

SISTEM INFORMASI KULIAH ONLINE RUANGGURU

SOFTWARE ARCHITECTURE DOCUMENT

VERSION 1.1
KELOMPOK B05

Project Manager : Saqib

Anggota :
Mira Pravitasari Yunus
Vasko Dagama
Wahyu Retnowati

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Revision History

Date	Version	Description	Author
20/03/2015	1.0	<ul style="list-style-type: none"> • Membuat UCRS • Membuat <i>Test Plan</i> 	Mira Pravitasari Yunus
		Membuat UCRS	Saqib
		<ul style="list-style-type: none"> • Membuat UCRS • Membuat Dokumen SAD 	Vasko Dagama
		Membuat UCRS	Wahyu Retnowati
28/03/2015	1.1	<ul style="list-style-type: none"> • <i>Merevisi</i> UCRS • <i>Merevisi</i> Skema Relasional 	Mira Pravitasari Yunus
		<ul style="list-style-type: none"> • <i>Merevisi</i> UCRS • <i>Merevisi physical architecture</i> 	Saqib
		<i>Merevisi</i> UCRS	Vasko Dagama
		<ul style="list-style-type: none"> • <i>Merevisi</i> UCRS • Mengubah format penulisan • Merapikan ejaan penulisan • <i>Finishing</i> standardisasi <i>Sequence Diagram</i> 	Wahyu Retnowati

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Table Of Contents

1. <i>Software Architecture Document</i>	3
2. <i>Use Case Realization Specification</i>	6

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

SISTEM INFORMASI KULIAH ONLINE RUANGGURU

SOFTWARE ARCHITECTURE DOCUMENT

VERSION 1.1

KELOMPOK B05

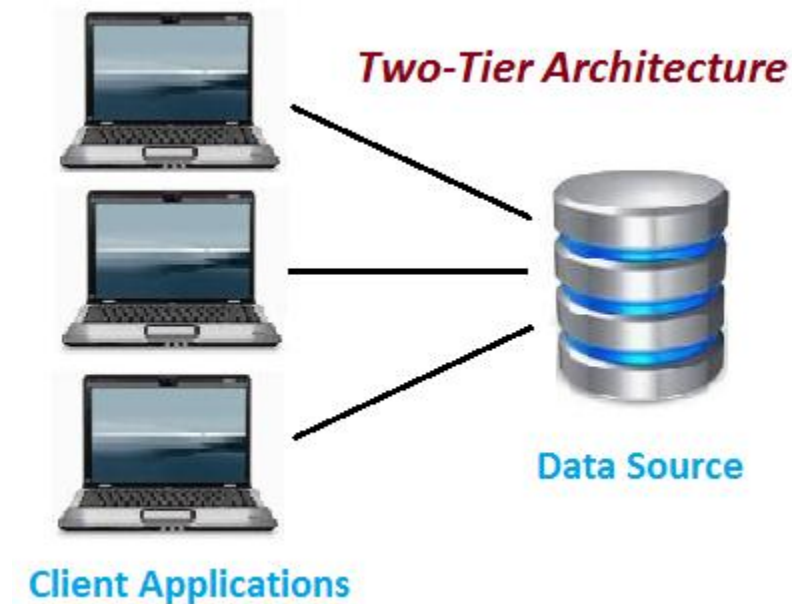
SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Table Of Contents

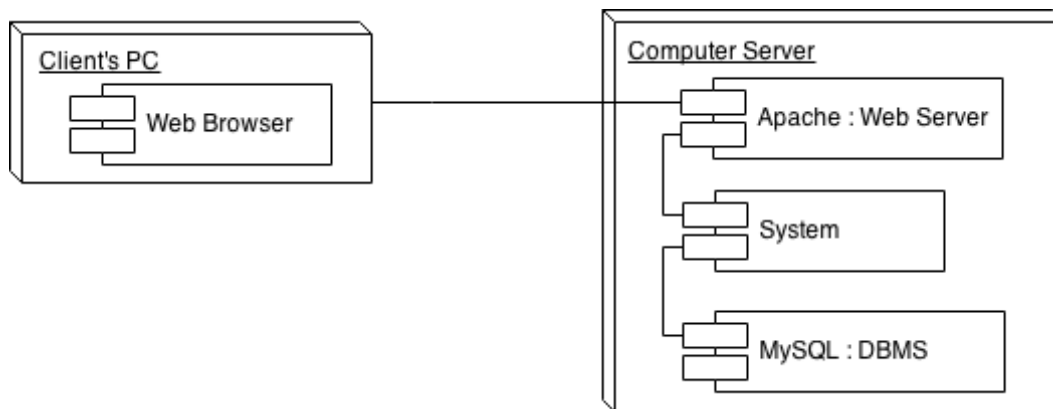
1. Rancangan <i>Physical Architecture</i>	5
2. Rancangan Basis Data.....	6
2.1 Skema Relasional.....	6
2.2 Spesifikasi.....	7
3. <i>Design Class Diagram</i>	9
4. Rancangan Kebutuhan Fungsional	10
5. <i>Test Plan</i>	10

1. Rancangan Physical Architecture

Architecture : 2tier
Operating System : LINUX
DBMS : MySQL
Web Server : Apache



Gambar 1. Pysical Design

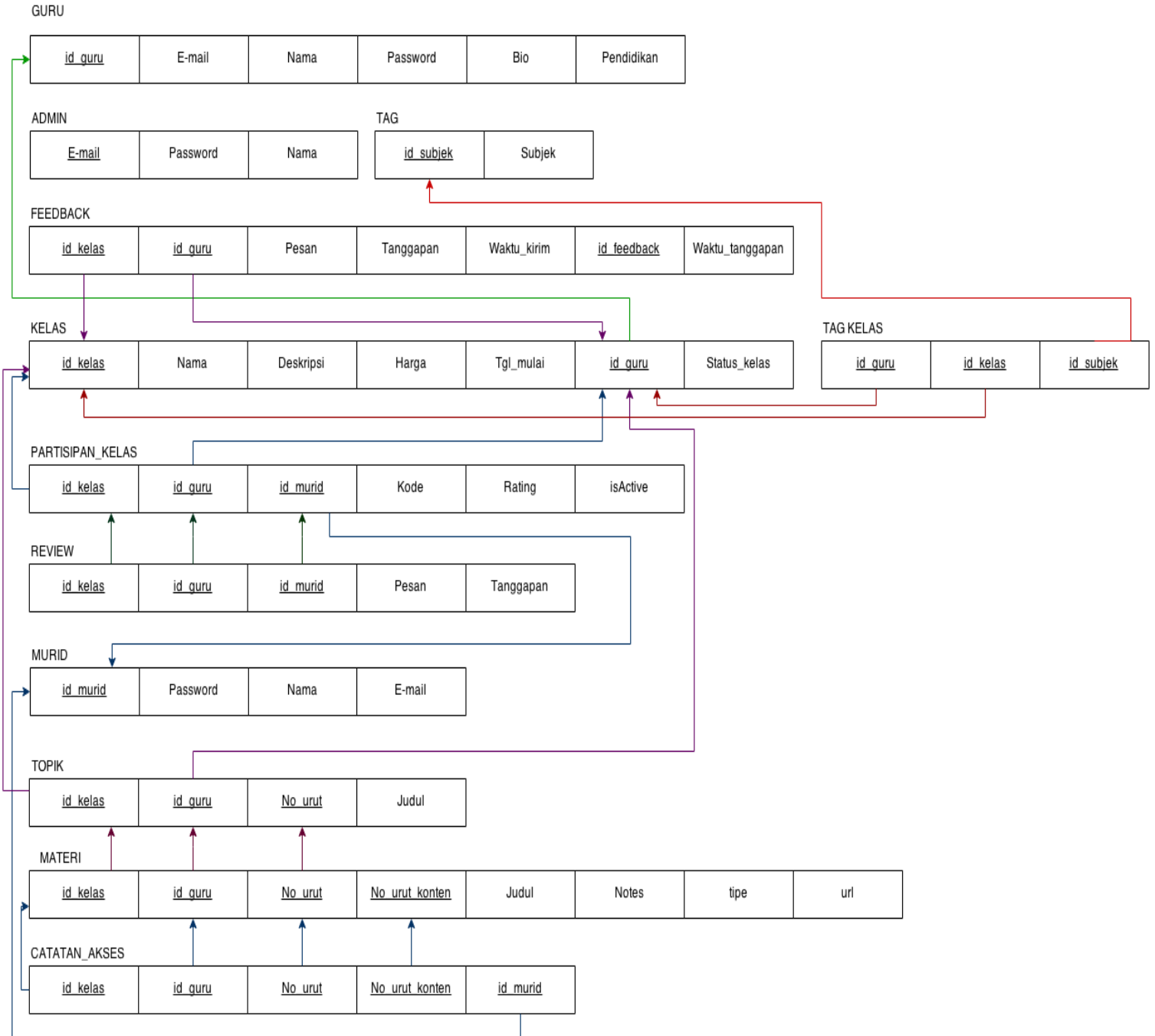


Gambar 2. Deployment Diagram

Dua gambar di atas menggambarkan tentang adanya koneksi antara *client* dengan *server*. Di mana *client* harus menggunakan *web browser* untuk mengakses sistem hanya ada satu *server* yang terdiri dari tiga bagian dan diakses secara berurutan yaitu Apache sebagai *web server*, sistem yang dikembangkan lalu MySQL sebagai DBMS.

2. Rancangan Basis Data

2.1 Skema Relasional



2.2 Spesifikasi

Berikut ini merupakan penjelasan untuk setiap tabel dari *relational schema* diatas:

- GURU

Tabel ini merepresentasikan *user* dengan *role* Guru dari sistem. Atributnya antara lain:

Nama Atribut	Tipe Atribut	Keterangan
<i>Id_guru</i>	<i>Int</i>	<i>Primary Key, Not Null</i>
<i>E-mail</i>	<i>Varchar(35)</i>	<i>Unique, Not Null</i>
<i>Password</i>	<i>Varchar(20)</i>	<i>Not Null</i>
Nama	<i>Varchar(35)</i>	<i>Not Null</i>
Bio	<i>Varchar(1500)</i>	<i>Not Null</i>
Pendidikan	<i>Varchar(500)</i>	

- MURID

Tabel ini merepresentasikan *user* dengan *role* Murid dari sistem. Atributnya antara lain:

Nama Atribut	Tipe Atribut	Keterangan
<i>Id_murid</i>	<i>Int</i>	<i>Primary Key, Not Null</i>
<i>E-mail</i>	<i>Varchar (35)</i>	<i>Unique, Not Null</i>
<i>Password</i>	<i>Varchar (20)</i>	<i>Not Null</i>
Nama	<i>Varchar (35)</i>	<i>Not Null</i>

- ADMIN

Tabel ini *user* dengan *role* Admin dari sistem. Atributnya antara lain:

Nama Atribut	Tipe Atribut	Keterangan
<i>E-mail</i>	<i>Varchar (35)</i>	<i>Primary Key, Unique, Not Null</i>
Nama	<i>Varchar (35)</i>	<i>Not Null</i>
<i>Password</i>	<i>Varchar (20)</i>	<i>Not Null</i>

- TAG

Tabel ini merepresentasikan URL milik Murid dari sistem. Atributnya antara lain:

Nama Atribut	Tipe Atribut	Keterangan
<i>Id_subjek</i>	<i>Int</i>	<i>Primary Key, Not Null</i>
Subjek	<i>Varchar (35)</i>	<i>Not Null</i>

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

- KELAS

Tabel ini merepresentasikan Kelas. Atributnya antara lain:

Nama Atribut	Tipe Atribut	Keterangan
<i>Id_kelas</i>	<i>Char (15)</i>	<i>Primary Key, Not Null</i>
<i>Id_guru</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>
<i>Nama</i>	<i>Int</i>	<i>Not Null</i>
<i>Deskripsi</i>	<i>Varchar(1000)</i>	<i>Not Null</i>
<i>Harga</i>	<i>Int</i>	<i>Not Null</i>
<i>Tgl_mulai</i>	<i>Date</i>	<i>Not Null</i>
<i>Status_kelas</i>	<i>Tiny Int</i>	
<i>Id_subjek</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>

- TAG KELAS

Tabel ini merepresentasikan URL milik Murid dari sistem. Atributnya antara lain:

Nama Atribut	Tipe Atribut	Keterangan
<i>Id_subjek</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>
<i>Id_guru</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>
<i>Id_kelas</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>

- PARTISIPAN_KELAS

Tabel ini merepresentasikan Kelas dan keterkaitannya dengan Murid yang mendaftar ke dalam kelas tersebut. Atributnya antara lain:

Nama Atribut	Tipe Atribut	Keterangan
<i>Id_kelas</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>
<i>Id_guru</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>
<i>Id_murid</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>
<i>Kode</i>	<i>Char (15)</i>	<i>Not Null</i>
<i>Rating</i>	<i>Tiny Int</i>	<i>Not Null</i>
<i>isActive</i>	<i>Boolean</i>	<i>Not Null</i>

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

- REVIEW

Tabel ini merepresentasikan komentar yang diberikan oleh Murid yang telah mendaftar di sebuah Kelas:

Nama Atribut	Tipe Atribut	Keterangan
<i>Id_kelas</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>
<i>Id_guru</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>
<i>E-mail</i>	<i>Varchar (35)</i>	<i>Foreign Key, Not Null</i>
<i>Rating</i>	<i>Float</i>	<i>Not Null</i>
<i>Pesan</i>	<i>Varchar (1500)</i>	
<i>Tanggapan</i>	<i>Varchar (1500)</i>	

- FEEDBACK

Tabel ini merepresentasikan *feedback* yang didapat Kelas. Atributnya antara lain:

Nama Atribut	Tipe Atribut	Keterangan
<i>Id_feedback</i>	<i>Int</i>	<i>Primary Key, Not Null</i>
<i>Id_guru</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>
<i>Id_kelas</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>
<i>Pesan</i>	<i>Varchar (1500)</i>	<i>Not Null</i>
<i>Tanggapan</i>	<i>Varchar (1500)</i>	
<i>Waktu_kirim</i>	<i>Date</i>	<i>Not Null</i>
<i>Waktu_tanggapan</i>	<i>Date</i>	

- TOPIK

Tabel ini merepresentasikan Topik yang dimiliki kelas. Atributnya antara lain:

Nama Atribut	Tipe Atribut	Keterangan
<i>No_urut</i>	<i>Int</i>	<i>Primary Key, Not Null</i>
<i>Id_kelas</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>
<i>Id_guru</i>	<i>Int</i>	<i>Foreign Key, Not Null</i>
<i>Judul</i>	<i>Varchar (100)</i>	<i>Not Null</i>

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

- MATERI

Tabel ini merepresentasikan Materi yang dimiliki kelas. Atributnya antara lain:

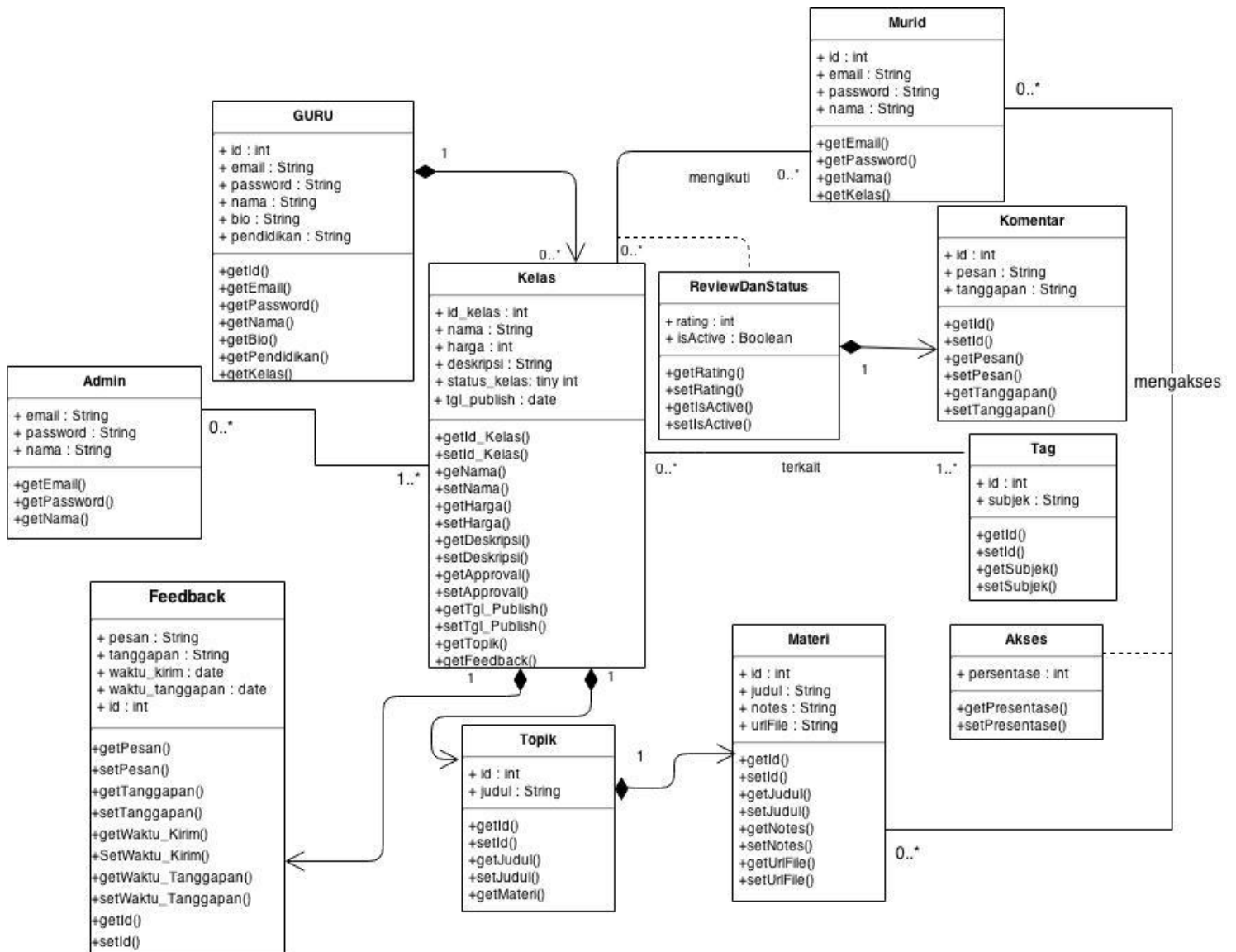
Nama Atribut	Tipe Atribut	Keterangan
No_urut_konten	<i>Int</i>	<i>Primary Key, Not Null</i>
No_urut	<i>Int</i>	<i>Foreign Key, Not Null</i>
Id_kelas	<i>Int</i>	<i>Foreign Key, Not Null</i>
Id_guru	<i>Int</i>	<i>Foreign Key, Not Null</i>
Judul	<i>Varchar (100)</i>	<i>Not Null</i>
URL	<i>Varchar (200)</i>	<i>Not Null</i>
Notes	<i>Varchar (1500)</i>	

- CATATAN_AKSES

Tabel ini merepresentasikan keterkaitan antara Materi dengan Murid untuk memastikan murid tersebut telah membuka materi. Atributnya antara lain:

Nama Atribut	Tipe Atribut	Keterangan
No_urut	<i>Int</i>	<i>Foreign Key, Not Null</i>
Id_kelas	<i>Int</i>	<i>Foreign Key, Not Null</i>
Id_guru	<i>Int</i>	<i>Foreign Key, Not Null</i>
No_urut_konten	<i>Int</i>	<i>Foreign Key, Not Null</i>
E-mail	<i>Varchar (35)</i>	<i>Foreign Key, Not Null</i>

3. Design Class Diagram



4. Rancangan Kebutuhan Fungsional

Berikut daftar *Use Case Realization Specification* (UCRS) yang akan dikerjakan pada iterasi 1. Dokumen UCRS disediakan pada dokumen terpisah.

