***Aplikasi Jasa Wedding Organizer dan Event Organizer***

***Berbasis Android***

**Diajukan untuk memenuhi tugas UTS Peraktikum Rekayasa Perangkat Lunak Lanjut**

*https://github.com/ramdhan5370/tugasbesar*



Oleh:

Nama : Ramdhan Suseto

Nim : 1137050178

Kelas : IF-D

**Teknik Informatika**

**UIN Sunan Gunung Djati Bandung**

**2015**

# Latar Belakang

Seiring berjalannya waktu kita membutuhan pelayanan yang lebih cepat dan akurat, maka pengembangan aplikasi perlu dilakukan. Hal ini sangat wajar terjadi, karena sebuah aplikasi dalam kurun waktu tertentu harus di perbaharuhi sesuai perkembangan zaman dan kebutuhan.

Dengan demikian *smartphone* juga bisa di jadikan sebagai alat yang dapat memudahkan aktifitas dengan mengolahan data yang cepat, tepat, dan mempunyai sistem penyimpanan data. Yang didukung keamanan dengan sistem komputerisasi yang begitu efisien sehingga menghasilkan suatu aplikasi yang baik dan memberikan pelayanan yang memuaskan kepada pengguna dan tentunya dapat saling menguntungkan satu sama lain, salah satu contohnya adalah usaha online yang khususnya bergerak di bidang ekonomi.

Permasalahan yang terjadi saat ini yaitu belum adanya kemudahaan untuk *user* yang membutuhkan jasa WO(*Wedding Organizers*) atau EO(*Event Organizers*) secara cepat. Padahal ada beberapa aplikasi berbasis *mobile* ada yang dapat memudahkan *user* mencari jasa seperti Gojek, Grabike, Traveloka, dll, akan tetapi untuk jasa WO atau EO masih belum tersedia. Bila ada sistem yang sesuai dengan aplikasi yang mendukung maka kegiatan bisnis akan lebih efektif.

Oleh karena itu, kita membutuhkan sebuah aplikasi yang dapat mempermudah dalam melakukan pencarian jasa. Itulah mengapa penulis memilih judul “Aplikasi Jasa Wedding Organizer dan Event Organizer Berbasis Android”.

# Rumusan Masalah

Saat ini aplikasi penyedia jasa dinilai masih kurang efisien sehingga muncul beberapa masalah antara lain:

* Bagaimana aplikasi berbasis android lebih efisien di banding yang lain?
* Apakah aplikasi ini memudahkan user untuk mencari jasa yang diinginkan?

# Tujuan

Tujuan dari pembuatan aplikasi ini ialah:

* Banyaknya user yang menggunakan smartphone adalah alasan utama mengapa penulis membuat aplikasi ini.
* Aplikasi ini dapat memudahkan user untuk mencari jasa WO/EO yang diinginkan.

# Batasan

Batasan aplikasi ini adalah :

* Platform Android.
* Koneksi dengan internet.

# Deskripsi Perangkat Lunkak

## Deskripsi Umum

Aplikasi Jasa Wedding Organizers dan Event Organizers digunakan untuk melakukan pencarian jasa WO dan EO. P/L ini dilengkapi menu Login, Sign in, Pencarian jasa, WO, EO, Perjanjian dan Log out. Ketika user sudah Log in/Sign up maka masuk ke menu utama yang menyajikan pilihan Pencarian jasa, Perjanjian dan Log out. User memilih menu Pencarian jasa lalu akan muncul dua pilihan WO dan EO. Setelah itu user memasukkan keyword untuk menemukan jasa yang diinginkan. Lalu user memilih jasa yang diinginakan. Penyedia jasa akan mendapatkan notifikasi untuk melakukan perjanjian dan itu masuk kedalam menu Perjanjian.



