# Proposal Tugas Besar

Aplikasi Event Organizer



Anggota :

Muhammad Afif – 1103120019

Taufik Akbar – 1103120020

Xxxx

Xxxx

Program Studi Sarjana Teknik Informatika

Fakultas Informatika

Universitas Telkom

Bandung

2015

Daftar Isi

[Proposal Tugas Besar I](#_Toc437467017)

[**1.** **Pendahuluan** 1](#_Toc437467018)

[**1.1** **Latar Belakang** 1](#_Toc437467019)

[**1.2** **Rumusan Masalah** 1](#_Toc437467020)

[**1.3** **Tujuan** 1](#_Toc437467021)

[**1.4** **Batasan Masalah** 2](#_Toc437467022)

[**1.5** **Metode Pengerjaan** 2](#_Toc437467023)

[**2.** **Analisis Kebutuhan Sistem** 4](#_Toc437467024)

[**2.1** **Use Case Model** 4](#_Toc437467025)

[**2.1.1 Use Case Diagram** 4](#_Toc437467026)

[**2.1.2 Skenario Use Case** 4](#_Toc437467027)

[**2.2 Class Diagram** 8](#_Toc437467028)

[**2.3 Sequence Diagram** 9](#_Toc437467029)

[**2.3.1 Login Admin** 9](#_Toc437467030)

[**2.3.2 Kelola User** 10](#_Toc437467031)

[**2.3.3 Verifikasi** 11](#_Toc437467032)

[**2.4 Kebutuhan Perangkat Keras dan Perangkat Lunak** 11](#_Toc437467033)

[**2.4.1** **Kebutuhan Perangkat keras** 11](#_Toc437467034)

[**2.4.2** **Kebutuhan Perangkat Lunak** 11](#_Toc437467035)

[**3** **Testing** 13](#_Toc437467036)

# **Pendahuluan**

## **Latar Belakang**

Sebuah komunitas ataupun Unit Kegiatan Mahasiswa (UKM) pasti memiliki *event-event* yang diselenggarakan baik secara berkala dalam kurun waktu tertentu ataupun yang diselenggarakan hanya sekali. Telkom University sebagai salah satu perguruan tinggi yang memiliki jumlah komunitas dan UKM yang banyak tentu memiliki *event-event* yang banyak juga.

Penyelenggaraan *event* tidak akan terlepas dengan kebutuhan properti yang tidak sedikit. Sebuah komunitas bahkan UKM yang besarpun tidak akan sanggup dalam memenuhi properti yang dibutuhkan dalam menyelenggarakan sebuah *event* dengan cara membelinya. Oleh karena itu, komunitas dan UKM membutuhkan sebuah alternative dalam memenuhi sarana dan properti tersebut, yaitu dengan cara menyewa. Dengan menyewa pengeluaran yang dibutuhkan dapat dihemat. Vendor merupakan sebutan untuk sebuah organisasi ataupun perseorangan yang memiliki usaha dalam bidang jasa penyewaan properti, seperti kursi, panggung, tenda, dan lain sebagainya.

Namun, mencari dan memilih sebuah vendor akan cukup sulit terutama bagi komunitas dan UKM yang baru dan belum meimiliki pengalaman dalam menyelenggarakan sebuah *event*, karena setiap vendor akan memberikan penawaran yang berbeda-beda pada properti yang sama baik dalam masalah harga ataupun kualitas. Selain itu, banyak vendor yang dapat dikatakan kurang bertanggung jawab dalam melaksanakan tugasnya dan hal tersebut dapat merugikan bagi penyelenggara *event* terutama yang belum memiliki pengalaman dalam memilih sebuah vendor. Oleh karena itu dibutuhkan sebuah sarana yang dapat digunakan sebagai wadah berkumpulnya komunitas, UKM, dan vendor. Maka dibuatlah aplikasi Event Organizer, aplikasi berbasis web yang berfungsi sebagai wadah dan forum yang dapat digunakan sebagai media promosi dan pencarian, sehingga komunitas dan UKM tidak perlu repot-repot dalam mencari informasi mengenai vendor dan vendor tidak akan sulit dalam mempromosikan jasa yang mereka sediakan.

