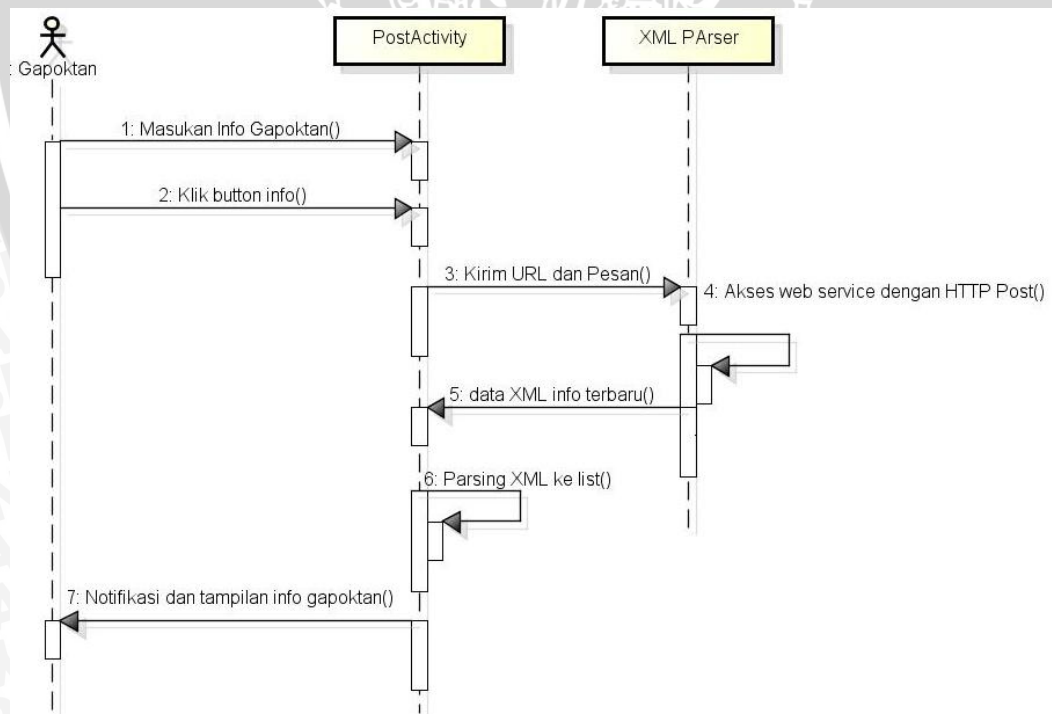
Gambar 4.35 *Sequence diagram* login *SI Gapoktan Mobile*

Sumber : [Perancangan]

1. User akan masuk ke dalam kelas `NotifikasiActivity`.
2. Notifikasi kemudian memanggil kelas XML parser untuk mengunduh data ke *web service*.
3. Kemudian *web service* memberi respon dengan mengirimkan data berupa XML.
4. Kemudian XML di parsing menjadi list di kelas `NotifikasiActivity` dengan bantuan kelas `element`.
5. Jika XML berhasil diparsing, maka data akan ditampilkan di *interface* dan muncul notifikasi kepada user.

C. *Sequences diagram* posting info gapoktan

Sequences diagram posting info gapoktan menggambarkan proses ketika user akan menginfokan kegiatan yang dilakukan gapoktan saat ini. User akan masuk terlebih dulu ke halaman Info gapoktan atau kelas `PostActivity`. Gambar 4.36 menjelaskan *sequences diagram* posting info gapoktan.

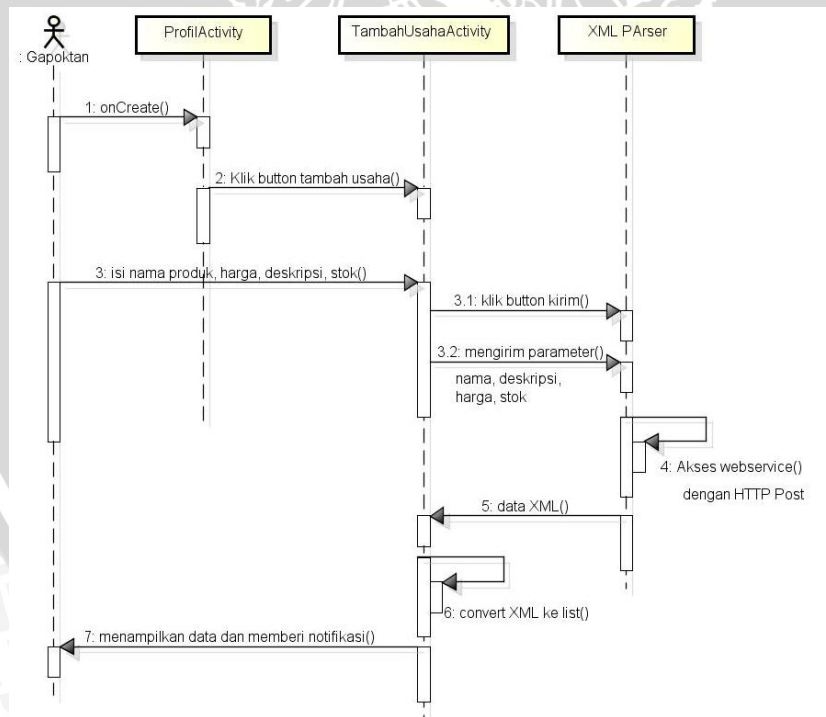


Gambar 4.36 *Sequence diagram* posting info gapoktan SI Gapoktan Mobile
Sumber : [Perancangan]

1. User mengisikan info terbaru dikolom info yang telah disediakan.
2. Info akan dikirimkan setelah user menekan button info.
3. Kelas PostActivity akan memanggil kelas XML Parser dan mengirimkan URL serta info yang telah ditulis oleh user.
4. XML melakukan request ke URL *web service* dengan parameter info gapoktan menggunakan method HTTP Post.
5. *Web service* memberikan balasan berupa data XML yang berisi kumpulan info gapoktan terbaru.
6. Data XML diparsing ke list view.
7. Proses terakhir adalah memunculkan notifikasi dan menampilkan data posting ke *interface*.

D. *Sequences diagram* posting produk baru

Sequences diagram posting produk baru menggambarkan proses ketika user akan menambah produk terbaru. Gambar 4.37 menjelaskan *sequences diagram* tambah produk gapoktan.