Kode UCRS	Nama Use Case	Author
UCRS-01	Melihat Daftar Calon Partisipan	Wahyu Retnowati
UCRS-02	Melihat Daftar Partisipan	Wahyu Retnowati
UCRS-03	Mengaktifkan Partisipan	Wahyu Retnowati
UCRS-04	Menonaktifkan Partisipan	Wahyu Retnowati
UCRS-05	Mendaftar Kelas	Wahyu Retnowati
UCRS-06	<i>Retrieve</i> Kelas	Saqib
UCRS-07	<i>Create</i> Kelas	Saqib
UCRS-08	<i>Update</i> Kelas	Saqib
UCRS-09	<i>Delete</i> Kelas	Saqib
UCRS-10	Melihat Daftar Kelas	Saqib
UCRS-11	Mengubah Status Kelas	Mira Pravita Sari
UCRS-12	Melihat <i>List</i> Feddback	Mira Pravita Sari
UCRS-13	Memberi <i>Feedback</i>	Mira Pravita Sari
UCRS-14	Menanggapi <i>Feedback</i>	Mira Pravita Sari
UCRS-15	Melihat Daftar Topik Kelas	Vasko Dagama
UCRS-16	Melihat Daftar Materi	Vasko Dagama
UCRS-17	<i>Retrieve</i> Materi	Vasko Dagama
UCRS-18	<i>Create</i> Materi	Vasko Dagama
UCRS-19	<i>Delete</i> Materi	Vasko Dagama

5. Test Plan

Untuk memastikan sistem memiliki kualitas dan kinerja yang baik, terdapat tahapan-tahapan *testing* yang akan dilakukan dalam proyek pengembangan Sistem Informasi Kuliah Online Ruangguru ini. *Testing* akan dilakukan sebanyak dua kali, yaitu pada akhir setiap iterasi. Tahapan *testing* ini bertujuan untuk memastikan fitur-fitur yang ada pada sistem diimplementasikan dan dapat dijalankan dengan benar serta sesuai dengan kebutuhan klien. Terdapat tiga jenis *testing* yang akan dilakukan, yaitu *Unit Testing*, *System Testing* dan *User Acceptance Test* (UAT).

5.1 Unit Testing

Unit testing akan dilakukan pada satu modul, yaitu modul pemberian *feedback*. Objektif dari *unit testing* ini adalah untuk melakukan validasi akan kerja sebuah unit yang diimplementasi secara terisolir agar dapat melihat dependensi pada *source code* yang tidak diperlukan. Berikut *unit testing* pada modul pemberian *feedback*.

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Class	Event	Data	Expected Result
Feedback	Memberi <i>Feedback</i>	• Pesan	Sistem dapat melakukan validasi input dan jika valid menyimpan pemberian <i>feedback</i> .
Feedback	Melihat <i>List Feedback</i>		Sistem dapat menampilkan daftar <i>feedback</i> kelas sesuai dengan hak akses setiap <i>user</i> .
Feedback	Memberi Tanggapan	• Tanggapan	Sistem dapat melakukan validasi input dan jika <i>valid</i> menyimpan tanggapan yang diberikan terhadap sebuah <i>feedback</i> .

5.2 System Testing

System testing dilakukan pada sistem lengkap yang semua unitnya telah terintegrasi untuk mengevaluasi kesesuaian sistem dengan *requirement*. *System testing* pada Sistem Kelas Online Ruangguru terdiri dari tiga komponen dan dijalankan dengan konteks yang berdasarkan pada dokumen *System Requirement Specification* (SRS). *System testing* juga diharapkan dapat mengukur batasan atas yang didefinisikan pada SRS.

1. Usability

Usability testing merupakan *test* yang akan dilakukan untuk mengukur seberapa baik pengguna pada umumnya dapat menggunakan sistem. Terdapat beberapa langkah-langkah dalam melakukan *usability test* yaitu:

- (1) Menyusun skenario *test* dan kuesioner pemakaian
- (2) Mengobservasi pelaksanaan skenario oleh *user* secara realistis
- (3) Mengumpulkan data lewat kuesioner pemakaian
- (4) Mengukur tingkat *usability* sistem

2. Performance

Performance testing adalah pengujian umum untuk mengukur kinerja Sistem Kelas Online Ruangguru khususnya dalam hal *responsiveness* dan stabilitas sistem pada *workload* tertentu. Pengujian ini juga dapat mengukur dan memvalidasi kualitas lainnya terkait sistem, seperti *reliability* dan *resource utilization*.

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

a. Load Testing

Load testing merupakan sebuah proses yang memberikan sistem sebuah *demand* dan kemudian mengukur respon yang diberikan oleh sistem tersebut. Pada tahapan *testing*, *load testing* akan membantu menentukan perilaku sistem pada kondisi normal ataupun *peak load* yang diharapkan. Berikut hal-hal yang akan diukur pada *load testing*.

- **Response Time for a Transaction**

Dengan mengasumsikan tidak ada gangguan pada koneksi internet, pergantian *interface* pada sistem diperkirakan dapat berlangsung selama 0.5 – 3 detik. Sedangkan untuk menjalankan fitur yang melakukan akses ke *database* sistem dapat berjalan dalam waktu 0.5 – 4 detik dengan asumsi *user* tidak mengalami gangguan koneksi internet.

- **Throughput**

Transfer data yang terjadi diperkirakan dapat mencapai 1000kb untuk halaman yang tidak menampilkan materi. Untuk halaman yang menampilkan materi, transfer data akan sesuai dengan besarnya berkas materi. Setiap operasinya diperkirakan akan membutuhkan transfer data sebesar 50kb. Diharapkan sistem ini dapat menangani hingga 80 transaksi/detik.

- **Capacity**

Kapasitas dari Sistem Kelas Online Ruangguru dapat diperkirakan dengan menggunakan persamaan *Little's Law* sebagai berikut:

$$N = X/\lambda$$

dimana N menyatakan jumlah *concurrent user* (*user* yang menggunakan sistem secara bersama-sama), λ menyatakan jumlah kedatangan rata-rata (*average arrival rate*), dan X menyatakan *throughput*. Akan diperiksa lebih lanjut dalam pelaksanaan *testing* nilai-nilai yang diperkirakan akan memenuhi persamaan di atas untuk mengukur *capacity* dari sistem.

- **Degradation Modes**

Untuk penanggulangan masalah, sistem akan bereaksi secara tergantung dengan masalah yang dihadapi. Untuk masalah umum seperti masalah koneksi internet, sistem akan menampilkan pesan notifikasi yang menandakan bahwa koneksi internet tidak tersambung dan apabila terjadi masalah secara teknis di dalam sistem, sistem akan menampilkan pesan notifikasi *error*.

- **Resource Utilization**

Agar dapat memastikan sistem tidak mengalami kegagalan, perlu diamati penggunaan *resource* yang ada. Terlebih lagi, seluruh data yang ada di dalam *database* sistem akan disimpan ke dalam *server* milik klien sehingga

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

harus dipastikan tidak ada kesalahan/ketidakwajaran dalam utilisasi *resource* pada sistem.

b. Stress Testing

Stress testing dilakukan untuk menguji batas kemampuan sistem. Berbeda dengan *load testing* yang dilakukan untuk mengukur respon dan perilaku sistem pada kondisi normal dan *peak load* yang telah diantisipasi, *stress testing* dilakukan pada kondisi yang tidak standar dan tidak diantisipasi untuk menentukan *breaking point* ataupun *safe usage limits* serta untuk melihat *modes of failure*, yaitu berbagai cara di mana sistem dapat mengalami kegagalan.

Berikut ini adalah garis besar langkah-langkah yang akan dilakukan dalam melakukan *performance testing* pada Sistem Kelas Online Ruangguru:

- (1) Mengidentifikasi *test environment*
- (2) Mengidentifikasi *performance acceptance criteria*
- (3) Merencanakan dan mendesain *test*
- (4) Melakukan konfigurasi *test environment*
- (5) Mengimplementasikan *test design*
- (6) Mengeksekusi *test*
- (7) Menganalisis dan melaporkan hasil *test* serta melakukan ulang *test* (jika diperlukan)

3. Compatibility

Karena sistem merupakan sistem yang berbasis *web*, perlu dilakukan *test compatibility* sistem pada *browser web* yang berbeda-beda. Dalam sistem ini akan dilakukan *test* pada *browser* Chrome dan Firefox.

5.3 User Acceptance Test

User Acceptance Test merupakan *test* yang akan dilakukan klien ketika sistem telah selesai dikembangkan. Sebelumnya, akan pula dilakukan demonstrasi dan proses *testing* kepada asisten dosen. Setelahnya, barulah akan dilaksanakan kegiatan *testing* kepada klien untuk memastikan sistem telah sesuai dengan *requirement* dari pihak klien. Akan disediakan skenario pengujian yang akan dilakukan oleh klien. UAT ini akan menampilkan kemungkinan skenario-skenario dari setiap *use-case* beserta hasil yang diharapkan terjadi berdasarkan tiap skenario yang dilakukan. Klien dapat pula memberikan skenario tambahan selama masih sesuai dengan cakupan proyek. Testing ini diperkirakan akan dilakukan dalam waktu 1 hari. Tahapan-tahapan yang dilakukan dalam menyusun UAT ini meliputi:

- (1) Analisis *requirement*
- (2) Membuat *Test Plan* UAT

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

- (3) Membuat Skenario Pengujian
- (4) Menjalankan Skenario
- (5) Dokumentasi
- (6) Persetujuan dari klien.

Berikut daftar *use-case* yang akan diuji beserta skenario *class*, data, dan hasil yang diinginkan.

Skenario	Data yang terkait	Hasil yang diinginkan
Create Kelas	<ul style="list-style-type: none"> • Nama • Deskripsi • Harga 	Sistem dapat melakukan validasi <i>input</i> dan jika <i>valid</i> menyimpan pembuatan kelas.
Retrieve Kelas		Sistem dapat menampilkan kelas sesuai dengan hak akses setiap <i>user</i> .
Update Kelas	<ul style="list-style-type: none"> • Nama • Deskripsi • Harga 	Sistem dapat melakukan validasi <i>input</i> dan jika <i>valid</i> menyimpan perubahan yang dilakukan pada kelas tersebut.
Delete Kelas		
Melihat Daftar Kelas		Sistem dapat menampilkan daftar kelas sesuai dengan hak akses setiap <i>user</i> .
Menambah Topik	<ul style="list-style-type: none"> • Judul 	Sistem dapat melakukan validasi <i>input</i> dan jika <i>valid</i> menyimpan penambahan topik.
Melihat Daftar Topik Kelas	<ul style="list-style-type: none"> • Judul 	Sistem dapat menampilkan daftar topik untuk setiap kelas.
Create Materi	<ul style="list-style-type: none"> • Judul • Notes • Berkas 	Sistem dapat melakukan validasi input dan jika <i>valid</i> menyimpan pembuatan materi.
Melihat Daftar Materi		Sistem dapat menampilkan daftar materi sesuai dengan hak akses setiap <i>user</i> .
Retrieve Materi		Sistem dapat menampilkan materi sesuai dengan hak akses setiap <i>user</i> .
Mendaftar Kelas	<ul style="list-style-type: none"> • <i>isActive</i> 	Sistem dapat menyimpan pendaftaran yang dilakukan.
Melihat Daftar Calon Partisipan		Sistem dapat menampilkan daftar calon partisipan.
Melihat Daftar Partisipan		Sistem dapat menampilkan daftar partisipan kelas.
Mengaktifkan	<ul style="list-style-type: none"> • <i>isActive</i> 	Sistem dapat menyimpan perubahan status calon

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Partisipan		partisipan.
Memberi Feedback	<ul style="list-style-type: none"> • Pesan 	Sistem dapat melakukan validasi <i>input</i> dan jika <i>valid</i> menyimpan pemberian <i>feedback</i> .
Melihat <i>List Feedback</i>		Sistem dapat menampilkan daftar <i>feedback</i> kelas sesuai dengan hak akses setiap <i>user</i> .
Memberi Tanggapan	<ul style="list-style-type: none"> • Tanggapan 	Sistem dapat melakukan validasi input dan jika valid menyimpan tanggapan yang diberikan terhadap sebuah <i>feedback</i> .
Melakukan akses	<ul style="list-style-type: none"> • Persentase 	Sistem dapat meng <i>update</i> persentase berdasarkan akses materi yang telah dilakukan.

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

SISTEM INFORMASI KULIAH ONLINE RUANGGURU

USE CASE REALIZATION SPECIFICATION

VERSION 1.1

KELOMPOK B05

Project Manager : Saqib

Anggota :

Mira Pravita Sari

Vasko Dagama

Wahyu Retnowati

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: Melihat Daftar Calon Partisipan

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case* Melihat Daftar Calon Partisipan, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Wahyu Retnowati

1.2 Actor

Admin

2. Flow of Events – Design

2.1 Brief Description

Use case ini bertujuan untuk menjelaskan *scenario* ketika Admin meminta sistem untuk menampilkan daftar semua partisipan yang telah mendaftar kelas namun belum disetujui oleh Admin. Admin mengelola partisipan dari *dashboard* Admin.

2.2 Use case Scenario

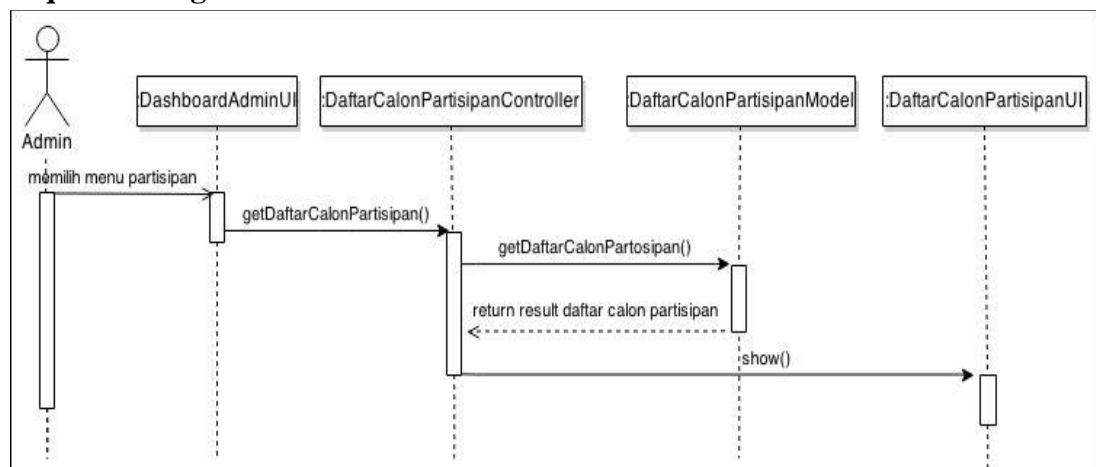
2.2.1 Basic Flow

Actor Actions	System Responses
1. Admin memilih menu partisipan	
	2. Sistem menampilkan daftar calon partisipan

2.2.2 Alternative Flows

N/A

2.3 Sequence Diagram



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.4 Explanation View class and Controller class

View Class :

- DashboardAdminUI : merupakan halaman utama Admin untuk mengelola partisipan dan kelas



- DaftarCalonPartisipanUI : merupakan halaman daftar calon partisipan yang terdapat di *dashboard* Admin



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Controller Class :

- DaftarCalonPartisipanController, yang memiliki *method* :
 - getDaftarCalonPartisipan() : merupakan *method* untuk mengambil semua data calon partisipan yang belum disetujui oleh Admin untuk memasuki suatu kelas dari *database* dan menampilkannya ke halaman DaftarCalonPartisipanUI

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

Admintelah melakukan *login* ke dalam sistem ruangguru.com

3.3 Post-Conditions

Sistem menampilkan daftar semua calon partisipan suatu kelas

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: Melihat Daftar Partisipan

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case* Melihat Daftar Partisipan, *sequence diagram*, *class view* dan *class controller*. *Use case* ini melibatkan tiga *use case* yaitu *use case* melihat daftar semua kelas, *retrieve* kelas, dan melihat daftar partisipan itu sendiri. Aktor *use case* ini adalah *user* yang mempunyai *role* sebagai Admin, Guru dan Murid.

1.1 Author

Wahyu Retnowati

1.2 Actor

User

2. Flow of Events – Design

2.1 Brief Description

Use case ini bertujuan untuk menjelaskan *scenario* ketika *user* meminta sistem untuk menampilkan daftar semua partisipan yang terdapat pada suatu kelas yang telah dibuat oleh Guru dan telah *publish* oleh Admin.

2.2 Use case Scenario

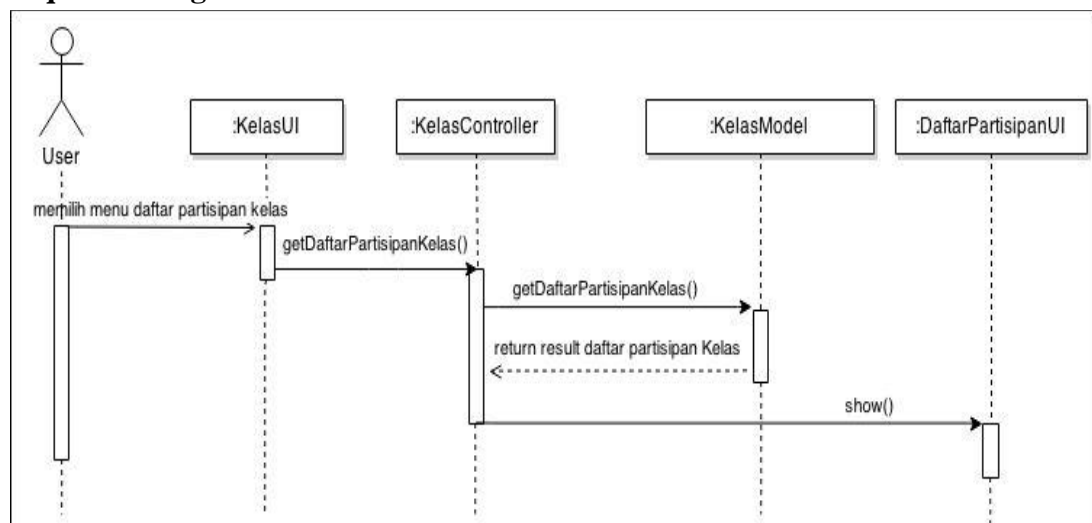
2.2.1 Basic Flow

Actor Actions	System Responses
1. User memilih menu daftar partisipan	
	2. Sistem menampilkan daftar partisipan

2.2.2 Alternative Flows

N/A

2.3 Sequence Diagram

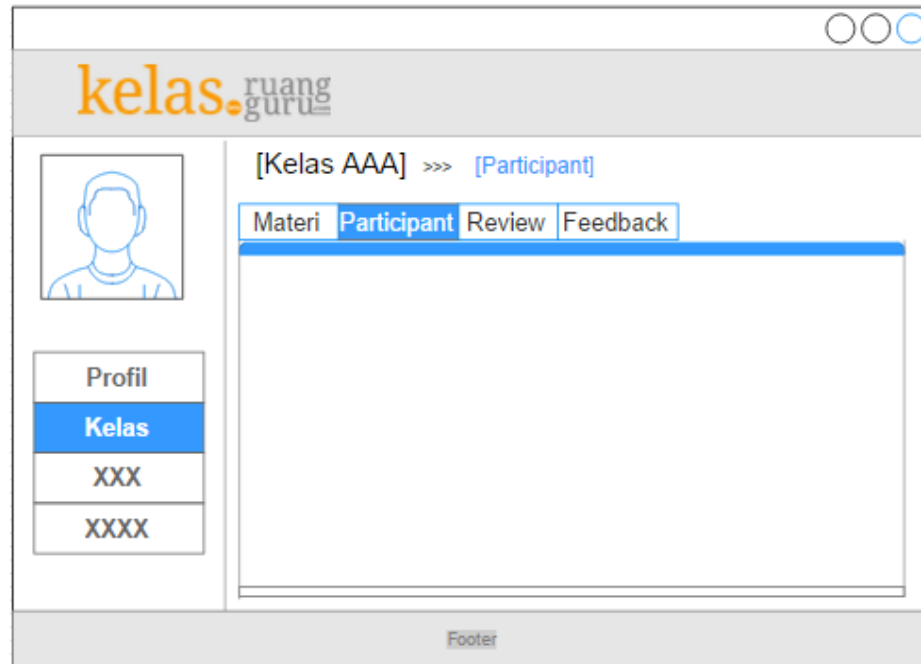


SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.4 Explanation View class and Controller class

