

## Pengembangan Sistem Manajemen Seminar Interaktif Berbasis Web (Studi Kasus: Universitas Brawijaya Televisi)

Ananda Putra Alfa Robby<sup>1</sup>, Achmad Arwan<sup>2</sup>, Djoko Pramono<sup>3</sup>

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>ananda\_alfa@student.ub.ac.id, <sup>2</sup>arwan@ub.ac.id, <sup>3</sup>djoko.jalin@ub.ac.id

### Abstrak

Penyelenggaraan kegiatan akademik pada Universitas Brawijaya televisi memiliki banyak agenda untuk mengadakan seminar dan sosialisasi tentang proses penyelesaian pekerjaan maupun proses penyampaian ilmu-ilmu terkait. Banyaknya Jadwal masuk penyelenggaraan seminar pada UBTV membuat pimpinan UBTV tidak bisa melihat sejauh mana perkembangan pengajuan seminar yang diproses. Kemudian pada pelaksanaan seminar, banyaknya peserta atau pembicara yang memiliki gelar dan jabatan tinggi membuat para peserta seminar malu untuk bertanya bilamana tidak memiliki tingkat percaya diri yang tinggi. Kurangnya waktu dalam sesi tanya jawab kepada pemateri menjadi faktor untuk tidak menjawab semua pertanyaan yang diajukan para peserta. Hal tersebut berefek kepada pengetahuan para peserta tersebut bila tidak mengetahui jawaban-jawaban dari pertanyaan yang ingin ditanyakan saat acara seminar maupun sosialisasi yang berlangsung. Berdasarkan permasalahan tersebut, dalam penelitian skripsi ini dibuat sistem manajemen seminar interaktif berbasis web. Diharapkan dengan adanya sistem tersebut peserta dapat bertanya sesuai dengan ketidaktahuan dari peserta tersebut tanpa ada rasa malu dalam bertanya dan dapat mengefisienkan waktu pelaksanaan seminar maupun sosialisasi di instansi ini. Sistem ini dirancang dengan metode *Object-Oriented* yang diimplementasikan dengan framework laravel berbahasa pemrograman PHP serta database MySQL. Dalam pengujian akan dilakukan dengan pendekatan kotak hitam (*black box*) dan kotak putih (*white box*) untuk mendapatkan hasil uji yang valid sehingga dapat disimpulkan bahwa sistem bekerja sesuai dengan kebutuhan dan menyelesaikan permasalahan pengguna.

**Kata kunci:** Pertanyaan, *Object-Oriented*, *Laravel*, *Web-Base*

### Abstract

*In organizing academic activities, University of Brawijaya Television has many agendas to hold seminars and socialization about the process of completion of work and the process of delivery of related sciences. The number of seminar schedules at UBTV makes its leaders unable to see how far the progress of seminar submissions is processed. Then in the implementation of the seminar, the number of participants or speakers who have high degrees and positions makes the seminar participants ashamed to ask if they do not have a high level of confidence. The lack of time in the question and answer session became a problem for participants to ask questions, so there were participants that were still wondering since their questions are not answered yet. This has an effect on the knowledge of the participants if they do not know the answers to the questions that want to be asked during the seminar and socialization. Based on these problems, this study proposed an interactive seminar management system based on the web that is expected, with the system, participants can ask anything they don't understand about the seminar without any embarrassment. Besides, they also can manage the time of the seminars and socialization in this institution. This system is designed along with Object-Oriented method implemented with Laravel framework in PHP programming language as well as MySQL database. The tryout is done with the approach of black box and white box to get a valid results so that it can be concluded that the system works according to the needs and solve user problems.*

**Keywords:** Seminar, Question, *Object-Oriented*, *Laravel*, *Web-Base*

### 1. PENDAHULUAN

Universitas Brawijaya Televisi (UBTV) adalah sebuah Unit kerja dibawah Badan Usaha Akademik yang memiliki target pendapatan pada lembaga pendidikan Universitas Brawijaya. Berdasarkan SK Menteri PTIP Nomor: 1/1963 dan Kepres RI Nomor: 196/1963 UBTV sebagai televisi komunitas kampus yang memiliki tugas utama yaitu menyampaikan informasi-informasi yang aktual pada masyarakat. UBTV juga bertugas sebagai penyedia layanan dalam pelaksanaan sebuah pertemuan berskala besar seperti seminar, kunjungan studi, dan sebuah event lainnya seperti kuliah tamu yang diselenggarakan oleh badan internal Universitas Brawijaya atau eksternal Universitas Brawijaya. Pada penelitian ini penulis akan berfokus kepada proses persiapan dan pelaksanaan pertemuan seminar yang ada di UBTV