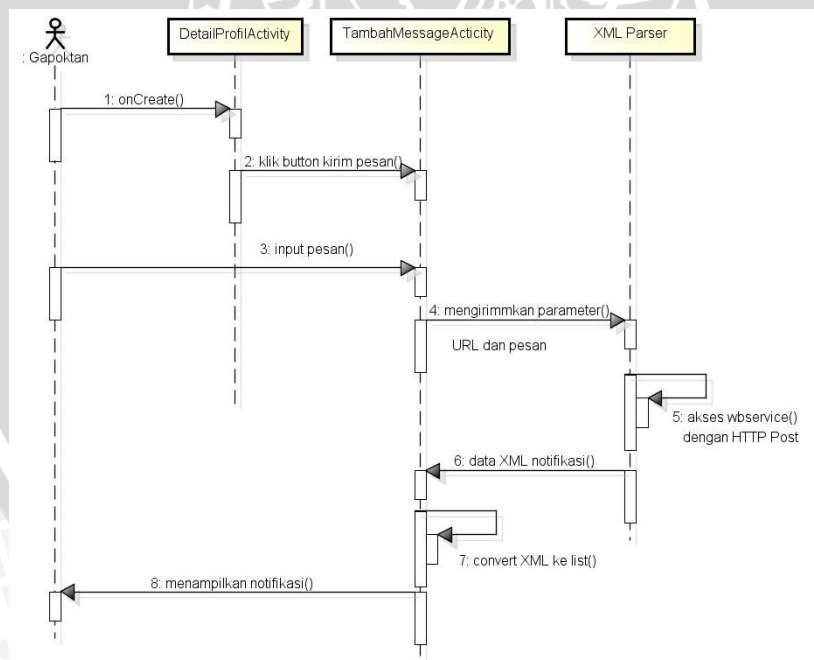
Gambar 4.37 *Sequence diagram* tambah produk gapoktan SI Gapoktan Mobile
Sumber : [Perancangan]

1. User masuk terlebih dahulu ke halaman profil atau ProfilActivity.
2. user menekan button tambah produk.

3. Setelah masuk ke kelas Tambah-ProdukActivity, user mengisikan nama produk, deskripsi, harga, dan jumlah stok produk baru serta mengunggah foto dari produk tersebut. (3.1)Kemudian user menekan button kirim untuk memproses posting produk terbaru. (3.2)TambahProdukActivity akan memanggil kelas XML Parser untuk mengirimkan URL dan parameter.
4. XML Parser mengirimkan parameter ke URL *web service* menggunakan method HTTP Post.
5. Jika data berhasil di inputkan maka akan ada notifikasi berupa data XML bahwa data berhasil di inputkan.
6. Kemudian data XML diparsing ke String.
7. Data string ditampilkan dalam bentuk Toast di *interface*.

E. *Sequences diagram* kirim pesan rahasia

Sequences diagram kirim pesan rahasia menggambarkan proses ketika user akan mengirim pesan rahasia ke gapoktan lain. Gambar 4.38 menjelaskan *sequences diagram* tambah kirim pesan rahasia.



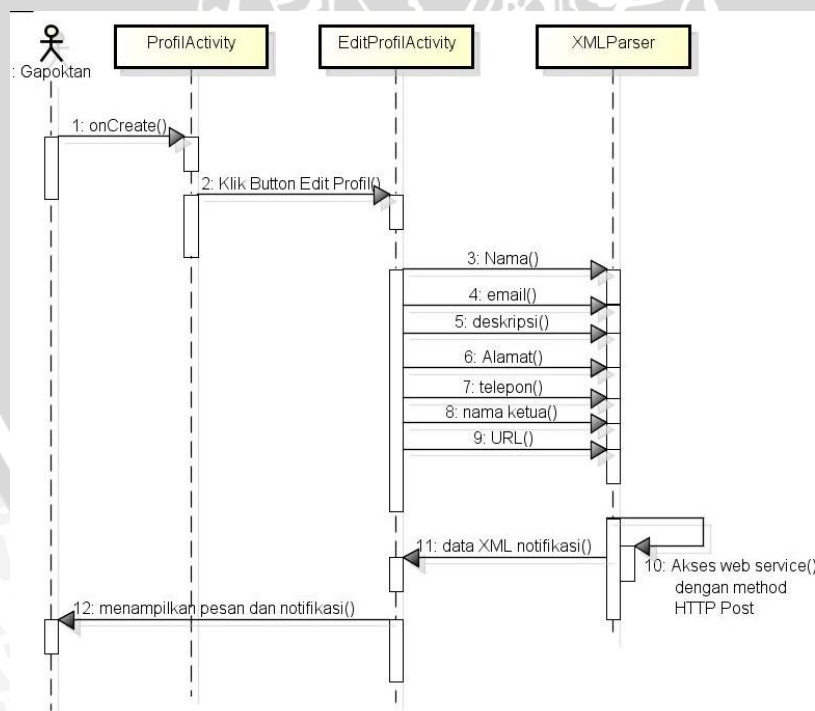
Gambar 4.38 *Sequence diagram* kirim pesan rahasia SI Gapoktan Mobile
Sumber : [Perancangan]

1. User masuk terlebih dahulu memilih gapoktan yang akan dikirim pesan atau masuk ke kelas DetailGapoktanActivity.

2. User kemudian menekan button kirim pesan.
3. Setelah masuk ke kelas KirimPesanActivity, user menuliskan pesan yang akan dikirim . Kemudian user menekan button kirim untuk mengirim pesan rahasia.
4. KirimPesanActivity akan memanggil kelas XML Parser untuk mengirimkan URL dan isi pesan.
5. XML Parser mengirimkan isi pesan ke URL *web service* menggunakan method HTTP Post.
6. Jika data berhasil di inputkan maka akan ada notifikasi berformat XML bahwa data berhasil di inputkan.
7. Data XML kemudian diubah ke format string.
8. Data string ditampilkan pada *interface*.

F. *Sequences diagram* edit profil

Sequences diagram edit profil menggambarkan proses ketika user akan mengedit biodata profilnya. Gambar 4.39 menjelaskan *sequences diagram* edit profil.

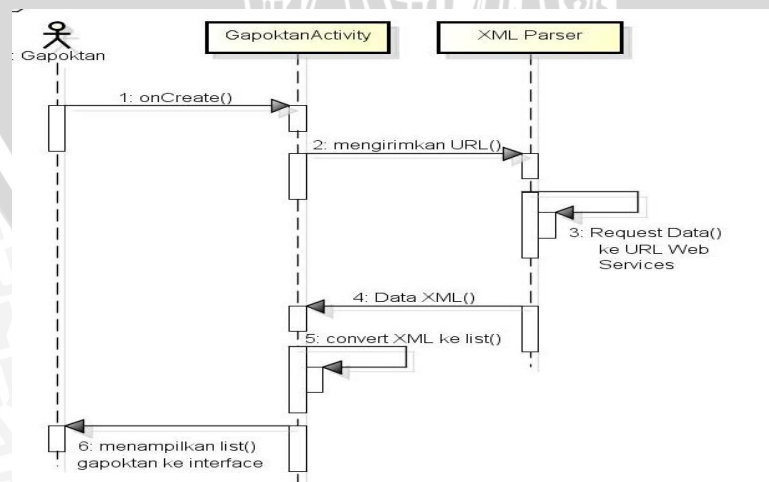


Gambar 4.39 *Sequence diagram* kirim pesan rahasia SI Gapoktan Mobile
Sumber : [Perancangan]

1. User masuk terlebih masuk ke halaman profil atau kelas ProfilActivity.
2. User kemudian menekan button edit profil.
3. User mengubah data nama pada kolom yang telah disediakan.
4. *Email* mengubah data nama pada kolom yang telah disediakan.
5. Deskripsi mengubah data nama pada kolom yang telah disediakan.
6. Alamat mengubah data nama pada kolom yang telah disediakan.
7. Telepon mengubah data nama pada kolom yang telah disediakan.
8. Ketua mengubah data nama pada kolom yang telah disediakan.
9. Kemudian user menekan button kirim untuk mengirim data profil yang baru. EditProfilActivity akan memanggil kelas XML Parser untuk mengirimkan URL dan biodata baru.
10. XML Parser mengirimkan biodata baru ke URL *web service* menggunakan method HTTP Post.
11. Jika data berhasil di inputkan maka akan ada notifikasi berupa XML bahwa biodata berhasil di ubah.
12. Notifikasi kemudian ditampilkan dalam bentuk toast dan tampilan pada halaman profil akan berubah menjadi data baru.

