**RANCANG BANGUN APLIKASI PESANTREN BERBASIS *WEB***

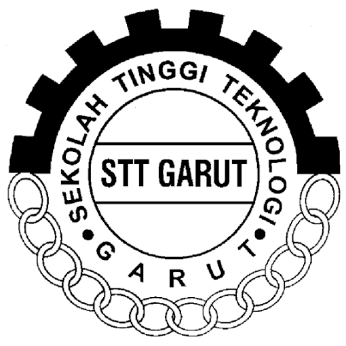
**LAPORAN KERJA PRAKTEK**

Dibuat untuk menyajikan hasil penalaran dari proses pengalaman kerja

**Oleh :**

**Marwah Maulana Sidik ( 1506020 )**

**Satria Adhi Nugraha ( 1506059 )**



**PROGRAM STUDI TEKNIK INFORMATIKA**

**SEKOLAH TINGGI TEKNOLOGI GARUT**

**2018**

**RANCANG BANGUN APLIKASI PESANTREN BERBASI WEB**

**LAPORAN KERJA PRAKTEK**

Dibuat untuk menyajikan hasil penalaran dari proses pengalaman kerja

**Oleh :**

**Marwah Maulana Sidik ( 1506020 )**

**Satria Adhi Nugraha ( 1506059 )**



**PROGRAM STUDI TEKNIK INFORMATIKA**

**SEKOLAH TINGGI TEKNOLOGI GARUT**

**2018**

**PERSYARATAN KEASLIAN**

Saya yang bertanda tangan di bawah ini :

Nama : Satria Adhi Nugraha

NIM : 1506059

Adalah wakil kelompok kerja, dengan ini menyatakan bahwa laporan kerja praktik yang dibuat belum pernah diajukan oleh siapapun, serta mengandung kutipan yang telah dilengkapi dengan sitasi dan tercantum dalam daftar pustaka secara memadai. Kami bersedia menerima sangsi akademik berupa nilai E apabila terbukti melakukan plagiasi, sesuai Peraturan Menteri Pendidikan Nasional Republik Indonesia nomor 17 tahun 2010.

Garut, 23 Februari 2019

Yang Membuat Pernyataan

Satria Adhi Nugraha

1506059

LEMBAR PENGESAHAN

**RANCANG BANGUN APLIKASI PESANTREN BERBASIS *WEB***

**LAPORAN KERJA PRAKTEK**

**Disusun oleh:**

**Marwah Maulana Sidik (1506020)**

**Satria Adhi Nugraha (1506059)**

Telah diseminarkan secara terbuka pada tanggal 4 Februari 2019

|  |  |
| --- | --- |
| Pembimbing  Aldy Rialdy Atmadja, M.T.  NIDN: 0426088801 | Penguji  Leni Fitriani, ST. M.Kom  NIDN: 0429058704 |
|  |  |
|  | |

Mengetahui,

Ketua Program Studi Teknik Informatika,

Rinda Cahyana, M.T.

NIDN: 0017107906

# ABSTRAK

Saat ini media informasi yang dapat diakses secara *online* sangat dibutuhkan untuk mengetahui informasi secara cepat yang dapat diakses dimana saja selama masih ada jaringan yang dapat menghubungkannya. Seperti halnya pesantren di Indonesia yang masih banyak belum memiliki sebuah website sebagai media penyebaran informasi, dengan demikian dibutuhkan aplikasi pesantren berbasis web ini untuk pesantren-pesantren seluruh indonesia yang dapat diperoleh secara gratis dan digagas oleh perusahaan MOBIDU Sinergi. Aplikasi ini dibuat saat pelaksanaan kerja praktik dengan memiliki hak akses sebagai admin dan user. Metodologi yang digunakan adalah *Rational Unified Process* sebagaisalah satu tahapan pengembangan sistem secara interaktif khusus untuk pemrograman berorientasi objek dengan empat fase yaitu *inception, elaboration, construction* dan *transition* dengan menggunakan pemodelan *Unified Modelling Language*, serta menggunakan metode *Black-Box Testing* sebagai pengujian aplikasi nantinya. Adapun hasil yang diharapkan dari aplikasi ini yaitu sebagai media penyebaran informasi pesantren yang dapat diakses secara online dan cepat selain itu, dapat memudahkan masyarakat untuk memperoleh informasi yang diinginkan.

**Kata Kunci :** Aplikasi Pesantren, *Laravel, Rational Unified Process*, *Black Box Testing.*

# KATA PENGANTAR

Puji syukur kita panjatkan Ke hadirat Allah SWT atas semua berkat dan karunia-NYA sehingga Laporan Tugas Besar ini dapat diselesaikan tepat waktu.Penyusun juga ingin mengucapkan rasa terima kasih sebesar-besarnya kepada pihak-pihak yang telah ikut membantu dalam pembuatan Laporan Kerja Praktek ini, antara lain kepada :

1. Bapak Dr. H. Hilmi Aulawi, ST., MT. selaku Ketua Sekolah Tinggi Teknologi Garut.
2. Bapak Rinda Cahyana, ST., MT. selaku Ketua Program Studi Informatika Sekolah Tinggi Teknologi Garut.
3. Aldy Rialdy Atmadja, MT. selaku dosen yang selalu membimbing kami saat menyusun laporan ini.
4. Ayah ibu dan saudara-saudaraku yang telah mencurahkan do’a, perhatian dan kasih sayangnya selama ini dan semoga untuk selama-lamanya.
5. Teman-temanku di Jurusan Informatika yang telah memberikan semangat dan do’a dalam pengerjaan laporan ini.
6. Semua pihak yang telah ikut terlibat secara langsung maupun tidak langsung dalam proses pembuatan laporan ini yang tidak dapat disebutkan satu persatu.

Akhir kata penyusun berharap agar laporan ini dapat bermanfaat bagi berbagai pihak yang berkepentingan. Penyusun juga menyadari masih adanya kekurangan dan ketidak sempurnaan pada Laporan Kerja Praktek ini, sehingga tidak menutup adanya pengambangan lebih lanjut dari tugas yang telah dibuat ini. Oleh karena itu penyusun senantiasa mengharapkan adanya kritik dan masukan yang membangun.

Garut,9 Oktober 2018

Penyusun

# DAFTAR ISI

[ABSTRAK i](#_Toc535825124)

[KATA PENGANTAR ii](#_Toc535825125)

[DAFTAR ISI iii](#_Toc535825126)

[DAFTAR GAMBAR v](#_Toc535825127)

[DAFTAR TABEL vi](#_Toc535825128)

[DAFTAR LAMPIRAN vii](#_Toc535825129)

[1. PENDAHULUAN 1](#_Toc535825130)

[1.1. Latar Belakang 1](#_Toc535825131)

[1.2. Tujuan Pekerjaan 1](#_Toc535825132)

[1.3. Ruang Lingkup Pekerjaan 2](#_Toc535825133)

[1.4. Tahapan Dan Waktu Kerja 2](#_Toc535825134)

[1.5. Sistematika Penulisan 2](#_Toc535825135)

[2. LANDASAN TEORI 4](#_Toc535825136)

[2.1. Rancang Bangun 4](#_Toc535825137)

[2.2. Pengertian Aplikasi 4](#_Toc535825138)

[2.3. Pesantren 4](#_Toc535825139)

[2.4. *Web* 5](#_Toc535825140)

[2.5. *Rational Unfied Process* 5](#_Toc535825141)

[2.6. *Unified Modeling Language* 6](#_Toc535825142)

[2.6.1. *Use Case* 7](#_Toc535825143)

[2.6.2. *Activity Diagram* 9](#_Toc535825144)

[2.6.3. *Sequence Diagram* 9](#_Toc535825145)

[2.6.4. *Class Diagram* 10](#_Toc535825146)

[2.7. Perangkat Lunak Pendukung 11](#_Toc535825147)

[2.7.1. *Laravel* 11](#_Toc535825148)

[2.7.2. *PHP* 12](#_Toc535825149)

[2.7.3. *MySQL* 12](#_Toc535825150)

[2.7.4. *Php MyAdmin* 12](#_Toc535825151)

[2.7.5. *HTML* 12](#_Toc535825152)

[2.7.6. *Adobe Ilustrator* 13](#_Toc535825153)

[3. METODOLOGI PEKERJAAN 14](#_Toc535825154)

[3.1. *Work Breakdown Structur* 14](#_Toc535825155)

[3.2. *Gantt Chart* 15](#_Toc535825156)

[4. HASIL DAN PEMBAHASAN 17](#_Toc535825157)

[4.1. Hasil Pekerjaan 17](#_Toc535825158)

[4.1.1. *Inception* 17](#_Toc535825159)

[4.1.1.1. Peroses Bisnis Yang Sedang Berjalan 17](#_Toc535825160)

[4.1.1.2. Proses Bisnis Yang Diajukan 18](#_Toc535825161)

[4.1.1.3. Spesifikasi Sistem 19](#_Toc535825162)

[4.1.1.4. Identifikasi Aktor 20](#_Toc535825163)

[4.1.1.5. Identifikasi *Use Case* Awal 20](#_Toc535825164)

[4.1.1.6. *Use Case* 21](#_Toc535825165)

[4.1.1.7. *Activity Diagram* 23](#_Toc535825166)

[4.1.2. *Elaboration* 25](#_Toc535825167)

[4.1.2.1. *Sequence Diagram* 25](#_Toc535825168)

[4.1.2.2. *Class Diagram* 27](#_Toc535825169)

[4.1.3. *Construction* 27](#_Toc535825170)

[4.1.3.1. Perancangan *Layout* 27](#_Toc535825171)

[4.1.3.2. Rancangan Aplikasi Versi Beta 30](#_Toc535825172)

[4.1.4. Pengujian 31](#_Toc535825173)

[4.2. Pengetahuan dan Keterampilan Empiris 32](#_Toc535825174)

[4.3. Etika Kerja 33](#_Toc535825175)

[5. KESIMPULAN DAN SARAN 34](#_Toc535825176)

[5.1. Kesimpulan 34](#_Toc535825177)

[5.2. Saran 34](#_Toc535825178)

[DAFTAR PUSTAKA 35](#_Toc535825179)

# DAFTAR GAMBAR

[Gambar 2.1. Tahapan *Rational Unified Process* 5](#_Toc535825101)

[Gambar 2.2 Notasi *Sequence Diagram* 10](#_Toc535825102)

[Gambar 3.1. *Work Breakdown Structure* 14](#_Toc535825103)

[Gambar 3.2. Diagram Alur Aktivitas 15](#_Toc535825104)

[Gambar 4.1. *Floechart* Peroses Bisnis Yang Berjalan. 17](#_Toc535825105)

[Gambar 4.2. *Flowchart* Proses Bisnis yang diajukan 18](#_Toc535825106)

[Gambar 4.3. *Use Case* Diagram Aplikasi Pesantren Berbasis Web 21](#_Toc535825107)

[Gambar 4.4. *Activity Diagram* Kelola *Post* 23](#_Toc535825108)

[Gambar 4.5. *Activity Diagram* Melihat Pengumuman 24](#_Toc535825109)

[Gambar 4.6. *Sequence Diagram* Kelola *Post* 26](#_Toc535825110)

[Gambar 4.7. *Sequence Diagram* Melihat Pengumuman 26](#_Toc535825111)

[Gambar 4.8. *Class Diagram* Aplikasi Pesantren Berbasis Web 27](#_Toc535825112)

[Gambar 4.9. Struktur Menu Aplikasi Pesantren 28](#_Toc535825113)

[Gambar 4.10. Tampilah Halaman Utama 28](#_Toc535825114)

[Gambar 4.11. Tampilan Halaman Tambah *Post* 29](#_Toc535825115)

[Gambar 4.12. Tampilan Halaman Single Kategori 29](#_Toc535825116)

[Gambar 4.13. Tampilan Halaman *Home* 30](#_Toc535825117)

[Gambar 4.14. Tampilan Halaman Kategori 31](#_Toc535825118)

# DAFTAR TABEL

[Tabel 2.1 Simbol-Simbol Diagram *Use Case* 7](#_Toc535825183)

[Tabel 2.2. Notasi *Activity Diagram* 9](#_Toc535825184)

[Tabel 2.3. Simbol-Simbol Pada *Class Diagram* 10](#_Toc535825185)

[Tabel 3.1. Sumber Daya Penelitian 16](#_Toc535825186)

[Tabel 4.1. Aktivitas Aktor 20](#_Toc535825187)

[Tabel 4.2. Skenario Use Case Kelola Post 21](#_Toc535825188)

[Tabel 4.3. Skenario *Use Case* Melihat Pengumuman 22](#_Toc535825189)

[Tabel 4.4. Deskripsi *Activity* *Diagram* Kelola *Post* 23](#_Toc535825190)

[Tabel 4.5. Deskripsi *Activity* *Diagram* Melihat Pengumuman Pesantren 25](#_Toc535825191)

[Tabel 4.6. Form Pengujian Aplikasi Pesantren Berbasis *Web* 31](#_Toc535825192)

# DAFTAR LAMPIRAN

[LAMPIRAN A : LANDASAN PEKERJAAN 36](#_Toc534700005)

[LAMPIRAN B : PRESENSI KERJA 38](#_Toc534700008)

[LAMPIRAN C : HASIL PEKERJAAN 42](#_Toc534700013)

# PENDAHULUAN

## Latar Belakang

Di kota garut ini terdapat Startup Company yaitu Mobidu Sinergi yang bertepatan di Jl. Rancabango *Estate* Ruko A2 Tarogong Kaler. Mobidu sendiri berdiri pada tahun 2017, kata mobidu berasal dari nama mobile education yang menjadi sebuah inovasi dari bidang tersebut salah satunya diciptakannya SDLC yang telah dijadikan hak paten Mobidu oleh Yusep Maulana selaku inovator. Sedangkan nama sinergi memiliki makna saling mengisi dan melengkapi perbedaan untuk mencapai hasil lebih besar dalam bidang teknologi informasi dan umum. dimana perusahaan ini merupakan perusahaan penyedia produk dan jasa pembuat dan pengembang teknologi informasi dan multimedia yang memberikan solusi pengembangan perangkat lunak aplikasi baik berbasis desktop, web, android dan iso sebagai penunjang system teknologi informasi yang akan dibangun.