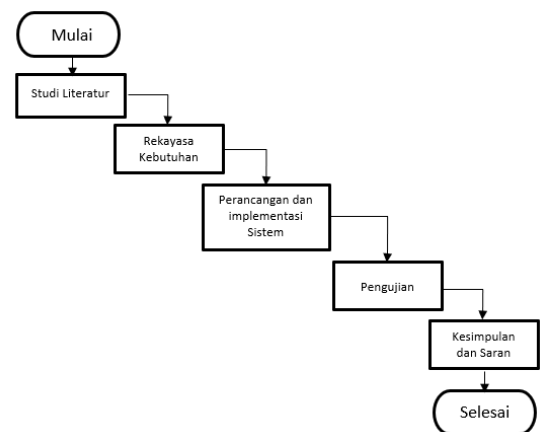
Dalam pelaksanaan seminar, kunjungan studi, dan kuliah tamu di UBTV, jumlah peserta digolongkan dalam berbagai paket berdasarkan jumlah yaitu paket small berjumlah 100 peserta, paket large berjumlah 150 peserta, dan extra large yaitu berjumlah 200 peserta. UBTV dapat membantu atau menyelenggarakan sebuah seminar dan kunjungan studi dengan rata-rata dalam 1 bulan terdapat 1 atau 2 pertemuan. Seminar dapat diselenggarakan apabila terdapat permintaan dari instansi pendidikan maupun permintaan dari produser penyelenggara program (Shelly, 2020).

Dalam sebuah seminar dan kunjungan studi yang diselenggarakan atau dibantu oleh UBTV, banyak hal-hal yang harus dipersiapkan seperti penentuan tempat, menyampaikan pendaftaran dan undangan seminar untuk peserta seminar serta kunjungan studi. Pada seminar tentu memiliki seorang moderator yang memimpin jalannya seminar. Moderator harus berkompeten dengan tema dan kegiatan seminar sehingga pada proses pelaksanaannya seminar dapat berjalan sesuai susunan acara yang sudah dijadwalkan dari sesi penyampaian materi hingga sesi tanya jawab. Pada sesi tanya jawab, dimungkinkan terdapat peserta yang ingin bertanya namun memiliki rasa malu untuk bertanya terhadap hal yang tidak dipahami. Dalam penjelasan (Hidayati, 2016) shyness dapat dijelaskan sebagai rasa cemas atau gugup yang terjadi pada seseorang saat berada di situasi yang baru atau berada di suatu kondisi dimana

seseorang tersebut bertemu orang lain yang tidak dikenal. Timbulnya rasa malu tersebut menjadikan peserta yang sungkan atau enggan memberikan pertanyaan untuk menjawab keraguan dari dirinya dalam bertanya atau menjawab sebuah kuis yang diberikan secara lisan dari pemateri.

Berdasarkan permasalahan yang telah terjadi, peneliti membuat sistem manajemen seminar interaktif berbasis web. Sistem berbasis web dipilih untuk memajemen proses persiapan hingga pelaksanaan pada seminar dan dapat digunakan oleh pegawai, peserta ataupun pemateri yang hadir dalam seminar tersebut untuk menyampaikan pertanyaan atau memberikan sebuah kuis melalui gawai dengan browser tanpa harus mengunduh dan memuat aplikasi terlebih dahulu.

### 2. METODOLOGI



Gambar 1. Metodologi Penelitian

Dalam melaksanakan penelitian ini, penulis mengadopsi *system life development waterfall*, terdapat berbagai langkah-langkah yang telah ditetapkan di alur penelitian ini.

#### 2.1 Studi Literatur

Dasar-dasar dalam pembuatan sistem pada UBTV yang terdiri dari semua pustaka berhubungan dengan:

1. Seminar
2. Rekayasa Perangkat lunak
3. Bahasa pemrograman PHP
4. Platform Web framework Laravel
5. Database (MySQL)
6. Unified Modelling Language (UML)
7. Pengujian

#### 2.2 Rekayasa Kebutuhan

Rekayasa kebutuhan dilakukan peneliti untuk menggali dan menganalisis serta dapat

memodelkan kebutuhan-kebutuhan yang akan diimplementasikan kedalam sistem. Rekayasa kebutuhan yang dilakukan memiliki berbagai tahap yaitu elisitasi kebutuhan sistem, analisis kebutuhan, dan pemodelan kebutuhan.