## **Rumusan Masalah**

Dari latar belakang yang sudah dipaparkan sebelumnya, masalah yang dapat dirumuskan adalah sebagai berikut.

1. Bagaimana membuat aplikasi yang berisi informasi mengenai properti yang dibutuhkan sebuah komunitas atau UKM untuk menyelenggarakan sebuah *event*?
2. Bagaimana membuat aplikasi yang berisi informasi mengenai vendor-vendor dan jasa yang mereka sediakan?
3. Bagaimana membuat aplikasi yang berisi informasi mengenai tingkat kepercayaan yang didapat sebuah vendor?

## **Tujuan**

Aplikasi yang dibangun memiliki tujuan sebagai berikut.

1. Memberikan layanan kepada komunitas atau UKM untuk memberi informasi mengenai properti yang dibutuhkan untuk menyelenggarakan *event*.
2. Memberikan layanan kepada vendor-vendor untuk memberi informasi mengenai jasa yang mereka sediakan.
3. Memberikan rating kepada setiap vendor berdasarkan penilaian yang diberikan komunitas dan UKM yang sudah pernah menggunakan jasa vendor tersebut.

## **Batasan Masalah**

Aplikasi ***event organizer*** memiliki batasan sebagai berikut.

1. Aplikasi pada bagian admin berbasis web desktop.
2. Aplikasi pada bagian user berbasis web mobile (android).
3. Aplikasi membutuhkan koneksi internet.
4. Aplikasi dikhususkan untuk mahasiswa Telkom University dan vendor yang berlokasi di Bandung.
5. Pengaksesan web hanya dapat dilakukan oleh user yang terdaftar dan sudah memiliki akun.
6. Admin web tidak ikut campur tangan dalam kegiatan transaksi.

## **Metode Pengerjaan**

1. **Perencanaan**

Pada tahap ini ditentukan tujuan dari aplikasi yang akan dibuat. Tujuan proyek ini adalah membuat web yang digunakan sebagai forum pencarian dan promosi penyewaan properti yang digunakan untuk sebuah acara/*event* secara online.

1. **Analisis Kebutuhan**

Pada tahap ini dilakukan semua kebutuhan user yang berkaitan dengan aplikasi yang akan dibangun.

1. Wawancara adalah proses mencari informasi dengan cara melakukan tanya jawab dengan narasumber mengenai penyewaan properti untuk menyelenggarakan sebuah *event*.
2. Studi Literatur adalah proses mencari informasi dengan cara mengumpulkan data dari buku, internet, dan pendapat orang lain mengenai penyewaan properti untuk menyelenggarakan sebuah *event*.
3. **Desain**

Pada tahap ini informasi yang didapat dari tahap analisis kebutuhan dirubah menjadi sebuah desain yang akan diimplementasikan oleh programer.

1. **Pengkodean**

Pada tahap ini desain yang sudah ada diimplementasikan kedalam aplikasi yang dibangun. Pada pengimplementasian digunakan bahasa pemrograman yang sesuai dengan aplikasi berbasis web, yaitu PHP, CSS, javascript, HTML, Jquery, dan database MySQL.

1. **Pengujian**

Pengujian dilakukan setelah aplikasi selesai, metode yang digunakan adalah blackbox testing dimana tujuan dari pengujian adalah menemukan kesalahan yang ada program baik input ataupun output yang tidak sesuai dengan yang direncanakan.

