

BAB III

PEMBAHASAN

3.1 Jadwal Kerja Praktek

Pelaksanaan kerja praktek dilakukan di BBIC ang beralamat di Jl. Gedung Achmad Bakrie, Labtek VIII lantai 3 jalan Ganecha no 10 Bandung. Adapun waktu pelaksanaan kerja praktek dimulai dari 15 Juli sampai dengan 8 September 2013 dan jadwal kehadiran kerja praktek pukul 09:00 sampai dengan 16:00 WIB.

3.2 Cara dan Teknik Kerja Praktek

Pelaksanaan kerja praktek ini dibagi kedalam beberapa tahapan kegiatan, antara lain:

1. Pembuatan surat Permohonan Kerja Praktek
2. BBIC yang beralamat di Gedung Achmad Bakrie, Labtek VIII lantai 3 jalan Ganecha no 10 Bandung.
3. Pelaksanaan Kerja Praktek

3.3 Data Hasil Kerja Praktek

3.3.1 Analisis Sistem

Analisis sistem merupakan penguraian dari suatu sistem yang utuh ke dalam bagian-bagian komponennya yang dimaksudkan untuk mengidentifikasi dan mengevaluasi segala permasalahan dan hambatan yang terjadi serta kebutuhan yang diharapkan dapat menjadi acuan untuk diusulkannya perbaikan-perbaikan.

3.3.2 Analisis Masalah

Masalah yang terdapat dalam adalah bagaimana membuat fitur alarm dan notifikasi yang nantinya akan digunakan dalam aplikasi layanan kesehatan imunisasi. Dengan adanya fitur ini diharapkan dapat membantu pengguna (orang tua) untuk mengetahui jadwal imunisasi anak dan mengetahui informasi perkembangan anak secara pasti.

3.3.3 Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan *non-fungsional* menggambarkan kebutuhan luar sistem yang diperlukan untuk menjalankan aplikasi yang dibangun. Adapun kebutuhan *non-fungsional* pada fitur alarm dan notifikasi ini meliputi kebutuhan perangkat keras, kebutuhan perangkat lunak dan pemain sistem yang akan memakai aplikasi. Analisis kebutuhan *non-fungsional* bertujuan agar aplikasi yang dibangun dapat digunakan sesuai dengan kebutuhan.

3.3.3.1 Analisis Perangkat Keras

Untuk menjalankan suatu aplikasi maka diperlukan perangkat keras yang dapat mendukung proses kerja dari sistem itu sendiri. Pada dasarnya aplikasi ini dapat dijalankan di semua perangkat *mobile android*. Berikut ini spes

Tabel 3.1 Analisis Perangkat Keras

No	Spesifikasi	Keterangan
1	<i>Chipset</i>	Adreno
2	<i>CPU</i>	832 MHz processor
3	<i>GPU</i>	Adreno6
4	<i>Type</i>	TFT capacitive touchscreen, 256K colors
5	<i>Size</i>	240 x 320 pixels, 3.0 inches (~133 ppi pixel density)
6	<i>Multitouch</i>	TouchWiz UI - Touch sensitive controls - Multi-touch input method
7	<i>OS</i>	Android OS, v2.3.5 (Gingerbread)

3.3.3.2 Analisis Perangkat Lunak

Perangkat lunak atau *software* merupakan hal yang terpenting dalam mendukung kinerja sebuah sistem. Perangkat lunak digunakan dalam sebuah sistem merupakan perintah-perintah yang diberikan kepada perangkat keras agar dapat saling

berinteraksi diantara keduanya. Perangkat lunak yang dibutuhkan untuk menjalankan aplikasi ini adalah minimal sistem operasi *android* versi 2.3 *Gingerbread*.

3.3.3.3 Analisis User

Pada analisis *user* ini akan mencakup analisis terhadap *user* melalui beberapa parameter

1. *User Knowledge and Experience* dari *user* yang akan menggunakan aplikasi ini

Aplikasi ini bisa dilakukan oleh kalangan apapun, tetapi pengetahuan dan pengalaman akan memudahkan *user* untuk menggunakan aplikasi ini. Berikut ini klasifikasi *knowledge and experience* dari *user*:

Tabel 3.2 Analisis Klasifikasi *Knowledge and Experience*

<i>Education Level</i>	<i>Reading Level</i>	<i>Typing Skills</i>
Aplikasi ini bisa digunakan oleh berbagai kalangan seperti pelajar, mahasiswa hingga masyarakat awam	Aplikasi ini bisa digunakan oleh berbagai <i>level</i> pendidikan dengan <i>reading level</i> tingkat menengah	Aplikasi ini tidak memerlukan <i>typing skill</i> yang tinggi
<i>Computer Literacy</i>	<i>Task Experience</i>	<i>System Experience</i>
Aplikasi ini tidak memerlukan kemampuan komputer yang terlalu tinggi	Aplikasi ini bisa digunakan oleh pengguna dengan pengalaman penggunaan aplikasi yang menengah.	Aplikasi ini bisa digunakan oleh pengguna dengan pengalaman penggunaan aplikasi yang menengah.
<i>Application Experience</i>	<i>Native Language</i>	<i>Use of Other System</i>
Aplikasi ini bisa digunakan oleh	Aplikasi ini menggunakan bahasa	Aplikasi ini dapat digunakan tanpa perlu

pengguna dengan pengalaman penggunaan aplikasi yang menengah	Inggris, sehingga lebih mudah digunakan oleh pengguna yang memiliki kemampuan berbahasa Inggris menengah.	memasang aplikasi lain
--	---	------------------------

2. *User Physical Characteristic*

Keadaan fisik seseorang mungkin akan berpengaruh pada penggunaan aplikasi ini.

Tabel 3.3 Analisis *Users Physical Characteristic*

Age	8 tahun keatas
Gender	Pria dan Wanita
Handedness	Kanan dan Kiri
Color Blind	Pengguna yang tidak bisa membedakan warna yang satu dengan yang lainnya (butta warna) masih mampu menggunakan aplikasi ini, karena tidak ada <i>indicator</i> warna-warna khusus yang membedakan antara fungsional yang satu dengan fungsional yang lainnya

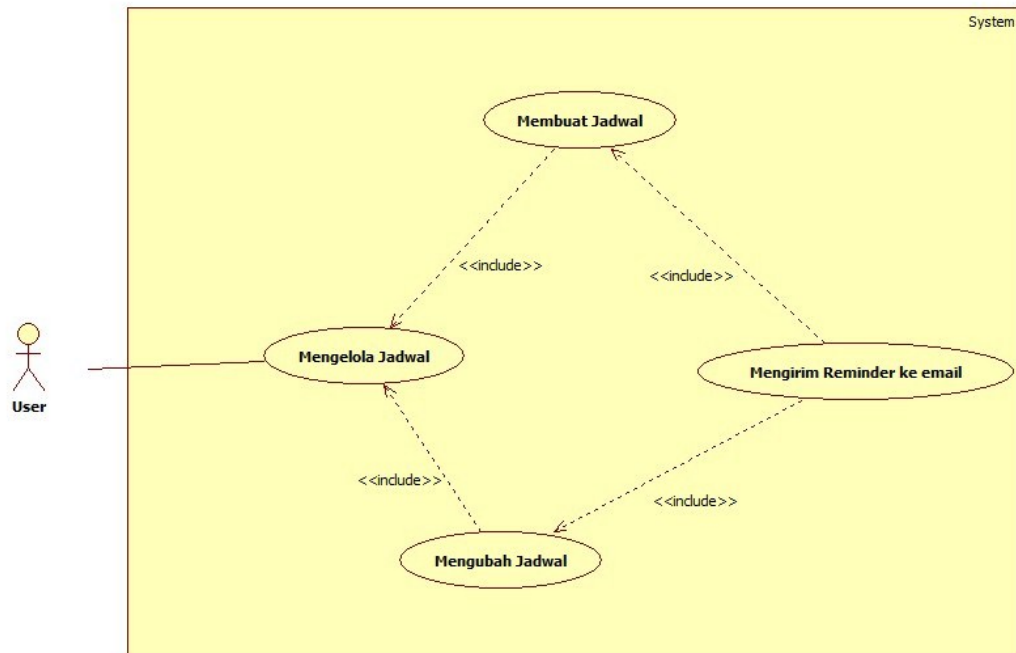
3.3.4 Analisis Kebutuhan Fungsional

Identifikasi aktor dapat dilakukan dalam analisis berorientasi objek dengan menggunakan UML yaitu menentukan aktor atau user sistem. Aktor dalam konteks UML menampilkan peran pemain atau sesuatu diluar sistem yang dikembangkan dapat berupa perangkat keras, *end user*, sistem yang lain dan sebagainya.