**2.3 Pemodelan Kebutuhan**

Pada tahapan pemodelan kebutuhan, hasil dari analisis kebutuhan akan dimodelkan dengan *object oriented*. Tahapan ini dilakukan untuk menjadi sebuah dasar dalam pengembangan yang dilakukan oleh peneliti untuk diimplementasikan kepada sistem manajemen seminar interaktif berbasis web. Pada pemodelan kebutuhan akan menghasilkan diantaranya *use case diagram* dan *use case scenario*.

**2.4 Perancangan Sistem dan Implementasi**

Setelah semua kebutuhan untuk sistem manajemen seminar interaktif berbasis web didapatkan melalui tahap rekayasa kebutuhan, tahap selanjutnya adalah melakukan perancangan sistem dan implementasi pada sistem manajemen seminar interaktif dengan berorientasi objek yang terdiri dari beberapa tahapan perancangan yaitu *sequence diagram*, *class diagram*, perancangan basis data, perancangan komponen, hingga perancangan *user interface*.

Pada tahap implementasi atau coding, penulis mengubah hasil perancangan-perancangan ke dalam kode program untuk membuat perangkat lunak yang diinginkan. Implementasi pada penelitian ini menggunakan bantuan kerangka kerja laravel untuk membangun web.

**2.5 Pengujian**

Bentuk pengujian menggunakan metode *blackbox testing* dan *whitebox testing* dengan menguji beberapa fungsionalitas utama dengan fungsi dari *user* dan *admin*. Bilamana dalam pengujian ditemukan sebuah kesalahan maupun terdapat kebutuhan yang tidak berjalan dengan sesuai, maka akan dilakukan perbaikan dari kesalahan tersebut. Pengujian yang akan dilakukan yaitu pengujian unit, pengujian integrasi, pengujian validasi.

**2.6 Kesimpulan dan Saran**

Pengambilan kesimpulan dilakukan setelah semua tahapan perancangan, dan implementasi sistem aplikasi telah selesai dilakukan. Kesimpulan diambil dari analisis terhadap

sistem yang dibangun. Tahap terakhir dari penulisan adalah saran yang dimaksudkan untuk memperbaiki kesalahan-kesalahan yang terjadi dan menyempurnakan penulisan serta untuk memberikan pertimbangan ataupun saran atas pengembangan aplikasi selanjutnya.

**3. HASIL DAN PEMBAHASAN**

Pada tahap ini, penulis akan membahas mengenai tahapan-tahapan di awal pada proses suatu pengembangan perangkat lunak.

**3.1. Rekayasa Kebutuhan**

Elisitasi kebutuhan dilakukan dengan teknik wawancara kepada kepala bidang digital branding UBTV yaitu Ibu Amalia Permata Risky, S.I.Kom, M.I.Kom, Bapak Anton selaku kepala sarana dan prasarana, Bapak Adi Satriya, A.Md sebagai kepala produser, Bapak Goldi Sena, A.Md, selaku produser, dan 1 orang sebagai moderator yaitu Ibu Dinia Sari.

Hasil dari elisitasi kebutuhan mendapatkan sebuah aktor-aktor yang akan berperan kedalam sistem pada tabel 1 berikut.

Tabel 1. Aktor yang berperan

No	Kategori pengguna
1	Guest
2	Pengaju
3	Kepala Digital Branding
4	Kepala Sarana dan Prasarana
5	Kepala Produksi
6	Kepala Produser
7	Produser
8	Moderator
9	Pemateri
10	Peserta

**3.1.1. Identifikasi Kebutuhan**

**3.1.1.1. Kebutuhan Fungsionalitas**

Kebutuhan fungsional merupakan kebutuhan yang dapat dikatakan sebagai kebutuhan utama dalam sistem ini karena menggambarkan fungsi-fungsi apa saja yang tersedia dalam sistem yang akan dibangun ini.

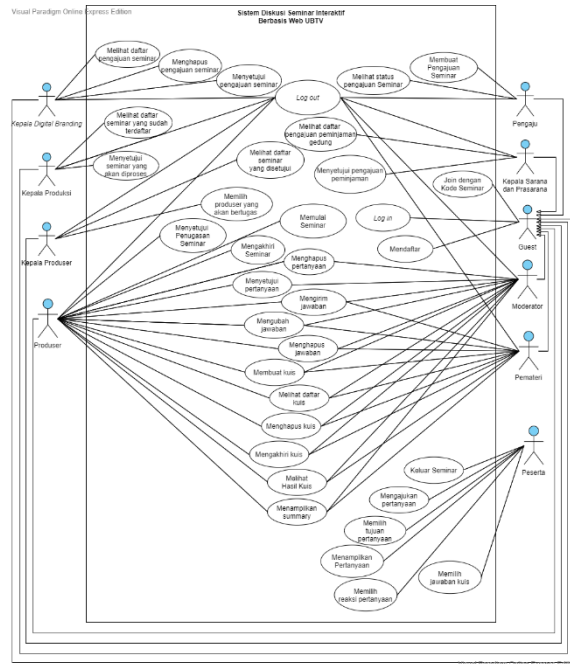
**3.1.1.2. Kebutuhan Non-fungsionalitas**