## Pengguna

| Kategori Pengguna | Tugas | Hak Akses ke aplikasi |
| --- | --- | --- |
| Admin | * Mengelola data pengguna * Mengelola data pengusaha | Memiliki hak akses kesemuanya kecuali merubah pilihaan pengguna dan data pengusaha. |
| Pengguna | * Memilih menu * Memilih tipe jasa * Memilih pengusaha jasa | Dapat memilih menu apapun yang tersedia diaplikasi. |
| Pengusaha | * Memberikan keputusan penerimaan jasa | Memberikkan jawaban untuk pengusaha |

# Requirement

## Functional Requirement

| ID | Kebutuhan | Penjelasan |
| --- | --- | --- |
| FR-001 | Membuat system login untuk pengguna | Admin membuat system login supaya account pengguna lebih aman. |
| FR-002 | Menampilkan menu utama | Sistem menampilkan menu utama yang terdiri cari, info, user dan logout. |
| FR-003 | Menampilkan pilihan jasa | Sistem menampilkan pilihan jasa antara WO/EO. |
| FR-004 | Menampilkan pilihan pengusaha jasa | System menampilkan pengusaha-pengusaha jasa yang tersedia. |
| FR-005 | Menampilkan data pengusaha jasa | Sistem menampilkan detail tiap pengusaha jasa secara terperinci. |

## Non Functional Requirement

| ID | Parameter | Kebutuhan |
| --- | --- | --- |
| NFR-001 | Availability | Tersedia waktu kerja (07.00-17.00) |
| NFR-002 | Reliability | Tidak pernah boleh gagal |
| NFR-003 | Ergonomy | N/A |
| NFR-004 | Portability | N/A |
| NFR-005 | Memory | N/A |
| NFR-006 | Response time | Menampilkan pop up pesan “Berhasil melakukan perjanjian” dalam interval waktu 10-20 detik. |
| NFR-007 | Safety | N/A |
| NFR-008 | Security | Menggunakan Username dan Password |
| NFR-009 | Bahasa komunikasi | * Bahasa indonesia * Bahasa inggris |

# Usecase

## Usecase Diagram

## Definisi Actor

|  |  |  |
| --- | --- | --- |
| No | Actor | Deskripsi |
| 1 | Pengguna | Pengguna memiliki 2 kategori, member dan baru. |
| 2 | Pengguna Baru | Pengguna baru dapat mencari, memilih serta melakukan perjanjian tetapi harus mendaftar menjadi member terlebih dahulu. |
| 3 | Pengguna Member | Pengguna member dapat mencari jasa, memilih jasa dan melakukan perjanjian setelah login terlebih dahulu. |
| 4 | Admin | Admin mengelola data pengguna member dan pengguna baru. |
| 5 | Pengusaha Jasa | Pengusaha penyediakan jasa WO/EO. |

## Definisi Usecase

|  |  |  |
| --- | --- | --- |
| No | Use Case | Deskripsi |
| 1 | Mencari | Sistem memberikan menu pencarian jasa |
| 2 | Memilih | Sistem memberikan menu memilih jasa |
| 3 | Perjanjian | Sistem memberikan menu melakukan perjanjian jasa |
| 4 | Mendaftar Member | Sistem menampilkan form perndaftaran yang harus diisi oleh pengguna baru |
| 5 | Akses Masuk | Sistem meminta Pengguna member untuk memasukkan username dan password |
| 6 | Mencari Jasa | Sistem memberikan menu pencarian jasa |
| 7 | Memilih Jasa | Sistem memberikan menu memilih jasa |
| 8 | Melakukan Perjanjian | Sistem memberikan menu melakukan perjanjian jasa |
| 9 | Validasi Data Member | Sistem memeriksa apakah data member yang masuk sesuai yang ada di database. |

## Skenario Usecase

Use case : Mencari

Aktor : Pengguna Baru

|  |  |
| --- | --- |
| Pengguna Baru | Sistem |
| 1. Klik menu cari |  |
| 2. Masukkan jasa yang dicari |  |
|  | 3. Tampilkan hasil cari |
|  | 4. Jika hasil cari tidak ditemukan kembali ke 2 |