Dimana saat ini perusahaan sedang membutuhkan aplikasi berbasi web untuk pesantren yang memiliki hak akses admin dan user, dan fitur berita. Karena masih banyak pesantren-pesantren di indonesia khusus-nya di garut yang masih belum memiliki web pesantren yang dapat di akses secara *online* untuk memudahkan masyarakat umum dalam mendapatkan informasi suatu pesantren, dalam hal ini peran mahasiswa yang melakukan kerja praktek di Mobidu Sinergi dibutuhkan untuk melakukan pembuatan web pesantren sebagai *designer* dan *programmer*. Berdasarkan latar belakang diatas, maka dari itu, penyusun akan merancang sistem pesantren berbasis *web* yang berjudul **“Rancang Bangun Aplikasi Pesantren Berbasis *Web*”.**

## Tujuan Pekerjaan

Adapula tujuan dari pekerjaan yang telah ditugaskan oleh perusahaan Mobibu Sinergi kepada mahasiswa adalah :

1. Mendesain tampilan web.
2. Membuat aplikasi pesantren berbasis web yang memiliki hak akses admin dan user serta terdapat fitur berita.

## Ruang Lingkup Pekerjaan

Agar dalam pembuatan aplikasi berjalan dengan baik dan lancar kami berkerjasama dengan memulai pembagian tugas, tugas yang pertama yaitu mendesain tampilan web dan yang kedua membuat membuat aplikasi web berdasarkan desain .Adapula spesifikasi perangkat yang dibutuhkan dalam pekerjaan yang kami lakukan diantaranya :

1. Spesifikasi perangkat yang digunakan *designer*, menggunakan *windows* 8.1 Pro 64-bit, *System Model* 80E4, *BIOS* *Insyde*H20, *Processor Intel* *Core* i3, 2048MB RAM.
2. Spesifikasi perangkat yang digunakan *programmer*, menggunakan *windows* 10 64-bit, *Processor AMD* A10, 4096MB RAM.

## Tahapan Dan Waktu Kerja

Pelaksanaan Kerja Praktek (KP) ini dilaksanakan di Perusahaan CV. Mobidu Sinergi, Jalan Rancabango Estate Blok A2 Tarogong Kaler Kabupaten Garut. Adapula waktu kerja yang dilakukan saat kerja peraktek yaitu hari senin sampai dengan hari jum’at dimulai jam 09:00 sampai jam 15:30.

## Sistematika Penulisan

Laporan kerja praktik ini disusun dengan sistematika sebagai berikut :

1. **PENDAHULUAN**, berisi latar belakang, tujuan pekerjaan, ruang lingkup pekerjaan, tempat dan waktu kerja, serta sistematika penulisan;
2. **LANDASAN TEORI**, berisi kutipan konsep, teori dan metode terkait tujuan pekerjaan yang bersumber dari sejumlah pustaka;
3. **METODOLOGI PEKERJAAN**, berisi penjelasan tentang tahapan kerja berikut aktivitas dan teknik yang digunakan, serta pelaksana pekerjaan, waktu, tempat, dan sumber daya yang digunakan, dengan memperhatikan landasan teori;
4. **HASIL DAN PEMBAHASAN**, berisi penjelasan tentang hasil pekerjaan, pengetahuan dan keterampilan empiris yang diperoleh, serta bagaimana kearifan lokal diperhatikan di dalam pekerjaan dengan merujuk kepada pustaka;
5. **KESIMPULAN DAN SARAN,** berisi ringkasan penelitian berdasarkan hasil dan pembahasan, serta rekomendasi pekerjaan terkait kelemahankemampuan mahasiswa atau hambatan di tempat kerja.

# LANDASAN TEORI

## Rancang Bangun

Rancang merupakan serangkaian prosedur untuk menerjemahkan hasil analisis dari sebuah sistem kedalam bahasa pemrograman yaitu untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem diimplementasikan. Sedangkan pengertian bangun atau pembangunan sistem adalah kegiatan menciptakan baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun sebagian (Pressman, 2012).

Jadi Rancang Bangun yaitu Rancangan yang dibuat untuk menciptakan hal yang baru yang dapat memudahkan seseorang, melalui pembuatan sebuah sistem yang dibutuhkan.

## Pengertian Aplikasi

Aplikasi yaitu merupakan salah satu faktor kebutuhan di zaman sekarang untuk mempermudah suatu pekerjaan. Perangkat lunak aplikasi adalah program-program mandiri yang menjawab kebutuhan bisnis yang terperinci. Aplikasi-aplikasi dalam bidang ini melakukan pemrosesan data bisnis atau data teknis yang mendukung berjalannya operasi-operasi bisnis atau pengaturan / pengambilan keputusan teknis. Selain ada aplikasi pemrosesan data konvensional, perangkat lunak aplikasi juga digunakan untuk mengendalikan fungsi-fungsi bisnis secara actual (Pressman, 2012).

## Pesantren

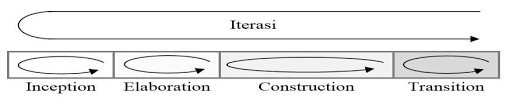
Pesantren adalah salah satu institusi yang unik dengan ciri-ciri khas yang sangat kuat dan lekat. Peran yang diambil adalah upaya-upaya pencerdasan bangsa yang telah turun temurun tanpa henti. Pesantrenlah yang memberikan pendidikan pada masa-masa sulit, masa perjuangan melawan kolonial dan merupakan pusat studi yang tetap *survive* sampai masa kini. Tujuan pendidikan pesantren bukanlah untuk mengejar kepentingan kekuasaan, uang dan keagungan duniawi, tetapi ditanamkan kepada mereka bahwa belajar adalah semata-mata kewajiban dan pengabdian kepada Tuhan. Oleh karena itu, sebagai salah satu lembaga pendidikan, pesantren juga mempunyai tanggung jawab yang tidak kecil dalam membentuk karakter para santri. (Zuhriy, 2011).

## *Web*

*World Wide Web* sendiri yaitu merupakan alat bantu yang sangat dibutuhkan oleh manusia dari sejak *web* muncul hingga sekarang, sebagai penunjang bagi semua orang agar dapat terjun ke dalam dunia maya dengan mencantumkan *hyperlink*. Dengan pendekatan *hyperlink* ini,seseorang dapat memperoleh informasi dengan meloncat dari satu halaman ke halaman lain (Kadir, 2005).

## *Rational Unfied Process*

*Rational Unfied Process* (RUP) adalah salah satu tahapan pengembangan sistem secara interaktif khusus untuk pemrograman berorientasi objek. RUP merupakan pendekatan pengembangan perangkat lunak yang dilakukan berulang-ulang (*interative*), fokus pada arsitektur.dan lebih diarahkan berdasarkan penggunaan kasus. (Sukamto & Shalahuddin, 2016).

****

**Gambar 2.1. Tahapan *Rational Unified Process***

(Sukamto & Shalahuddin, 2016)**.**

Penjelasan untuk setiap fase pada RUP yaitu (Sukamto & Shalahuddin, 2016).

1. *Inception* (pemulaan)

Tahapan ini lebih pada pemodelan proses bisnis yang dibutuhkan (*business modeling*) dan mendefinisikan kebutuhan akan sistem yang akan dibuat (*requirements*).

Berikut adalah tahapan yang dbutuhkan pada tahapan ini:

1. Memahami ruang lingkup dari proyek (termasuk pada biaya, waktu, kebutuhan, resiko dan lain sebagainya)
2. Membangun kasus bisnis yang dibutuhkan.
3. *Elaboration* (perluasan/perencanaan)

Tahapan ini lebi difokuskan pada perencanaan arsitektur sistem. Tahap ini juga dapat mendeteksi apakah arsitektur sistem yang diinginkan dapat dibuat atau tidak. Mendeteksi resiko yang mungkin terjadi dari arsitektur yang dibuat. Tahap ini lebih pada analisis dan desain sistem serta implementasi sistem yang fokus pada purwarupa sistem (prototype).

1. *Construction* (konstruksi)

Tahap ini fokus pada pengembangan komponen dan fitur-fitur sistem. Tahap ini lebih pada implementasi dan pengujian sistem yang fokus pada implementasi perangkat lunak pada kode program. Tahap ini menghasilkan produk perangkat lunak dimana menjadi syarat *Initial Operational Capability Milestone* atau batas/tonggak kemampuan operasional awal.

1. *Transition* (transisi)

Tahap ini lebih pada *deployment* atau instalasi sistem agar dapat dimengerti oleh *user*. Tahap ini menghasilkan produk perangkat lunak dimana menjadi syarat dari *Initial Operational Capability Milestone* atau batas/tonggak kemampuan operasional awal. Aktifitas pada tahap ini termasuk pada pelatihan *user*, pemeliharaan dan pengujian sistem apakah sudah memenuhi harapan *user.* Tahapan uji coba ini menggunakan metode *black-box* *testing,* yang bertujuan untuk menguji sistem dengan analisis atau tahapan yang ada pada metodologi *Rational Unfied Process*.

Akhir dari keempat fase ini adalah produk perangkat lunak yang sudah lengkap. Keempat fase pada RUP dijalankan secara berurutan dan iteratif dimana sistem interasi dapat digunakan untuk memperbaiki interasi berikutnya.

## *Unified Modeling Language*

Salah satu pemodelan yang saat ini paling banyak digunakan salah satunya yaitu UML (*Unified Modeling Language*), UML adalah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.

UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataanya UML paling banyak digunakan pada metodologi berorientasi objek. (Sukamto & Shalahuddin, 2016)

### Use Case

*Use case* merupakan pemodelan kelakukan (*behavior*) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. (Sukamto & Shalahuddin, 2016).

Tabel 2.1 Simbol-Simbol Diagram *Use Case*

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| Use case | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal di awal frase nama use case. |
| Asosiasi / *association* | Komunikasi antara aktor dan *use case* yang berpartisipasi pada *use case* atau *use case* memiliki intraksi dengan aktor |

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| Aktor/*actor* | Orang, proses, atau sistem lain yang berintraksi dengn sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang ,tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awl frase nama aktor. |
| Ekstensi/ *extend*  << extend >> | Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan. |
| Generalisasi/ *generalization* | Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya. |
| Menggunakan / *include / uses*  <<include>>    <<uses>> | Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan *use case* ini. |

**Sumber :** (Sukamto & Shalahuddin, 2016).

### Activity Diagram

*Activity Diagram*  menggambarkan *Workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu di perhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dilakukan oleh sistem.