Pada sistem ini memiliki kebutuhan non-fungsionalitas yang mendukung dalam kinerja dari berjalannya sistem ini. Penulis melakukan satu pengujian non-fungsionalitas yaitu *Compatibility*.

**3.2. Pemodelan Kebutuhan**

3.2.1. Use Case Diagram

Berdasarkan kebutuhan fungsional yang sudah didapatkan, maka selanjutnya dibuat use case diagram dengan aktor yang sesuai pada tabel 1. Pada gambar 2 berikut dijelaskan use case diagram pada sistem manajemen seminar interaktif.



Gambar 2. Use Case Diagram

3.2.2. Use Case Scenario

Use Case Scenario merupakan sebuah langkah-langkah yang dilakukan aktor dalam sebuah sistem berdasarkan use case yang ada. Pada use case scenario dijelaskan perilaku sistem disetiap urutan perilaku yang dijalankan oleh aktor tersebut, pada use case scenario terdapat pre-conditions yaitu syarat kondisi sebelum alur dijalankan, kemudian main flow yaitu alur utama perilaku sistem yang dijalankan, lalu terdapat alternative flows yaitu sebuah alternatif alur dan kondisi bilamana salah satu alur main flow tidak terpenuhi. Berikut dijelaskan pada tabel 2.

Tabel 2. Use Case Scenario

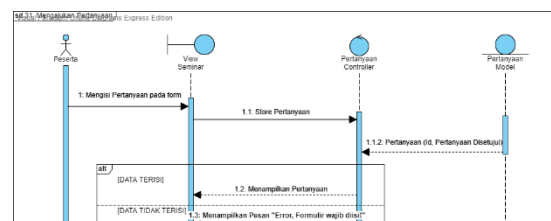
Use Case Code	SNRUB_F-030
Use Case Name	Mengajukan Pertanyaan
Actor	Peserta
Description	Use case ini menggambarkan proses untuk mengajukan pertanyaan pada sistem.
Pre-Condition	Aktor telah masuk ke seminar dengan kode seminar
Main Flow	1. Aktor mengisi pertanyaan dan tujuan

	<p>pertanyaan pada kolom pertanyaan</p> <p>2. Sistem mengecek kelengkapan form</p> <p>3. Sistem berhasil menyimpan pertanyaan pada basis data</p>
Alternative flows	<p>2a. Data Tidak Terisi</p> <p>Jika data yang dimasukkan pada formulir tidak terisi, sistem akan menampilkan pesan error dan peserta harus mengisi formulir.</p>
Post-Condition	<p>1. Sistem menampilkan pertanyaan yang dikirim</p> <p>2. Peserta berhasil mengirim pertanyaan</p>

3.3. Perancangan Sistem

3.3.1. Sequence Diagram

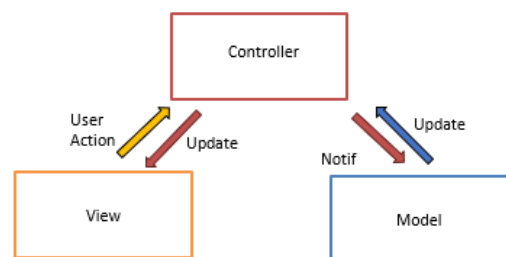
Pada sequence diagram ini dijelaskan alur-alur sistem untuk peserta dalam mengajukan pertanyaan di sistem yang akan penulis buat. Peserta akan mengirimkan pertanyaan sesuai dengan tujuan yang mereka pilih ataupun tanpa tujuan. Berikut sequence diagram pada gambar 3.



Gambar 3 Sequence Diagram Mengajukan Pertanyaan

3.3.2. Class Diagram

Class diagram merupakan sebuah kumpulan objek, method serta attribute yang diimplementasikan dan saling berinteraksi ke dalam sistem untuk mencapai sebuah tujuan tertentu. Pada class diagram ini telah mengimplementasikan pola perancangan Model-View-Controller (MVC). Pada gambar 4 dijelaskan interaksi antar kelas dengan pola perancangan MVC.