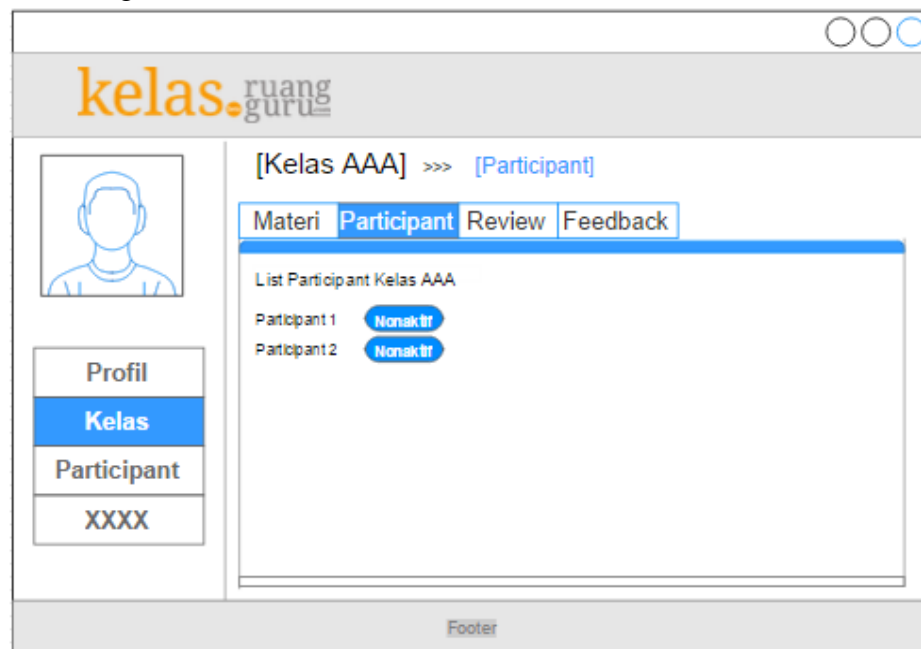
View Class :

- KelasUI : merupakan halaman utama kelas yang telah dibuat oleh guru dan telah *publish* oleh Admin



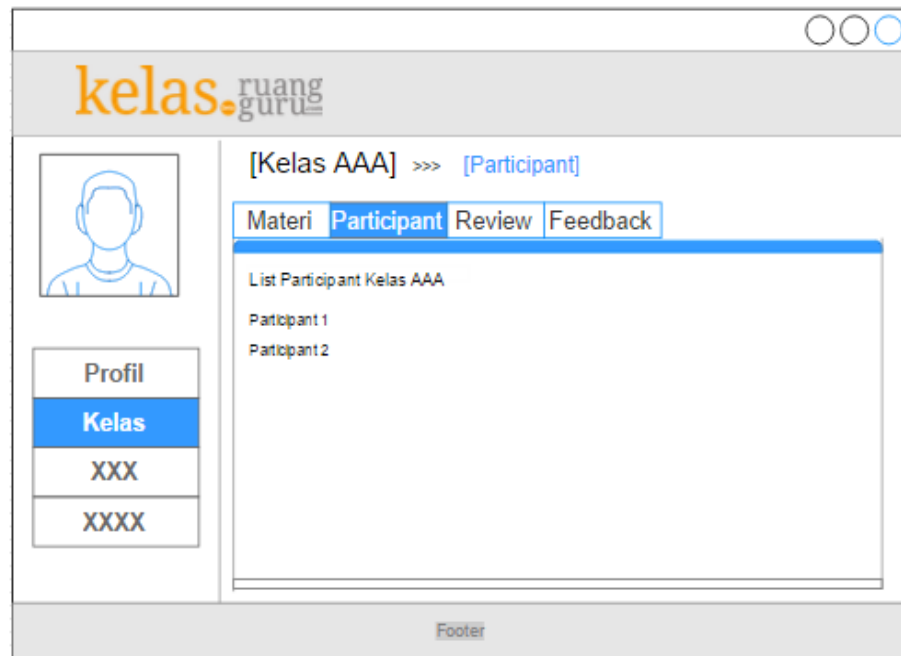
- DaftarPartisipanUI : merupakan halaman daftar semua partisipan yang terdapat di suatu kelas

User dengan role Admin



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

User dengan role murid dan guru



Controller Class:

- KelasController, yang memiliki *method* :
 - getDaftarPartisipanKelas() : merupakan *method* untuk mengambil semua data partisipan yang berada di suatu kelas yang dipilih *user* dari *database* dan menampilkannya ke halaman DaftarPartisipanUI

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- Users telah melakukan *login* ke dalam sistem ruangguru.com
- User telah melakukan *use case* “Retrieve Kelas”

3.3 Post-Conditions

Sistem menampilkan daftar semua partisipan suatu kelas

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: Mengaktifkan Partisipan

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case* Mengaktifkan Partisipan, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Wahyu Retnowati

1.2 Actor

Admin

2. Flow of Events – Design

2.1 Brief Description

Use case MengaktifkanPartisipan ini menjelaskan *scenario* ketika murid telah mendaftar kelas tertentu akan tetapi statusnya masih belum aktif sebagai partisipan kelas sehingga membutuhkan *Admin* untuk mengaktifkan status murid tersebut agar dapat mengikuti kelas yang telah didaftarnya.

2.2 Use case Scenario

2.2.1 Basic Flow

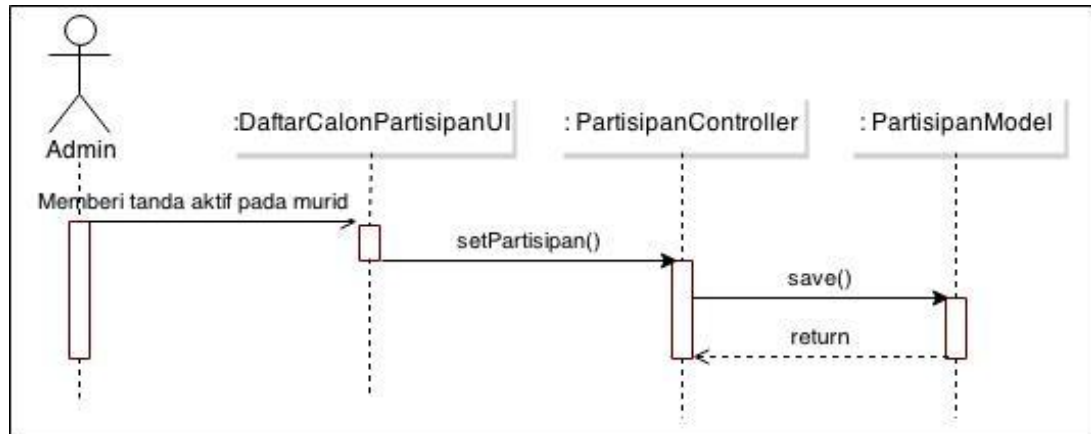
Actor Actions	System Responses
1. <i>Admin</i> memberikan tanda aktif kepada partisipan	
	2. Sistem akan menambah murid yang telah diaktifkan ke dalam kelas yang telah didaftar dan menyimpannya ke dalam <i>database</i> sebagai partisipan diterima
	3. Sistem memberikan notifikasi diterima kepada murid tersebut melalui <i>e-mail</i> minimal 1 jam setelah proses <i>approve</i>

2.2.2 Alternative Flows

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

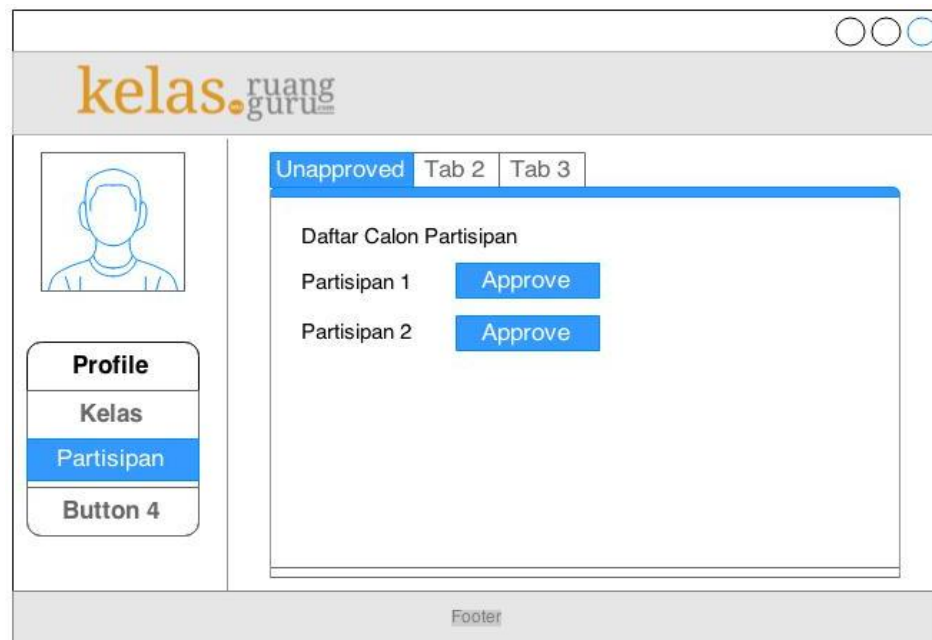
2.3 Sequence Diagram



2.4 Explanation View class and Controller class

View Class :

- DaftarCalonPartisipanUI : merupakan halaman daftar calon partisipan yang berada di menu partisipan pada *dashboard* Admin



Controller Class :

- PartisipanController, yang memiliki *method* :
 - setPartisipan() : merupakan *method* yang digunakan untuk mengubah status calon partisipan menjadi aktif

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- *Admin* telah melakukan *login* ke dalam sistem ruangguru.com
- *Admin* sudah melakukan *use case* “Melihat daftar calon partisipan”

3.3 Post-Conditions

Murid berhasil menjadi partisipan aktif

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: Menonaktifkan Partisipan

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case* Menonaktifkan Partisipan, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Wahyu Retnowati

1.2 Actor

Admin

2. Flow of Events – Design

2.1 Brief Description

Use case Menonaktifkan Partisipan ini menjelaskan *scenario* ketika murid telah mendaftar kelas tertentu akan tetapi statusnya masih belum aktif sebagai partisipan kelas sehingga membutuhkan *Admin* untuk mengaktifkan status murid tersebut agar dapat mengikuti kelas yang telah didaftarnya.

2.2 Use case Scenario

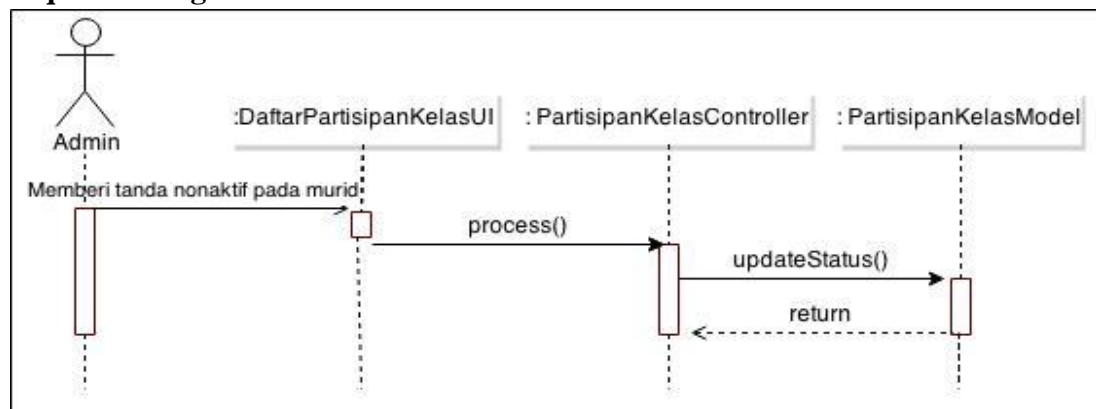
2.2.1 Basic Flow

Actor Actions	System Responses
1. Admin memberikan tanda nonaktif kepada partisipan	
	2. Sistem akan mengubah status aktif murid menjadi tidak aktif dari suatu kelas
	3. Sistem memberikan notifikasi kepada murid tersebut melalui <i>email</i> minimal 1 jam setelah proses nonaktif.

2.2.2 Alternative Flows

N/A

2.3 Sequence Diagram

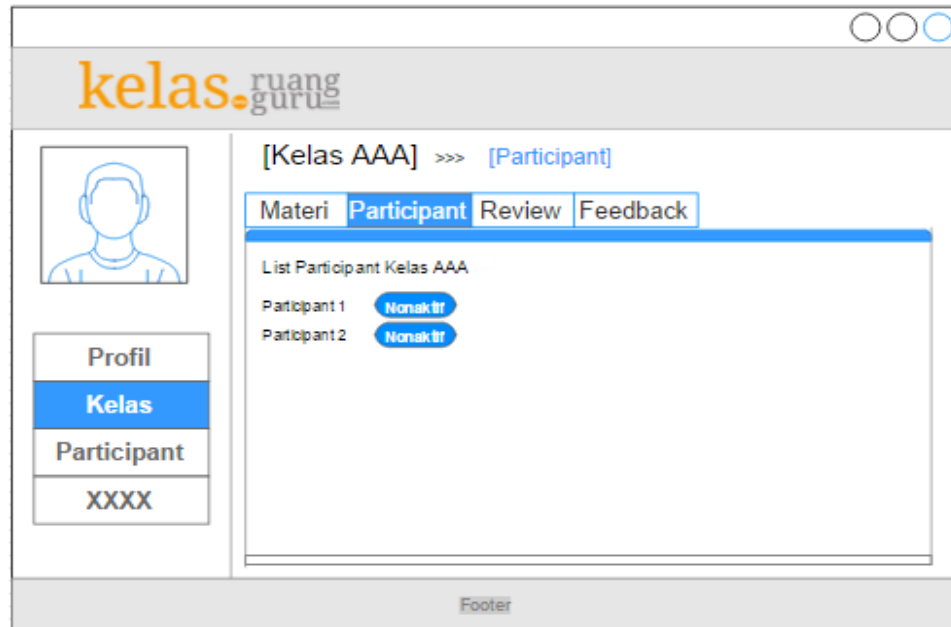


SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.4 Explanation View class and Controller class

View Class :

- DaftarPartisipanKelasUI : merupakan halaman daftar partisipan suatu kelas tertentu



Controller Class :

- PartisipanKelasController, yang memiliki *method* :
 - Process() : merupakan *method controller* yang digunakan untuk memproses nonaktif murid

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- Admin telah melakukan *login* ke dalam sistem ruangguru.com
- Admin sudah melakukan *use case* “Melihat daftar partisipan”

3.3 Post-Conditions

Murid berhasil dinonaktifkan

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: Mendaftar Kelas

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case* Mendaftar kelas, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Wahyu Retnowati

1.2 Actor

Murid

2. Flow of Events – Design

2.1 Brief Description

Use case Mendaftar kelas menjelaskan mengenai skenario proses murid dalam mendaftar kelas yang telah dibuat Guru dan telah dipublish Admin.

2.2 Use case Scenario

2.2.1 Basic Flow

Actor Actions	System Responses
1. Murid memilih menu pendaftaran	
	2. Sistem menampilkan term dan pilihan metode pembayaran
3. Murid menyetujui term dan memilih salah satu metode pembayaran	
	4. Sistem akan menampilkan notifikasi bahwa pendaftaran berhasil
	5. Sistem akan menyimpan data murid sebagai partisipan <i>pending</i> .

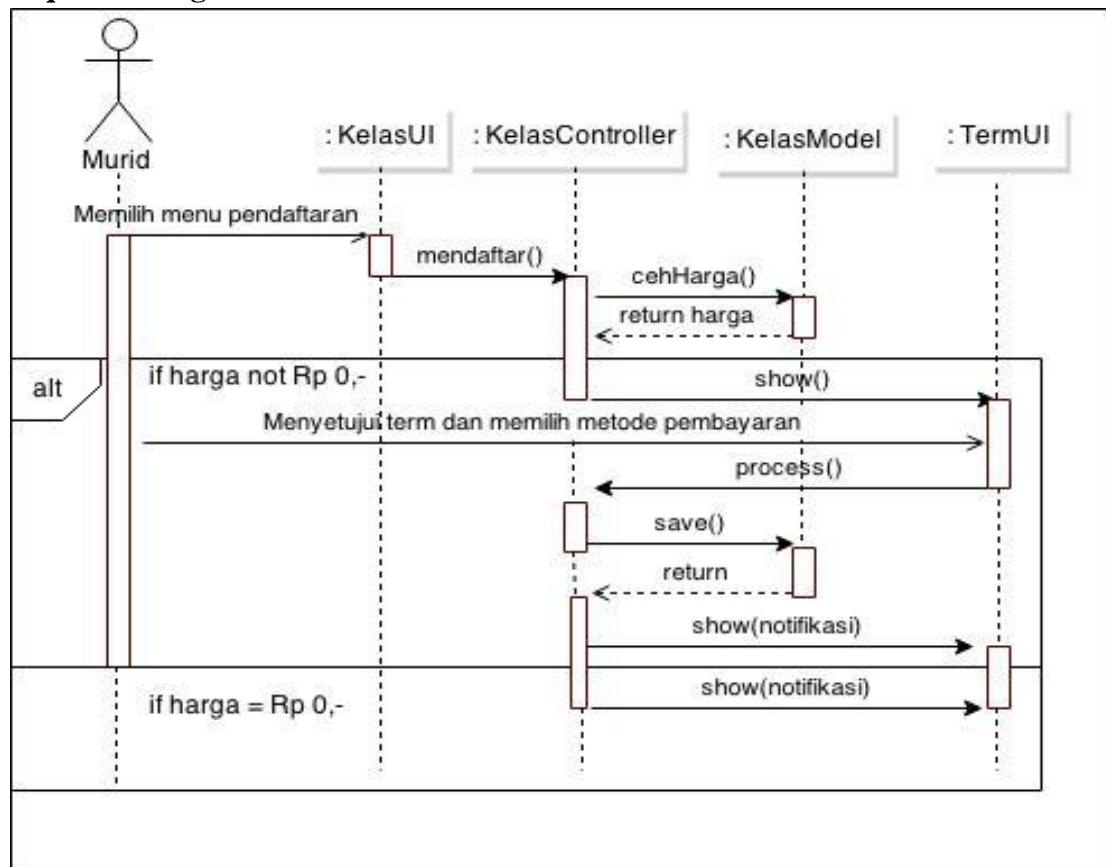
2.2.2 Alternative Flows

2.2.2.1 Alternatif Flows 2A

Jika harga kelas sama dengan Rp 0,- maka sistem akan langsung mengaktifkan murid dan memberi hak akses kepada murid untuk memasuki kelas tersebut.

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.3 Sequence Diagram



2.4 Explanation View class and Controller class

View Class

- KelasUI : merupakan halaman kelas utama



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

- TermUI : merupakan halaman untuk menampilkan persetujuan dan cara pembayaran

Controller Class :

- KelasController, yang memiliki *method* :
 - cekHarga() : merupakan *method* untuk mengatur proses mengecek harga ke *database*
 - show() : merupakan *method* menampilkan harga
 - save() : merupakan *method* untuk menyimpan *action* term dan cara pembayaran murid
 - show(notifikasi) : merupakan *method* untuk menampilkan notifikasi bahwa pendaftaran berhasil

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- Murid telah melakukan *login* ke dalam sistem ruangguru.com
- Murid sudah melakukan *use case* “Retrieve kelas”

3.3 Post-Conditions

Murid berhasil mendaftar kelas

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: *Create Kelas*

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case create kelas*, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Saqib

1.2 Actor

Guru

2. Flow of Events – Design

2.1 Brief Description

Use case ini menerangkan skenario ketika guru membuat sebuah kelas baru pada sistem. Guru mengisi isian mengenai informasi kelas dan sistem akan melakukan validasi *output*. Jika data yang diisi *valid*, sistem akan menyimpan sebuah kelas baru.