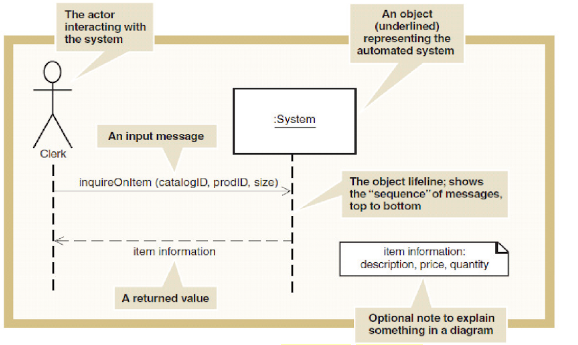
Tabel 2.2. Notasi Activity Diagram

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| status awal | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal. |
| aktivitas  aktivitas | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja. |
| percabangan / *decision* | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu. |
| penggabungan / *join* | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu. |
| status akhir | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir. |

**Sumber:** (Sukamto & Shalahuddin, 2016)

### Sequence Diagram

*Sequence Diagram* adalah diagram yang menggambarkan interaksi antar objek. *Sequence Diagram* secara khusus menjabarkan *behavior* sebuah skenario tunggal. *Diagram* tersebut menunjukkan sejumlah objek contoh dan pesan-pesan yang melewati objek ini dalam sebuah *use case*. (Mulyani, 2016)



Gambar 2.2 Notasi *Sequence Diagram*

(Triandini & Suardika, 2012)

### Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian dari kelas-kelas yang akan dibuat untuk membangun sistem. Diagram kelas dibuat agar pembuatan program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron (Sukamto & Shalahuddin, 2016).

Tabel 2.3. Simbol-Simbol Pada *Class Diagram*

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| |  | | --- | | Nama\_Kelas | | +Atibut | | +operasi() | | Kelas pada struktur sistem. |
|  | Sama dengan konsep interface dalam pemrograman berorientasi objek. |
| **Simbol** | **Deskripsi** |
|  | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity.* |
|  | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity.* |
|  | Relasi antar kelas dengan makna generalisasi-spesialisasi |
|  | Relasi antar kelas dengan makna kebergantungan antar kelas |
|  | Relasi antar kelas dengan makna semua-bagian. |

**Sumber :** (Sukamto & Shalahuddin, 2016)**.**

## Perangkat Lunak Pendukung

Perangkat lunak pendukung ini yaitu sebagai sarana untuk membantu dalam menyelesaikan pembuatan aplikasi pesantren berbasis web. Adapun perangkat lunak pendukung tersebut diantaranya *Laravel,* PHP, MySQL, PhpMyAdmin, HTML dan Adobe Ilustrator.

### Laravel

Laravel dibuat oleh Taylor Otwell, proyek Laravel ini dimulai pada April 2011. Larapel adalah sebuah framework PHP yang dirilis dibawah lisensi MIT, dengan kode sumber yang sudah disediakan oleh Github, sama halnya seperti framework-framework yang lain, Laravel dibangun dengan konsep MVC (Model-Controller-View), kemudian Laravel dilengkapi juga *command line tool* yang bernama Artisan yang bisa digunakan untuk *packaging bundle* dan *instalasi bundle* melalui command prompt. (Aminudin, 2015).

### PHP

PHP Merupakan bahasa pemrograman atau *script* yang digunakan untuk mengola data dari server untuk ditampilka ke *website.* Menurut dokumen resmi PHP, PHP singkatan dari *Hypertext PreProcessor*. PHP ini merupakan bahasa berbentuk skrip yang di tempatkan dalam server dan diperoses di server. Hasilnyalah yang di kirimkan ke klien,tempat pemakai menggunakan *browser*. (Kadir, 2002).

### MySQL

MySQL merupakan sebuah database yang sering digunakan oleh banyak orang untuk pembuatan database kedalam aplikasi yang akan dibuat, MySQL sendiri adalah sebuah program database server yang mampu menerima dan mengirimkan datanya dengan sangat cepat, multi user serta menggunakan perintah standar SQL ( Structured Query Language ). (Nugroho, 2005).

### Php MyAdmin

Php MyAdmin adalah suatu program Open Sorce berbasis web.dimana Program ini berguna untuk mengakses database MySQL. Dengan adanya Program ini akan mempermudah dan mempersingkat kerja kita, dengan beberapa kelebihan , antara lain pengguna awam tidak harus mengenal syntax-syntax SQL dalam pembuatan database dan tabel. (Nugroho, 2005).

### HTML

Semua pemrograman web, pasti memiliki kemampuan untuk mengolaborasikan kodenya dengan html. Html sendiri kepanjangan dari *Hyper Text Markup Language,* yaitu suatu bahasa pemrograman *Hyper Text.* Html bisa digunakan untuk pembuatan aplikasi karena berfungsi untuk membangun kerangka ataupun format web berbasis html. Begitu juga Html merupakan bahasa yang dapat digunakan untuk menampilkan dan mengelola *hypertext*. (Saputra, Subagio, & Saluky, 2012).

### Adobe Ilustrator

Adobe Ilustrator merupakan salah satu aplikasi pendukung untuk membuat desain aplikasi, karna aplikasi ini adalah salah satu program pembuat dan pengelola vektor. Program ini banyak memiliki fasilitas, di antaranya kemampuan untuk membuat objek tiga dimensi secara cepat dan juga memiliki keragaman warna yang lebih kompleks, serta fasilitas-fasilitas lainnya. (Prabawati, 2010).

# METODOLOGI PEKERJAAN

## Work Breakdown Structur (WBS)

*Work Breakdown Structure* (WBS) merupakan suatu metode pengorganisasian yang digunakan untuk memecahkan tiap peroses pekerjaan menjadi lebih detail. Dengan metode ini kami dapat menggambarkan tahapan-tahapan dalam pembuatan aplikasi yang di berikan oleh perusahaan Mobidu Sinergi.

|  |  |
| --- | --- |
| ***Work Breakdown Structure*** | |
|  | |
| Tujuan | Rancang Bangun AplikasiPesantrenBerbasis *Web* |
| Tahapan | ***Construction***  ***Elaboration***  **Transition**  ***Inception*** |
| Aktivitas | Merancang Sequence Diagram  Merancang Class Diagram  Pengujian *Black-box*  Pengimplementasian kedalam bahasa pemrograman  Merancang *Aktivity* Diagram  Merancang *Use Case* Diagram  Identifikasi Aktor  Merancang Spesifikasi Sistem  Indentifikasi Peroses Bisnis Yang Sedang Berjalan  Perancangan interface  Perancangan struktur menu |

Gambar 3.1. *Work Breakdown Structure*

Susunan pada tahapan-tahapan *Work Breakdown Structure* adalah *Inception,Elaboration,Constuction dan Transition.* Dimana tahapan-tahapan ini merupakan tahapan dan Aktivitas yang akan di buat untuk peroses perancangan aplikasi yang berjudul **”Rancang Bangun Aplikasi Pesantren Berbasis *Web*”.**

## *Gantt Chart*

Berdasarkan *Work Breakdown Structure* yang sudah digambarkan maka aktivitas penelitian akan di sajikan dalam diagram alur aktivitas berikut ini;

23 Juli 2018

23 Juli 2018

6 Agustus 2018

13 Agustus 2018

3 September 2018

20 Agustus 2018

14 Agustus 2018

6 Agustus 2018

6 Agustus 2018

23 Juli 2018

23 Juli 2018

23 Juli 2018

23 Juli 2018

23 Juli 2018

23 Juli 2018

23 Juli 2018

23 Juli 2018

23 Juli 2018

Gambar 3.2. Diagram Alur Aktivitas

Mengacu pada tahapan yang sudah dijelaskan oleh (Kruchten, 2004) mengenai tahapan RUP (*Rational Unified Process*)yaitu *inception*, *elaboration*, *construction* dan *transition.* Dimulai dari tahapan *inception* dimana tahapan ini mengidentifikasi proses bisnis yang sedang berjalan yang didapat dari hasil wawancara, dokumen dan buku referensi atau jurnal penelitian yang sudah ada.

Tahap selanjutnya yaitu *elaboration* tahapan ini menentukan gambaran secara rinci suatu sistem dan pembuatan rancangan dengan membuat *sequence* diagram, *class* diagram, struktur menu dan rancangan *layout*.

Pada tahapan *construction* peneliti melakukan perancangan dan pembuatan aplikasi yang nantinya akan di implementasikan ke bahasa pemrograman, yang kemudian pada tahap terakhir akan dilakukan pengujian.

Tabel 3.1. Sumber Daya Penelitian

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Aktivitas** | **Manusia** | **Perangkat** | **Tanggal** |
| 1. | Pengumpulan kebutuhan untuk aplikasi yang akan dibuat | Marwah maulana sidik & Satria Adhi Nugraha | Laptop,seperangkat alat tulis | 23 Juli 2018 |
| 2. | Identifikasi actor | Marwah Maulana Sidik & Satria Adhi Nugraha | Laptop, ArgoUML | 23 Juli 2018 |
| 3. | Desain rancangan aplikasi | Marwah Maulana Sidik | Laptop, Aplikasi Pendukung | 23 Juli 2018 |
| 4. | Implementasi desain ke bahasa pemrograman | Satria Adhi Nugraha | Laptop,XAMPP, Cmd | 20 Agustus 2018 |
| 5. | Perancangan dan pembuatan *database* | Satria Adhi Nugraha | Laptop,XAMPP, Cmd | 20 Agustus 2018 |
| 6. | Penanganan terjadinya error saat pembuatan aplikasi | Satria Adhi Nugraha | Laptop,XAMPP, Cmd | 20 Agustus 2018 |
| 7. | *Black Box Testing* | Marwah Maulana Sidik & Satria Adhi Nugraha | Laptop | 3 September 2018 |

# HASIL DAN PEMBAHASAN

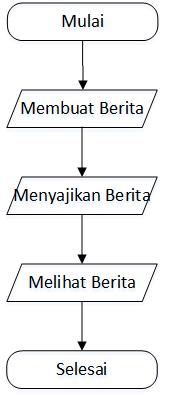
## Hasil Pekerjaan

### Inception

Pada tahapan ini dilakukan pemodelan proses bisnis yang sedang berjalan (*business modeling*) dan menentukan kebutuhan sistem yang akan dibuat (*requirements*).

#### Proses Bisnis Yang Sedang Berjalan

Adapun proses bisnis yang sedang berjalan pada pesantren saat ini dapat digambarkan pada *flowchart* sebagai berikut :



Gambar 4.1. *Flowchart* Proses Bisnis Yang Berjalan.

Berdasarkan *flowchart* yang telah dibuat pada gambar 4.1 dimana seorang petugas dari pesantren membuat sebuah berita atau pengumuman yang kemudian disajikan pada sebuah mading agar dapat dilihat oleh para santri pesantren tersebut.

#### Proses Bisnis Yang Diajukan

Proses bisni yang diajukan untuk media informasi pada pesantren dibuat dalam bentuk gambar *flowchart* berikut ini :



Gambar 4.2. *Flowchart* Proses Bisnis yang diajukan

Adapun penjelasan pada gambar 4.2 adalah untuk melihat informasi atau berita dipesantren yang sudah menggunakan media informasi yaitu web pesantren yang telah dibuat. User akan membuka web pesantren terlebih dahulu kemudian akan tampil halaman utama dari web pesantren yang didalamnya user dapat melihat berita yang meliputi informasi diluar bidang akademik seperti prestasi, kemudian pengumuman yang menginformasikan berita-berita seputar akademik seperti jadwal ujian dan lain-lain, adapula informasi basiswa. Dari kategori-kategori berita seperti pengumuman dan beasiswa user juga dapat melihat berdasarkan tag siswa dan guru sehingga ketika user ingin melihat info untuk siswa maka seluruh kategori berita, pengumuman dan basiswa yang ditujukan untuk siswa akan ditampilkan begitu juga dengan tag guru.

#### Spesifikasi Sistem

1. Rincian Persyaratan

Tahapan ini akan menjelaskan mengenai kebutuhan pada perancangan aplikasi yang akan dibuat.

1. Persyaratan Tampilan
2. Menyediakan fitur berita.
3. Terdapat hak akses admin dan user.
4. Persyaratan Sistem

Agar bisa memenuhi kebutuhan dari pengguna, mempunyai spesifikasi sebagai berikut:

1. Spesifikasi Fungsional Sistem
2. Sistem menyediakan fitur untuk mengupload berita agar dapat disajikan secara online.
3. Terdapat admin dan user yang memiliki hak akses yang berbeda.
4. Spesifikasi Non Fungsional Sistem
5. Aplikasi pesantren yang dibuat berbasis web dengan menggunakan *framework* *Laravel* dan *database* menggunakan *MySQL.*
6. Perangkat komputer yang dibutuhkan untuk aplikasi yang dibuat pertama untuk yang nantinya akan mengembangkan program (pengembangan) dan orang yang hanya akan menggunakan nya saja (*user*).
7. Sisi pengembang

Perangkat : Laptop Asus atau lainnya.

Processor : Inter Core i5 atau AMD A10

Ram : 4Gb

1. Sisi Pemakai

Perangkat : Laptop Asus atau lainnya.

Processor : Intel Core i5 atau AMD A10.

Ram : 4GB DDR4

1. Persyaratan Pengembangan
2. Menggunakan *usecase* diagram untuk menggambarkan interaksi antar pengguna dan sistem.
3. Perancangan database menggunakan MySQL.
4. Dalam melakukan pemrograman menggunakan framework laravel
5. Menggunakan metodologi dan tools yang tepat.