G. *Sequences diagram* lihat list gapoktan

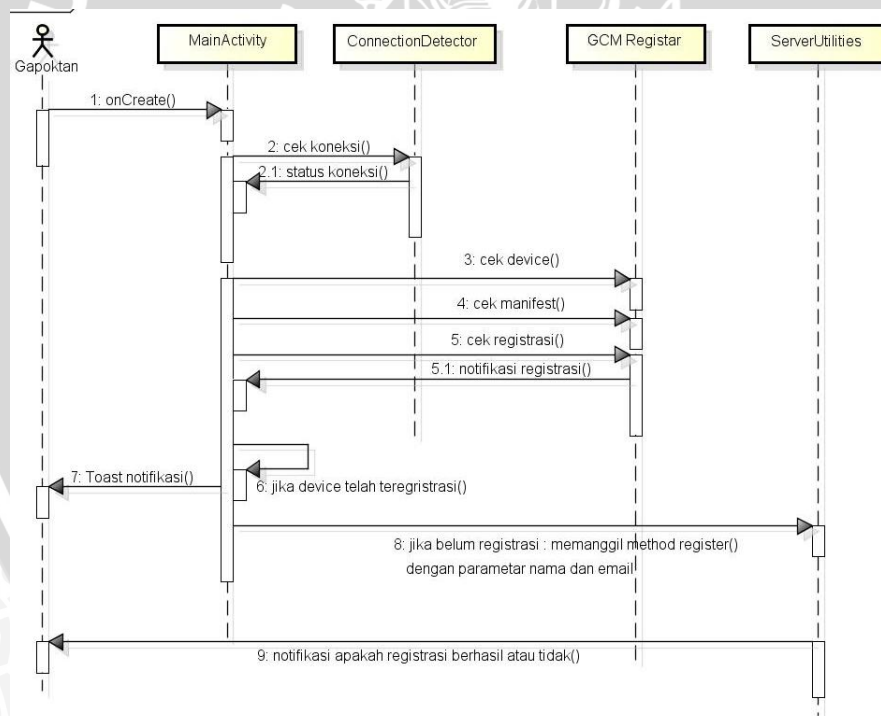
Sequences diagram lihat list gapoktan menggambarkan proses user melihat seluruh gapoktan yang terdaftar pada SI Gapoktan *Mobile*. Gambar 4.40 menjelelaskan *sequences diagram* lihat list gapoktan.



Gambar 4.40 *Sequence diagram* kirim pesan rahasia SI Gapoktan *Mobile*
Sumber : [Perancangan]

1. User akan masuk ke dalam halaman list gapoktan atau kelas GapoktanActivity.
 2. GapoktanActivity kemudian memanggil kelas XML parser untuk mengirimkan URL.
 3. XML Parser kemudian mengakses URL.
 4. *Web service* memberikan response data berformat XML.
 5. XML kemudian diparsing di kelas GapoktanActivity dengan bantuan kelas Element.
 6. Proses terakhir adalah menampilkan data list gapoktan ke *interface*
- H. *Sequences diagram* daftar GCDM

Sequences diagram daftar GCDM menggambarkan proses user mendaftarkan diri ke *web service* google push notification. Gambar 4.41 menjelaskan *sequences diagram* daftar GCDM.



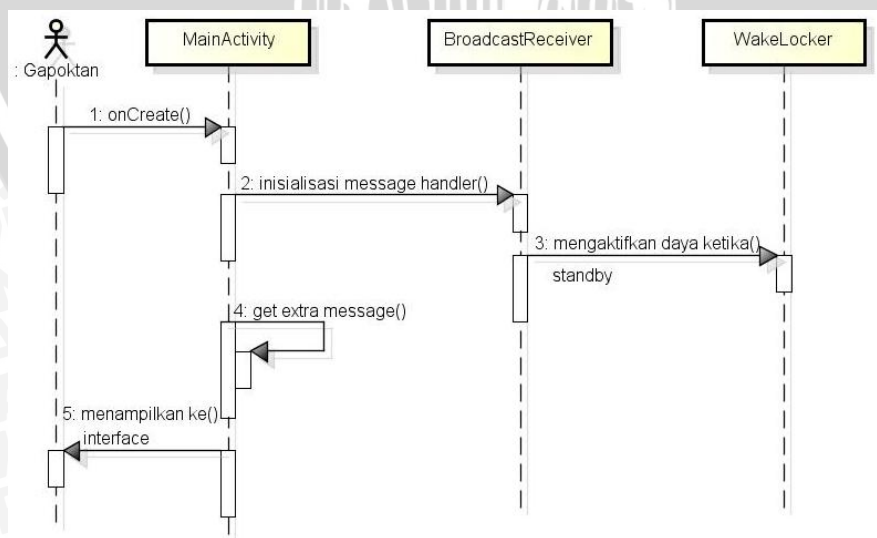
Gambar 4.41 *Sequence diagram* daftar GCDM SI Gapoktan Mobile
Sumber : [Perancangan]

1. User akan masuk ke kelas MainActivity.
2. Sistem kemudian melakukan pengecekan status koneksi.
3. Memanggil kelas GCM Registrar untuk melakukan pengecekan device

4. Melakukan pengecekan permission pada manifest.
5. Jika device sudah diketahui mendukung, maka akan dilakukan pengecekan apakah user sudah terdaftar pada google push notification. Kelas GCM Register merupakan kelas yang terdapat pada *package* `com.google.android.gcm.GCMRegistrar`. Kelas ini disediakan oleh google untuk membantu developer dalam mengakses method-method yang berkaitan dengan google push notification.
6. Mengecek apakah user sudah terregistrasi.
7. Menampilkan notifikasi berupa toast di *interface* sistem.
8. Jika user belum teregistrasi, maka akan ada dua proses. Proses pertama registrasi ke GCM Server dengan mengirimkan API_ID (SENDER_ID). Proses kedua mendaftarkan diri ke sistem informasi gapoktan menggunakan method register yang terdapat pada kelas ServerUtilities.
9. Jika pendaftaran telah selesai dilakukan, maka akan muncul pemberitahuan berupa toast pada *interface*.

I. *Sequences diagram* notifikasi GCDM

Sequences diagram daftar GCDM menggambarkan proses SI Gapoktan *Mobile* menerima notifikasi dari server. Gambar 4.42 menjelaskan *sequences diagram* notifikasi GCDM.