2.2 Use case Scenario

2.2.1 Basic Flow

Actor Actions	System Responses
1. Memilih menu untuk membuat kelas	
	2. Menampilkan halaman untuk membuat kelas baru
3. Mengisi nama kelas, deskripsi kelas, <i>tag</i> , memasukan judul topik-topik yang akan ada pada kelas	
4. Melakukan <i>upload</i> minimal 5 materi	
5. Melakukan <i>submit</i>	
	6. Sistem membuatkan kelas baru dengan rincian sesuai input dari <i>actor</i> dan memiliki status <i>unapprove</i>
	7. Memberi notifikasi pada aktor bahwa kelas sudah berhasil tetapi detil kelas tidak bisa diubah sampai pihak ruangguru memberi persetujuan

2.2.2 Alternative Flows

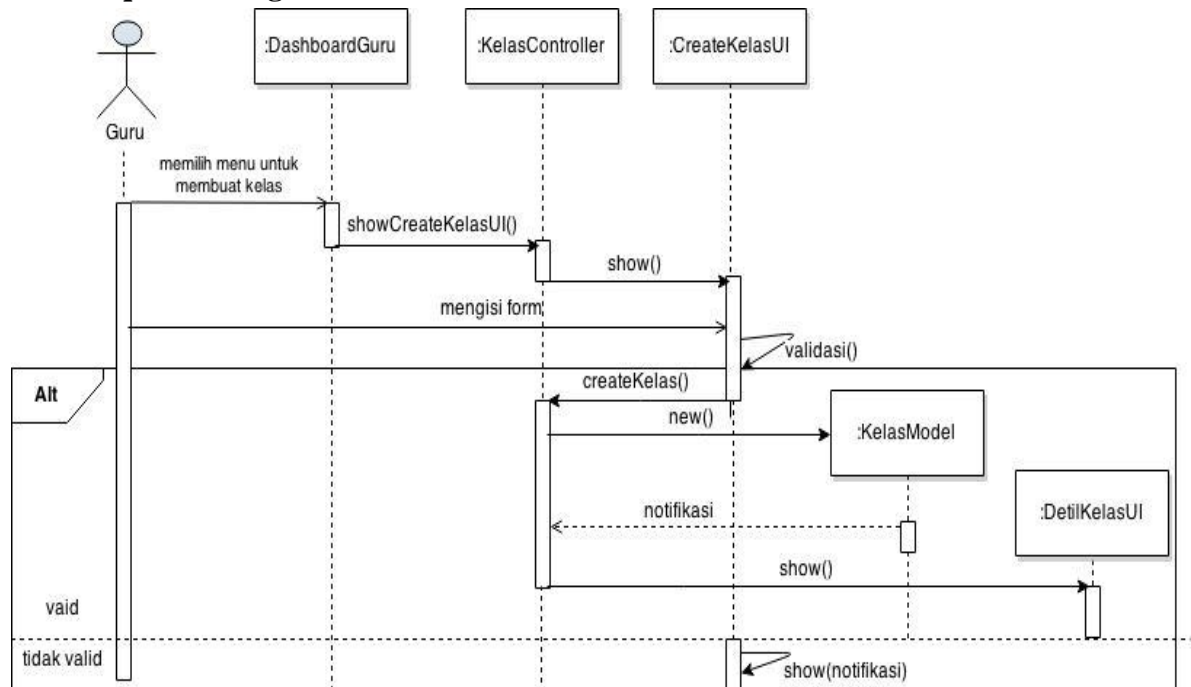
2.2.2.1 Alternatif Flows 5A

Jika data yang dimasukan aktor tidak *valid* maka sistem akan

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

memberi notifikasi kesalahan dan akan dikembalikan pada detail kelas.

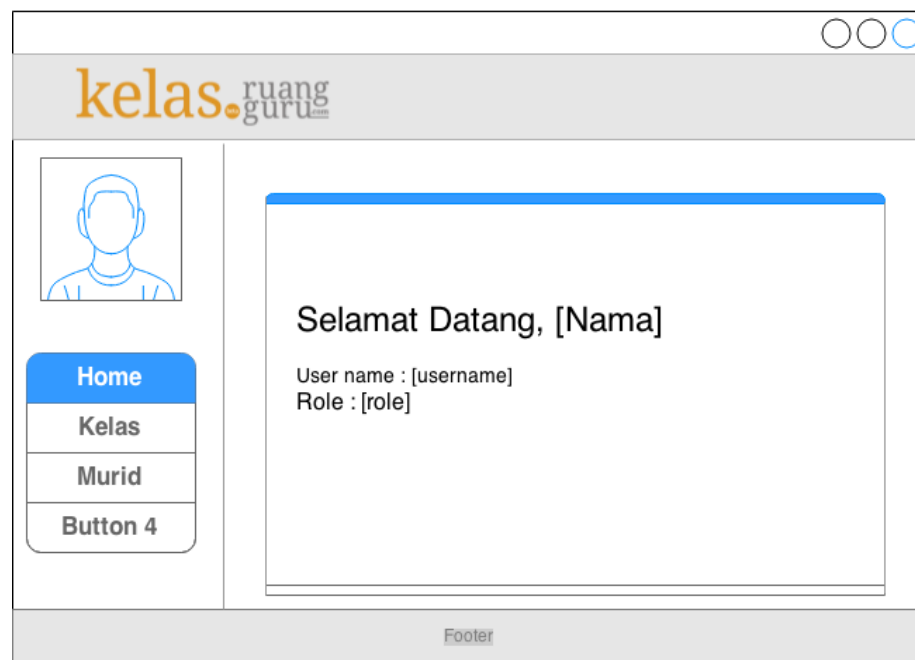
2.3 Sequence Diagram



2.4 Explanation View class and Controller class

View Class :

- DashboardGuru : merupakan halaman utama guru dalam mengelola kelas



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

- CreateKelasUI : merupakan halaman pembuatan kelas pada *dashboard* guru

Membuat Kelas Baru

Nama Kelas:

Harga:

Deskripsi:

Topik:

Materi 1

Materi 2

Topik 2

Tag:

Upload Materi

Nama :

Materi : ☐ Video ☐ File

tidak ada file terpilih

Controller Class

- KelasController, mempunyai *method* sebagai berikut;
 - CreateKelas() adalah *method* untuk membuat kelas baru dan memanggil *method* membuat materi minimal sebanyak 5 kali.

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

Guru telah melakukan *login* ke dalam sistem ruangguru.com

3.3 Post-Conditions

Kelas baru berhasil dibuat sesuai keinginan guru dengan status *unapproved*

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: *Update Kelas*

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case update kelas*, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Saqib

1.2 Actor

Guru

2. Flow of Events – Design

2.1 Brief Description

Use case ini menerangkan skenario ketika guru mengubah detail kelas yang sebelumnya telah ditentukan oleh guru tersebut. Mencakup nama, deskripsi, harga, *tag* dan topik-topik pada kelas tersebut.

2.2 Use case Scenario

2.2.1 Basic Flow

Actor Actions	System Responses
1. Memilih menu untuk mengubah kelas	
	2. Menampilkan halaman untuk mengubah kelas yang sudah berisi rincian kelas yang terakhir disimpan oleh sistem

2.2.1.1 Sub Flow 1

Actor Actions	System Responses
1. Mengubah nama, deskripsi, harga, <i>tag</i> , atau topik pada kelas tersebut	
2. Melakukan <i>submit</i>	
	3. Melakukan validasi data yang dimasukan oleh aktor
	4. Sistem menyimpan rincian kelas sesuai input dari aktor
	5. Memberi notifikasi pada aktor bahwa rincian kelas sudah berhasil dirubah

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.2.1.2 Sub Flow 2

Actor Actions	System Responses
1. Mengubah status kelas	
	2. Mengubah status kelas menjadi menunggu konfirmasi dari <i>admin</i> untuk perubahan
	3. Memberi notifikasi pada aktor bahwa perubahan status kelas sedang dalam proses konfirmasi dari <i>admin</i>

2.2.2 Alternative Flows

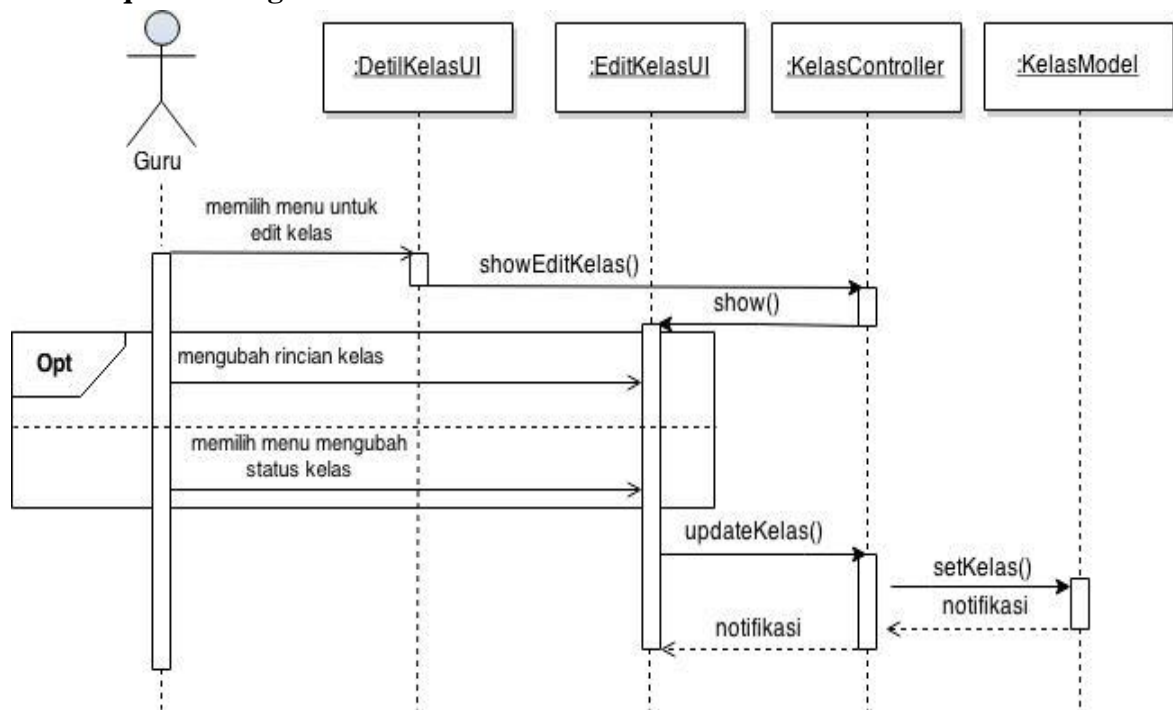
2.2.2.1 Alternative Sub Flow 1 tahap 1

Jika aktor memilih untuk membatalkan pengubahan rincian kelas maka sistem akan member notifikasi bahwa rincian kelas tidak dilakukan perubahan dan *use case* selesai

2.2.2.2 Alternative Sub Flow 1 tahap 3

Jika data yang dimasukan aktor tidak *valid* maka sistem akan memberi notifikasi kesalahan dan akan dikembalikan pada tahap ke-1

2.3 Sequence Diagram



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.4 Explanation View class and Controller class

View Class :

– DetilKelasUI

The screenshot shows a web application window titled 'kelas.ruangguru'. On the left is a sidebar with a user profile icon and buttons for 'Profile', 'Kelas' (highlighted), 'Murid', and 'Button 4'. The main content area is titled 'Nama Kelas 1' and has four tabs: 'Umum' (selected), 'Topik', 'Tab 3', and 'Tab 4'. Under the 'Umum' tab, there is a description: 'Deskripsi : ini adalah sebuah kelas yang mengajri tentang ABC untuk tingkat advance', a price: 'Harga : 100.000,00', and tags: 'Tag : #advance #mahal #ekskusif #ABC'. At the bottom right of the main area are 'EDIT' and '(X) DELETE' buttons. A 'Footer' is visible at the very bottom.

– EditKelasUI

The screenshot shows the 'Edit' version of the class details page. The title is 'Nama Kelas 1 >> Edit' with a status of 'Approved'. The sidebar is identical to the previous view. The main content area contains form fields for 'Nama Kelas' (filled with 'Nama Kelas 1'), 'Harga' (filled with 'Rp. 50.000,00'), 'Deskripsi' (filled with 'Kelas ini akan bla bla bla bla dimana anda akan bla bla'), and 'Topik' (with 'Topik 1' and 'Topik 2' selected, and a 'New Topik' button). There is also a 'Tag' field filled with '#mate #mipa #matdas'. At the bottom right are three buttons: '(X) BATAL', 'SAVE', and 'Request to Publish'. A 'Footer' is visible at the very bottom.

Controller Class :

- UpdateKelas() adalah *method* untuk melakukan perubahan pada sebuah kelas, baik untuk mengubah rinciannya atau pun mengubah status kelas agar tidak bisa diubah kembali

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- Guru telah melakukan *login* ke dalam sistem ruangguru.com
- Guru telah melihat detil sebuah kelas yang hanya berstatus *approved*

3.3 Post-Conditions

Kelas berhasil diubah

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: *Delete Kelas*

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case delete kelas*, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Saqib

1.2 Actor

Guru

2. Flow of Events – Design

2.1 Brief Description

Use case ini menerangkan skenario ketika guru ingin menghapus sebuah kelas yang pernah dibuatnya.

2.2 Use case Scenario

2.2.1 Basic Flow

Actor Actions	System Responses
1. Memilih menu untuk menghapus kelas	
	2. Meminta konfirmasi pada aktor untuk melakukan penghapusan kelas secara permanen
3. Memberi konfirmasi untuk penghapusan kelas secara permanen	
	4. Memberi notifikasi pada aktor bahwa kelas sudah berhasil dihapus
	5. Menampilkan halaman daftar kelas yang dimiliki

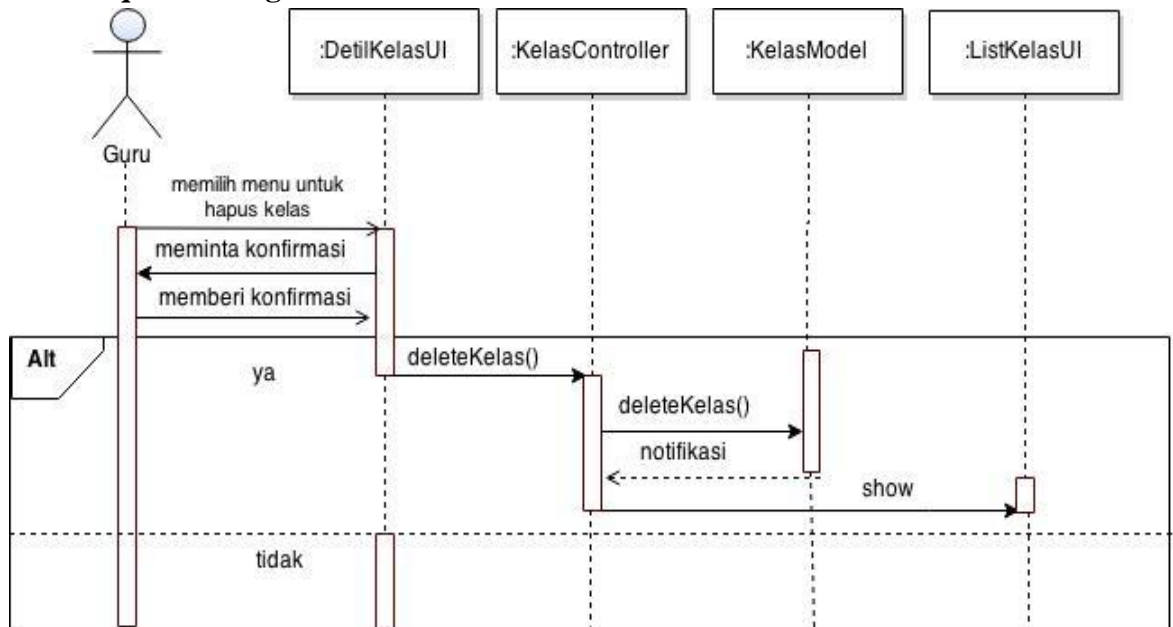
2.2.2 Alternative Flows

2.2.2.1 Alternative Flows 4A

Jika aktor tidak menyetujui untuk penghapusan kelas secara permanen maka akan langsung menuju tahap 5

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

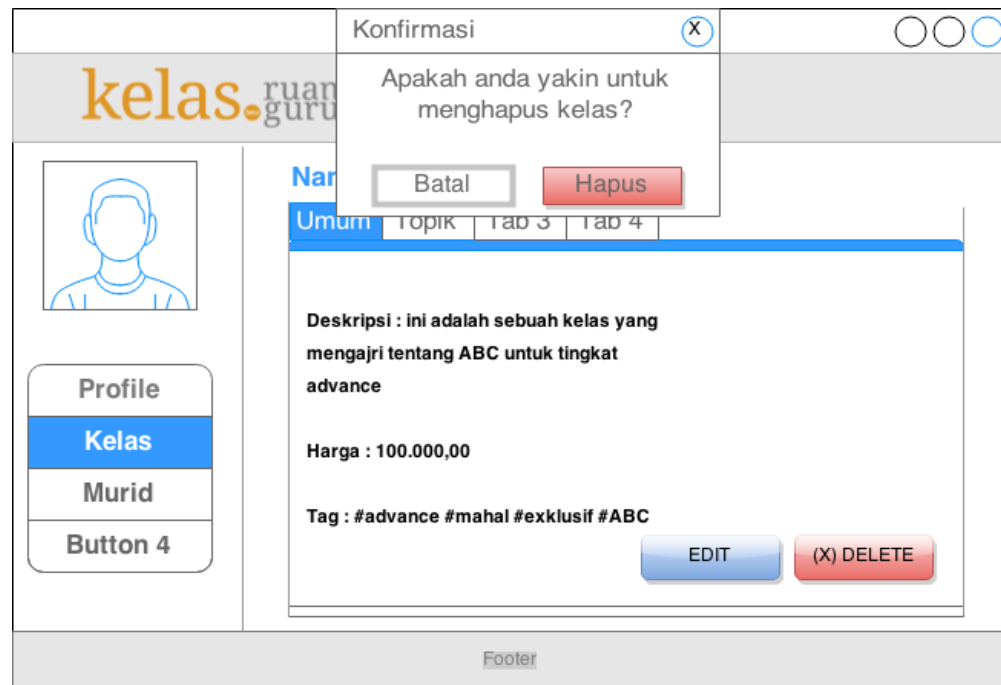
2.3 Sequence Diagram



2.4 Explanation View class and Controller class

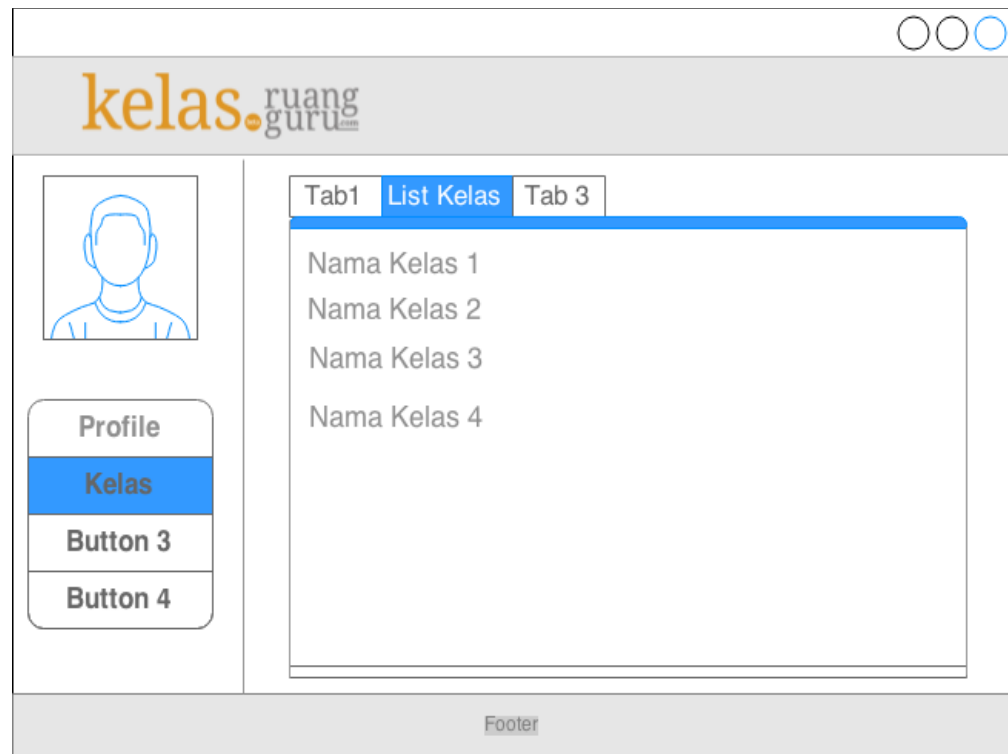
View Class :

– DetilKelasUI



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

– ListKelasUI



Controller Class :

- DeleteKelas() adalah *method* untuk menghapus sebuah kelas yang hanya berstatus *approved* yang dimiliki oleh guru tersebut

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- Guru telah melakukan *login* ke dalam sistem ruangguru.com
- Guru telah melihat detail sebuah kelas yang hanya berstatus *approved*

3.3 Post-Conditions

Kelas berhasil dihapus

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: *Retrieve Kelas*

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case retrieve kelas*, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Saqib