Gambar 4. Interaksi Class Model Controller View

### 3.3.3. Perancangan Komponen

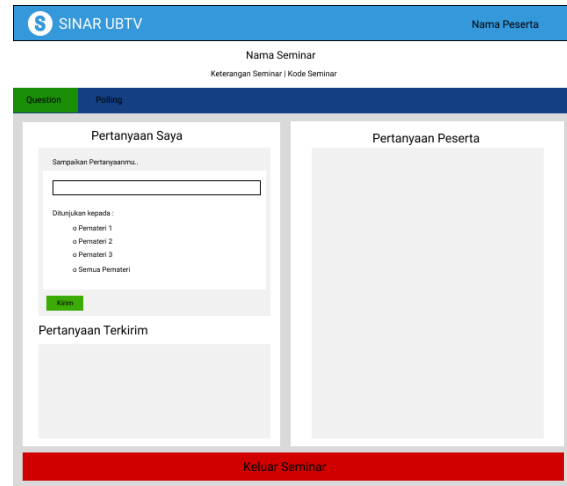
Pada tahap perancangan komponen, penulis menjelaskan rancangan algoritme terhadap beberapa fungsi *class*. Perancangan komponen diambil satu sampel uji untuk pembuatan rancangan algoritme. Sampel uji dipilih berdasarkan pada fungsi yang telah dirancang pada kebutuhan fungsionalitas. Berikut gambar 4 tentang algoritme mengajukan pertanyaan

Tabel 3. Algoritme Method Kirim Pertanyaan

Nama algoritma kirim_pertanyaan
<pre> Mulai Ambil data dari Form Variable user = ambil nama user dari sesion Pemateri = Ambil data dari Form If tanya tidak sama dengan 'tidak ada' Variabel Question=ambil data'nama_user' if (pemateri sama dengan "universal")     buat data     'Nama_User' = user     'Id_Seminar' = Id_Seminar     'Pertanyaan' = pertanyaan     'Pemateri' = Universal else     buat data     'Nama_User' = user     'Id_Seminar' = Id_Seminar     'Pertanyaan' = pertanyaan     'Pemateri' = Pemateri else     Kembali ke halaman Selesai                 </pre>

### 3.3.4. Perancangan Antarmuka

Perancangan antarmuka menggambarkan terkait rancangan dari sistem manajemen seminar berbasis web pada UBTV. Antarmuka dibuat dengan warna yang akan diimplementasikan oleh penulis kedalam sistem. Perancangan User Interface dilakukan oleh penulis dengan menggunakan referensi pada sistem yang telah ada sebelumnya seperti Sli.do. Pada referensinya, penulis hanya mencari warna-warna dan tata letak kebutuhan pada user yang sesuai dan terlihat nyaman untuk dilihat. Warna-warna yang soft dipilih penulis dikarenakan warna soft sering digunakan pada media-media pembelajaran. Berikut gambar 5 tentang perancangan antarmuka mengirim pertanyaan.



Gambar 5. Perancangan Antarmuka Mengirim Pertanyaan

### 3.4. Implementasi Sistem

Pada tahap implementasi, penulis menyampaikan detail tentang implementasi sistem Sinar-UBTV dengan *framework Laravel*. Berikut tabel 4 tentang source code program mengajukan pertanyaan.

#### 3.4.1. Implementasi Kode Program

Implementasi dilakukan dengan menggunakan *framework Laravel* dengan menggunakan Netbeans IDE 8.2 RC sebagai teks editor. Berikut salah satu *Source Code* yang diimplementasikan kedalam sistem.

Tabel 4. Source Code Mengajukan Pertanyaan

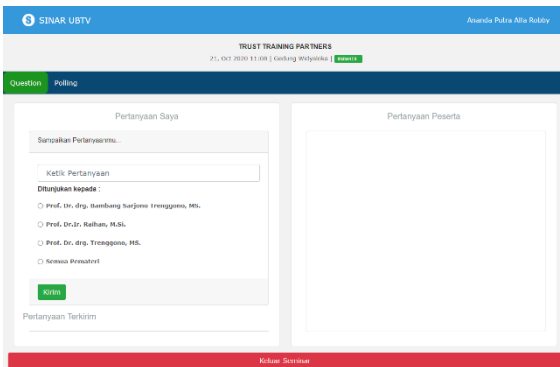
<pre> Nama Method kirim pertanyaan public function store(Request \$request) {     \$form = \$request-&gt;ask;     \$pemateri = \$request-&gt;pemateri;     if(\$pemateri == 'all'){         Pertanyaan::create([             'idPeserta'=&gt;\$request- &gt;session()-&gt;get('idPeserta'),             'idSeminar'=&gt;\$request- &gt;session()-&gt;get('idSeminar'),             'idPemateri'=&gt; 0 ,             'pertanyaan'=&gt; \$form,             'like'=&gt; 0,             'dislike'=&gt; 0,             'status'=&gt; 0,         ]);         Session::flash('pertanyaan', 'Pertanyaan Terkirim!');     }else{         Pertanyaan::create([             'idPeserta'=&gt;\$request- &gt;session()-&gt;get('idPeserta'),             'idSeminar'=&gt;\$request- &gt;session()-&gt;get('idSeminar'),             'idPemateri'=&gt;             \$pemateri ,             'pertanyaan'=&gt; \$form,                 </pre>
--

```

        'like'=> 0,
        'dislike'=> 0,
        'status'=> 0,
    });
    Session::flash('pertanyaan','Pertanyaan
    Terkirim!');
    }
    return redirect()->back();
}
    
```