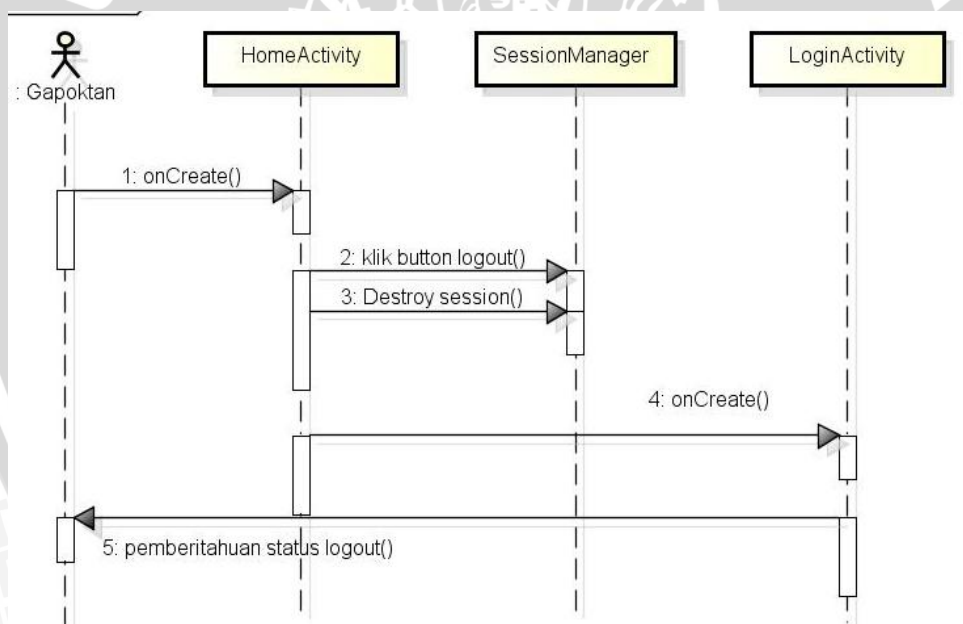


Gambar 4.42 *Sequence diagram* notifikasi GCDM SI Gapoktan *Mobile*
 Sumber : [Perancangan]

1. Pesan akan diterima oleh variabel `newMessage` pada kelas `MainActivity`.
2. Melakukan inisialisasi message handler sebagai objek penyimpanan pesan.
3. Kemudian sistem akan membangkitkan daya device jika device dalam mode standby.
4. Sistem mengakses method `getExtramessage` untuk mengambil data pesan.
5. Data ditampilkan ke *interface*.

J. *Sequences diagram logout*

Sequences diagram logout menggambarkan proses ketika user akan keluar dari SI Gapoktan *Mobile*. Gambar 4.43 menjelelaskan *sequences diagram logout*.



Gambar 4.43 *Sequence diagram logout* SI Gapoktan *Mobile*
Sumber : [Perancangan]

1. User harus berada pada halaman home atau kelas `HomeActivity`.
2. User menekan button logout yang telah disediakan.
3. Kelas `HomeActivity` akan memanggil kelas `session manager` untuk mendestroy session yang ada.

4. Jika session berhasil di destroy maka user akan masuk ke dalam halaman *login* atau kelas *LoginActivity*.
5. Proses terakhir adalah memunculkan pemberitahuan berupa toast ke user bahwa gapoktan telah berhasil keluar dari sistem.

4.2.4 Perancangan Data

Perancangan data bertujuan untuk mendapatkan struktur sumber data dan penyimpanan data yang dibutuhkan perangkat lunak. Data diperoleh dari penelitian terdahulu yang menjadi studi kasus. Data kemudian disimpan dalam database. Pemetaan didalam database direpresentasikan dalam bentuk pemodelan *entity relationship diagram* (ERD). Struktur data dari sistem informasi berbasis web ditunjukkan pada ERD berikut :

- a. Tabel Member

Tabel member berfungsi untuk menyimpan data-data member sistem informasi gapoktan berbasis *website*.

- b. Tabel *Posting*

Tabel *posting* berfungsi untuk menyimpan data-data posting *user*.

- c. Tabel Komentar

Tabel komentar berfungsi untuk menyimpan data komentar dari *posting user*

- d. Tabel Usaha

Tabel usaha berfungsi untuk menyimpan data usaha dari *user* gapoktan. Pada tabel ini dilengkapi dengan *field* keterangan yang menunjukkan usaha yang disimpan merupakan usaha primer atau sekunder.

- e. Tabel *Gcm_user*

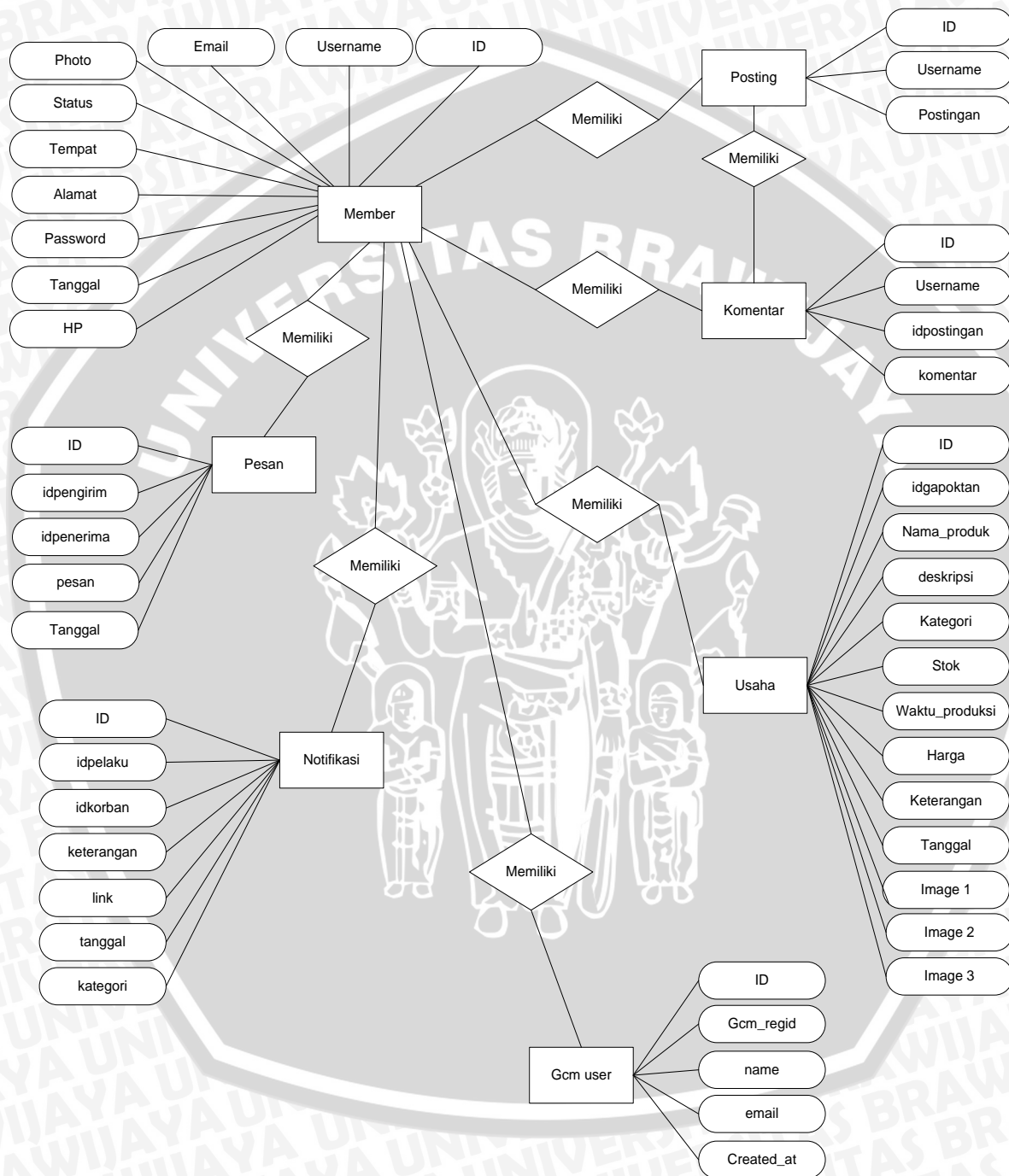
Tabel *gcm_user* berfungsi untuk menyimpan kode registrasi dari GCDM.