Use case : Memilih

Aktor : Pengguna Baru

|  |  |
| --- | --- |
| Pengguna Baru | Sistem |
| 1. Pilih menu yang tersedia |  |
|  | 2. Tampilkan menu yang dipilih |

Use case : Perjanjian

Aktor : Pengguna Baru

|  |  |
| --- | --- |
| Pengguna Baru | Sistem |
| 1. Masukkan nama jasa |  |
|  | 2. Tampilkan jasa yang dipilih |
|  | 3. Jika tidak ada kembali ke 2 |
| 4. Pilih nama jasa yang diinginkan |  |
| 5. Klik “Meetup” |  |
|  | 6. Menampilkan konfirmasi perjanjian |

Use case : Mendaftar Member

Aktor : Admin

|  |  |
| --- | --- |
| Admin | Sistem |
|  | 1. Menampilkan data pengguna baru |
| 2. Check pengguna |  |
|  | 1. Menampilkan data member |
|  | 1. Jika member tidak ada maka tidak dapat akses selanjutnya |

Use case : Akses Masuk

Aktor : Pengguna Member

|  |  |
| --- | --- |
| Pengguna Member | Sistem |
| 1. Klik Login |  |
| 2. Masukkan username dan password |  |
|  | 3. Masuk ke menu berikutnya |
|  | 4. Jika gagal masuk kembali ke 2 |

Use case : Mencari Jasa

Aktor : Pengguna Member

|  |  |
| --- | --- |
| Pengguna Member | Sistem |
| 1. Klik menu cari |  |
| 2. Masukkan jasa yang dicari |  |
|  | 3. Tampilkan hasil cari |
|  | 4. Jika hasil cari tidak ditemukan kembali ke 2 |

Use case : Memilih Jasa

Aktor : Pengguna Member

|  |  |
| --- | --- |
| Pengguna Member | Sistem |
| 1. Pilih menu yang tersedia |  |
|  | 2. Tampilkan menu yang dipilih |

Use case : Validasi Data Member

Aktor : Admin

|  |  |
| --- | --- |
| Pengguna Member | Sistem |
|  | 1. Menampilkan akses pengguna yang masuk |
| 1. Memeriksa data pengguna |  |
|  | 1. Menampilkan data member |
|  | 1. Jika data salah, tidak dapat mengakses menu berikutnya |

Use case : Melakukan Perjanjian

Aktor : Pengusaha Jasa

|  |  |
| --- | --- |
| Pengguna Member | Sistem |
|  | 1. Menampilkan pengguna yang ingin menggunakan jasa |
| 1. Mamutuskan terima/tidak |  |
|  | 1. Jika diterima pengguna dan pengusaha Meetup |
|  | 1. Jika tidak pengguna akan mendapatkan notifikasi penolakan |

# Class Diagram

1. Class diagram daftar pengguna baru



|  |  |  |
| --- | --- | --- |
| No | Nama Kelas | Jenis |
| 1 | displaySignin | View |
| 2 | DataBaru | Model |

1. Class diagram pengguna baru akses menu



|  |  |  |
| --- | --- | --- |
| No | Nama Kelas | Jenis |
| 1 | PenggunaBaru | Controller |
| 2 | displaySigin | View |
| 3 | MemilihBaru | Model |
| 4 | MencariBaru | Model |
| 5 | PerjanjianBaru | Model |

1. Class diagram pengguna member akses menu

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas | Jenis |
| 1 | PenggunaMember | Controller |
| 2 | displayMenu | View |
| 3 | MemilihMember | Model |
| 4 | MencariMember | Model |
| 5 | PerjanjianMember | Model |
| 6 | PengusahaJasa | Controller |
| 7 | Admin | Controller |
| 8 | ValidasiData | View |