#### Identifikasi Aktor

Tahapan ini bertujuan untuk mengidentifikasi pengguna (aktor). Adapun aktor yang teridentifikasi dalam pembuatan aplikasi ini adalah Admin (petugas) dan user , dimana actor ini adalah orang yang akan menggunakan aplikasi. Adapun aktivitas yang dilakukan oleh aktor adalah sebagai berikut:

Tabel 4.1. Aktivitas Aktor

| **Aktor** | **Aktivitas** |
| --- | --- |
| Admin (Petugas) | 1. Melakukan *login* agar bisa masuk kedalam system aplikasi. 2. Mengelola Post/Informasi data pesantren. 3. Mengelola Tag. 4. Mengelola Kategori. 5. Melakukan *logout* untuk keluar dari system aplikasi. |
| *User* | Melihat informasi pesantren. |

#### Identifikasi Use Case Awal

Adapun *use case* pada aplikasi yang dibuat adalah :

*Login* Admin

Kelola *Post*

Kelola *Tag*

Kelola Kategori

Lihat Berita

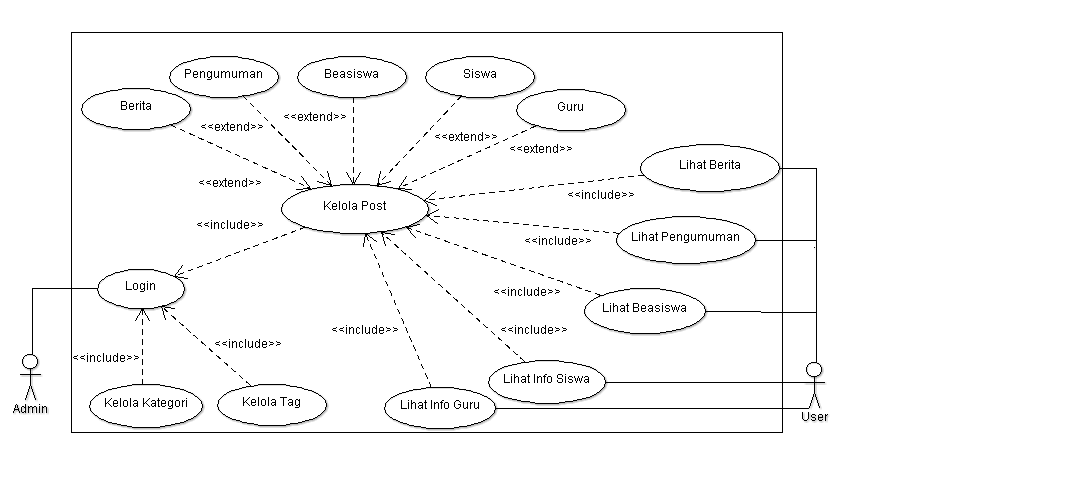
Lihat Pengumuman

Lihat Beasiswa

Lihat Info Siswa

Lihat Info Guru

#### Use Case

Berikut adalah *use case* diagram untuk aplikasi pesantren berbasis web : 

Gambar 4.3. *Use Case* Diagram Aplikasi Pesantren Berbasis Web

Adapun *scenario* dari *use* *case* diagram aplikasi pesantren berbasis web yang telah dibuat adalah :

Skenario *Use Case* Kelola *Post*

Tabel 4.2. Skenario Use Case Kelola Post

|  |  |  |
| --- | --- | --- |
| **Identifikasi** | | |
| Nama | Kelola Post | |
| Tujuan | Untuk mengelola *post/*data informasi pesantren. | |
| Aktor | Admin(Petugas) | |
| **Skenario** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| Admin memilih menu tambah post | |  |
|  | | Sistem menampilkan halaman tambah post |

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| 3. Admin memilih salah satu menambah, mengubah atau menghapus |  |
|  | 4. Sistem menampilkan form untuk menambah post |
| 5. Admin Mengisi form lalu mengklik tombol save |  |
|  | 6. Sistem menyimpan data yang telah ditambahkan |

Skenario *Use Case* Melihat Pengumuman

Tabel 4.3. Skenario *Use Case* Melihat Pengumuman

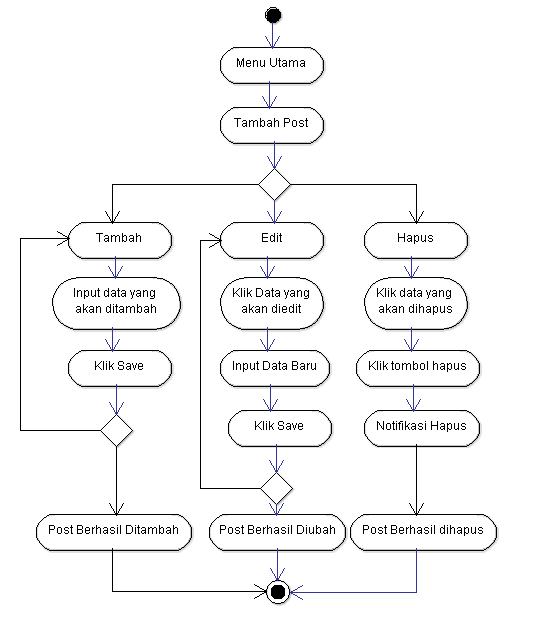
|  |  |  |
| --- | --- | --- |
| **Identifikasi** | | |
| Nama | Melihat Pengumuman | |
| Tujuan | Untuk melihat pengumuman yang ditampilkan pada web pesantren. | |
| Aktor | User dan Admin | |
| **Skenario** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| *User* mengklik pada kategori pengumuman | |  |
|  | | Sistem menampilkan halaman pengumuman berita yang dipilih *user* |
| User memilih pengumuman yang ingin dilihat lalu mengklik judul berita | |  |
|  | | Sistem menampilkan halaman pengumuman yang dipilih |

Adapun untuk tabel skenario *use case* lainnya dilampirkan pada **LAMPIRAN C**.

#### Activity Diagram

Diagram aktifitas ini dibuat berdasarkan skenario *use case* yang mengambarkan kerja admin atau user terhadap suatu sistem. Berikut adalah aktifitas diagram aplikasi pesantren berbasis *web* :

1. *Activity* *Diagram* Kelola *Post*



Gambar 4.4. *Activity Diagram* Kelola *Post*

Berikut adalah penjelasan mengenai deskripsi *activity diagram* kelola *post* :

Tabel 4.4. Deskripsi *Activity* *Diagram* Kelola *Post*

|  |  |  |
| --- | --- | --- |
| **No** | **Aktivitas** | **Deskripsi** |
| 1. | Menu Utama | Sistem menampilkan halaman utama aplikasi |
| 2. | Tambah *Post* | Admin mengklik menu tambah *post* kemudian sistem menampilkan halaman untuk tambah *post* |
| 3. | *Input* Data | Memasukkan data baru pada form isian agar data diubah atau di tambahkan. |
| **No** | **Aktivitas** | **Deskripsi** |
| 4. | Klik Data | Memilih dan mengklik data yang akan diubah atau dihapus. |
| 5. | Klik Save | Menyimpan hasil yang telah diubah pada sistem |
| 6. | Klik Hapus | Sistem akan menghapus data yang telah dipilih oleh admin |
| 7. | Notofikasi Hapus | Sistem menampilkan notifikasi hapus untuk memastikan data yang akan dihapus sesuai dengan yang diinginkan |
| 8. | Perintah Berhasil Dieksekusi | Sistem akan menampilkan notifikasi perintah berhasil dieksekusi saat mengklik tombol save atau hapus. |

1. *Activity Diagram* Melihat *Pengumuman*



Gambar 4.5. *Activity Diagram* Melihat Pengumuman

Berikut adalah penjelasan deskripsi *activity diagram* melihat pengumuman pesantren :

Tabel 4.5. Deskripsi *Activity* *Diagram* Melihat Pengumuman Pesantren

|  |  |  |
| --- | --- | --- |
| **No** | **Aktivitas** | **Deskripsi** |
| 1. | Menu utama | Sistem menampilkan halaman utama aplikasi |
| 2. | Pada kategori pilih dan klik pengumuman | Pada kategori user/admin memilih pengumuman dan sistem akan menampilkan halaman pengumuman |
| 3. | Klik judul pengumuman yang ingin dilihat | User/admin memilih pengumuman yang ingin dilihat kemudian klik judul pengumuman yang dipilih |
| 4. | *Single post* melihat pengumuman | Jika judul pengumumantelah diklik maka akan muncul halaman atau *single post* yang menampilkan pengumuman. |

Adapun untuk gambar tampilan *activity diagram* lainnya dilampirkan pada **LAMPIRAN C**.

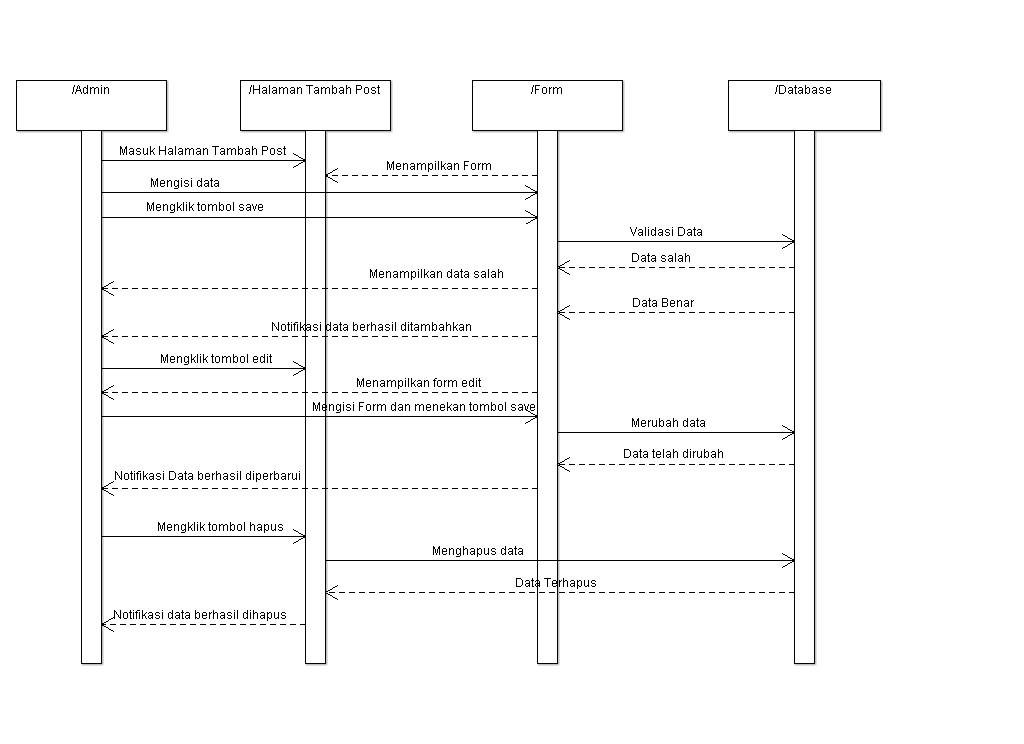
### Elaboration

Pada tahap ini merupakan tahap lanjutan dari *inception,* yaitu tahapan nalisis dan desain arsitektur.

#### Sequence Diagram

*Sequence diagram* digunakan untuk menggambarkan interaksi sistem dengan aktor. Berikut *sequence diagram* untuk aplikasi pesantren berbasis web :

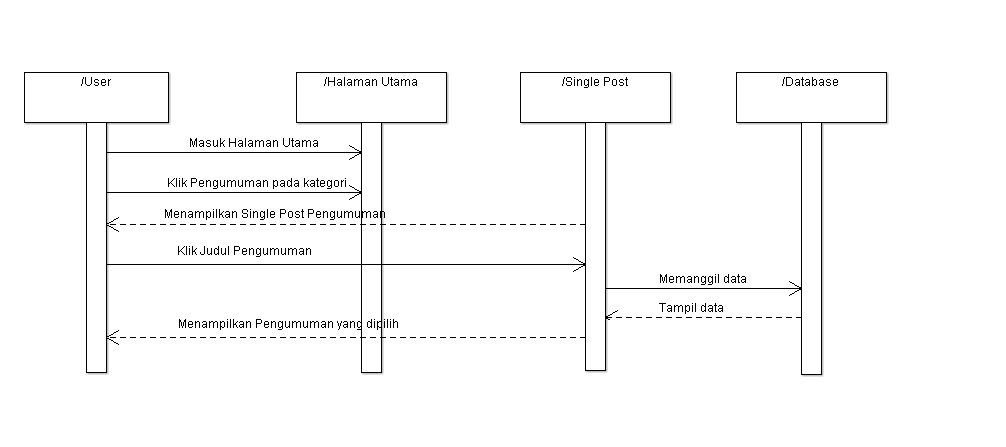
*Sequence Diagram* Kelola *Post*



Gambar 4.6. *Sequence Diagram* Kelola *Post*

Pada *sequence diagram* kelola *post* yang telah digambarkan, menjelaskan bagaimana interaksi antara admin dengan sistem ketika admin sedang mengelola *post*, dimulai dari menambah post, mengedit post sampai dengan menghapus post. Setiap perubahan yang yang terjadi sistem akan memberikan notifikasi kepada admin bahwa perubahan yang terjadi telah berhasil dieksekusi.