- f. Tabel Notifikasi

Tabel notifikasi berfungsi untuk menyimpan notifikasi yang diberikan kepada user.

g. Tabel Pesan

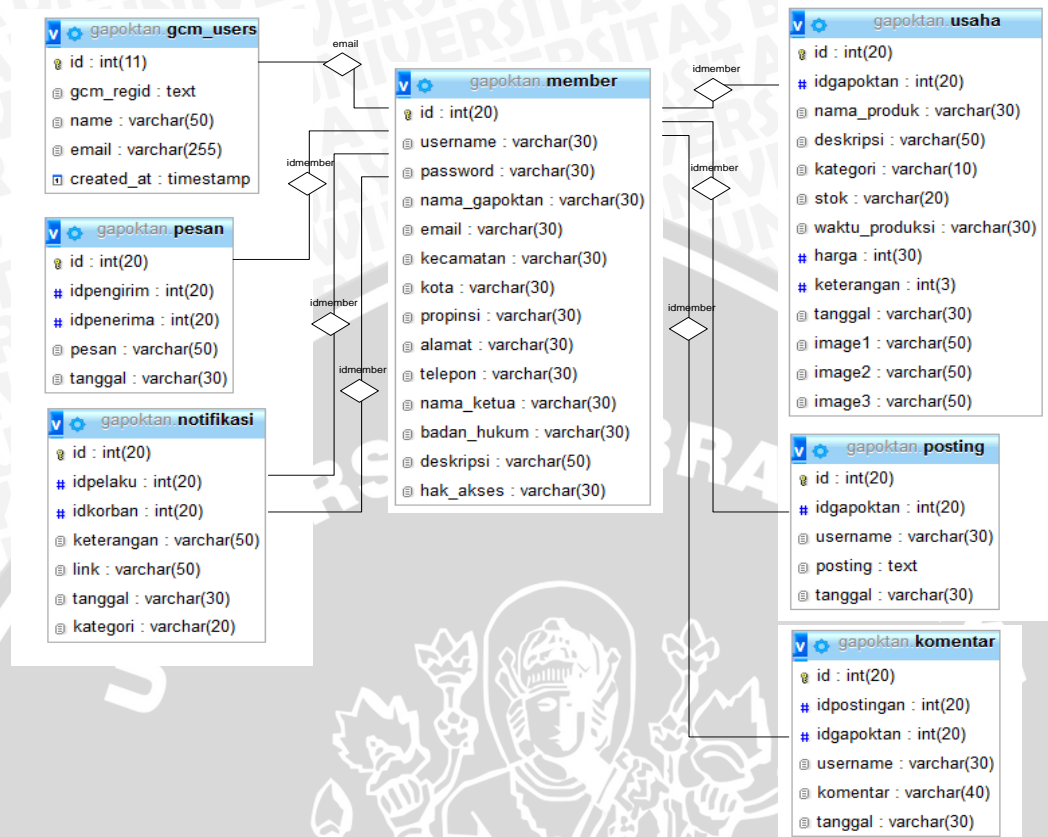
Tabel pesan berfungsi untuk menyimpan pesan rahasia antar gapoktan



Gambar 4.44 Entity relationship diagram sistem
Sumber : Perancangan

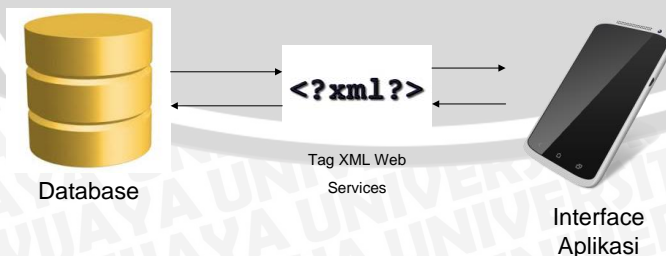


Physical Diagram dirancang berdasarkan ERD pada gambar 4.45



Gambar 4.45 Physical diagram sistem Sumber : Perancangan

Data dari database kemudian akan dipetakan ke dalam web services restfull dengan format data XML. Web services ini kemudian yang digunakan untuk SI Gapoktan *Mobile* dalam mengakses data sistem informasi gapoktan. User melakukan request data ke URL *web services*, kemudian *web services* mengambil data dari *database*. Gambar 4.8 menjelaskan diagram blok pengambilan dan penyimpanan data.



Gambar 4.46 Diagram blog penyimpanan dan pengambilan data perangkat lunak SI Gapoktan *Mobile* Sumber : [Perancangan]



Data XML kemudian diubah ke bentuk list yang nantinya ditampilkan di antarmuka SI Gapoktan *Mobile*. Berikut adalah URL *web services* yang digunakan oleh SI Gapoktan *Mobile*.

1. Data notifikasi

Data notifikasi akan dikirim ke *web service* notifikasi dan akan diteruskan ke tabel notifikasi. Tabel 4.18 menjelaskan alur data notifikasi dari *interface* ke *web service* ke XML dan *database*.

Tabel 4.18 Alur pemetaan data notifikasi

No.	Nama <i>Field database</i> notifikasi	Tag XML notifikasi	Atribut SI Gapoktan
1	Id	<No>	Id
2	Tanggal	<Tanggal>	KEY_TANGGAL
3	Keterangan	<Keterangan>	KEY_KET
4	Link	<Link>	KEY_LINK

Sumber : [Perancangan]

2. Data produk gapoktan

Data produk gapoktan akan dikirim ke *web service* produk gapoktan dan akan diteruskan ke tabel usaha. Tabel 4.19 menjelaskan alur data produk gapoktan dari *interface* ke *web service* ke XML dan *database*.