1. **Dokumentasi**

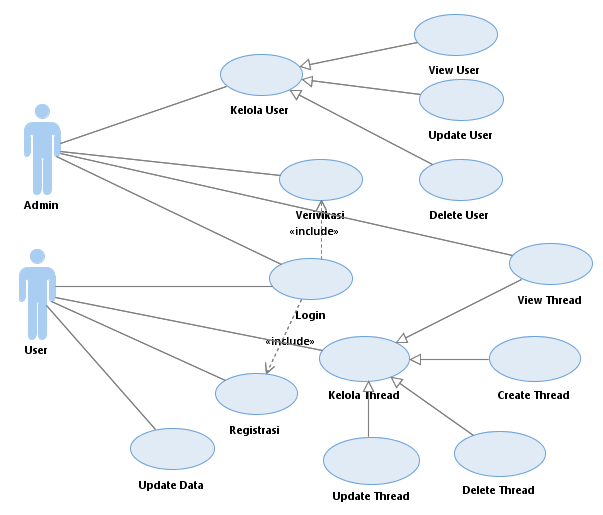
Proses pencatatan/dokumentasi dilakukan setiap tahap sebagai catatan mengenai pembangunan dari aplikasi.

# **Analisis Kebutuhan Sistem**

## **Use Case Model**

### **2.1.1 Use Case Diagram**

Secara umum, fungsionalitas yang akan dibangun memiliki 2 user yakni admin dan user. Hubungan antara usecase dan user akan digambarkan seperti gambar dibawah ini.



### **2.1.2 Skenario Use Case**

Berikut ini adalah skenario use case yang digunakan aplikasi event organizer, yaitu:

#### **2.1.2.1 Kelola User**

Nama *Use* *Case* : Kelola User

Aktor : Admin

Tipe : Primer

Tujuan : Untuk mengelola data user seperti mengedit, melihat, dan menghapus data.

Kondisi Awal : Admin sudah *log in* ke sistem.

Deskripsi : Admin bertindak sebagai pengawas yang berhak melihat, mengubah, dan menghapus data user.

|  |  |
| --- | --- |
| Aktor | Sistem |
| 1. Aktor mengakses menu verivikasi |  |
|  | 1. Sistem menampilkan seluruh data user. |
| 1. Aktor memilih edit atau hapus user. |  |
|  | 1. Sistem menampilkan *form* edit data user atau menghapus data user dan menyimpan ke database. |
| 1. Aktor mengisi *form* edit user*.* |  |
|  | 1. Sistem melakukan validasi *form* dan menyimpan data ke *database*. |

#### **2.1.2.2 View User**

Nama *Use* *Case* : View User

Aktor : Admin

Tipe : Sekunder

Tujuan : Untuk melihat data user secara lengkap.

Kondisi Awal : Admin sudah *log in* ke sistem.

Deskripsi : Admin bertindak sebagai pengawas yang berhak melihat seluruh data user.

|  |  |
| --- | --- |
| Aktor | Sistem |
| 1. Aktor mengakses menu verivikasi |  |
|  | 1. Sistem menampilkan seluruh data user. |

#### **2.1.2.3 Update User**

Nama *Use* *Case* : Update User

Aktor : Admin

Tipe : Sekunder

Tujuan : Untuk mengedit data user.

Kondisi Awal : Admin sudah *log in* ke sistem.

Deskripsi : Admin bertindak sebagai pengawas yang berhak mengubah data user.

|  |  |
| --- | --- |
| Aktor | Sistem |
| 1. Aktor mengakses menu verivikasi |  |
|  | 1. Sistem menampilkan seluruh data user. |
| 1. Aktor memilih edit user. |  |
|  | 1. Sistem menampilkan *form* edit data user. |
| 1. Aktor mengisi *form.* |  |
|  | 1. Sistem melakukan validasi *form* dan menyimpan data ke *database*. |

#### **2.1.2.4 Delete User**

Nama *Use* *Case* : Delete User

Aktor : Admin

Tipe : Sekunder

Tujuan : Untuk menghapus data user.

Kondisi Awal : Admin sudah *log in* ke sistem.