*Sequence Diagram* Melihat Pengumuman



Gambar 4.7. *Sequence Diagram* Melihat Pengumuman

Pada gambar 4.7 dijelaskan bagaimana interaksi yang terjadi antara user dengan sistem ketika melihat pengumuman.

Adapun untuk gambar *sequence diagram* lainnya dilampirkan pada **LAMPIRAN C**.

#### Class Diagram

*Class* *Diagram* dibuat agar pembuatan program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron. Berikut merupakan *class* dari aplikasi pesantren berbasis web :



Gambar 4.8. *Class Diagram* Aplikasi Pesantren Berbasis Web

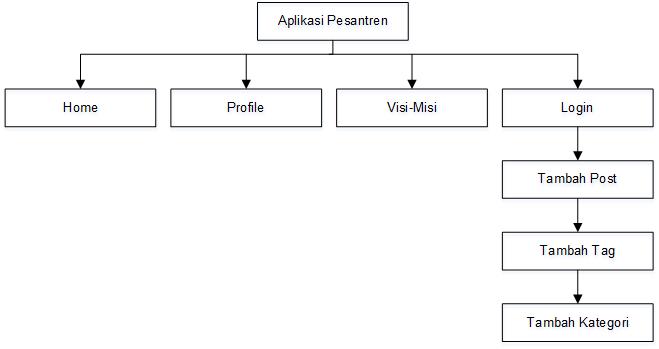
### Construction

Pada tahap ni adalah lanjutan dari *elaboration,* untuk merancangan struktur menu aplikasi, *interface* dan pengimplementasian kedalam bahasa pemrograman.

#### Perancangan Layout

a. Membuat Rincian Sistem

Pada rician sistem ini akan digambarkan berupa struktur menu seperti pada gambar berikut :

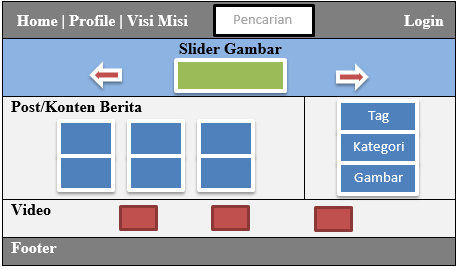


Gambar 4.9. Struktur Menu Aplikasi Pesantren

b. Membuat Rancangan *Layout*

Adapun rancangan *layout* aplikasi pesantren ini dapat dilihat pada gambar berikut :

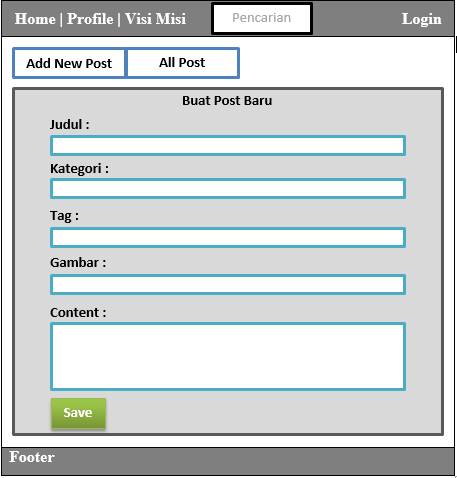
Halaman Utama



Sejarah

Gambar 4.10. Tampilah Halaman Utama

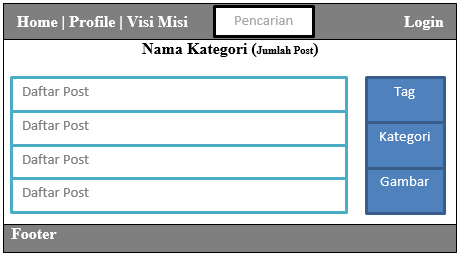
Halaman Kelola *Post*



**Sejarah**

Gambar 4.11. Tampilan Halaman Tambah *Post*

Halaman Single Kategori



**Sejarah**

Gambar 4.12. Tampilan Halaman Single Kategori

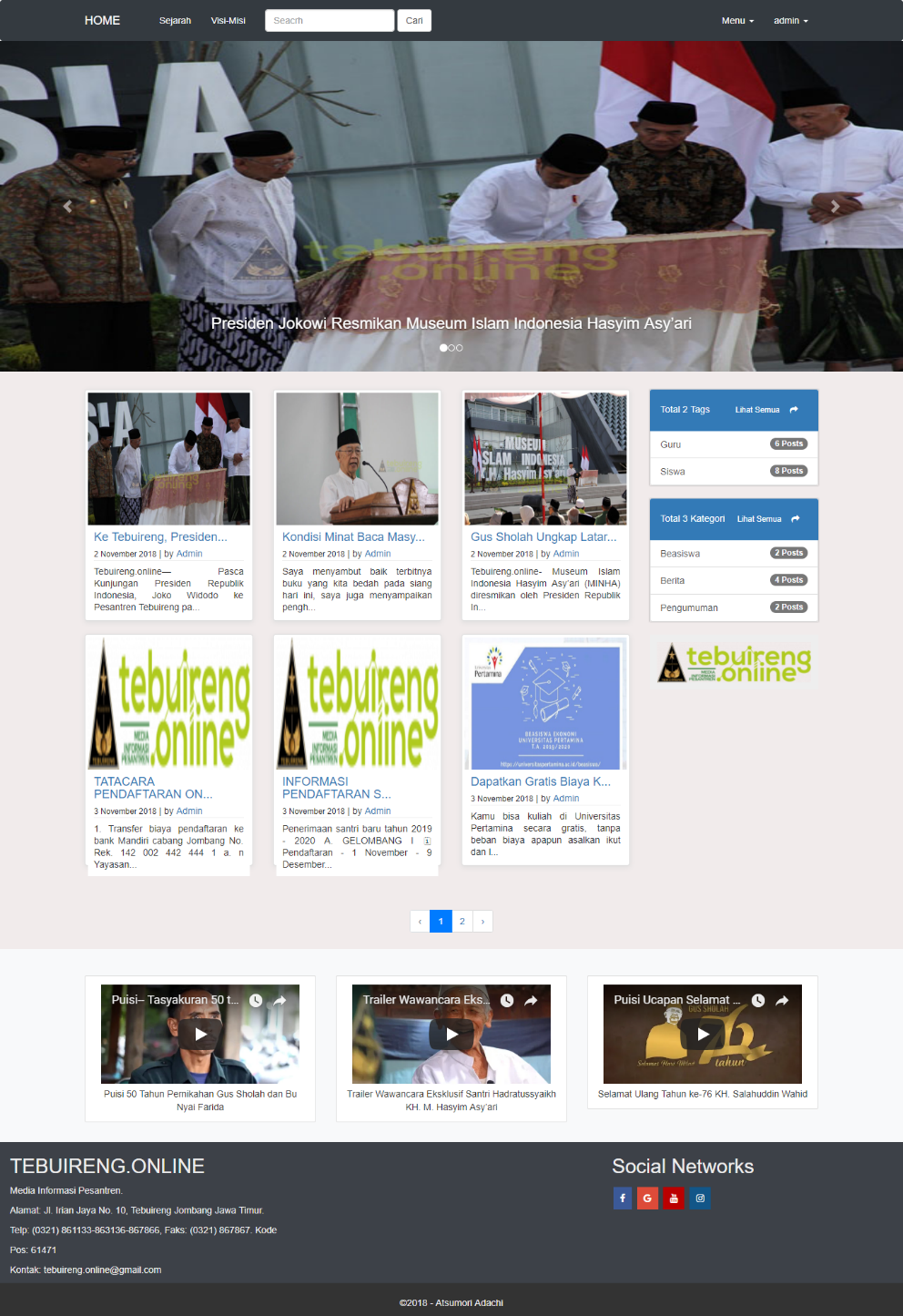
#### Rancangan Aplikasi Versi Beta

Aplikasi yang telah dibuat berjalan menggunakan perangkat lunak sebagai berikut :

1. Database menggunakan MySQL
2. Visual Studio Code sebagai text editor

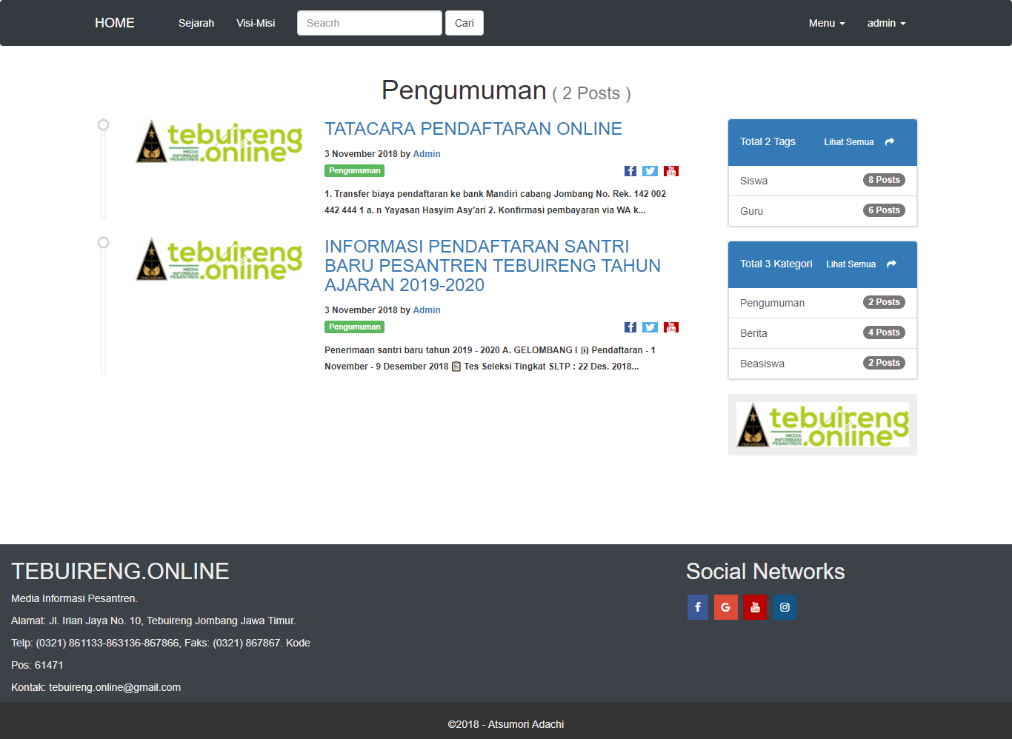
Adapun untuk tampilan halam utama/home pada aplikasi pesantren dapat dilihat pada gambar berikut :

* 1. Tampilan Halaman Utama



Gambar 4.13. Tampilan Halaman *Home*

* 1. Tampilan Halaman *Single* Kategori



Gambar 4.14. Tampilan Halaman Kategori

Adapun untuk tampilan lainnya dari aplikasi pesantren ini dilampirkan pada **LAMPIRAN C**.

### Pengujian

Adapun pengujian yang digunakan untuk menguji fitur-fitur dan fungsi aplikasi pesantren ini adalah *blaxk-box testing.*

Tabel 4.6. Form Pengujian Aplikasi Pesantren Berbasis *Web*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Aktifitas** | **Kelas Uji** | **Skenario Uji** | **Hasil yang diharapkan** | **Kesimpulan** |
| *Login* | Melakukan *login* yang benar | Masukan *username* dan *password* yang benar | *Login* Berhasil | Sesuai |
| Melakukan *Login* yang salah | Masukan *username* dan *password* yang salah | *Login* gagal | Sesuai |
| *Logout* | Melakukan *logout* | Mengklik menu *logout* | Keluar dari status admin | Sesuai |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Aktifitas** | **Kelas Uji** | **Skenario Uji** | **Hasil yang diharapkan** | **Kesimpulan** |
| Tambah Post | Mengisi form tambah post | Menambah data | Pengisian form berfungsi dengan baik | Sesuai |
| Edit data | Memilih data yang akan dirubah | Data akan berubah sesuai inputan | Sesuai |
| Hapus data | Memilih data yang akan dihapus | Data berhasil dihapus | Sesuai |
| Tambah Tag | Mengisi form tambah post | Menambah data | Pengisian form berfungsi dengan baik | Sesuai |
| Edit data | Memilih data yang akan dirubah | Data akan berubah sesuai inputan | Sesuai |
| Hapus data | Memilih data yang akan dihapus | Data berhasil dihapus | Sesuai |
| Tambah Kategori | Mengisi form tambah post | Menambah data | Pengisian form berfungsi dengan baik | Sesuai |
| Edit data | Memilih data yang akan dirubah | Data akan berubah sesuai inputan | Sesuai |
| Hapus data | Memilih data yang akan dihapus | Data berhasil dihapus | Sesuai |

Dari hasil pengujian *black-box testing* yang dilakukan pada aplikasi pesantren, fitur-fitur didalamnya dapat berjalan sesuai dengan fungsinya.