**3.4.2. Implementasi Antarmuka**

Pada sesi peserta, mengajukan pertanyaan adalah kunci utama dari penyelesaian masalah ini. Peserta dapat mengajukan pertanyaan dengan antarmuka yang sudah dirancang sebelumnya dengan kesesuaian form saat user memilih bagian pertanyaan. Form akan tertampil dengan ukuran layar yang sesuai dengan gawai para user sendiri. Berikut gambar 6 yang menampilkan antarmuka mengajukan pertanyaan pada sistem manajemen seminar interaktif berbasis web.



Gambar 6. Antarmuka Mengajukan Pertanyaan

**3.5. Pengujian Sistem**

Pengujian sistem dilakukan untuk memastikan bahwa sistem telah berjalan dengan semestinya dan sesuai dengan apa yang telah didefinisikan pada tahap analisis kebutuhan pada sistem manajemen seminar interaktif berbasis web pada UBTV ini.

**3.5.1. Pengujian Unit**

Pengujian sistem dilakukan untuk memastikan bahwa sistem telah berjalan dengan semestinya dan sesuai dengan apa yang telah didefinisikan pada tahap analisis kebutuhan pada sistem manajemen seminar interaktif berbasis web pada UBTV. Dalam pengujian ini dilakukan pengecekan terhadap algoritme yang telah dirancang pada sistem dengan menggunakan basis path testing. Pada pengujian ini dilakukan terhadap fungsionalitas utama yaitu mengajukan pertanyaan.

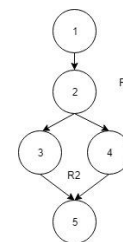
Pelaksanaan pengujian ini dilakukan pada

tahap peserta yang sudah bergabung pada Sinar-UBTV. Mengajukan pertanyaan diproses dalam controller *PertanyaanController* pada *method store()*. Algoritme pada *method store()* dijelaskan pada tabel 5 merupakan algoritme dari proses mengajukan pertanyaan untuk mengetahui peserta yang menyampaikan pertanyaan pada sistem.

Tabel 5 Pseudocode Method Store() Mengajukan Pertanyaan

Pseudocode	
1	Mulai → 1
2	Inisiasi Data Pertanyaan, pemateri
3	if(pemateri = 'all')
4	Insert Pertanyaan dengan } 2
5	tujuan semua pemateri → 3
6	else
7	Insert Pertanyaan dengan
8	pilihan pemateri → 4
	endif
	Selesai → 5

Berdasarkan pseudocode pada tabel 5 diatas, maka didapatkan bentuk flow graph berikut:



Gambar 7. Flow Graph pada Method Store() Mengajukan Pertanyaan

Berdasarkan flow graph pada gambar 7 tersebut, selanjutnya dapat ditentukan jumlah jalur independen dan perhitungan cyclomatic complexity pada persamaan 1, 2, dan 3.

Jalur 1 : 1-2-3-5

Jalur 2 : 1-2-4-5

Perhitungan cyclomatic complexity pada persamaan 1, 2, dan 3

$$\begin{aligned}
 V(G) &= E - N + 2 & (1) \\
 &= 5 - 5 + 2 \\
 &= 2
 \end{aligned}$$

$$\begin{aligned}
 V(G) &= P + 1 & (2) \\
 &= 1 + 1 \\
 &= 2
 \end{aligned}$$

$$\begin{aligned}
 V(G) &= R & (3) \\
 &= 2
 \end{aligned}$$

Berdasarkan jumlah independent path diatas, maka diperoleh kasus uji yang dijelaskan pada tabel sebagai berikut