3.3.4.1 Usecase Diagram

Use case diagram merupakan pemodelan untuk mendeskripsikan interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Berikut ini adalah *use case* diagram dari fitur yang dibuat:

Gambar 3.1 Usecase Diagram



1. Definisi aktor

Definisi Aktor berfungsi untuk menjelaskan aktor yang terdapat pada *use case* diagram. Definisi aktor diterangkan pada tabel berikut:

Tabel 3.4 Definisi Aktor

No	Aktor	Deskripsi
1	User	User yang membuat jadwal

2. Definisi Usecase

Definisi *use case* berfungsi untuk menjelaskan proses yang terdapat pada setiap *use case*. Definisi *use case* diterangkan pada

Tabel 3.5 Definisi Usecase

No	Usecase	Deskripsi
1	Menambah Jadwal	Agar user dapat menambah jadwal jadwal kegiatan. User harus masuk kedalam system kelola alarm dan notifikasi.
2	Membuat Jadwal	User dapat membuat jadwal dengan memasukan email, nama kegiatan, tanggal kegiatan, dan deskripsi lain dari kegiatan.
4.	Mengubah Jadwal	User dapat mengubah jadwal dengan memasukan email, nama kegiatan, tanggal kegiatan, dan deskripsi lain dari kegiatan.
5	Mengirim reminder ke email	System akan otomatis mengirim reminder ke email user pada tanggal kegiatan yang di buat oleh user.