1.2 Actor

User

2. Flow of Events – Design

2.1 Brief Description

Use case ini menerangkan skenario ketika *user* ingin melihat detil sebuah kelas

2.2 Use case Scenario

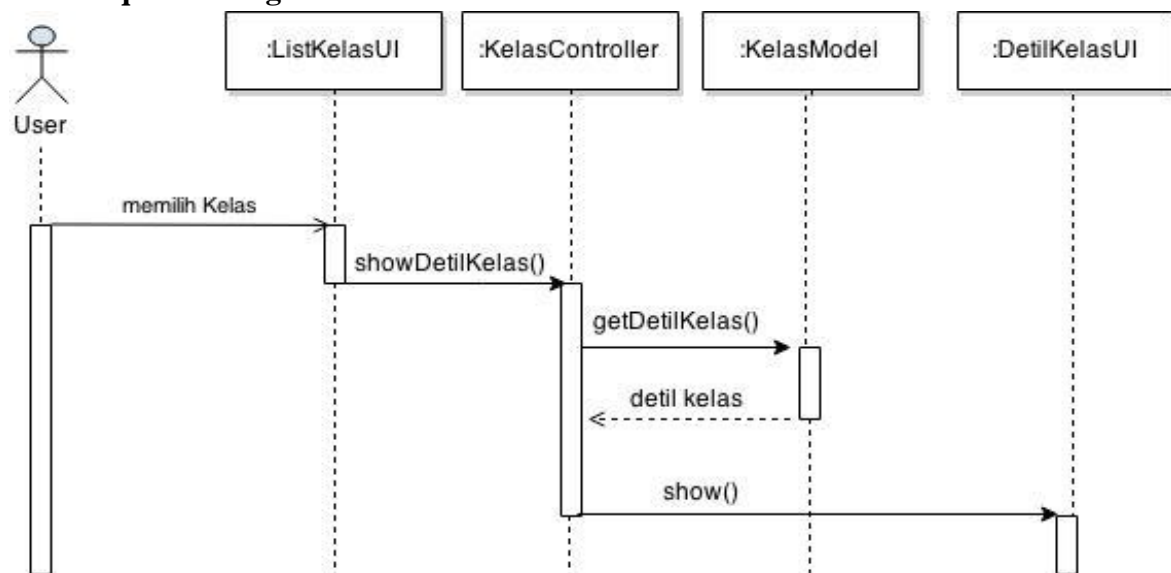
2.2.1 Basic Flow

Actor Actions	System Responses
1. Memilih menu untuk menampilkan detil sebuah kelas	
	2. Menampilkan detil sebuah kelas mencakup nama, deskripsi, harga, <i>tag</i> , topik pada kelas tersebut

2.2.2 Alternative Flows

N/A

2.3 Sequence Diagram

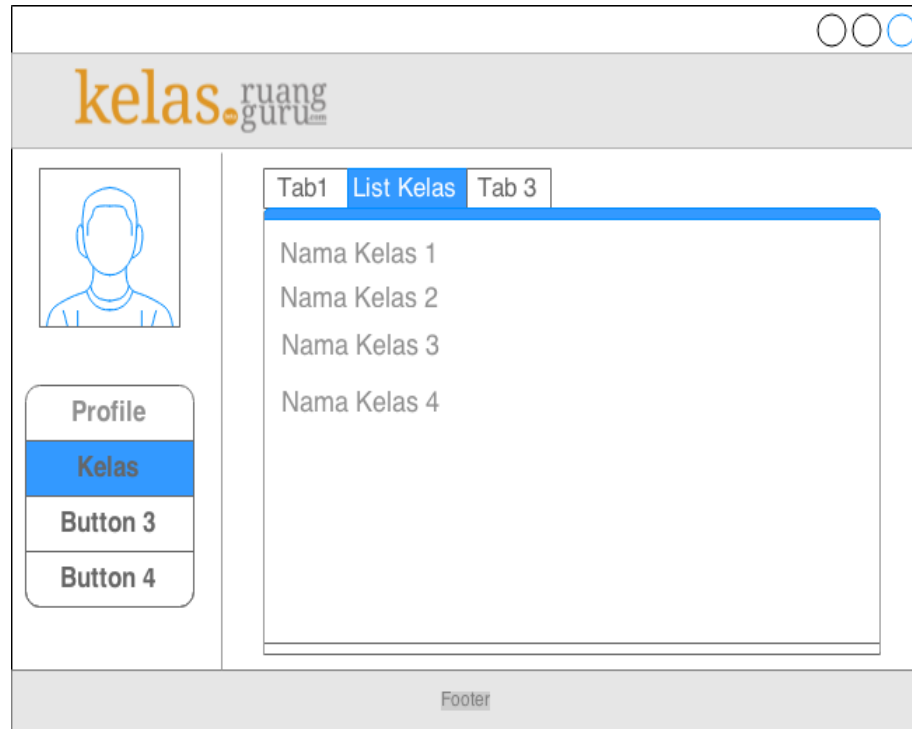


SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

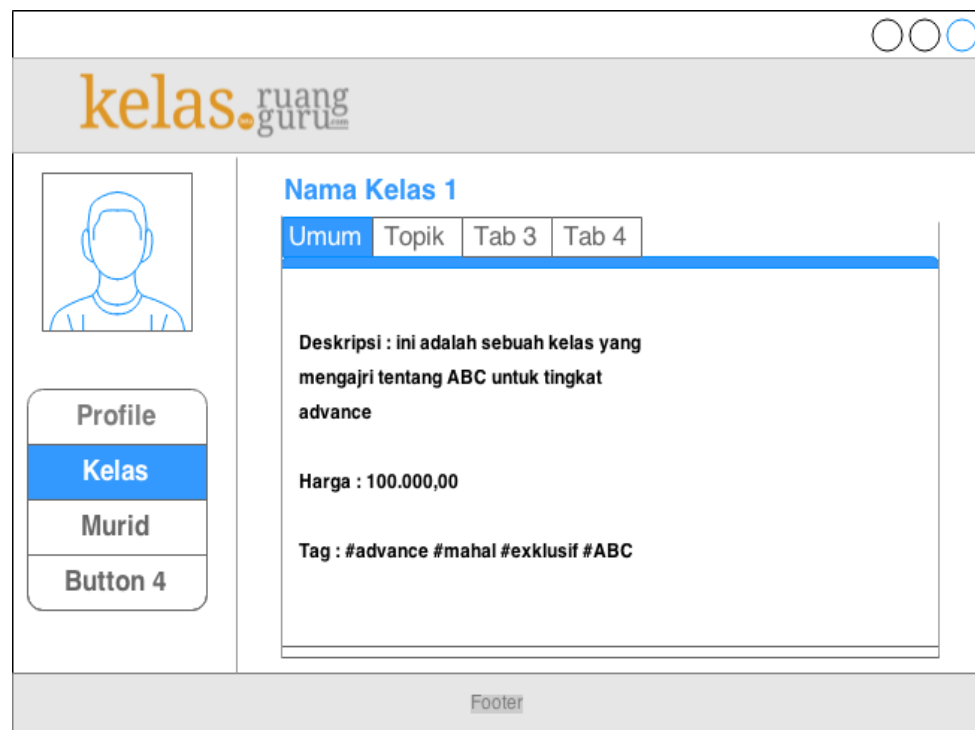
2.4 Explanation View class and Controller class

View Class:

- ListKelasUI



- DetilKelasUI



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Controller Class

- showDetilKelas() melakukan proses pengambilan data tentang rincian pada sebuah kelas, mencakup nama, deskripsi, *tag*, harga, dan topik-topik yang pada kelas tersebut

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- Guru telah melakukan *login* ke dalam sistem ruangguru.com
- Guru telah melihat daftar kelas

3.3 Post-Conditions

Kelas dapat dilihat *user*

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: Melihat Daftar Kelas

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case* melihat daftar kelas, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Saqib

1.2 Actor

User

2. Flow of Events – Design

2.1 Brief Description

Use case ini menerangkan skenario ketika *user* ingin melihat daftar kelas yang sesuai dengan hak akses *user* tersebut.

2.2 Use case Scenario

2.2.1 Basic Flow

Actor Actions	System Responses
1. Memilih menu untuk melihat daftar kelas	

2.2.1.1 Sub Flow 1 Menampilkan semua kelas

Actor Actions	System Responses
	1. Menampilkan halaman yang berisi daftar seluruh kelas yang ada pada sistem

2.2.1.2 Sub Flow 2 Menampilkan kelas yang sudah published

Actor Actions	System Responses
	1. Menampilkan halaman yang berisi daftar seluruh kelas yang berstatus <i>published</i>

2.2.1.3 Sub Flow 3 Menampilkan kelas yang dibuat user tersebut

Actor Actions	System Responses
	1. Menampilkan halaman yang berisi daftar seluruh kelas yang dibuat oleh <i>user</i>

2.2.1.4 Sub Flow 4 Menampilkan kelas yang diikuti

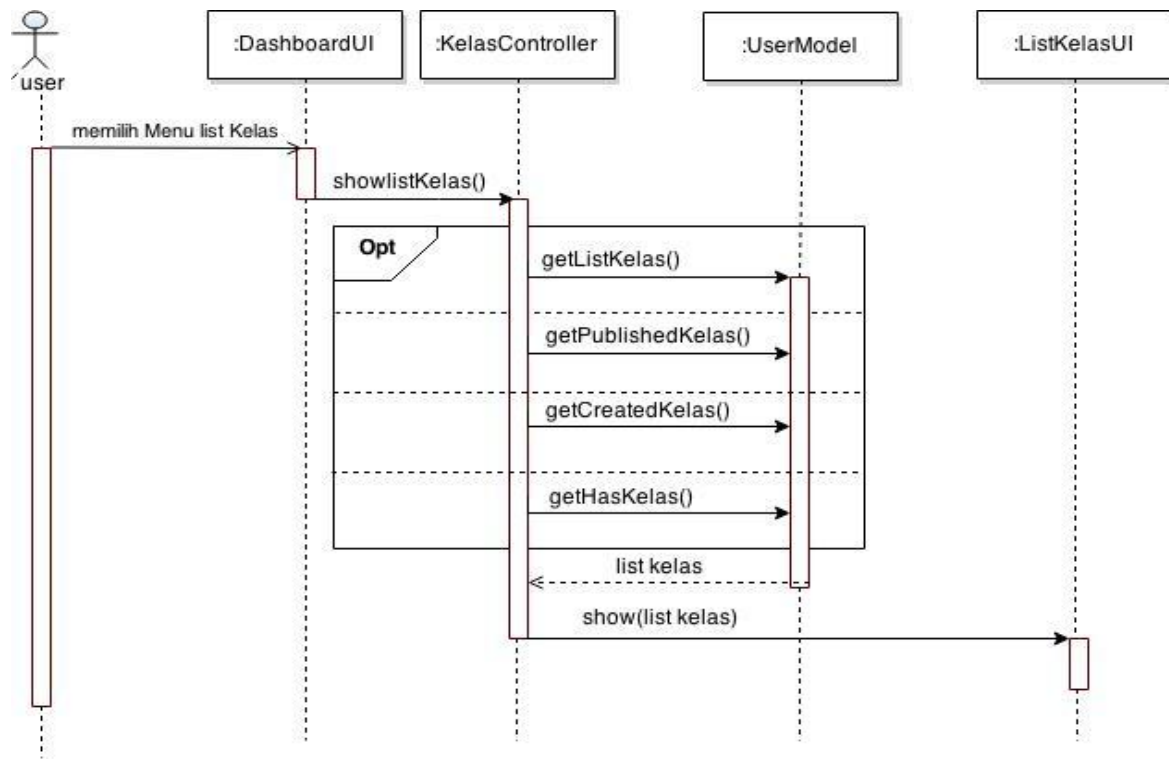
Actor Actions	System Responses
	1. Menampilkan halaman yang berisi daftar seluruh kelas yang diikuti oleh <i>user</i>

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.2.2 Alternative Flows

N/A

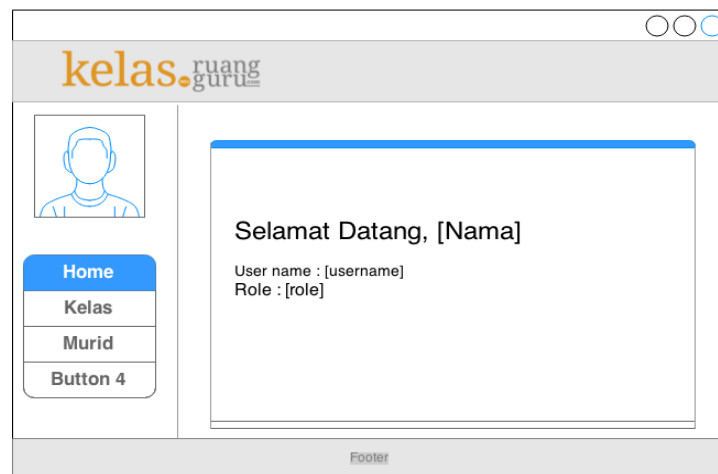
2.3 Sequence Diagram



2.4 Explanation View class and Controller class

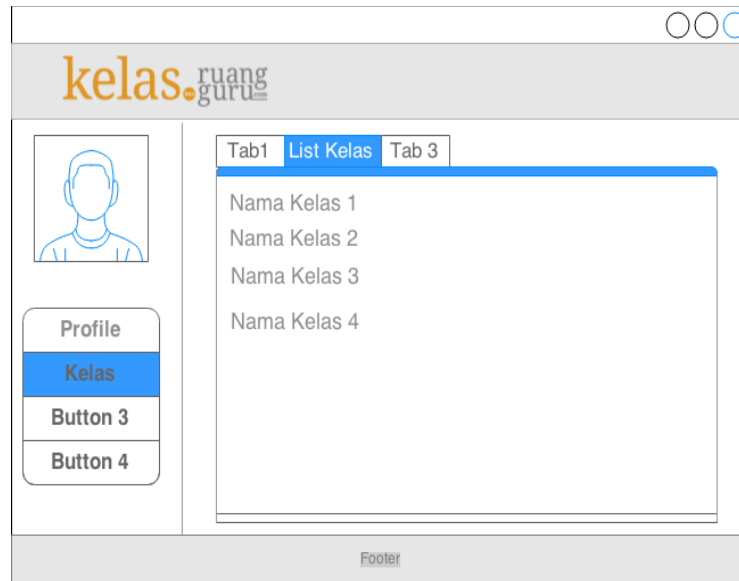
View Class :

- DasboardUI



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

– ListKelasUI



Controller Class:

- ShowListKelas() melakukan pengecekan hak akses seorang *user* untuk menentukan kelas dengan status apa yang bisa *user* tersebut lihat

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

User telah melakukan *login* ke dalam sistem ruangguru.com

3.3 Post-Conditions

User dapat melihat daftar kelas

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: Mengubah Status Kelas

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case* mengubah status kelas, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Mira Pravitasari

1.2 Actor

Admin

2. Flow of Events – Design

2.1 Brief Description

Use-case ini menerangkan tentang skenario ketika *admin* mengubah status kelas. Dianggap *admin* telah membuka sebuah kelas dan mengakses materi pada kelas tersebut. Jika *admin* menganggap materi di kelas tersebut layak, *admin* akan mengubah status kelas dan sistem akan menyimpan perubahan status tersebut.

2.2 Use case Scenario

2.2.1 Basic Flow

Actor Actions	System Responses
1. Memilih pilihan ubah status kelas	
	2. Menampilkan pilihan Pengubahan Status Kelas
3. Jika Kelas berstatus <i>Pending Approve</i> , lakukan <i>Sub Flow 1</i> Jika Kelas berstatus <i>Pending Publish</i> , lakukan <i>Sub Flow 2</i>	

2.2.1.1 Sub Flow 1 Approve Kelas

Actor Actions	System Responses
1. Memilih pilihan <i>Approve</i> kelas.	
	2. Menyimpan perubahan status kelas

2.2.1.2 Sub Flow 2 Publish/Unpublish Kelas

Actor Actions	System Responses
1. Memilih pilihan <i>Publish</i> kelas.	
	2. Menyimpan perubahan status kelas

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.2.2 Alternative Flows

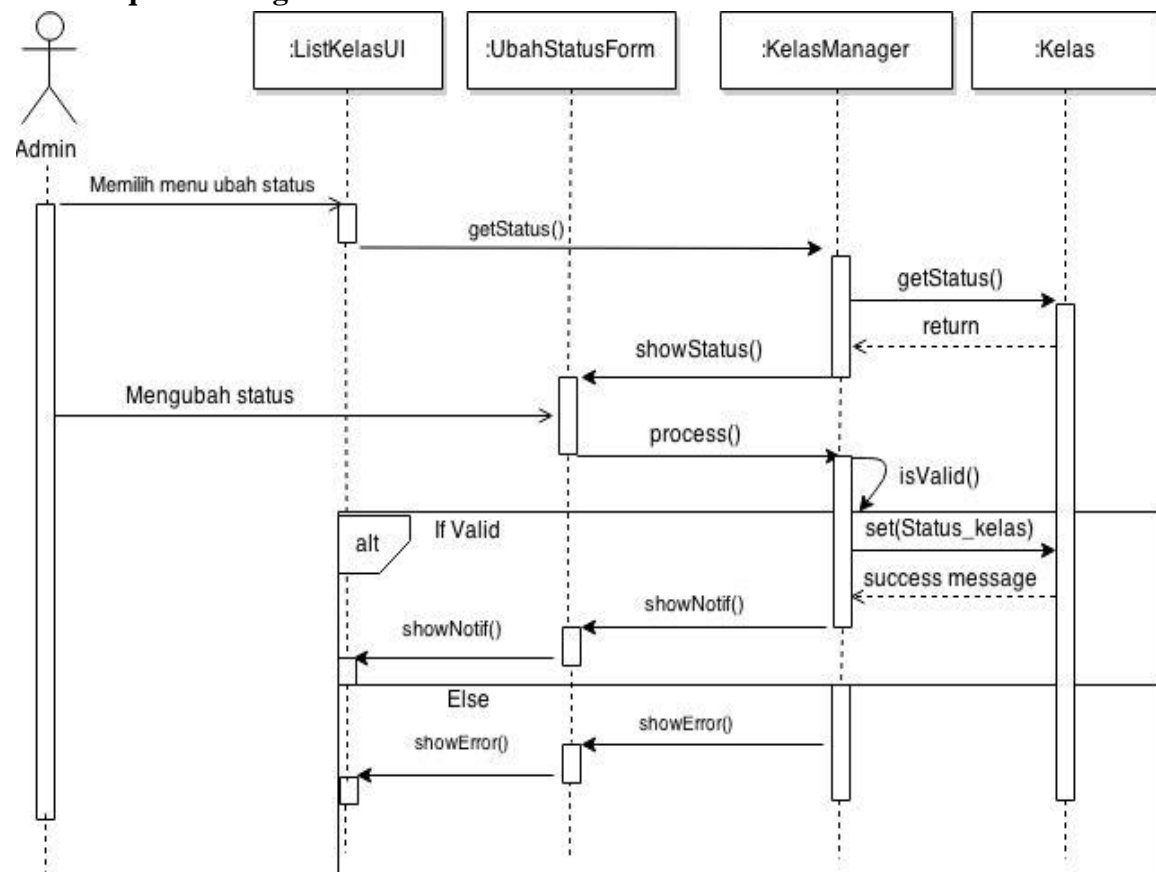
2.2.2.1 Alternative flows Sub Flow 1 1A

Jika materi kelas dianggap tidak layak, *Admin* dapat mengubah status kelas dari *pending approve* kembali menjadi *unapproved* dengan memilih pilihan *unapprove*. Langkah ini dilanjutkan kelangkah S1-2.

2.2.2.2. Alternative flows Sub Flow 2 1A

Jika materi kelas dianggap tidak layak, *Admin* dapat mengubah status kelas dari *pending publish* kembali menjadi *Approved* dengan memilih pilihan *Unpublish*. Langkah ini dilanjutkan kelangkah S2-2.

2.3 Sequence Diagram



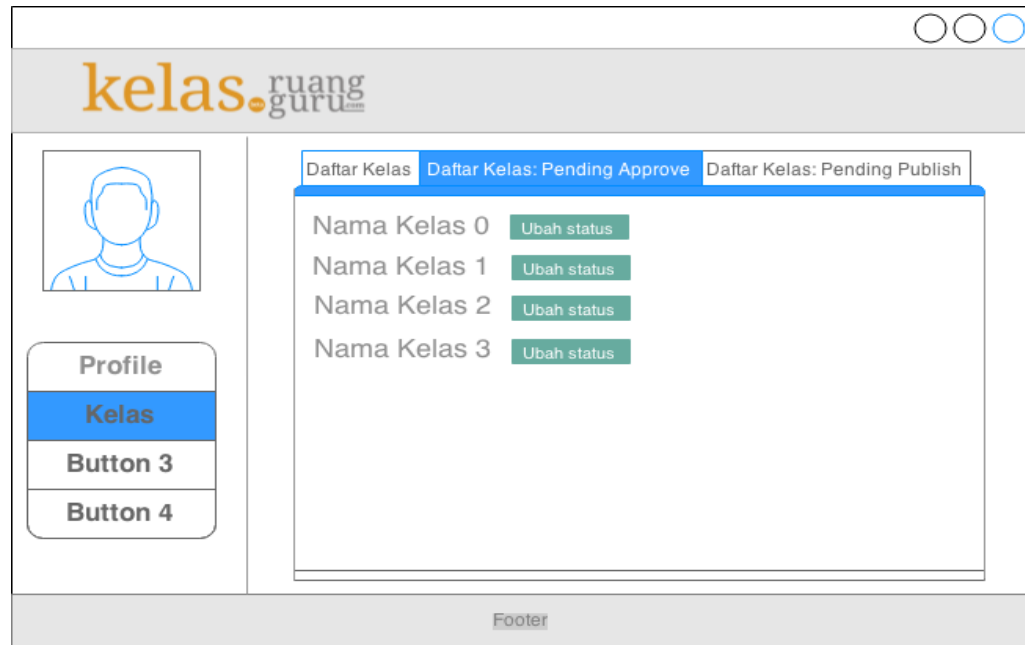
SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.4 Explanation View class and Controller class

View Class :

- ListKelasUI

Merupakan kelas yang menampilkan daftar seluruh kelas dan dapat difilter berdasarkan status kelas. Berikut tampilan dari ListFeedbackUI.