Tabel 4.19 Alur pemetaan data produk gapoktan

No.	Nama <i>Field database</i> usaha	Tag XML produk gapoktan	Atribut SI Gapoktan
1	id	<No>	Id
2	Idgapoktan	<Idgapoktan>	Idgapoktan
3	Nama_produk	<Nama>	KEY_NAMA
4	Deskripsi	<Deskripsi>	KEY_DES
5	Harga	<Harga>	KEY_HARGA
6	Tanggal	<Tanggal>	KEY_TANGGAL
7	Keterangan	<Keterangan>	KEY_KET
8	Stok	<Stok>	KEY_STOK

Sumber : [Perancangan]

3. Data list gapoktan

Data list gapoktan akan dikirim ke *web service* list gapoktan dan akan diteruskan ke tabel member. Tabel 4.20 menjelaskan alur data list gapoktan dari *interface* ke *web service* ke XML dan *database*.

Tabel 4.20 Alur pemetaan data list gapoktan

No.	Nama <i>Field database</i> member	Tag XML list gapoktan	Atribut SI Gapoktan
1	Id	<No>	Id
2	<i>Username</i>	<Nama>	KEY_NAMA
3	<i>Email</i>	<Email>	KEY_EMAIL
4	Alamat	<Alamat>	KEY_ALAMAT
5	Kecamatan	<Kecamatan>	KEY_KEC
6	Kota	<Kota>	KEY_KOTA
7	Propinsi	<Propinsi>	KEY_PROP
8	Telepon	<Telepon>	KEY_TELP

Sumber : [Perancangan]

4. Data semua pesan

Data semua pesan akan dikirim ke *web service* pesan dan akan diteruskan ke tabel pesan. Tabel 4.21 menjelaskan alur data pesan

Tabel 4.21 Alur pemetaan data semua pesan

No.	Nama <i>Field database</i> pesan	Tag XML pesan	Atribut SI Gapoktan
1	Id	<No>	Id
2	<i>Username</i>	<Pengirim>	KEY_NAMA_PENGINRIM
3	Tanggal	<Tanggal>	KEY_TANGGAL
4	Pesan	<Isi>	KEY_PESAN

Sumber : [Perancangan]

5. Data detail gapoktan

Data detail gapoktan akan dikirim ke *web service* detail gapoktan dan akan diteruskan ke tabel member. Tabel 4.22 menjelaskan alur data detail gapoktan dari *interface* ke *web service* ke XML dan *database*.

Tabel 4.22 Alur pemetaan data detail gapoktan

No.	Nama <i>Field database</i> member	Tag XML detail gapoktan	Atribut SI Gapoktan
1	Id	<No>	Id
2	<i>Username</i>	<Nama>	KEY_NAMA
3	<i>Email</i>	<Email>	KEY_EMAIL
4	Alamat	<Alamat>	KEY_ALAMAT
5	Kecamatan	<Kecamatan>	KEY_KEC
6	Kota	<Kota>	KEY_KOTA
7	Propinsi	<Propinsi>	KEY_PROP
8	Telepon	<Telepon>	KEY_TELP
9	Nama_ketua	<Ketua>	KEY_KETUA
10	Badan_hukum	<Badanhukum>	KEY_BH

11	Deskripsi	<Deskripsi>	KEYPDES
----	-----------	-------------	---------

Sumber : [Perancangan]

6. Data detail produk

Data detail produk akan dikirim ke *web service* detail produk dan akan diteruskan ke tabel usaha. Tabel 4.23 menjelaskan alur data detail produk dari *interface* ke *web service* ke XML dan *database*.

Tabel 4.23 Alur pemetaan data detail produk gapoktan

No.	Nama <i>Field database</i> usaha	Tag XML detail produk	Atribut SI Gapoktan
1	id	<No>	Id
2	Idgapoktan	<Idgapoktan>	Idgapoktan
3	Nama_produk	<Nama>	KEY_NAMA
4	Deskripsi	<Deskripsi>	KEY_DES
5	Harga	<Harga>	KEY_HARGA
6	Tanggal	<Tanggal>	KEY_TANGGAL
7	Keterangan	<Keterangan>	KEY_KET
8	Stok	<Stok>	KEY_STOK
9	Waktu_produksi	<Waktu_produksi>	KEY_PRODUKSI
10	Kategori	<Kategori>	KEY_KATEGORI

Sumber : [Perancangan]

7. Data komentar

Data komentar akan dikirim ke *web service* komentar dan akan diteruskan ke tabel komentar. Tabel 4.24 menjelaskan alur data komentar dari *interface* ke *web service* ke XML dan *database*.

Tabel 4.24 Alur pemetaan komentar

No.	Nama <i>Field database</i> komentar	Tag XML komentar	Atribut SI Gapoktan
1	id	<No>	Id
2	<i>Username</i>	<Username>	KEY_NAMA
3	Tanggal	<Tanggal>	KEY_TANGGAL
4	Komentar	<Isi>	KEY_KOMENTAR

Sumber : [Perancangan]

8. Data Login

Data *login* akan dikirim ke *web service login* dan akan diteruskan ke tabel *member*. Tabel 4.25 menjelaskan alur data *login* dari *interface* ke *web service* ke XML dan *database*.

Tabel 4.25 Alur pemetaan data produk gapoktan

No.	Nama Field database member	Tag XML login	Atribut SI Gapoktan
1	id	<No>	Id
2	Username	<Username >	KEY_NAMA
3	Password	<Password>	KEY_PASSWORD
4	Status login	<Status>	KEY_STATUS

Sumber : [Perancangan]

4.2.5 Perancangan Antarmuka SI Gapoktan Mobile

Pada bagian ini akan dijelaskan tentang perancangan antarmuka perangkat lunak SI Gapoktan *Mobile*. Antarmuka perangkat lunak ini akan digunakan oleh pengguna untuk berinteraksi dengan sistem perangkat lunak SI Gapoktan *Mobile*. Antarmuka perangkat ini terdiri dari antarmuka *login*, antarmuka *home*, antarmuka *notifikasi*, antarmuka *produk*, antarmuka *info gapoktan*, antarmuka *message*, antarmuka *profil*, antarmuka *list gapoktan*, antarmuka *detail gapoktan*, antarmuka *detail message*, antarmuka *setting*, antarmuka *GCDM*, dan antarmuka *tambah gapoktan*.

1. Antarmuka Login

Antarmuka *login* merupakan antarmuka yang pertama diakses oleh user saat pertama kali masuk aplikasi. Gambar 4.47 menunjukkan perancangan antarmuka *login*. Pada antarmuka *login* terdapat 2 kolom *EditText* yang berfungsi sebagai tempat user memasukan data *username* dan *password*. Pada antarmuka ini juga terdapat tombol *login* yang berisi logika jika user ingin masuk ke dalam aplikasi.

Logo Sistem Informasi Gapoktan Android	
Username	
Kolom pengisian username	
Password	
Kolom pengisian password	
Button Login	
About US	Help

Gambar 4.47 Antarmuka *login* SI Gapoktan *Mobile*
Sumber : [Perancangan]

2. Antarmuka Home

Antarmuka home merupakan antarmuka utama SI Gapoktan *Mobile*. Pada antarmuka ini terdapat navigasi ke semua fitur yang terdapat dalam sistem. Gambar 4.48 menunjukkan perancangan antarmuka home.