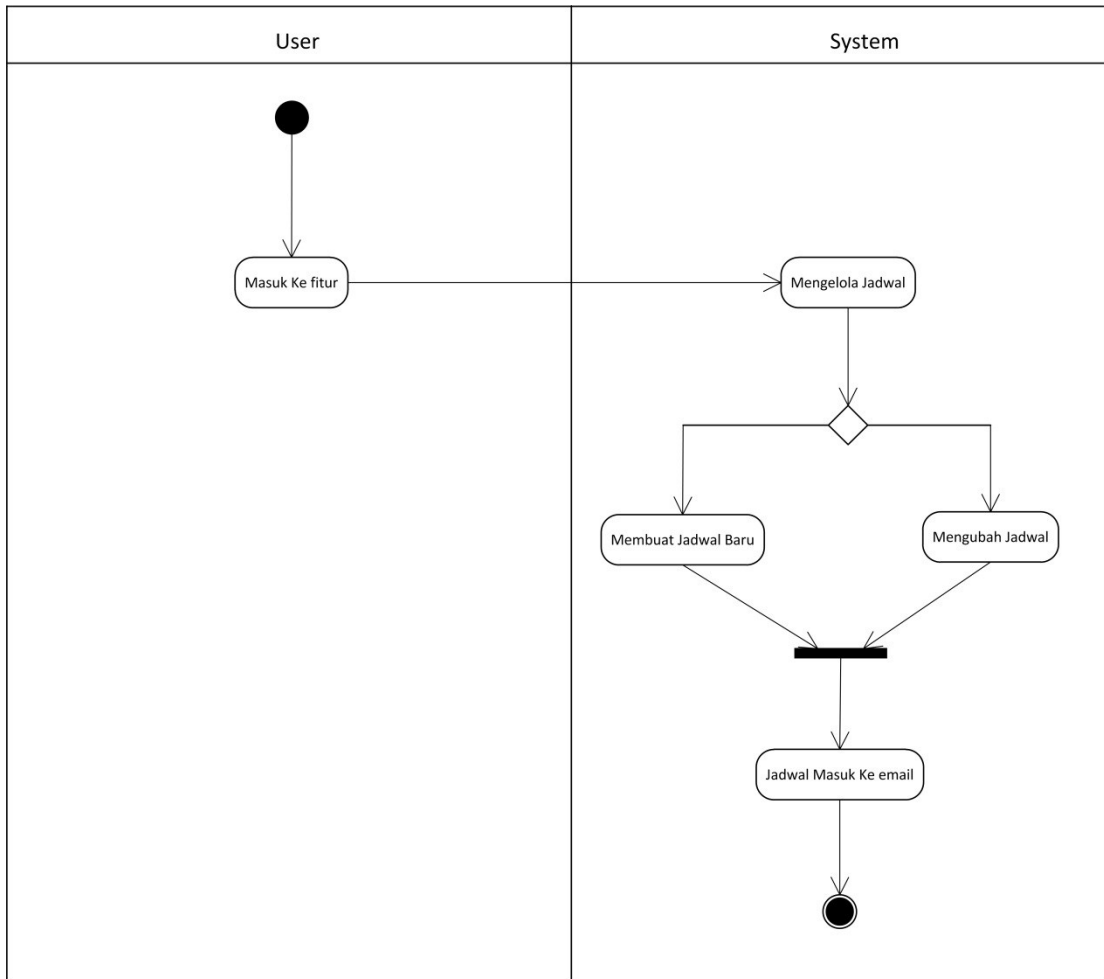
3. Skenario Usecase

Skenario *use case* merupakan bagian pada *use case* yang menunjukkan proses apa saja yang terjadi pada setiap bagian di dalam *use case*, dimana user memberikan perintah pada setiap bagian dan respon apa yang diberikan oleh sistem kepada user setelah user memberikan perintah pada setiap bagian-bagian *use case*. Berikut ini beberapa skenario *use case* berdasarkan *use case* yang ada yaitu :

3.3.4.2 Diagram Aktivitas

Activity diagram menggambarkan berbagai aliran aktivitas dalam sistem yang sedang dirancang. Berikut ini beberapa *activity diagram* yang terdapat pada fitur:

Gambar 3.2 Diagram Aktivitas

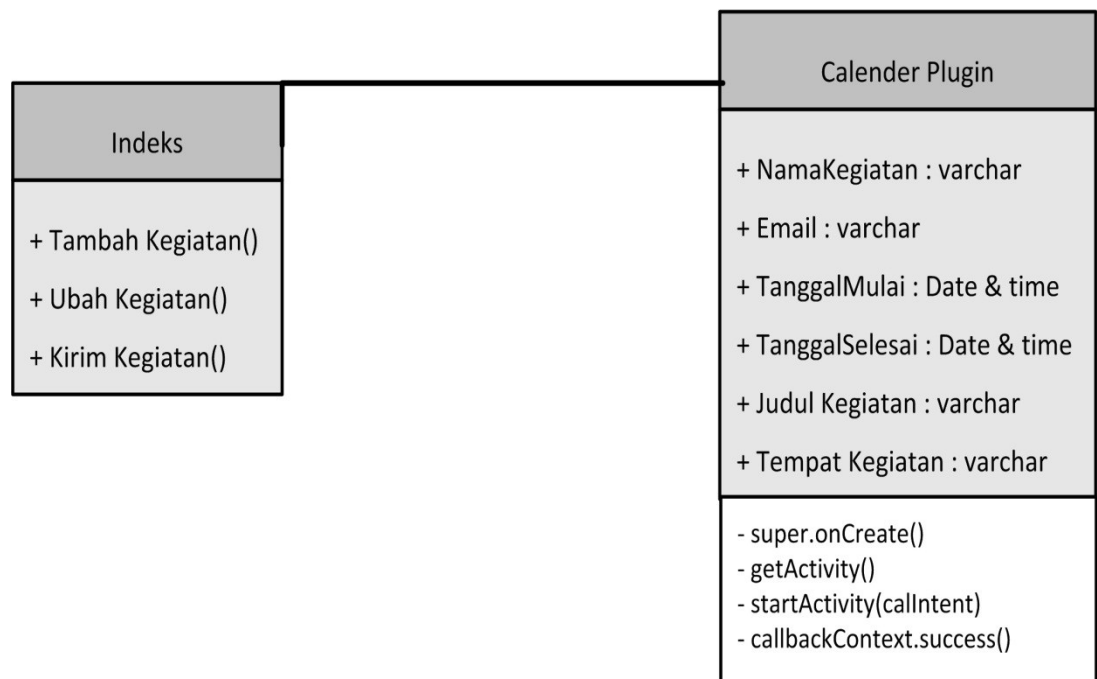


3.3.4.3 Diagram Class

Class diagram digunakan untuk menampilkan kelas-kelas dan paket-paket di dalam system. Class diagram memberikan gambaran system secara statis dan relasi antar mereka. Biasanya, dibuat beberapa class diagram untuk system tunggal. Beberapa diagram akan menampilkan subset dari kelas-kelas dan relasinya. Dapat dibuat beberapa diagram sesuai dengan yang diinginkan untuk mendapatkan gambaran lengkap terhadap system yang dibangun.

Class diagram adalah alat perancangan terbaik untuk tim pengembang. Diagram tersebut membantu pengembang mendapatkan struktur system sebelum kode ditulis, dan membantu untuk memastikan bahwa system adalah desain terbaik.