## Pengetahuan dan Keterampilan Empiris

*Laravel* adalah salah satu *framework* PHP yang digunakan programmer dalam membangun aplikasi berbasis web yang diterapkan pada perusahaan MOBIDU. Namun *laravel* masih asing untuk mahasiswa Teknik Informatika STT-Garut karena tidak dipelajari dan dikenalkan pada saat perkuliahan. Sehingga mahasiswa yang melaksanakan kerja praktek di MOBIDU diberikan tutorial dan arahan dalam membuat aplikasi berbasis *web* dengan menggunakan *framework laravel.* Pelaksanaan kerja praktik ini pun menambah pengetahuan yang sangat berharga untuk mahasiswa karena dapat mempelajari *framework* yang baru.

## Etika Kerja

Perusahaan MOBIDU memiliki lingkungan pekerjaan yang ramah dan memberikan kebebasan dalam berkreativitas dan berinovasi untuk meningkatkan kemampuan mahasiswa pada saat melaksanakan kerja praktik. Bekerja santai namun harus dapat selesai sesuai dengan target yang ditentukan, tidak ada patokan bekerja di kantor atau di rumah untuk mahasiswa yang melakukan kerja praktik maupun karyawan perusahaannya, sehingga tidak merasa tertekan dan dapat bekerja secara maksimal. Etika kerja dalam beribadah pun diberikan kebebasan misalnya dalam melaksanakan shalat wajib dapat dilakukan di masjid terdekat.

# KESIMPULAN DAN SARAN

## Kesimpulan

Aplikasi Pesantren Berbasis web ini dibangun dengan menggunakan *framework laravel* dengan *database* MySQL. Pada pengujiannya menggunakan metode *black box testing.* Adapun kesimpulan dari apa yang telah dikerjakan diantaranya adalah aplikasi pesantren berbasis web yang dibuat dapat mempermudah dalam menyampaikan informasi kepada santri, guru maupun masyarakat umum yang ingin mendapatkan informasi mengenai suatu pesantren.

## Saran

Berdasarkan kesimpulan yang telah dibuat, adapun saran untuk mengembangkan aplikasi agar menjadi lebih baik yaitu membuat versi *mobile* sehingga aplikasi ini nantinya dapat diakses dimana saja dengan lebih mudah.

# DAFTAR PUSTAKA

Aminudin. (2015). *Cara Efektif Belajar Framework Laravel.* (L. Hakim, Penyunt.) Yogyakarta: Lokomedia.

Kadir, A. (2002). *Dasar Pemrograman WEB Dinamis Menggunakan PHP* (2 ed.). Yogyakarta: ANDI.

Kadir, A. (2005). *Dasar pemrograman WEB dengan ASP.* Yogyakarta: ANDI.

Kruchten, P. (2004). *The Rational Unified Process : An Introduction, Third Edtion.* Addison-Wesley Professional.

Mulyani, S. (2016). *Metode Analisis dan Perancangan Sistem.* Bandung: Abdi Sistematika.

Nugroho, B. (2005). *Database Relasional dengan MySQL.* Yogyakarta: ANDI.

Prabawati, T. A. (Penyunt.). (2010). *Panduan Praktis Adobe Ilustrator CS 4* (1 ed.). Yogyakarta: ANDI,WAHANA KOMPUTER.

Pressman, R. S. (2012). *Rekayasa Perangkat Lunak* (7 ed.). (D. Hardjono, Penyunt.) Yogyakarta: ANDI.

Saputra, A., Subagio, R. T., & Saluky. (2012). *Membangun Aplikasi E-Library untuk panduan Skripsi.* Jakarta: PT Elex Media Komputindo.

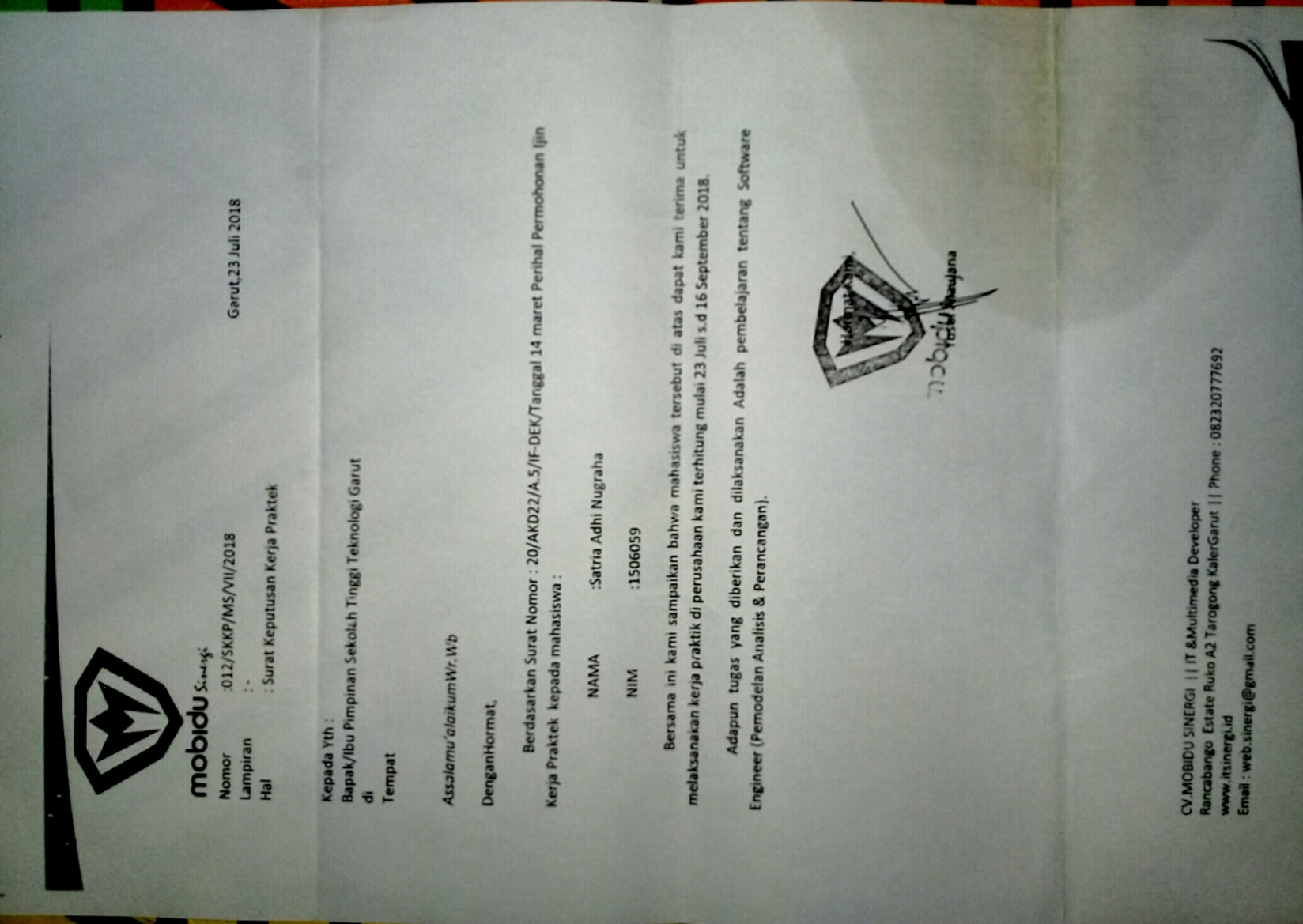
Sukamto, R. A., & Shalahuddin, M. (2016). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek.* Bandung: Informatika Bandung.

Triandini, E., & Suardika, I. G. (2012). *Step By Step Desain Proyek Menggunakan UML.* Yogyakarta: CV. ANDI OFFSET.

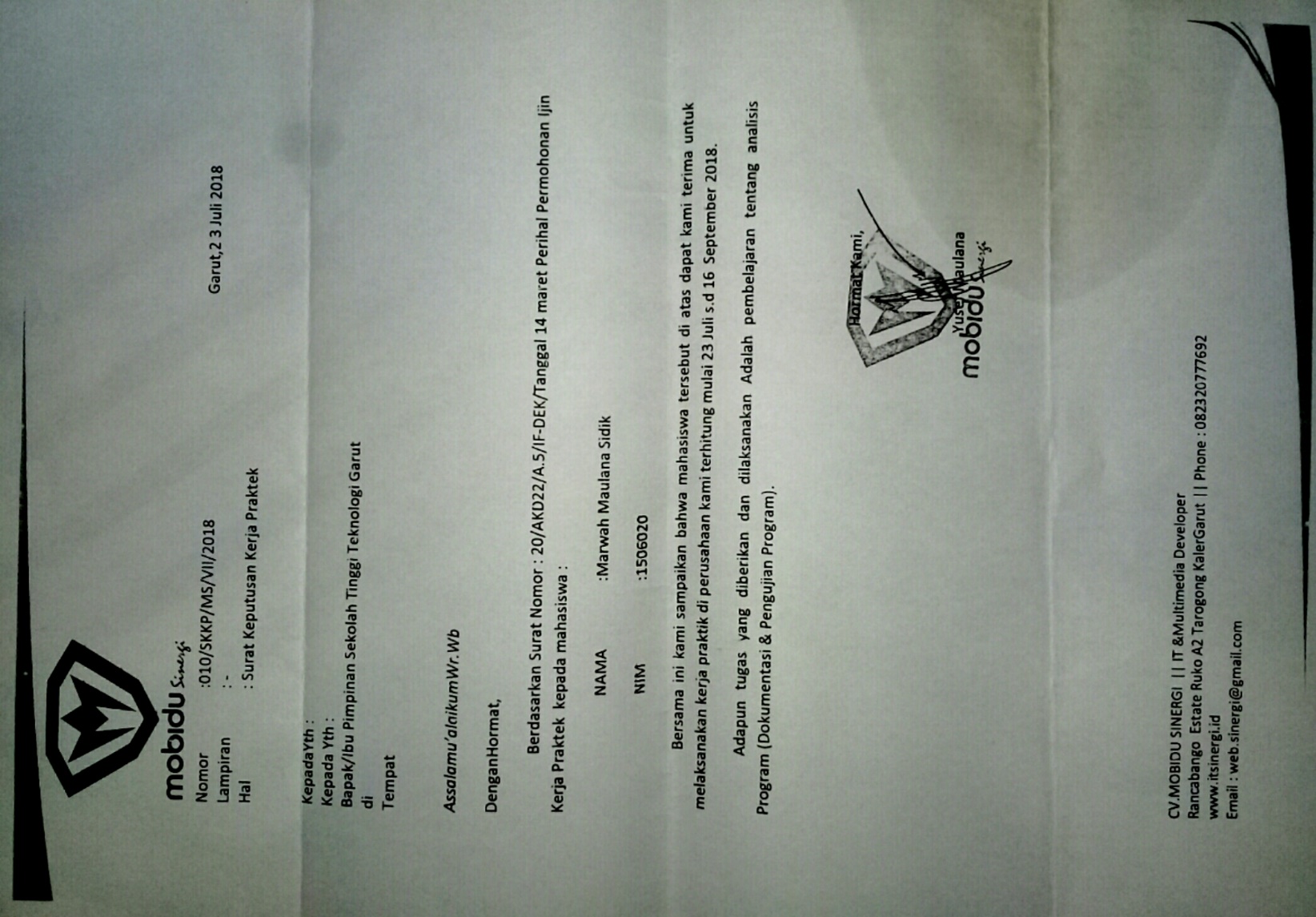
Zuhriy, M. S. (2011). Budaya Pesantren Dan Pendidikan Karakter. 2.