Deskripsi : Admin bertindak sebagai pengawas yang berhak menghapus data user.

|  |  |
| --- | --- |
| Aktor | Sistem |
| 1. Aktor mengakses menu verivikasi |  |
|  | 1. Sistem menampilkan seluruh data user. |
| 1. Aktor memilih hapus user. |  |
|  | 1. Sistem menghapus data user dan menyimpan ke database. |

#### **2.1.2.5 Verifikasi**

Nama *Use* *Case* : Verifikasi

Aktor : Admin

Tipe : Primer

Tujuan : Untuk menyetujui atau mendelete pendaftaran user baru.

Kondisi Awal : Admin sudah *log in* ke sistem.

Deskripsi : Admin bertindak sebagai pengawas yang berhak menyetujui atau menolak seorang calon user untuk bergabung kedalam sistem.

|  |  |
| --- | --- |
| Aktor | Sistem |
| 1. Aktor mengakses menu verivikasi |  |
|  | 1. Sistem menampilkan seluruh data user. |
| 1. Aktor memilih accept atau delete pada user yang belum diverifikasi. |  |
|  | 1. Sistem mengubah status user menjadi aktif atau menghapus data calon user dan menyimpan kedalam database. |

#### **2.1.2.6 Update Data**

Nama *Use* *Case* : Update Data

Aktor : User

Tipe : Primer

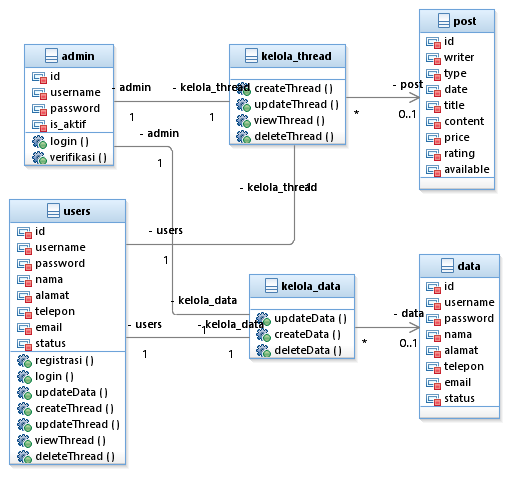
Tujuan : Untuk mengupdate data user.

Kondisi Awal : User sudah *log in* ke sistem.

Deskripsi : Melakukan perubahan data user.

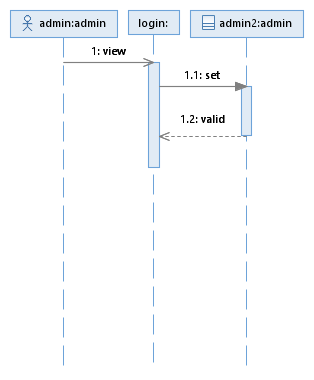
|  |  |
| --- | --- |
| Aktor | Sistem |
| 1. Aktor mengakses menu vendor atau logistik |  |
|  | 1. Sistem menampilkan seluruh thread. |
| 1. Aktor memilih edit user. |  |
|  | 1. Sistem menampilkan *form* edit data user. |
| 1. Aktor mengisi *form.* |  |
|  | 1. Sistem melakukan validasi *form* dan menyimpan data ke *database*. |