- UbahStatusFormUI

Merupakan kelas yang menampilkan pengubahan status sebuah kelas yang dapat dipilih oleh *Admin*. Berikut tampilan dari UbahStatusFormUI.



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Controller Class:

- KelasManager, yang memiliki *method* :
 - getStatus () : *Method* ini mengambil status kelas yang akan diubah.
 - set(Status_kelas) : *Method* ini menyimpan perubahan status yang dilakukan.

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- Admin telah melakukan *login* ke dalam sistem ruangguru.com
- Admin berada pada halaman yang menampilkan daftar kelas yang belum *publish*.

3.3 Post-Conditions

Sistem menampilkan isian *feedback*.

3.4 Extension Points

3.4.1 Include

Use-case ini memiliki *relationship include* dengan *use-case* Memberi *Feedback*, di mana Pengubahan Status Kelas harus diikuti dengan pemberian *feedback* terhadap kelas.

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: Melihat *List Feedback*

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case* melihat daftar *feedback*, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Mira Pravitasari

1.2 Actor

Admin

2. Flow of Events – Design

2.1 Brief Description

Use case ini menerangkan skenario ketika *user* ingin melihat daftar *feedback* yang diberikan kepada sebuah kelas.

2.2 Use case Scenario

2.2.1 Basic Flow

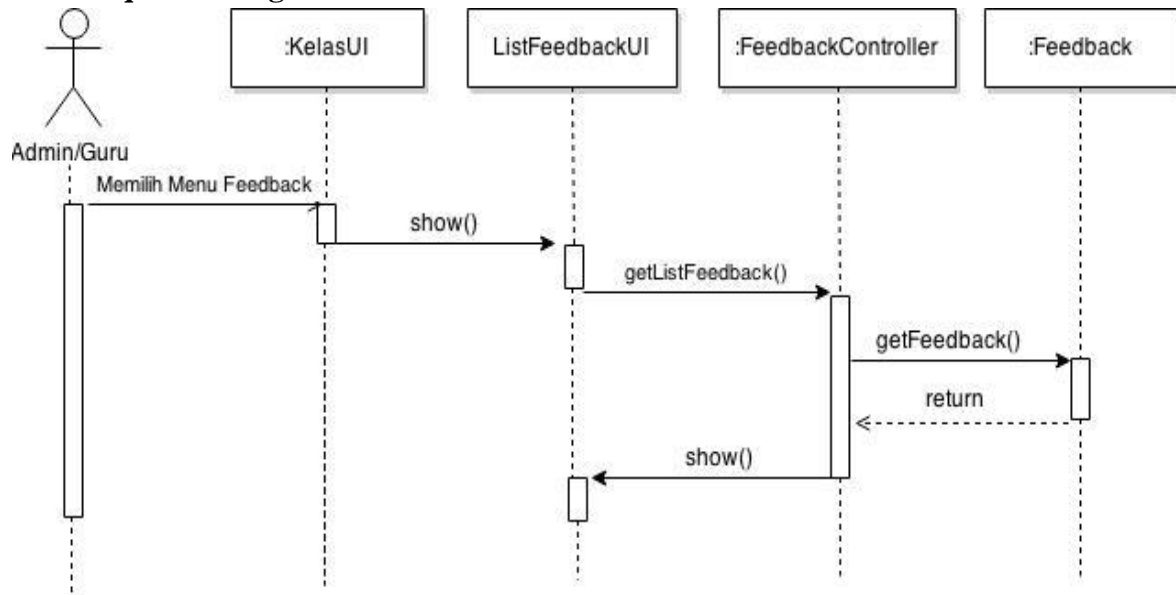
Actor Actions	System Responses
1. <i>User</i> memilih menu untuk melihat daftar <i>feedback</i> pada sebuah kelas.	
	2. Sistem menampilkan halaman yang berisi daftar <i>feedback</i> yang ada pada kelas tersebut

2.2.2 Alternative Flows

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

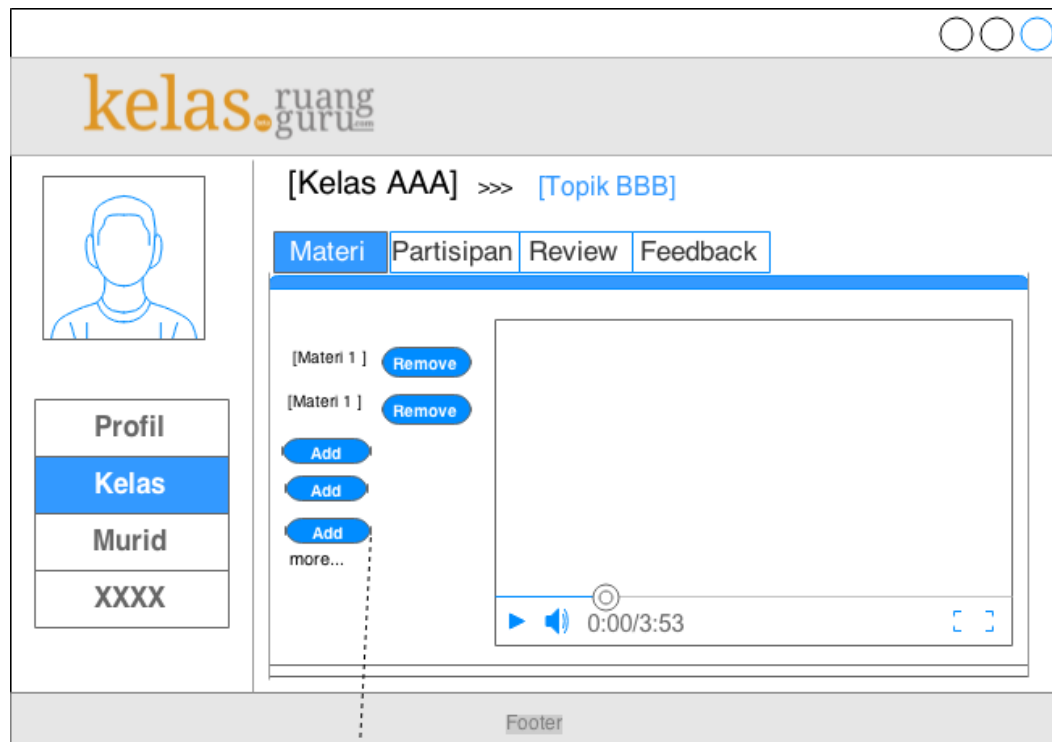
2.3 Sequence Diagram



2.4 Explanation View class and Controller class

View Class :

- KelasUI : Merupakan tampilan dari sebuah Kelas. Berikut tampilan dari KelasUI



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

- ListFeedbackUI

Merupakan tampilan daftar seluruh *feedback* dari sebuah kelas beserta tanggapannya. Berikut tampilan dari ListFeedbackUI.

The screenshot shows a web application interface for 'kelas.ruangguru.com'. On the left is a sidebar with a user profile icon and a menu containing 'Profile', 'Kelas' (highlighted), 'Murid', and 'Button 4'. The main content area is titled 'Kelas Xyz >> List Feedback' and features two tabs: 'List Feedback' (active) and 'Beri Feedback'. Below the tabs, there are two feedback entries. Each entry consists of a date label ('Tanggal Feedback 1' and 'Tanggal Feedback 2'), a text input field for 'Isi Feedback', and a text input field for 'Tanggapan 1' and 'Tanggapan 2' respectively. A 'Footer' label is at the bottom.

This screenshot shows the same interface as the previous one, but with additional navigation tabs in the main content area: 'Detil', 'Materi', 'Review', 'Partisipan', and 'Feedback' (highlighted). The feedback entries are identical, but the second entry ('Tanggal Feedback 2') now includes a 'Beri Tanggapan' button at the bottom right of its response field. The 'Footer' label remains at the bottom.

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Controller Class :

- FeedbackController, yang memiliki *method* :
 - `getFeedback()` : *Method* ini mengambil *feedback* yang akan ditampilkan dalam daftar.

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

Admin telah melakukan *login* ke dalam sistem ruangguru.com

3.3 Post-Conditions

Sistem menampilkan daftar *feedback*.

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: Memberi *Feedback*

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case* memberi *feedback*, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Mira Pravitasari

1.2 Actor

Admin

2. Flow of Events – Design

2.1 Brief Description

Use-case ini menerangkan tentang skenario ketika *admin* memberikan *feedback* terhadap kelas. *Admin* mengisi sebuah isian yang memuat pendapat dan keterangan terkait kelas serta materi kelas dan sistem akan melakukan validasi *input* dan jika *input valid* secara otomatis menyimpan data.

2.2 Use case Scenario

2.2.1 Basic Flow

Actor Actions	System Responses
1. <i>User</i> memilih menu Memberi <i>Feedback</i> pada halaman Daftar <i>Feedback</i> Kelas	
	2. Sistem menampilkan isian Memberi <i>Feedback</i>
3. <i>User</i> mengisi isian <i>feedback</i>	
	4. Sistem melakukan <i>validasi</i> data
	5. Sistem menampilkan notifikasi Pemberian <i>Feedback</i> berhasil

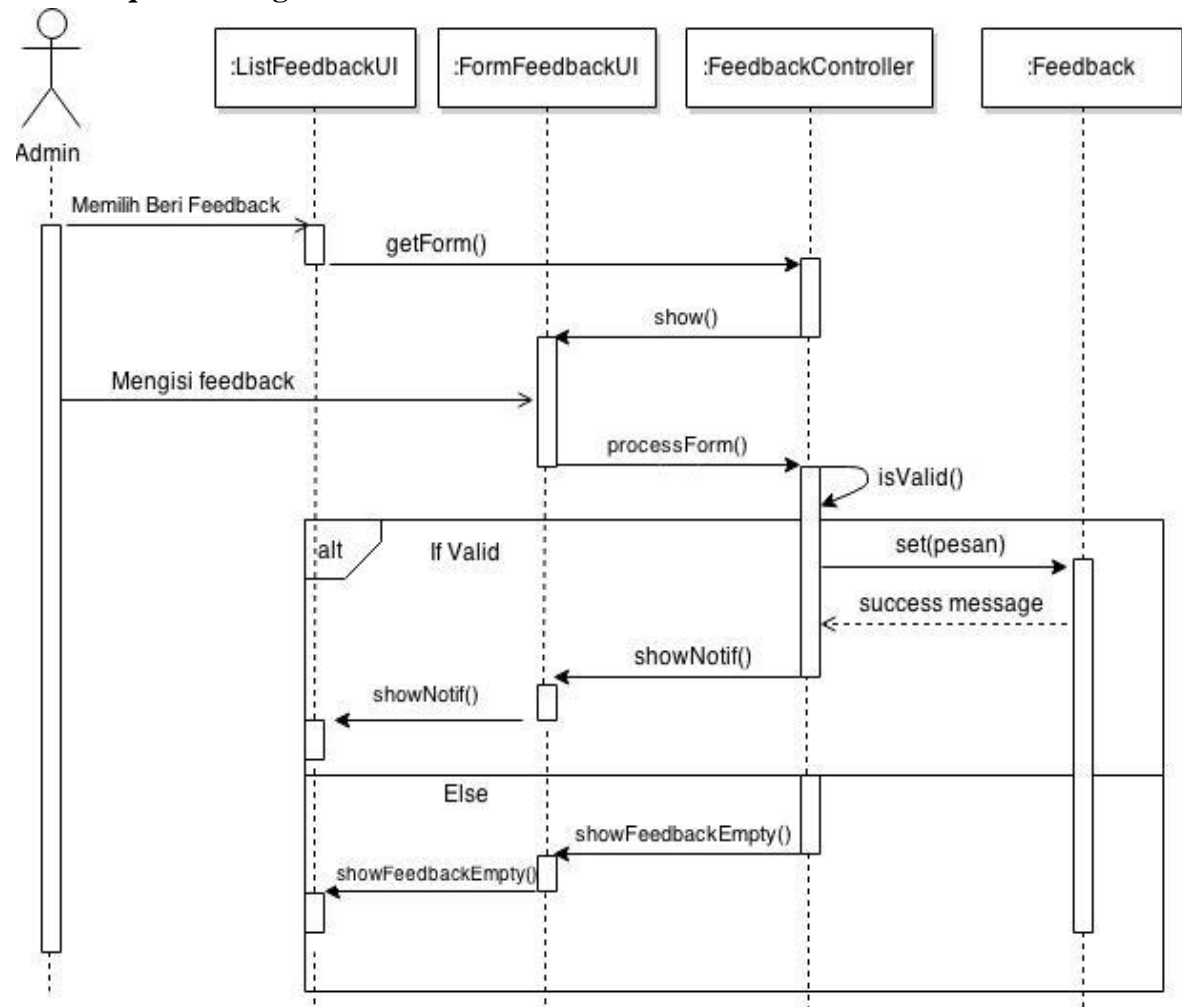
2.2.2 Alternative Flows

Alternative flow 4A

Apabila isian *feedback* kosong, sistem akan menampilkan pesan bahwa *feedback* belum diisi dan kembali menampilkan halaman Memberi *Feedback* (2).

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.3 Sequence Diagram



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.4 Explanation View class and Controller class

View Class :

- ListFeedbackUI

Merupakan kelas yang menampilkan daftar seluruh *feedback* dari sebuah kelas beserta tanggapannya. Berikut tampilan dari ListFeedbackUI

- FormFeedbackUI

Merupakan kelas yang menampilkan isian *feedback* terhadap sebuah kelas yang dapat diisi oleh *Admin*. Berikut tampilan dari FormFeedbackUI.

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Controller Class :

- FeedbackController, yang memiliki *method* :
 - set(pesan) : *Method* ini menyimpan *feedback* dari *admin*.

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

Admin telah melakukan *login* ke dalam sistem ruangguru.com

3.3 Post-Conditions

Admin berhasil memberi *feedback*.

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: Menanggapi *Feedback*

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case* menanggapi *feedback*, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Mira Pravitasari

1.2 Actor

Guru

2. Flow of Events – Design

2.1 Brief Description

Use-case ini menerangkan tentang skenario ketika guru memberikan tanggapan terhadap *feedback* yang didapat terhadap kelas yang ia buat. Guru mengakses halaman kelas kemudian melihat *feedback* dan memberikan tanggapan. Sistem akan melakukan validasi *input* dan jika *input valid* secara otomatis menyimpan data.

2.2 Use case Scenario

2.2.1 Basic Flow

Actor Actions	System Responses
1. <i>User</i> memilih menu Tanggapi <i>Feedback</i> .	
	2. Sistem menampilkan isian untuk tanggapan terhadap sebuah <i>feedback</i> .
3. <i>User</i> mengisi isian untuk menanggapi tanggapan.	
	4. Sistem melakukan validasi isian. Jika isian <i>valid</i> , akan disimpan
	5. Sistem menampilkan pesan sukses menyimpan tanggapan

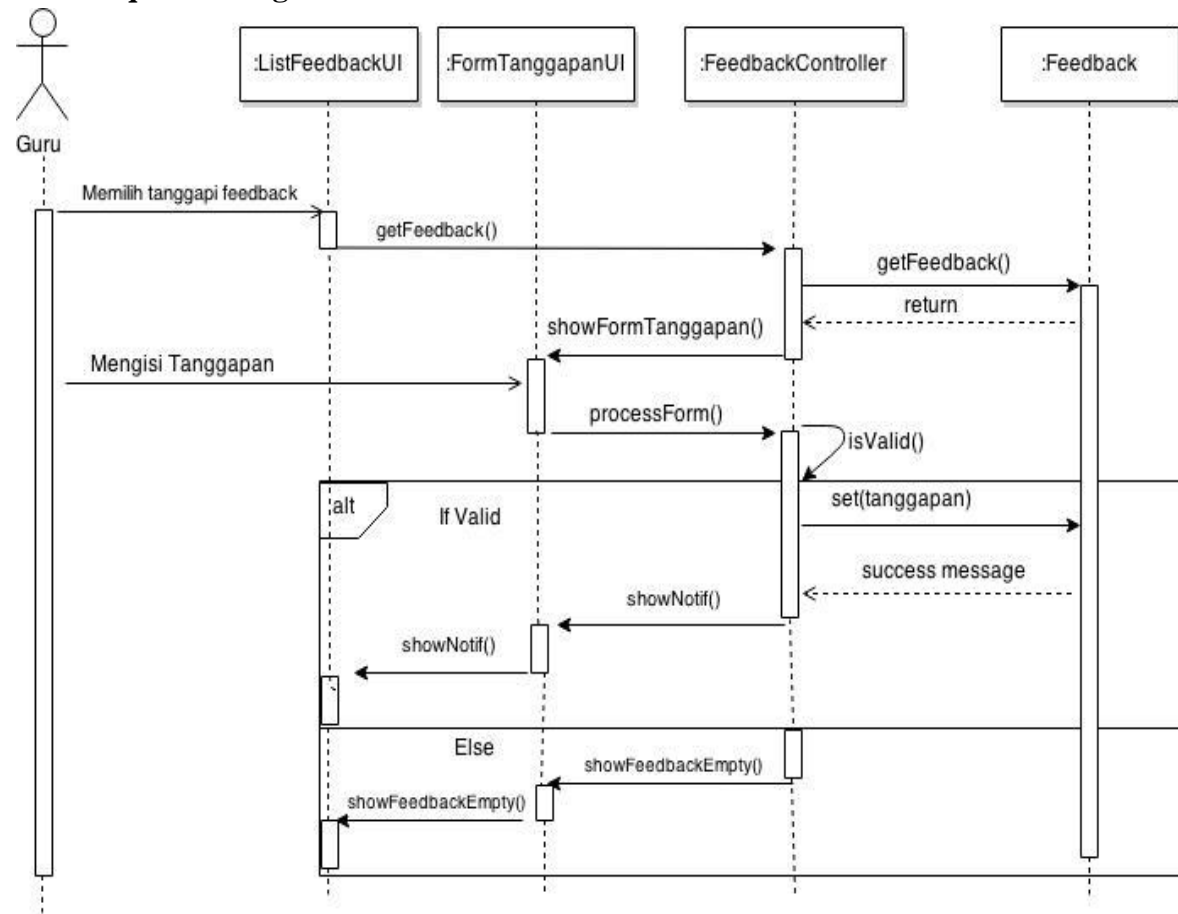
2.2.2 Alternative Flows

2.2.2.1 Alternative Flow 3A

Apabila isian tanggapan masih kosong, sistem akan menampilkan pesan bahwa isian *feedback* kosong dan kembali ke halaman isian tanggapan *Feedback* (2).

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.3 Sequence Diagram



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.4 Explanation View class and Controller class

View Class:

- ListFeedbackUI

Merupakan kelas yang menampilkan daftar seluruh *feedback* dari sebuah kelas beserta tanggapannya. Berikut tampilan dari ListFeedbackUI.

- FormTanggapanUI

Merupakan kelas yang menampilkan isian tanggapan terhadap sebuah *feedback* yang dapat diisi oleh guru. Berikut tampilan dari FormTanggapanUI.

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Controller Class :

- FeedbackController
 - getFeedback() : *Method* ini mengambil *feedback* yang akan diberikan tanggapan oleh guru
 - set(tanggapan) : *Method* ini menyimpan tanggapan guru terhadap sebuah *feedback*.

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- Guru telah melakukan *login* ke dalam sistem ruangguru.com
- Berada pada halaman yang memuat daftar *feedback* kelas

3.3 Post-Conditions

Guru berhasil menanggapi *feedback*.