Logo Sistem Informasi Gapoktan Android		
Button Logout		
Navigasi Notifikasi		
Navigasi Produk		
Navigasi Info Gapoktan		
Navigasi Profil		
Navigasi Message		
Navigasi List Gapoktan		
About US	Logout	Help

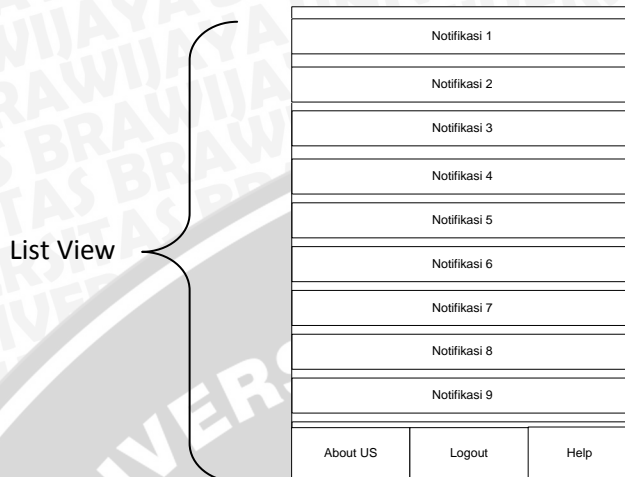
} Button navigasi

Gambar 4.48 Antarmuka *home* SI Gapoktan *Mobile*
Sumber : [Perancangan]



3. Antarmuka notifikasi

Antarmuka notifikasi berisi semua notifikasi yang berhubungan dengan user aktif.

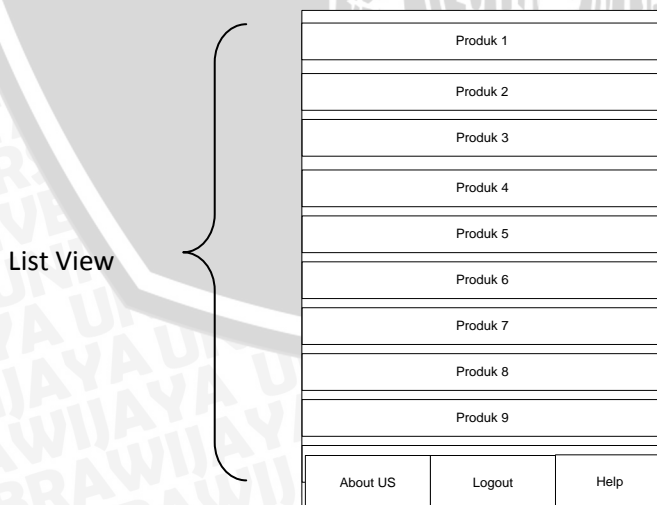


Gambar 4.49 Antarmuka notifikasi SI Gapoktan Mobile
Sumber : [Perancangan]

Notifikasi ini ditampikan dalam bentuk list mulai dari notifikasi pertama hingga terakhir yang di load. Gambar 4.46 menunjukkan perancangan antarmuka notifikasi.

4. Antarmuka produk

Antarmuka produk berisi semua produk yang di posting oleh gapoktan. Produk ditampilkan dalam bentuk list mulai dari produk pertama hingga produk terakhir yang berhasil di unduh. Gambar 4.50 menunjukkan perancangan antarmuka produk.



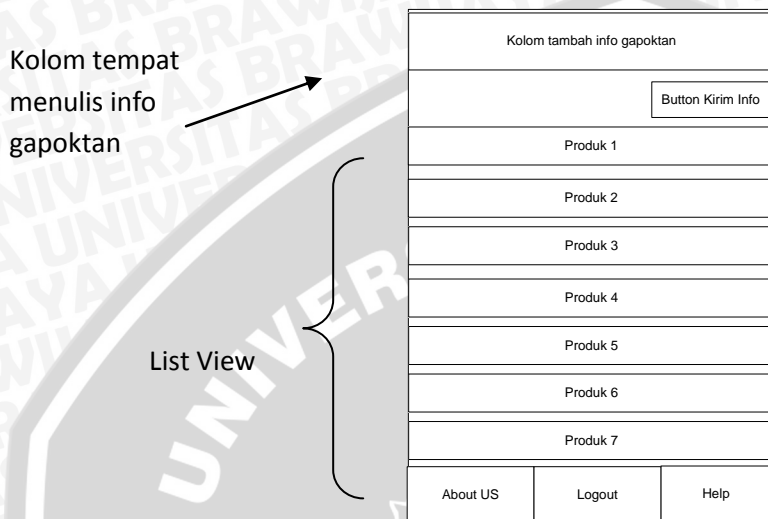
Gambar 4.50 Antarmuka produk SI Gapoktan Mobile
Sumber : [Perancangan]



5. Antarmuka Info Gapoktan

Antarmuka info gapoktan berisi semua informasi tentang gapoktan yang telah diposting. Pada antarmuka ini user yang aktif juga bisa menginfokan kegiatannya dengan mengetikkan tulisan pada kolom info.

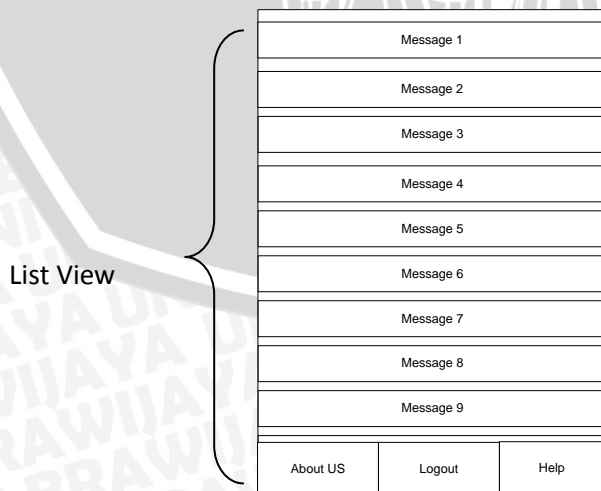
Gambar 4.51 menunjukkan perancangan antarmuka info gapoktan.



Gambar 4.51 Antarmuka info gapoktan SI Gapoktan Mobile
Sumber : [Perancangan]

6. Antarmuka Message

Antarmuka message berisi semua message yang masuk ke user yang sedang aktif. Message ditampilkan dalam bentuk list mulai dari message pertama hingga terakhir yang berhasil di unduh. Gambar 4.49 menunjukkan perancangan antarmuka message.



Gambar 4.52 Antarmuka message SI Gapoktan Mobile
Sumber : [Perancangan]

7. Antarmuka Profil

Antarmuka profil berisi biodata user yang sedang aktif. Pada antarmuka ini juga terdapat foto dari user. Gambar 4.50 menunjukkan perancangan antarmuka Profil.