# LAMPIRAN A : LANDASAN PEKERJAAN

Surat Balasan 1

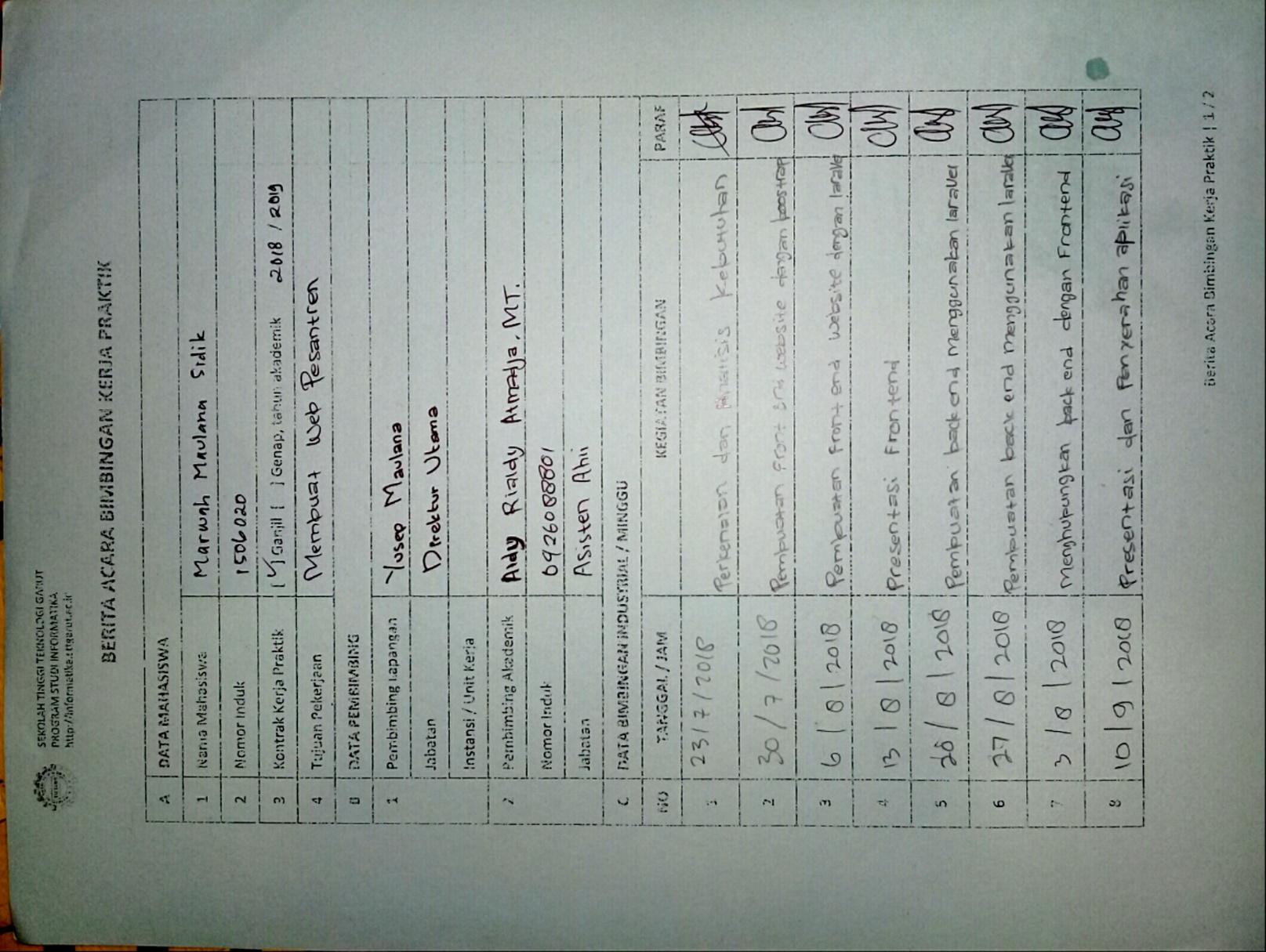


Surat Balasan 2

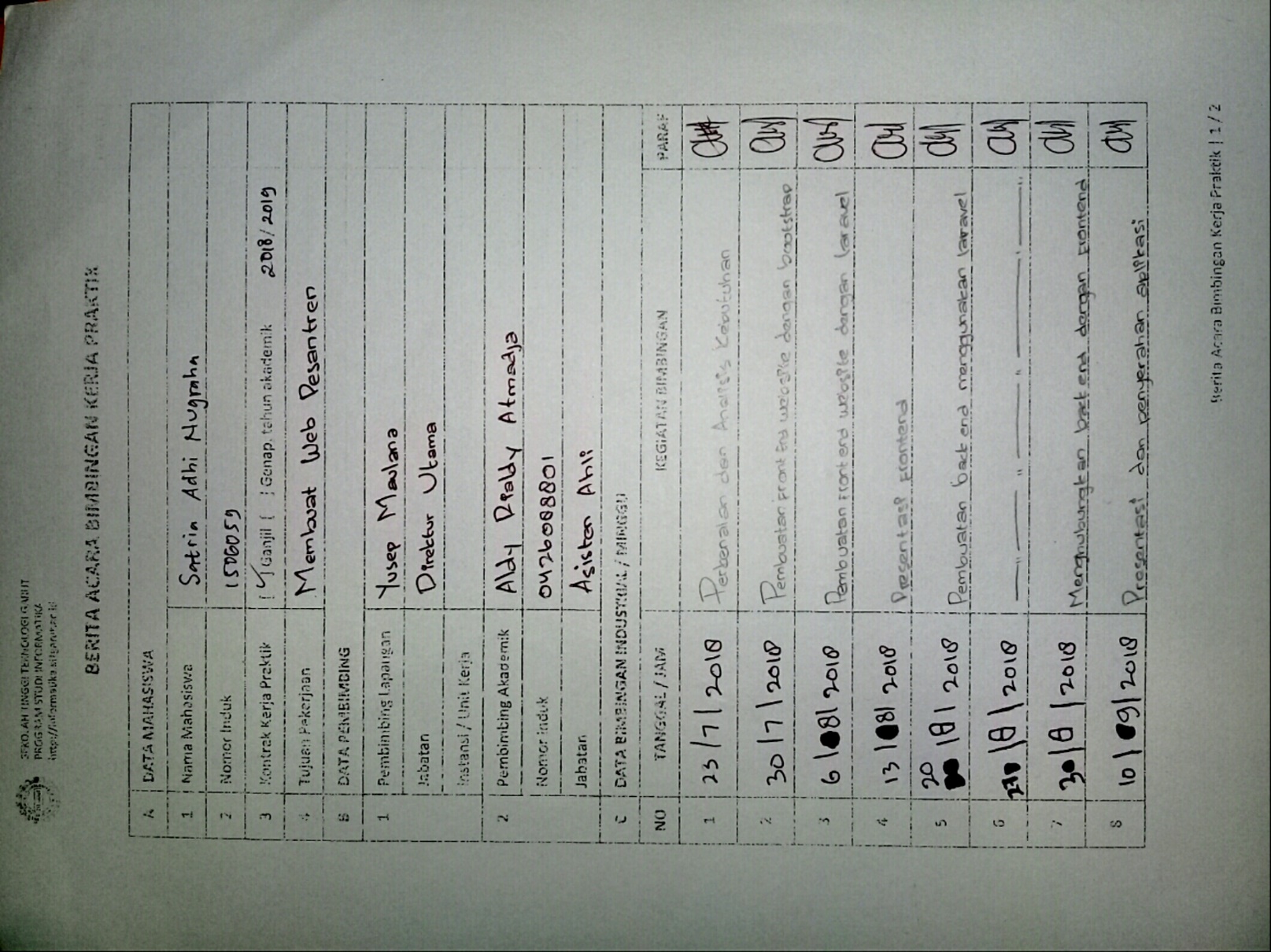


# LAMPIRAN B : PRESENSI KERJA

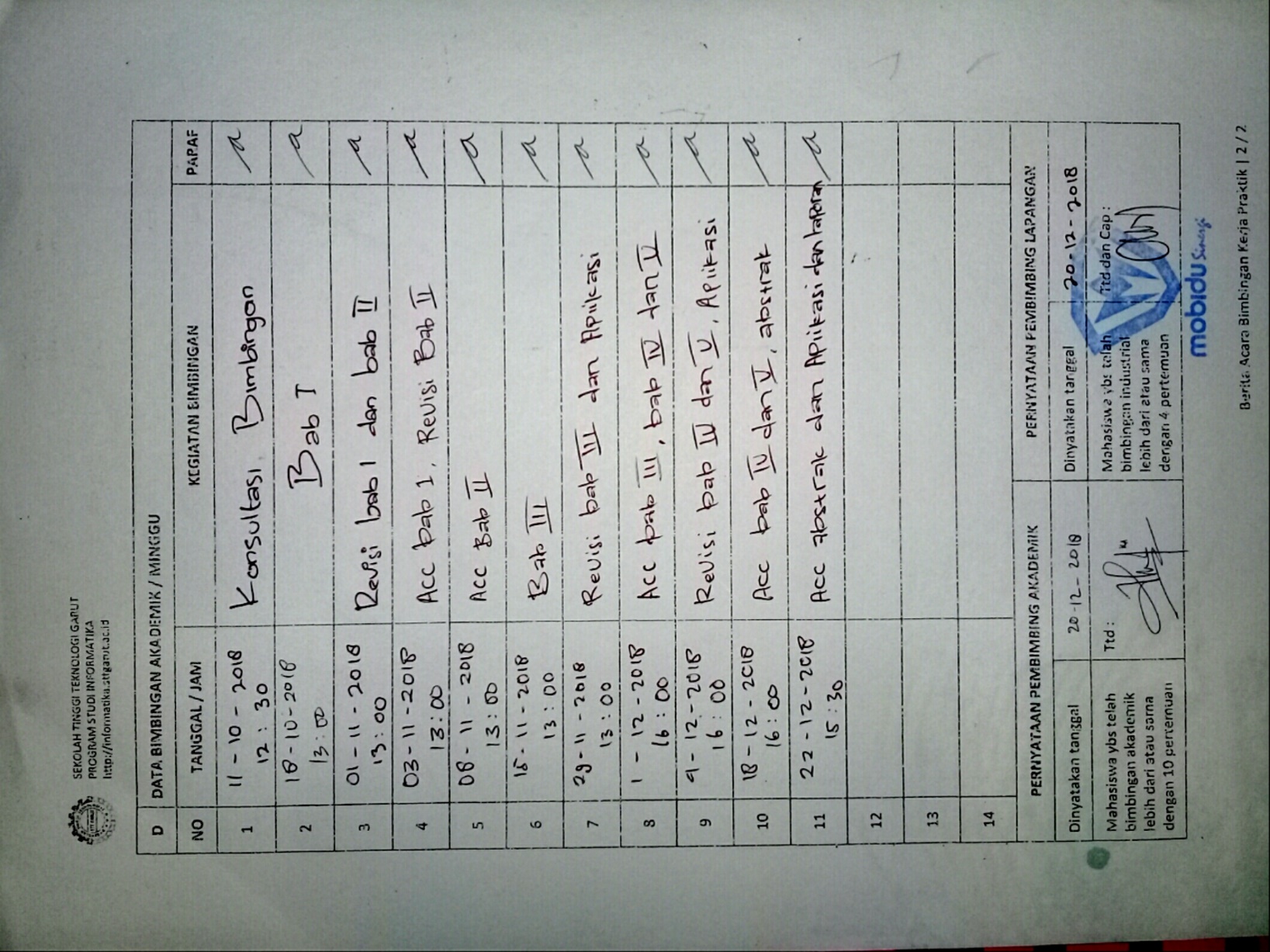
Kartu Bimbingan Lapangan Marwah



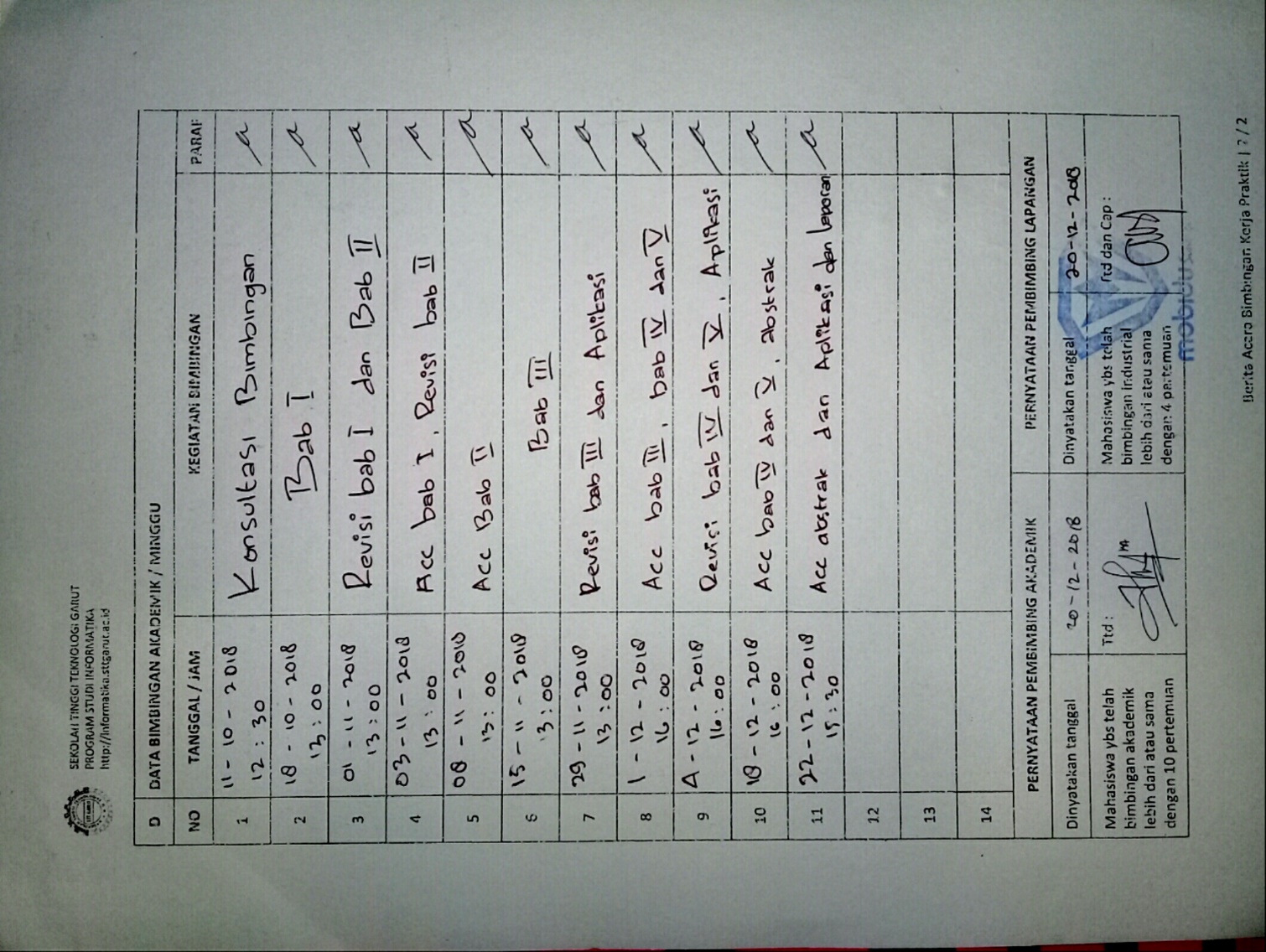
Kartu Bimbingan Lapangan Satria



Kartu Bimbingan Akademik Marwah



Kartu Bimbingan Akademik Satria



# LAMPIRAN C : HASIL PEKERJAAN

Skenario Use Case

a. Skenario *Use Case Login*

|  |  |  |
| --- | --- | --- |
| **Identifikasi** | | |
| Nama | Login | |
| Tujuan | Untuk melakukan login agar dapat mengelola aplikasi | |
| Aktor | Admin(Petugas) | |
| **Skenario** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| Admin membuka aplikasi dan mengklik tombol *login* | |  |
|  | | Sistem menampilkan halaman *login* |
| Admin mengisi username dan password lalu mengklik tombol *login* | |  |
|  | | Sistem memvalidasi username dan password. |
| Sistem menampilkan halaman utama |

b. Skenario *Use Case* Kelola Tag

|  |  |  |
| --- | --- | --- |
| **Identifikasi** | | |
| Nama | Kelola Tag | |
| Tujuan | Untuk mengelola Tag. | |
| Aktor | Admin(Petugas) | |
| **Skenario** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. Admin memilih menu tambah tag | |  |
|  | | 2. Sistem menampilkan halaman tambah tag |

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| 3. Admin memilih salah satu menambahkan, mengubah atau menghapus |  |
|  | 4. Sistem menampilkan form untuk mengubah tag |
| 5. Mengisi form lalu mengklik tombol save |  |
|  | 6. Sistem menyimpan data yang telah diubah |

c. Skenario *Use Case* Kelola Kategori

|  |  |  |
| --- | --- | --- |
| **Identifikasi** | | |
| Nama | Kelola Kategori | |
| Tujuan | Untuk mengelola Kategori. | |
| Aktor | Admin(Petugas) | |
| **Skenario** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| Admin memilih menu tambah kategori | |  |
|  | | 1. Sistem menampilkan halaman tambah kategori |
| 1. Admin memilih salah satu menambahkan, mengubah atau menghapus | |  |
| 1. Admin mengklik tombol hapus pada kategori yang ingin dihapus | |  |
|  | | 1. . Sistem menampilkan notifikasi hapus |
| 1. Admin mengklik tombol hapus | |  |
|  | | 1. Sistem menghapus kategori yang dipilih |

d. Skenario *Use Case* Melihat Berita

|  |  |  |
| --- | --- | --- |
| **Identifikasi** | | |