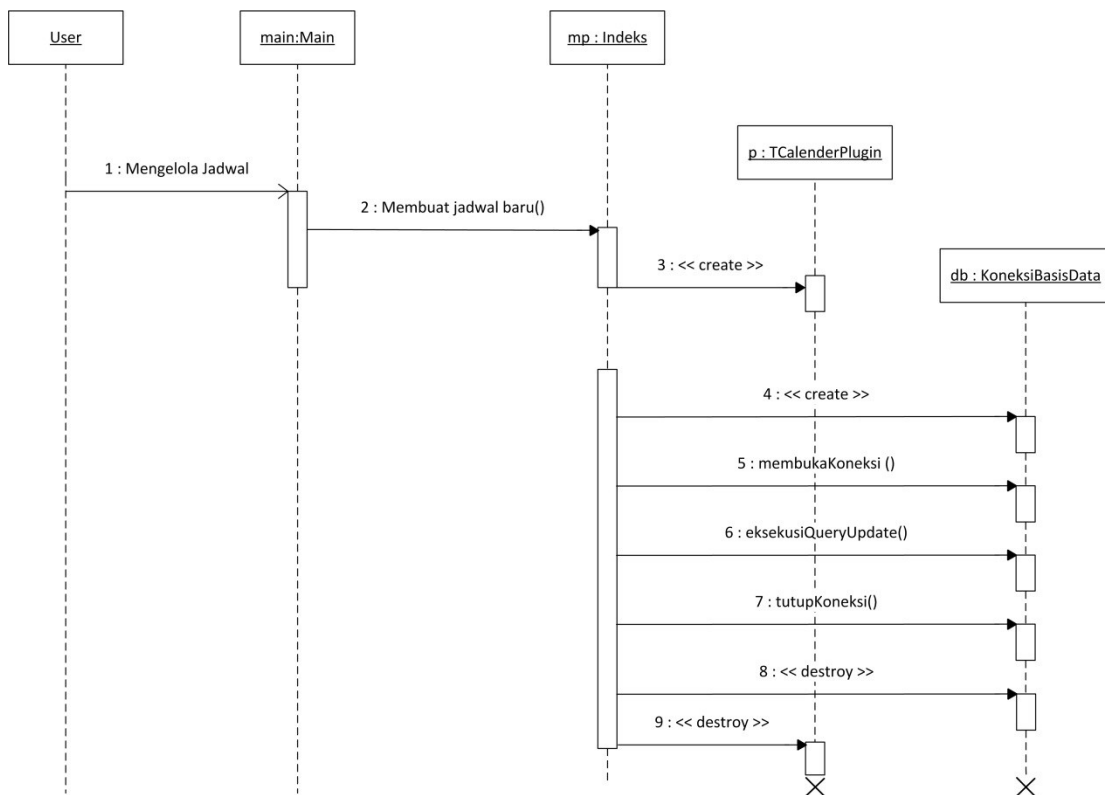
Gambar 3.3 Class Diagram



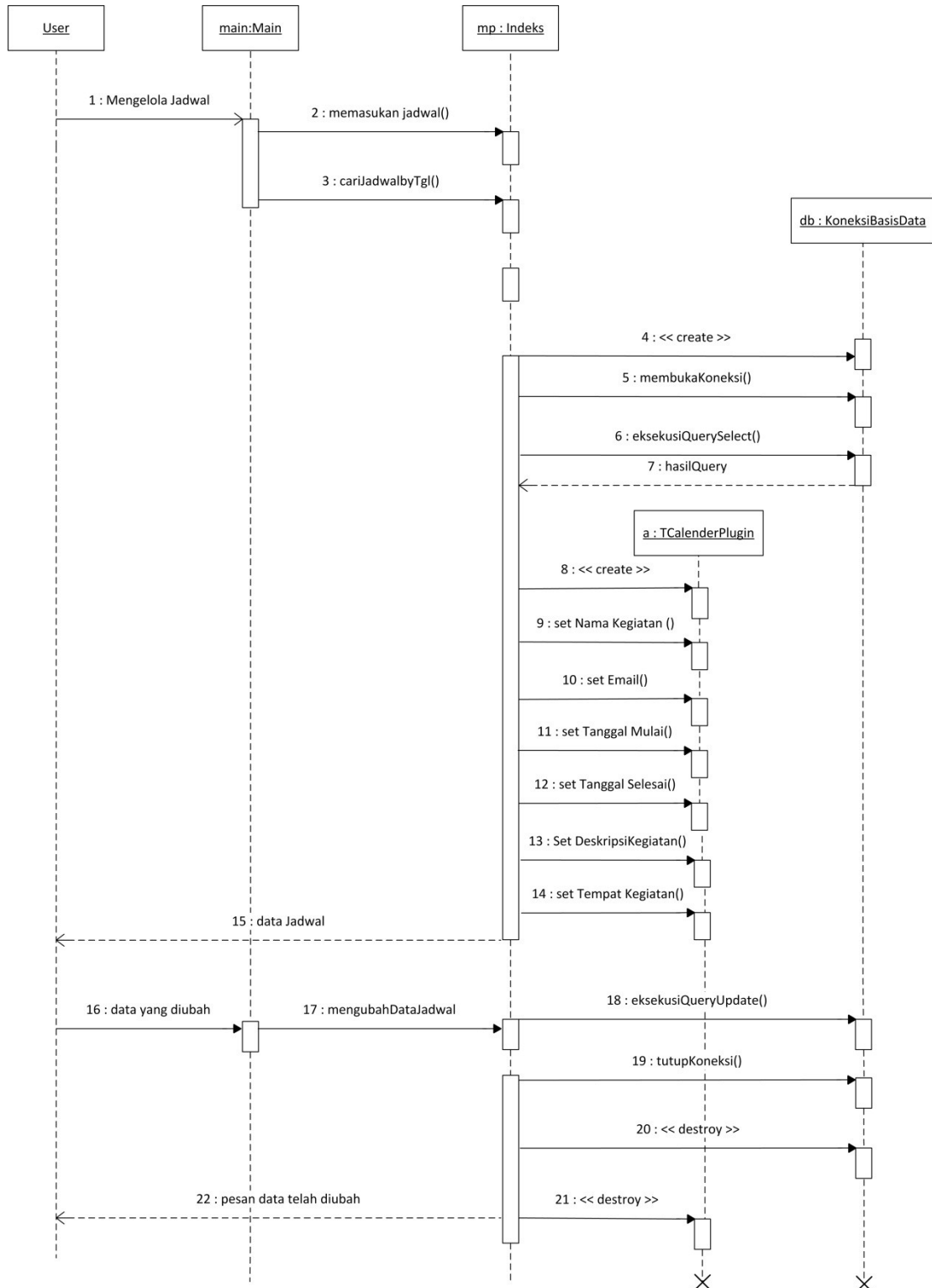
3.3.4.4 Diagram Sequential

Sequential diagram menggambarkan interaksi antar objek di dalam dan disekitar sistem berupa *message* yang digambarkan terhadap waktu. Berikut ini beberapa *sequential* diagram yang terdapat pada fitur:

Gambar 3.4 Diagram Sequential Buat Jadwal



Gambar 3.5 Diagram Sequential ubah jadwal



3.3.5 Perancangan System

Perancangan sistem merupakan metodologi pengembangan suatu perangkat lunak yang dilakukan setelah melalui tahapan analisis. Dalam tahap ini digambarkan rancangan sistem yang akan dibangun sebelum dilakukan pengkodean ke dalam suatu bahasa pemrograman.

3.3.5.1 Perancangan User Interface

Perancangan antarmuka dibutuhkan untuk mewakili keadaan sebenarnya dari aplikasi yang akan dibangun. Berikut ini beberapa perancangan antarmuka dari fitur yang akan dibangun yaitu :

Gambar 3.6 User Interface

T01

Fitur Alarm Dan Notifikasi

Email

Nama Kegiatan

Lokasi

Mulai

Hari

Tanggal

Bulan

Tahun

Jam

Selesai

Hari

Tanggal

Bulan

Tahun

Jam

Full Day

Oktober 13

S	S	R	K	J	S	M
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	2	3	

Deskripsi kegiatan

Durasi Alarm

Jenis Kegiatan

Keterangan

Klik menu Alarm dan Notifikasi untuk masuk ke menu T01

Klik Tombol Cari Untuk mencari jadwal kegiatan

Untuk membuat jadwal baru masukkan, email, nama kegiatan, lokasi, tanggal, deskripsi kegiatan, durasi kegiatan, dan jenis kegiatan

Klik Tombol simpan Untuk menyimpan jadwal kegiatan

Ukuran, Background,font dan warna disesuaikan