Tabel 6. Hasil Uji Mengajukan Pertanyaan

No mor Jalur	Data Input	Hasil yang diharapkan	Hasil yang diperoleh	Status
1	Data dimasukan Pertanyaan = "Apakah Perlu Blackmagic Design untuk Broadcast multi cam?", Pemateri = "All"	Sistem tetap di halaman <i>question</i> dan menampilkan pesan sukses "Pertanyaan Terkirim!" dengan keterangan "Semua Pemateri"	Sistem menampilkan pesan sukses "Pertanyaan Terkirim!"	Valid
2	Data dimasukan Pertanyaan = "Apakah Perlu Blackmagic Design untuk Broadcast multi cam?", Pemateri = "Dr. Riyanto, M.Hum"	Sistem tetap di halaman <i>question</i> dan menampilkan pesan sukses "Pertanyaan Terkirim!" dengan keterangan pemateri "Dr. Riyanto, M.Hum"	Sistem menampilkan pesan sukses "Pertanyaan Terkirim!" dengan keterangan pemateri "Dr. Riyanto, M.Hum"	Valid

**3.5.2. Pengujian Integrasi**

Pengujian integrasi merupakan pengujian yang menggunakan metode *basis path testing* dimana bertujuan untuk menguji interaksi antar *class*. Pengujian ini dilakukan dengan pendekatan *top-down* dengan pengujian relasi antar *class* pada *high level-*

*level design*. Method yang diujikan yaitu *method seminar()* pada *class SeminarController*. Pada *method seminar()* akan memanggil *method getSeminar()* pada *class Seminar*, *method insertPeserta()*, *method getPeserta()* dan *method getPertanyaan()*.

**3.5.3. Pengujian Validasi**

Pengujian blackbox atau validasi ini dilakukan untuk menguji fungsionalitas dari sistem apakah sudah sesuai dengan kebutuhan yang terdefinisi pada perancangan serta sistem dapat berjalan dengan baik atau tidak. Penulis membuat sebuah tabel berisi *test name*, *test case*, *expected result*, *result* dan status. Pengujian dilakukan di sisi user dan admin sesuai dengan kebutuhan yang diterapkan. Bilamana sistem sudah sesuai maka akan menghasilkan status valid, bilamana sebaliknya maka akan menghasilkan tidak valid dan akan dilakukan perbaikan pada sistem dan diuji ulang

**3.5.4. Pengujian Compatibility**

Hasil Pengujian *compatibility* pada sistem manajemen seminar interaktif berbasis web di Universitas Brawijaya Televisi memiliki hasil yang sesuai dan sistem dapat dijalankan pada *browser google chrome* dan *firefox*.

**3.6. Kesimpulan dan Saran**

**3.6.1. Kesimpulan**

1. Sistem manajemen interaktif berbasis web terbagi menjadi 2 tahap, yaitu proses pengajuan dan proses pelaksanaan seminar. Dalam proses pengajuan, sistem difungsikan untuk pimpinan UBTV yaitu kepala digital branding, kepala sarana dan prasarana, kepala produksi, kepala produser, dan produser. Pada tahap pelaksanaan seminar, sistem dioperasikan oleh produser dan moderator untuk memulai dan menyeleksi pertanyaan-pertanyaan yang masuk dari peserta. Dari kedua tahapan tersebut, penulis telah melakukan pemodelan kebutuhan yang menghasilkan sejumlah 35 kebutuhan fungsionalitas dan 1 kebutuhan non-fungsionalitas.
2. Di tahap perancangan penulis menggunakan *object oriented design*. Proses perancangan menghasilkan *sequence diagram*, *class diagram*, dan antarmuka perancangan. Kemudian di tahap implementasi, penulis

mengimplementasikan semua kebutuhan fungsional dan rancangan-rancangan dengan menerapkan *object oriented*.

3. Pada tahap pengujian, penulis sistem menggunakan pengujian unit dengan pendekatan *basis path testing*, pengujian integrasi, pengujian validasi dan pengujian non-fungsionalitas menggunakan pengujian *compatibility*. Hasil yang didapatkan dari pengujian *white box* menghasilkan 100% valid dari 11 kasus uji. Lalu, pengujian integrasi yang dilakukan penulis menghasilkan 100% valid dalam integrasi antar *method* dalam sistem serta Pengujian validasi yang menghasilkan 100% valid dalam *browser chrome* dan *firefox*

### 3.6.2. Saran

Berdasarkan kesimpulan yang dibuat penulis diatas, maka penulis menyarankan untuk pengembangan dan penelitian selanjutnya dalam bidang manajemen seminar interaktif adalah:

1. Perlu perbaikan kembali pada fitur melihat hasil kuis untuk pendalaman *auto update field*, sehingga pengguna dapat melihat hasil dengan keadaan terpisah dari proses berjalannya kuis.
2. Penambahan fungsi *summary* yang lebih detail atau pencetakan laporan hasil seminar agar pengguna dapat melihat hasil seminar dalam dokumen.
3. Integrasi proses *login* dengan akun lainnya untuk memudahkan pengguna dalam proses autentikasi kedalam sistem.
4. Perbaikan perancangan *user interface* yang lebih menarik sehingga pengguna semakin nyaman dalam menggunakan sistem ini.