## **2.2 Class Diagram**

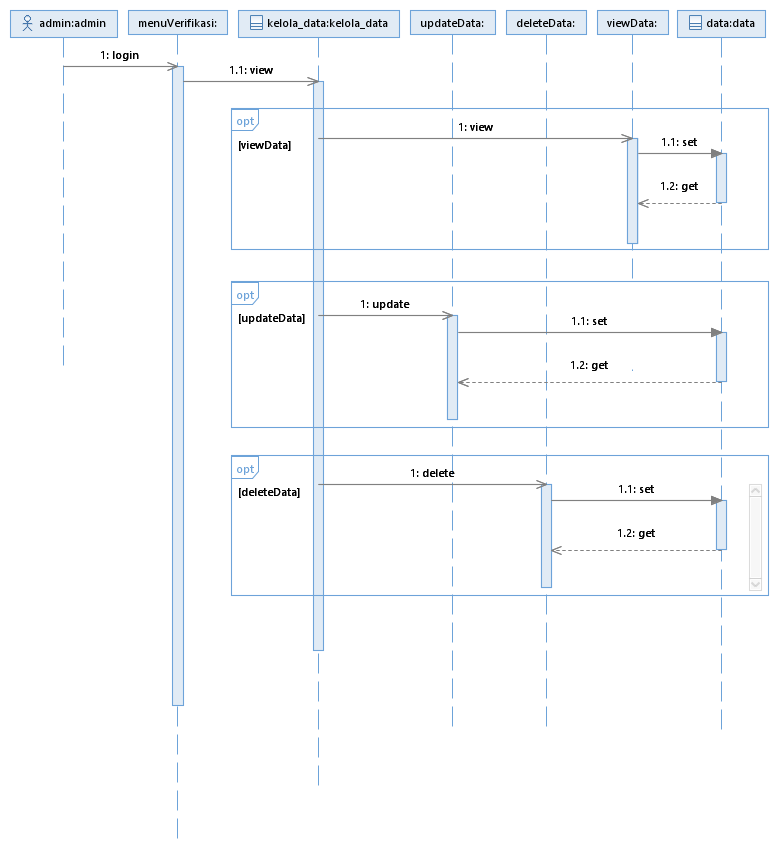


## **2.3 Sequence Diagram**

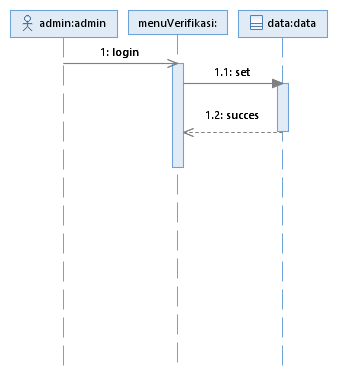
### **2.3.1 Login Admin**



### **2.3.2 Kelola User**



### **2.3.3 Verifikasi**



## **2.4 Kebutuhan Perangkat Keras dan Perangkat Lunak**

Kebutuhan perangkat keras dan perangkat lunak ini bertujuan untuk memberikan kemudahan serta kenyamanan dalam melakukan pembuatan aplikasi nantinya

### **Kebutuhan Perangkat keras**

Perangkat keras yang dibutuhkan sebisa mungkin mencapai batas-batas berikut.

|  |  |  |
| --- | --- | --- |
| No. | Spesifikasi Perangkat Keras | Kebutuhan |
| 1 | Laptop ( Notebook ) |  |
|  | 1. Ram | 4 GB |
|  | 1. Hardisk | 320 GB |
|  | 1. Processor | Intel(R) Core(TM) i5 |
|  | 1. Keyboard | 84 keys |
|  | 1. Monitor | 11.5 Inch |
|  | 1. Mouse atau pointing device lainnya |  |

### **Kebutuhan Perangkat Lunak**

Perangkat lunak yang diperlukan untuk mengerjakan tugas besar ini antara lain.

|  |  |  |
| --- | --- | --- |
| No. | Spesifikasi Perangkat Lunak | Kegunaan |
| 1 | Xampp 5.6.14 for windows | Penyedia layanan web server (MySQL) |
| 2 | Google Chrome | Web browser |
| 3 | Notepad++ / Sublime Text | Memudahkan dalam penulisan pengkodean dalam bentuk PHP ataupun HTML |
| 4 | Windows 7 atau ++ | Sebagai operating system |
| 5 | Eclips / netbeans | Untuk mengoperasikan serta membuat aplikasi android |

# **Testing**