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: Melihat Daftar Topik Kelas

1. Introduction

Use Case ini menjelaskan tentang bagaimana proses dalam melihat daftar topik yang dilakukan oleh *user/aktor*, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Vasko Dagama

1.2 Actor

User

2. Flow of Events – Design

2.1 Brief Description

Use Case ini menjelaskan tentang bagaimana aktor dapat melihat daftar topik yang ada pada suatu kelas. Aktor memilih salah satu kelas yang ingin dilihat daftar topiknya, kemudian sistem akan menampilkan semua daftar topik yang ada pada kelas tersebut

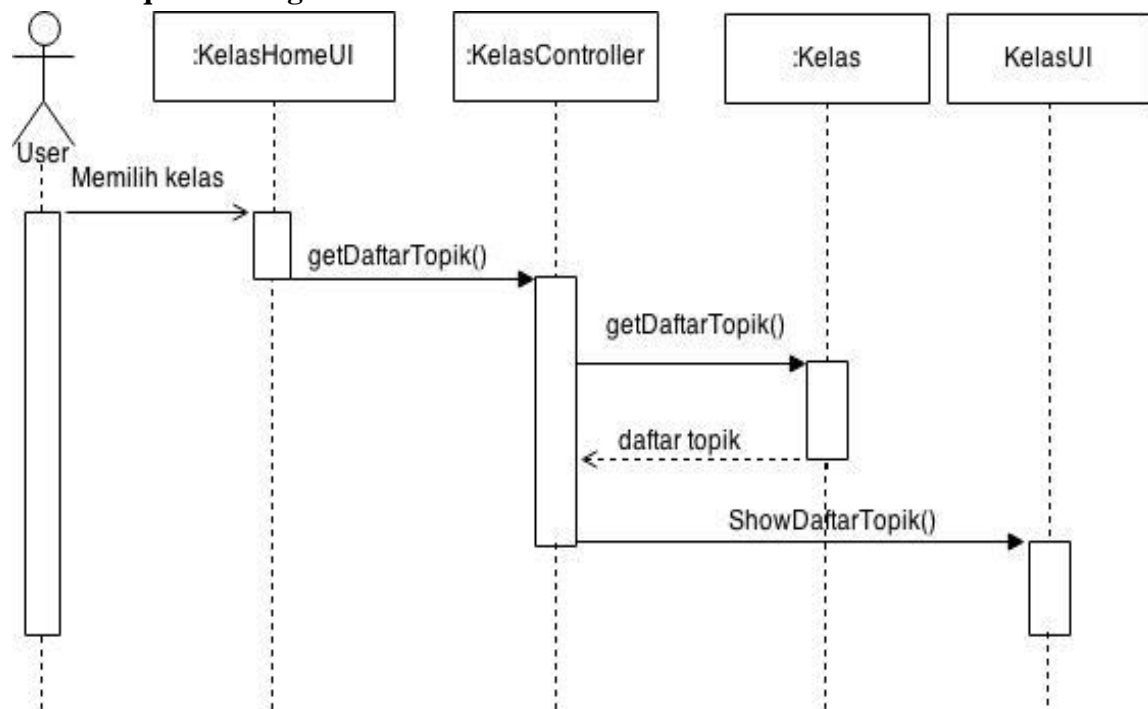
2.2 Use Case Scenario

2.2.1 Basic Flow

Actor Actions	System Responses
1. Memilih menu topic	
	2. Menampilkan topik-topik yang ada pada kelas tersebut

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

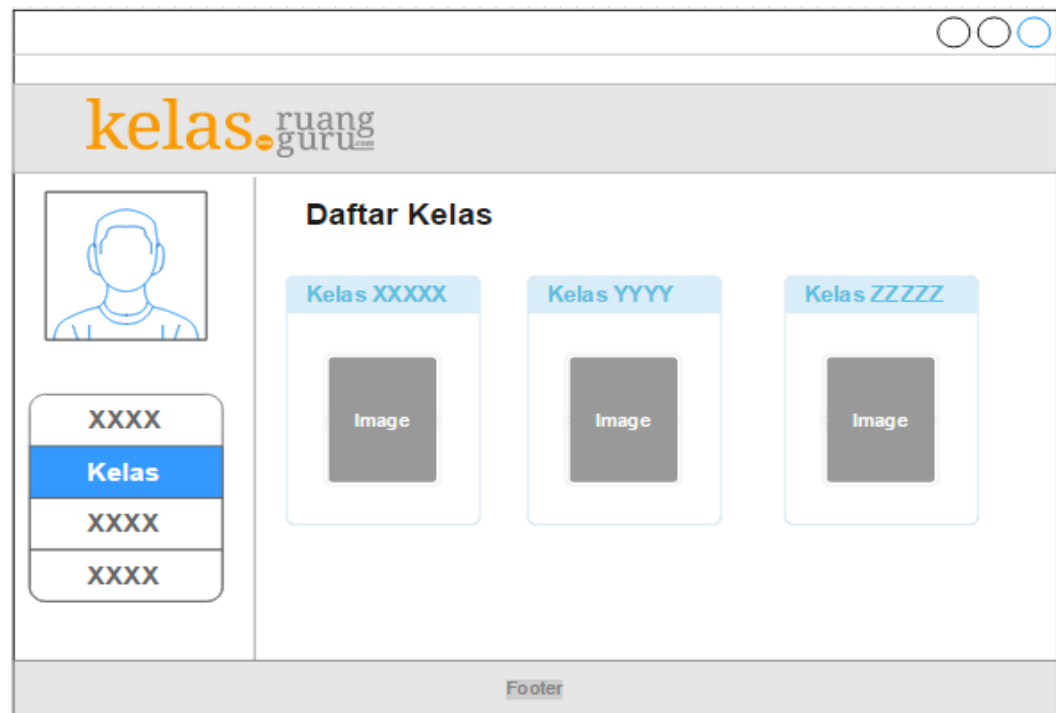
2.3 Sequence Diagram



2.4 Explanation of View Class and Controller Class

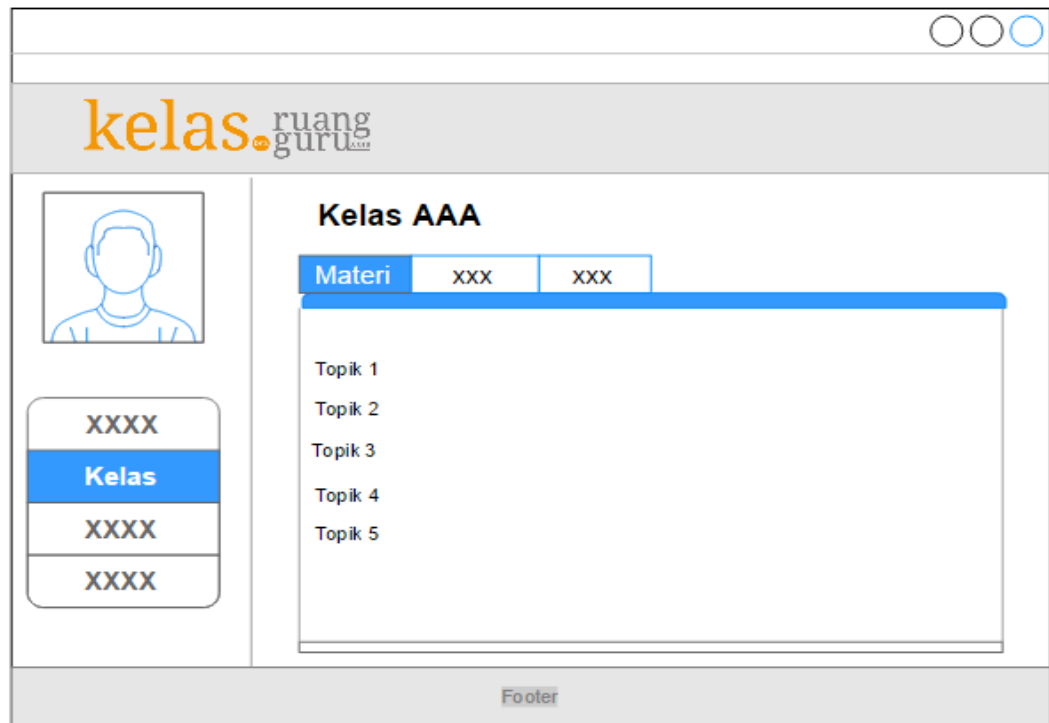
View Class:

- DashboardHomeUI : Merupakan halaman dimana terdapat daftar-daftar kelas.



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

- KelasUI : Merupakan halaman kelas dimana di dalamnya terdapat daftar-daftar topik dari suatu kelas



Controller Class :

- KelasController, yang memiliki *method* :
 - `getDaftarTopik()` : Berfungsi untuk *retrieve* daftar-daftar topik yang ada pada suatu kelas

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- *User* telah *login* dalam sistem Ruangguru.com
- *User* telah melakukan use case *retrieve* kelas

3.3 Post-Conditions

Sistem menampilkan daftar topik

3.4 Relationship

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: Melihat Daftar Materi

1. Introduction

Use Case ini menjelaskan scenario tentang bagaimana *user* melihat daftar materi pada suatu topik kelas tertentu, beserta *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Vasko Dagama

1.2 Actor

User

2. Flow of Events – Design

2.1 Brief Description

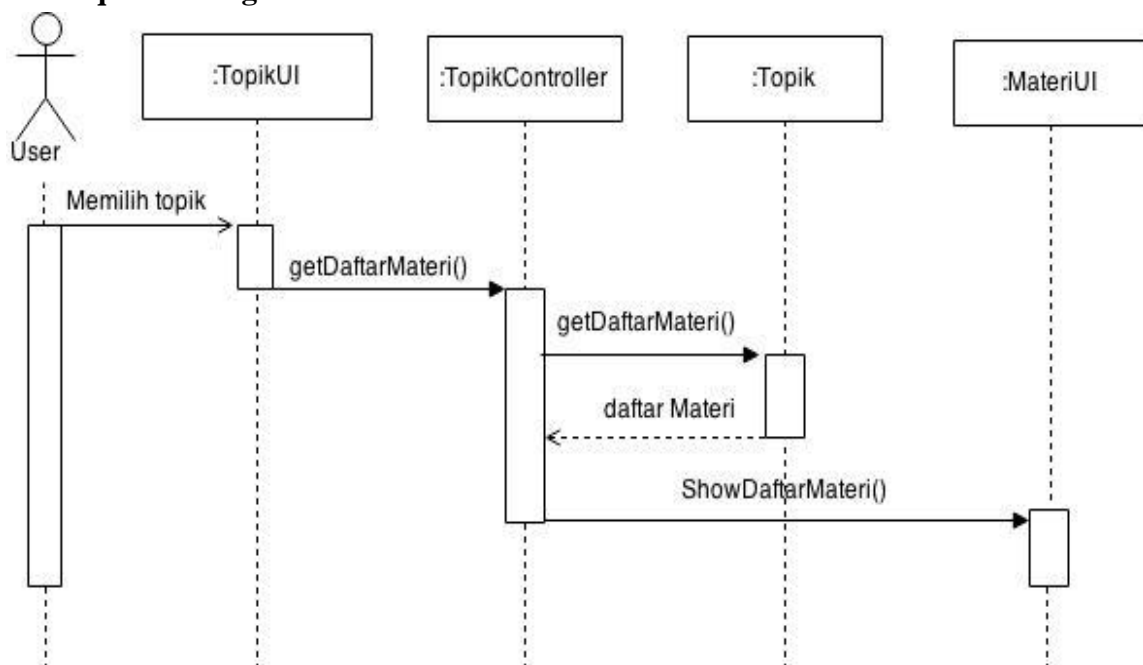
Use Case ini menjelaskan bagaimana aktor dapat melihat daftar materi pada suatu topik kelas. *User* memilih salah satu topik yang mana materinya ingin dilihat, kemudian sistem akan menampilkan daftar materi pada topik tersebut .

2.2 Use Case Scenario

2.2.1 Basic Flow

Actor Actions	System Responses
1. Memilih salah satu topik yang ingin dilihat materinya.	
	2. Menampilkan materi-materi yang ada pada kelas tersebut

2.3 Sequence Diagram

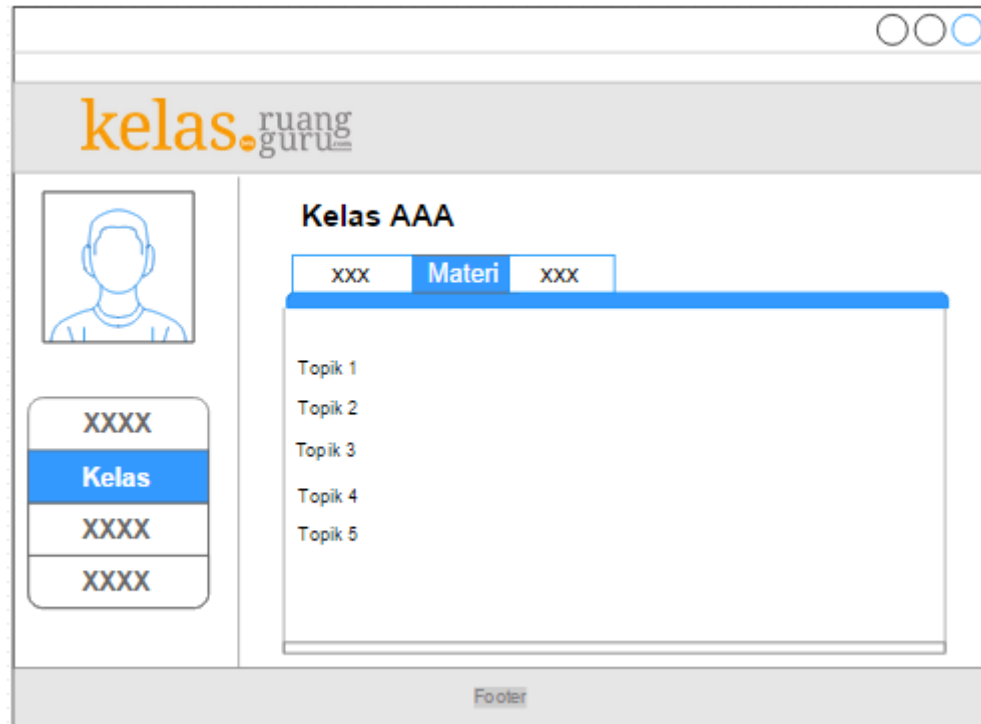


SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

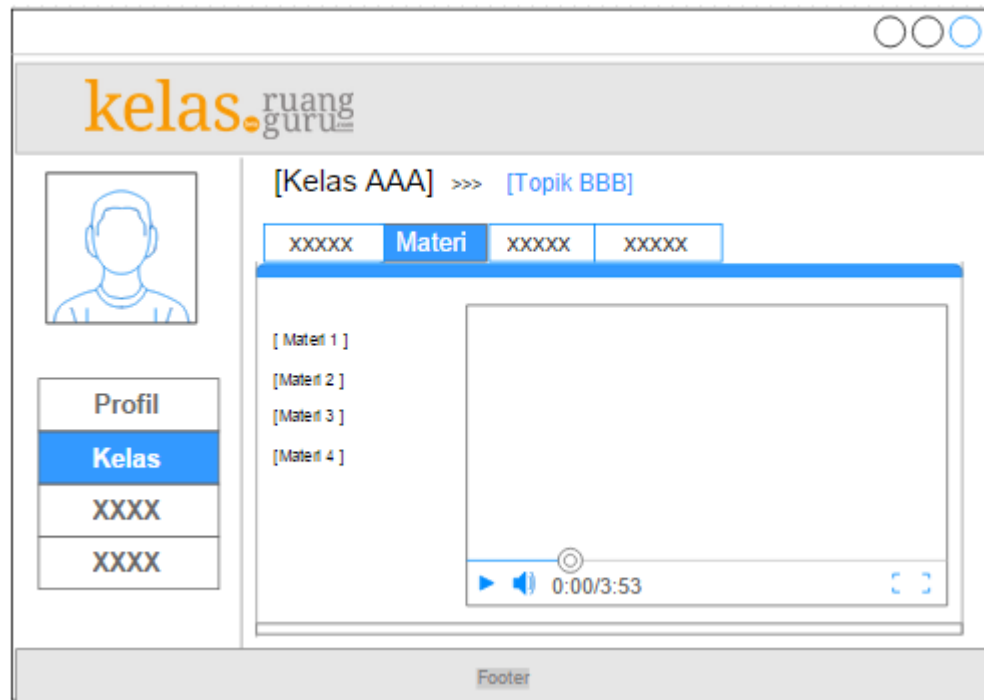
2.4 Explanation of View Class and Controller Class

View Class:

- TopikUI : Merupakan halaman dimana terdapat kumpulan topik-topik



- MateriUI : Merupakan halaman dimana terdapat kumpulan materi-materi



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Controller Class :

- *TopikController*, yang memiliki method :
 - *getDaftarMateri ()* : Berfungsi untuk *meretrieve* daftar-daftar materi.

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- *User* telah *login* dalam sistem Ruangguru.com
- *User* telah melakukan *use case* melihat daftar topik kelas tersebut

3.3 Post-Conditions

Sistem menampilkan daftar materi

3.4 Relationship

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: *Retrieve materi*

1. Introduction

Use Case ini menjelaskan scenario tentang bagaimana *user* mengakses atau *meretrieve* materi, beserta *sequence diagram*, *class view* dan *class controller*.

1.2 Author

Vasko Dagama

1.3 Actor

User

2. Flow of Events – Design

2.1 Brief Description

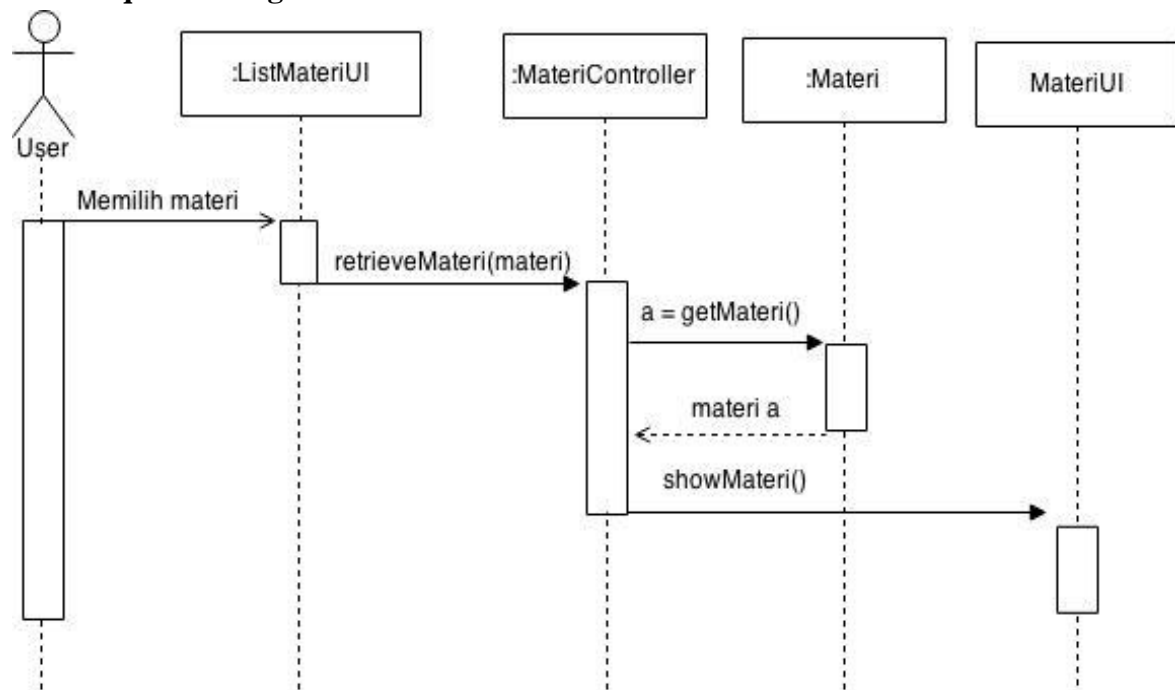
Use case ini menjelaskan bagaimana proses *user* mengakses materi pada suatu topik kelas tertentu. *User* memilih salah satu materi yang akan di akses, kemudian sistem akan menampilkan materi tersebut yang bisa berupa video atau *pdf*.