Nama Gapoktan		
Foto		
Username Gapoktan		
Ketua		
Alamat		
Email		
Telepon		
Buton Tambah produk primer	Buton Tambah produk sekunder	
About US	Logout	Help

Gambar 4.53 Antarmuka profil SI Gapoktan Mobile
Sumber : [Perancangan]

8. Antarmuka List Gapoktan

Antarmuka list gapoktan berisi daftar list semua gapoktan yang terdaftar di sistem informasi gapoktan. Gambar 4.54 menunjukkan perancangan antarmuka list gapoktan.

Kolom tempat menulis info gapoktan

List View

Kolom cari gapoktan		
		Buton cari
Gapoktan 1		
Gapoktan 2		
Gapoktan 3		
Gapoktan 4		
Gapoktan 5		
Gapoktan 6		
Gapoktan 7		
About US	Logout	Help

Gambar 4.54 Antarmuka list gapoktan SI Gapoktan Mobile
Sumber : [Perancangan]

9. Antarmuka Detail Gapoktan

Antarmuka detail gapoktan berisi biodata gapoktan yang dipilih oleh user. Gambar 4.55 menunjukkan perancangan antarmuka detail gapoktan.

Nama Gapoktan		
Foto		
Nama		
Ketua		
Alamat		
Email		
Telepon		
message		
About US	Logout	Help

Gambar 4.55 Antarmuka detail gapoktan SI Gapoktan Mobile
Sumber : [Perancangan]

10. Antarmuka Detail Message

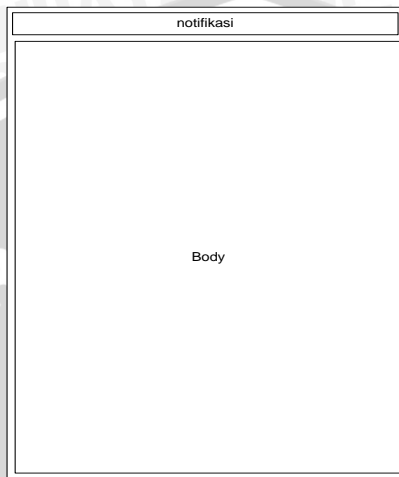
Antarmuka detail message berisi pesan rahasia anatar user aktif dan gapoktan yang dipilih. Gambar 4.56 menunjukkan perancangan antarmuka detail message.

Tulis pesan		
Button Kirim Pesan		
Message 1		
Message 2		
Message 3		
Message 4		
Message 5		
Message 6		
Message 7		
About US	Logout	Help

Gambar 4.56 Antarmuka detail message SI Gapoktan Mobile
Sumber : [Perancangan]

11. Antarmuka GCDM

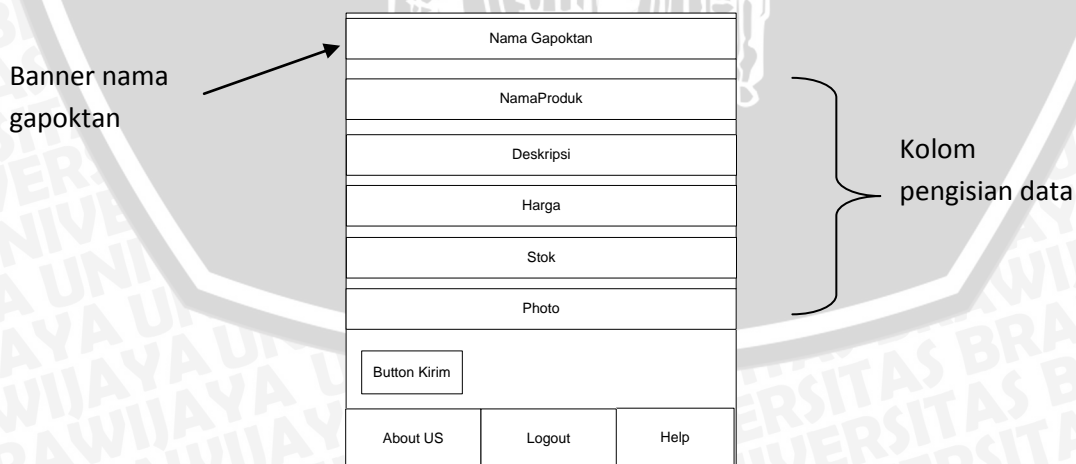
Antarmuka GCDM berisi alert notifikasi terbaru. Notifikasi ditampilkan pada header android sehingga apabila user tidak membuka aplikasi, maka notifikasi dapat langsung diketahui pada header android. Gambar 4.57 menunjukkan perancangan antarmuka GCDM.



Gambar 4.57 Antarmuka GCDM SI Gapoktan Mobile
Sumber : [Perancangan]

12. Antarmuka tambah produk

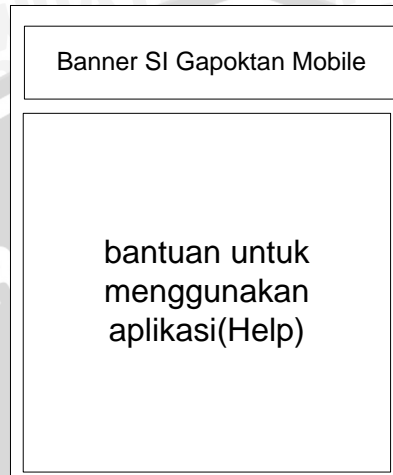
Antarmuka tambah produk berisi kolom pengisian data nama produk, deskripsi, harga, dan jumlah stok. Gambar 4.57 menunjukkan perancangan antarmuka tambah produk.



Gambar 4.58 Antarmuka tambah produk SI Gapoktan Mobile
Sumber : [Perancangan]

13. Antarmuka Help

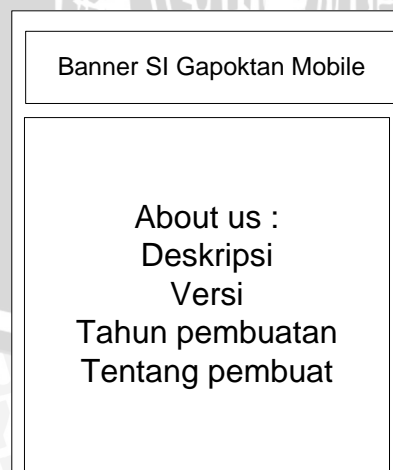
Antarmuka help berisi panduan tatacara menggunakan aplikasi. Pada antarmuka ini terdapat banner aplikasi dan text field untuk menampilkan bantuan. Gambar 4.58 menunjukkan perancangan antarmuka about us.



Gambar 4.59 Antarmuka *help* SI Gapoktan *Mobile*
Sumber : [Perancangan]

14. Antarmuka About Us

Antarmuka about us berisi tentang pembuat, versi aplikasi, deskripsi singkat, dan tahun pembuatan aplikasi. Pada antarmuka ini terdapat banner aplikasi dan text field untuk menampilkan data about us. Gambar 4.59 menunjukkan perancangan antarmuka about us.



Gambar 4.60 Antarmuka *about us* SI Gapoktan *Mobile*
Sumber : [Perancangan]