## 4. DAFTAR PUSTAKA

- Aminudin. (2015). Framework Laravel. *Ekp*, 13(3), 1576–1580.
- Derviş, B. (2013). JURNAL PSIKOLOGI Teori & Terapan. *Journal of Chemical Information and Modeling*, 53(9), 1689–1699.  
<https://doi.org/10.1017/CBO9781107415324.004>
- DeVolder, D., Ghazanshahi, S., & Zadeh, J. (2008). Software testing and quality assurance. In *WMSCI 2008 - The 12th World Multi-Conference on Systemics, Cybernetics and Informatics, Jointly with the 14th International Conference on Information Systems Analysis and Synthesis, ISAS 2008 - Proc.* (Vol. 1).  
<https://doi.org/10.1002/9780470382844>
- Gutiérrez, J. J., Escalona, M. J., Mejías, M., & Torres, J. (2006). An approach to generate test cases from use cases. *ICWE'06: The Sixth International Conference on Web Engineering, January*, 113–114.  
<https://doi.org/10.1145/1145581.1145606>
- Gutiérrez, J. J., Escalona, M. J., Mejías, M., Torres, J., Jia, K., Pang, X., Li, Z., Craig, D. C., Zuberek, W. M., Singh Ghuman, S., Shklar, L., Rosen, R., Robert K. Wysocki, DeVolder, D., Ghazanshahi, S., Zadeh, J., Kurniawan, T. A., Ladjamuddin B, A.-B., Henderi, ... Suryana, T. (2006). JURNAL PSIKOLOGI Teori & Terapan. *ICWE'06: The Sixth International Conference on Web Engineering, 1*(January 2005), 11–18.  
[https://doi.org/10.1061/\(ASCE\)IS.1943-555X.0000027](https://doi.org/10.1061/(ASCE)IS.1943-555X.0000027)
- Henderi. (2009). Object Oriented Modelling With Unified Modeling Language (Uml). 5 November 2009, June, 77.  
<https://doi.org/10.13140/RG.2.1.3464.4088>
- Hidayati, D. S. (2016). *Shyness dan Loneliness*. 19–20.
- Kurniawan, T. A. (2007). Pengujian Struktur Program Dengan Pengujian Jalur Dasar ( Basis Path Testing ) : Teori Dan Aplikasi. *Eccis*, 1(1), 29–32.  
<http://jurnaleccis.ub.ac.id/index.php/eccis/article/viewFile/357/266>
- Ladjamuddin B, A.-B. (2006). *Rekayasa Perangkat Lunak. 1*.
- Robert K. Wysocki. (2010). *Effective software project management* (1st editio). Wiley; 1 edition (September 29, 2010).
- Roger S. Pressman, P. D., & Bruce R. Maxim, P. D. (2015). *Software engineering : a practitioner's approach* (EIGHTH EDI). McGraw-Hill Education.
- Shklar, L., & Rosen, R. (2003a). Web Application Architecture: Principles, Protocols and Practices. In *Software Engineering*.  
<http://books.google.com/books?id=J0IZ3ZLzy2YC&pgis=1%5Cnftp://79.110.128.93/books/programming/Java/Web>



- Application Architecture - ISBN  
0471486566.pdf
- Shklar, L., & Rosen, R. (2003b). Web Application Architecture: Principles, Protocols and Practices. In *Software Engineering*.
- Solichin S.Kom, A. (2016). Pemrograman Web dengan PHP dan MySQL - Achmad Solichin - Google Buku. *Universitas Budi Luhur, January 2005*, 215.
- [https://doi.org/10.1061/\(ASCE\)IS.1943-555X.0000027](https://doi.org/10.1061/(ASCE)IS.1943-555X.0000027)
- Sommerville, I. (2016). Software engineering (10th edition). In *Pearson Education Limited*.
- Suryana, T. (2017). *Aplikasi Internet Menggunakan HTML , CSS , dan JavaScript ( Indonesian. April*.