2.2 Use Case Scenario

2.2.1 Basic Flow

Actor Actions	System Responses
1. Memilih salah satu materi yang ingin dilihat materinya	
	2. Menampilkan materi yang dipilih

2.3 Sequence Diagram

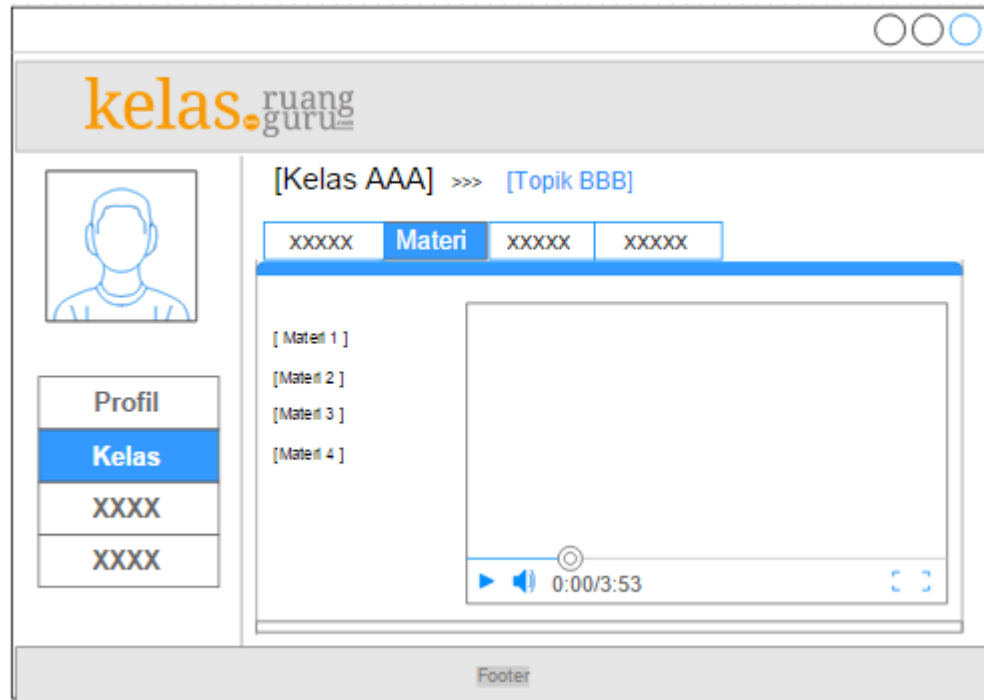


SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

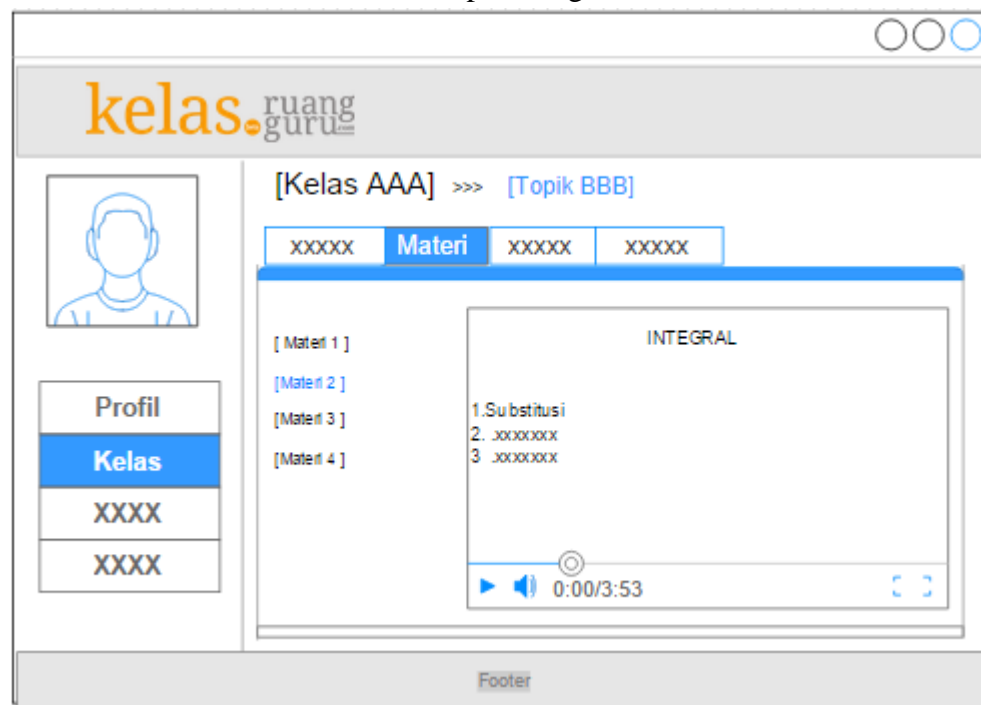
2.4 Explanation of View Class and Controller Class

View Class :

- ListMateriUI



- MateriUI : Halaman dimana *user* dapat mengakses materi



SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Controller Class :

- *MateriController* ,yang memiliki method :
 - *retrieveMateri(materi)* : Berfungsi untuk *meretrieve* sebuah materi yang akan ditampilkan

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- *User* telah *login* dalam sistem Ruangguru.com
- *User* telah melakukan *use case* melihat daftar topik kelas
- *User* memiliki hak akses untuk *meretrieve* materi

3.3 Post-Conditions

Sistem menampilkan materi

3.4 Relationship

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: *Create Materi*

1. Introduction

Pada bagian ini akan menjelaskan mengenai *use case create materi*, *sequence diagram*, *class view* dan *class controller*.

1.1 Author

Vasko Dagama

1.2 Actor

Guru

2. Flow of Events – Design

2.1 Brief Description

Use Case ini akan menjelaskan bagaimana proses guru menambah atau mengupload materi pada suatu kelas. Guru dapat mengupload materi berupa video atau *file* sesuai dengan format yang ditentukan. Setelah guru mengupload materi, sistem akan menyimpan materi tersebut.

2.2 Use Case Scenario

2.2.1 Basic Flow

Actor Actions	System Responses
1. Guru memilih menu untuk tambah materi	
	2. Sistem menampilkan isian nama materi dan slot untuk <i>upload</i> materi
3. Guru mengisi isian dan memilih jenis materi yang akan di <i>upload</i>	
4. Guru mengupload materi	
	5. Sistem memvalidasi format materi
	6. Sistem menampilkan pesan bahwa materi telah berhasil di <i>upload</i> .

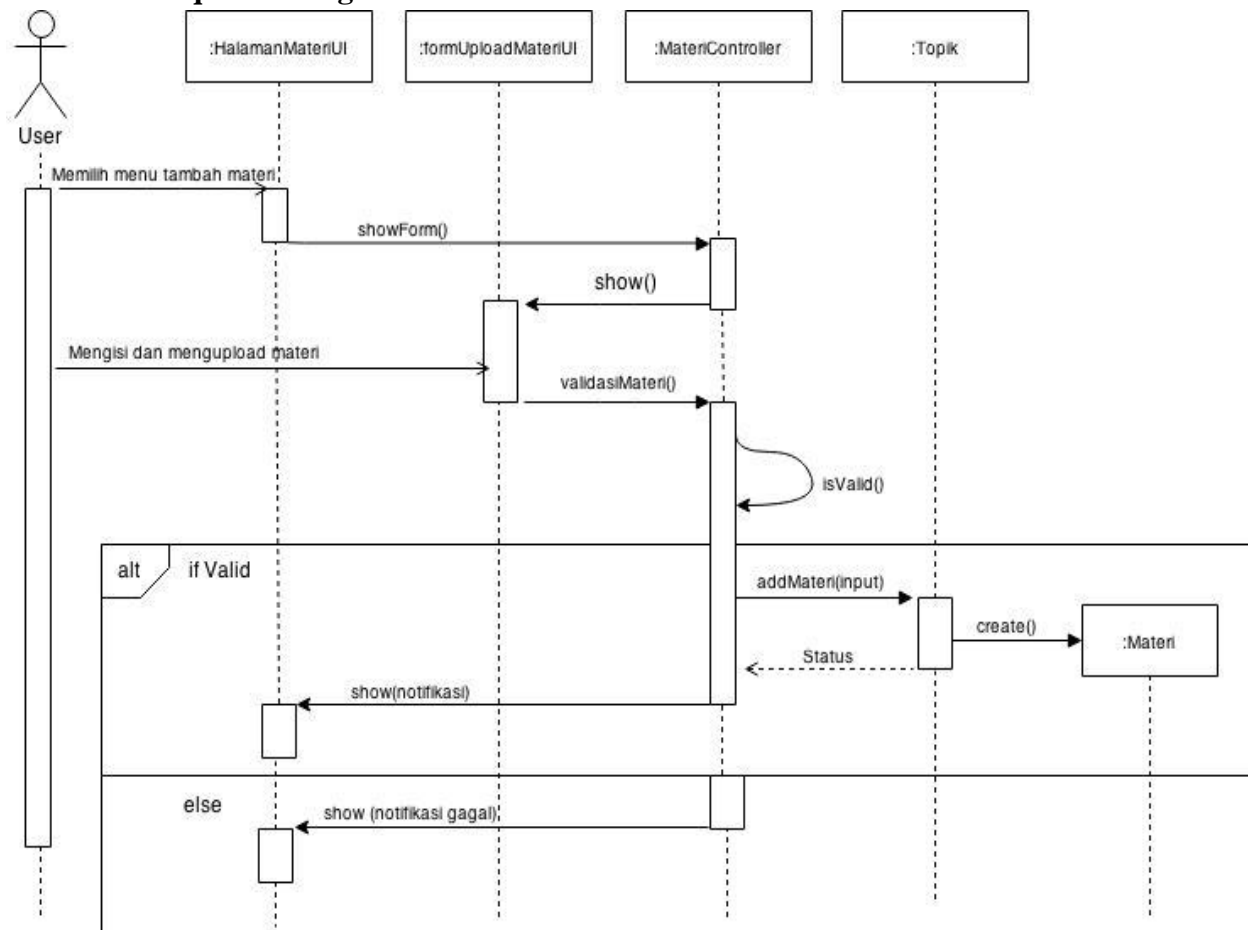
2.2.2 Alternative Flow

2.2.2.1 Alternative Flows 5A

Alternative Flows pada langkah 5, apabila format materi yang diupload tidak sesuai dengan format yang dibutuhkan, maka sistem akan menampilkan pesan kesalahan format. Aktor mengulangi lagi langkah ke 3.

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.3 Sequence Diagram

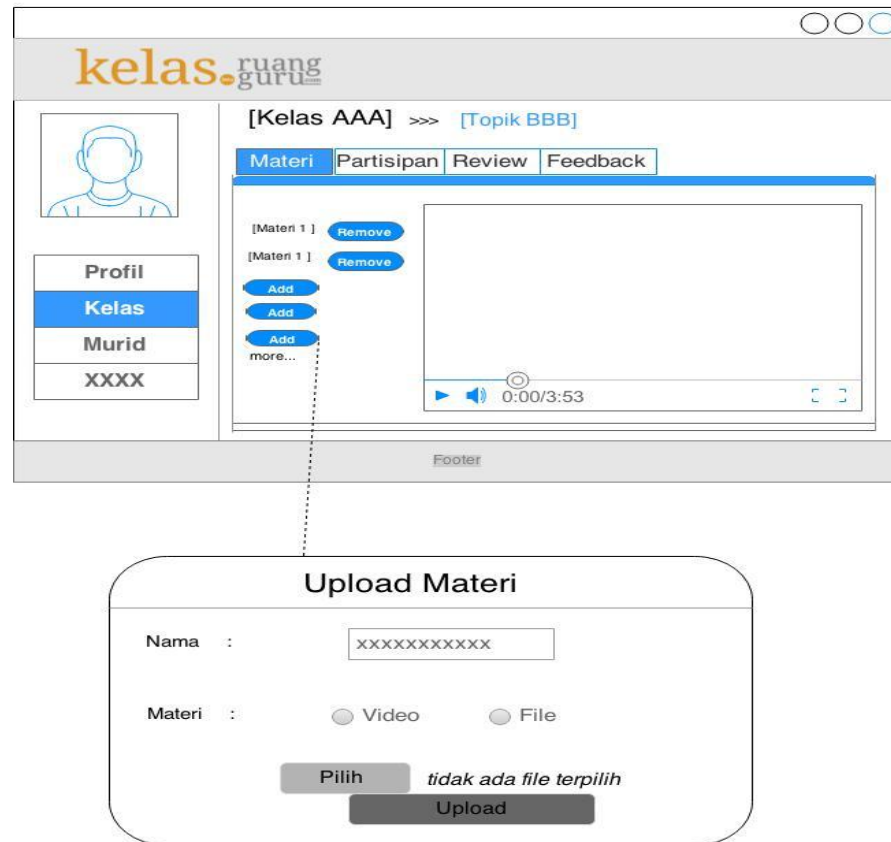


2.4 Explanation of View Class and Controller Class

View Class:

- HalamanMateriUI :Merupakan halaman dimana aktor dapat melihat kumpulan materi yang ada.
- FormUploadMateriUI : Merupakan halaman/sebuah jendela yang muncul ketika aktor melakukan penambahan materi baru (*Upload Materi*).

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015



Controller Class :

- *MateriController* ,yang memiliki *method* :
 - *Validasimateri()* : Berfungsi untuk melakukan validasi materi yang di *upload* oleh aktor.

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- Guru telah *login* dalam sistem Ruangguru.com
- Guru telah melakukan *use case retrieve* kelas

3.3 Post-Conditions

Sistem menyimpan materi baru

3.4 Extension Points

N/A

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

Use-Case-Realization Specification: *Delete materi*

1. Introduction

Use Case ini menjelaskan skenario tentang bagaimana guru menghapus atau mendelete materi, beserta *sequence diagram*, *class view* dan *class controller*.

1.2 Author

Vasko Dagama

1.3 Actor

Guru

2. Flow of Events – Design

2.1 Brief Description

Use case ini menjelaskan bagaimana skenario guru menghapus materi pada suatu kelas.

2.2 Use Case Scenario

2.2.1 Basic Flow

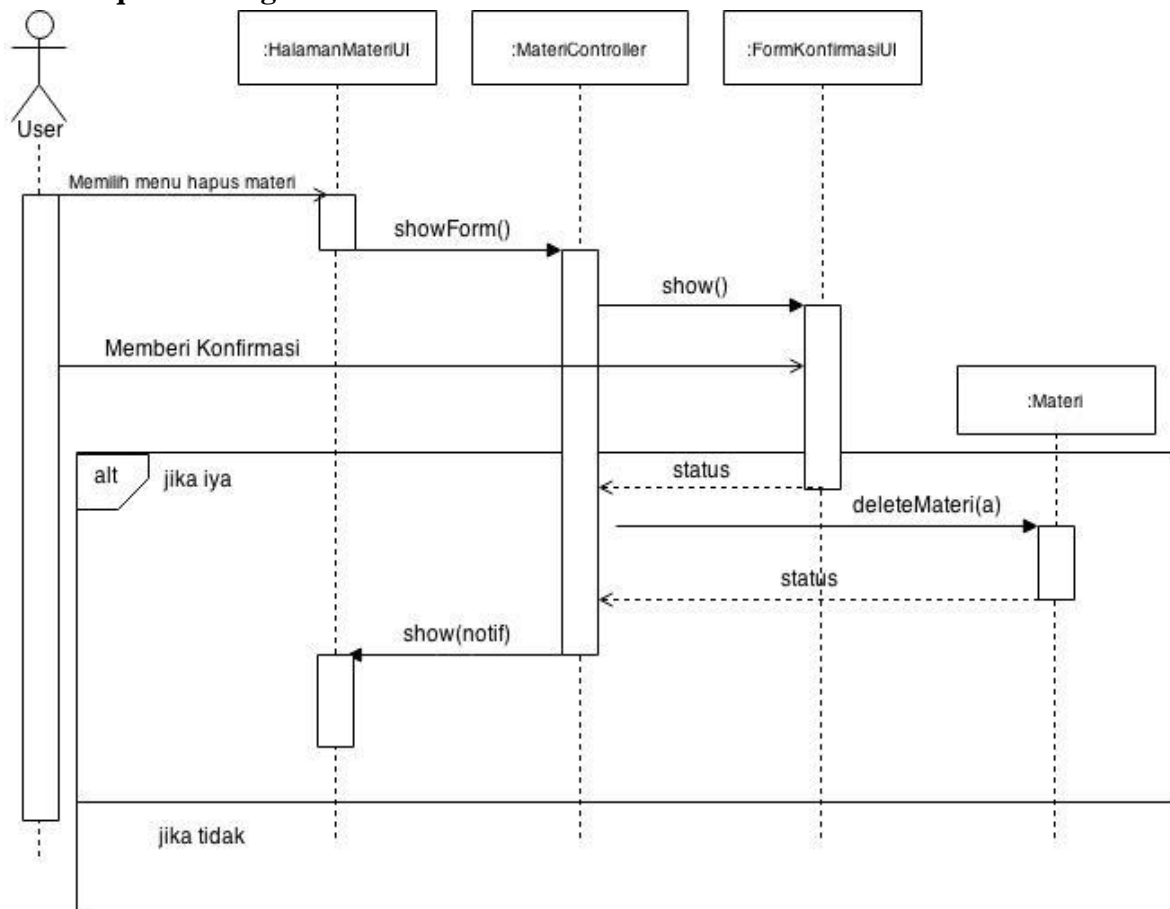
Actor Actions	System Responses
1. Memilih menu hapus pada salah satu materi.	
	2. Sistem menampilkan pertanyaan konfirmasi
3. Memberi konfirmasi	
	4. Sistem menghapus materi tersebut
	5. Sistem menampilkan pesan bahwa materi telah dihapus

2.2.2 Alternative Flows

Pada langkah 3, apabila aktor membatalkan konfirmasi maka materi tidak akan terhapus. Aktor bisa kembali ke langkah 1 ataupun berhenti di langkah 3 tersebut.

SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.3 Sequence Diagram

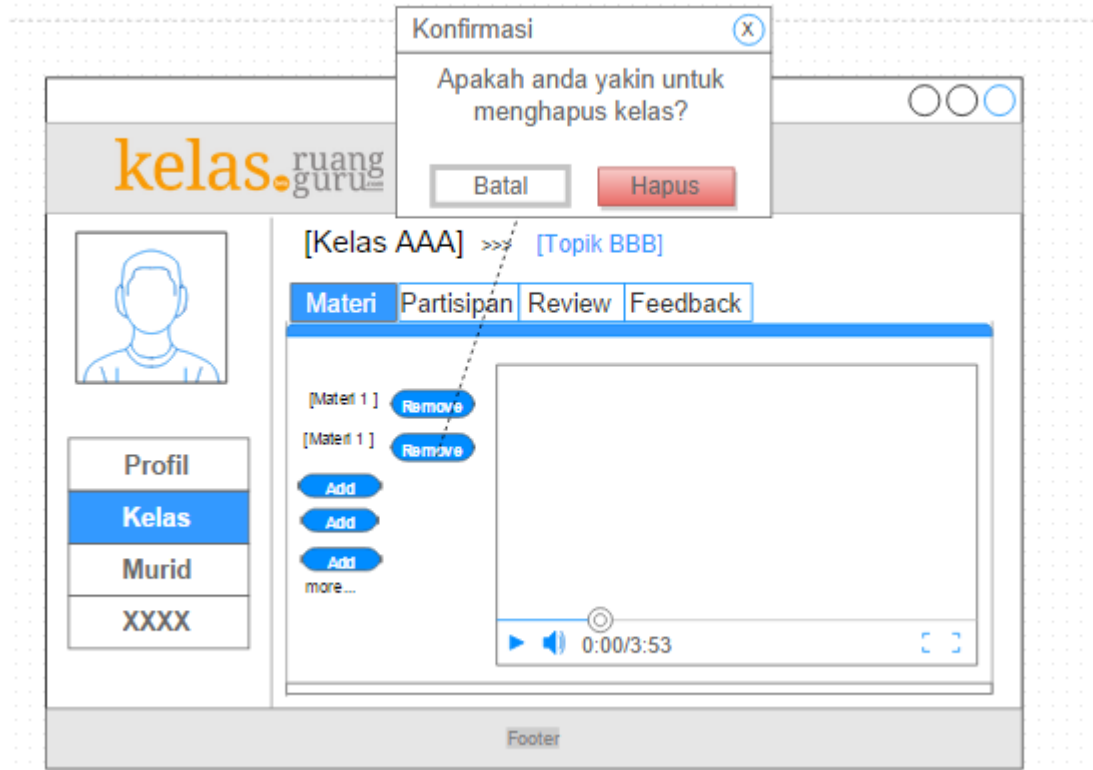


SISTEM INFORMASI KELAS ONLINE RUANGGURU	Version: 1.1
Software Architecture Document	Issue Date: 28/03/2015

2.4 Explanation of View Class and Controller Class

View Class :

- HalamanMateriUI : Merupakan halaman materi suatu kelas pada *dashboard* guru.



- FormKonfirmasiUI : Merupakan jendela/halaman untuk melakukan konfirmasi penghapusan.

Controller Class :

- MateriController, yang memiliki *method* :
 - showForm() : Berfungsi untuk menampilkan form konfirmasi penghapusan.

3. Derived Requirements

3.1 Special Requirements

N/A

3.2 Pre-Conditions

- Guru telah *login* dalam sistem Ruangguru.com
- Guru telah melakukan *use case retrieve* kelas

3.3 Post-Conditions

Sistem menghapus materi

3.4 Relationship

N/A