| Nama | Melihat Berita | |
| Tujuan | Untuk melihat berita yang ditampilkan pada web pesantren. | |
| Aktor | User dan Admin | |
| **Skenario** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. *User* mengklik pada kategori berita | |  |
|  | | 1. Sistem menampilkan halaman kategori berita yang dipilih *user* |
| 1. User memilih berita yang ingin dilihat lalu mengklik judul berita | |  |
|  | | 1. Sistem menampilkan halaman berita yang dipilih |

e. Skenario *Use Case* Melihat Beasiswa

|  |  |  |
| --- | --- | --- |
| **Identifikasi** | | |
| Nama | Melihat Beasiswa | |
| Tujuan | Untuk melihat beasiswa yang ditampilkan pada web pesantren. | |
| Aktor | User dan admin | |
| **Skenario** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| *1. User* mengklik pada kategori beasiswa | |  |
|  | | 2. Sistem menampilkan halaman kategori beasiswa yang dipilih *user* |
| 3. User memilih beasiswa yang ingin dilihat lalu mengklik judul berita | |  |
|  | | 4. Sistem menampilkan halam beasiswa yang dipilih |

f. Skenario *Use Case* Melihat Info Siswa

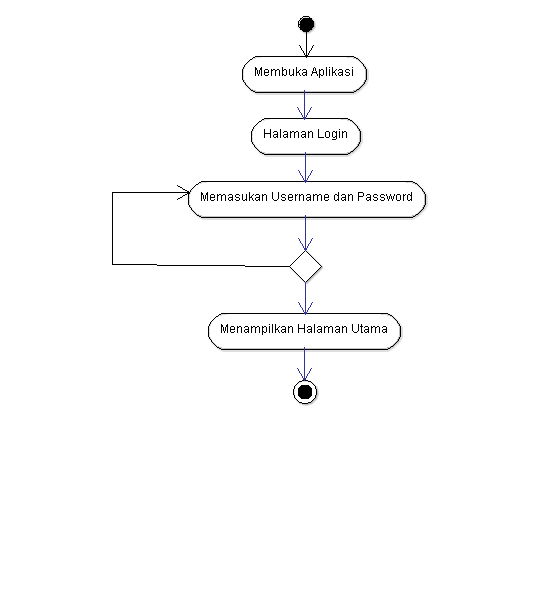
|  |  |  |
| --- | --- | --- |
| **Identifikasi** | | |
| Nama | Melihat Info Siswa | |
| Tujuan | Untuk melihat Info Siswa yang ditampilkan pada web pesantren. | |
| Aktor | User dan admin | |
| **Skenario** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. *User* mengklik pada tag siswa | |  |
|  | | 1. Sistem menampilkan halaman tag siswa yang dipilih *user* |
| 1. User memilih info siswa yang ingin dilihat lalu mengklik judul berita | |  |
|  | | 1. Sistem menampilkan halam info siswa yang dipilih |

g. Skenario *Use Case* Melihat Info Guru

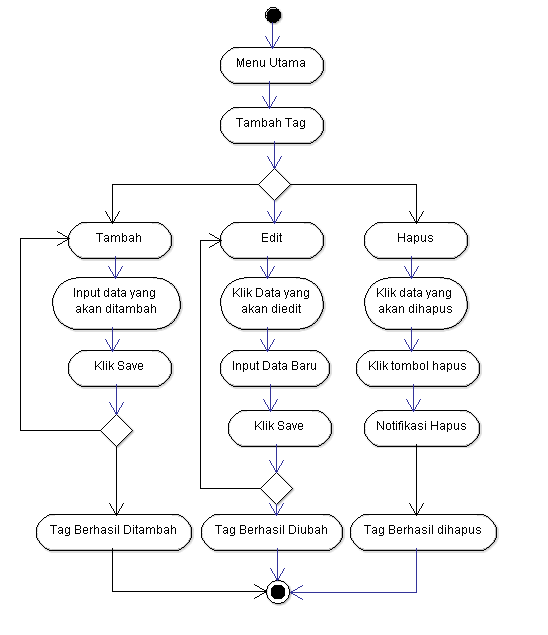
|  |  |  |
| --- | --- | --- |
| **Identifikasi** | | |
| Nama | Melihat Info Guru | |
| Tujuan | Untuk melihat info Guru yang ditampilkan pada web pesantren. | |
| Aktor | User dan admin | |
| **Skenario** | | |
| **Aksi Aktor** | | **Reaksi Sistem** |
| 1. *User* mengklik pada tag guru | |  |
|  | | 1. Sistem menampilkan halaman tag guru yang dipilih *user* |
| 1. User memilih info guru yang ingin dilihat lalu mengklik judul berita | |  |
|  | | 1. Sistem menampilkan halam berita yang dipilih |

2. Activity Diagram

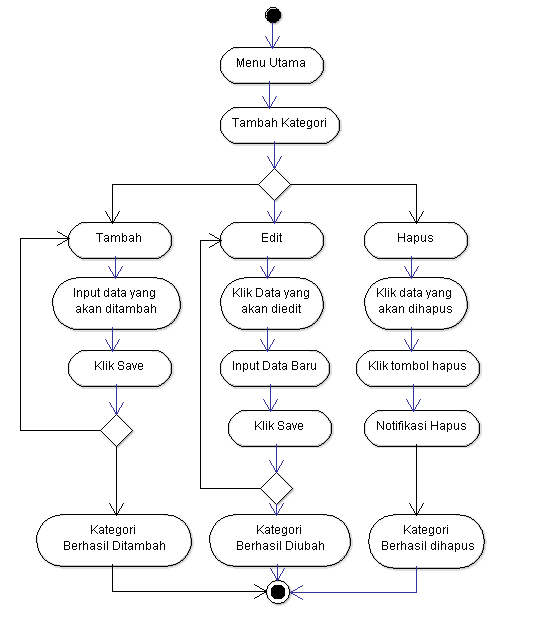
1. *Activity diagram* *login*



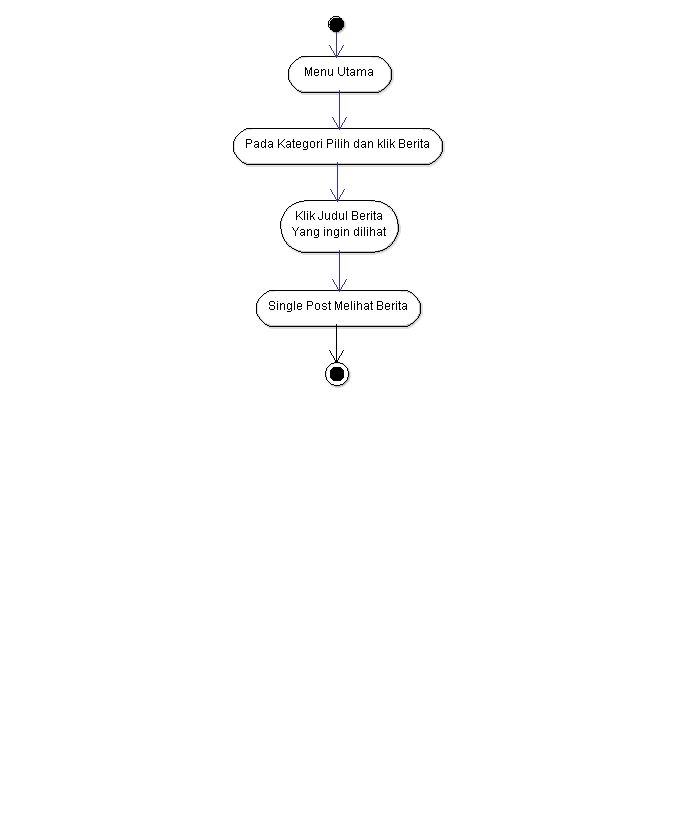
b*. Activity Diagram* Tag



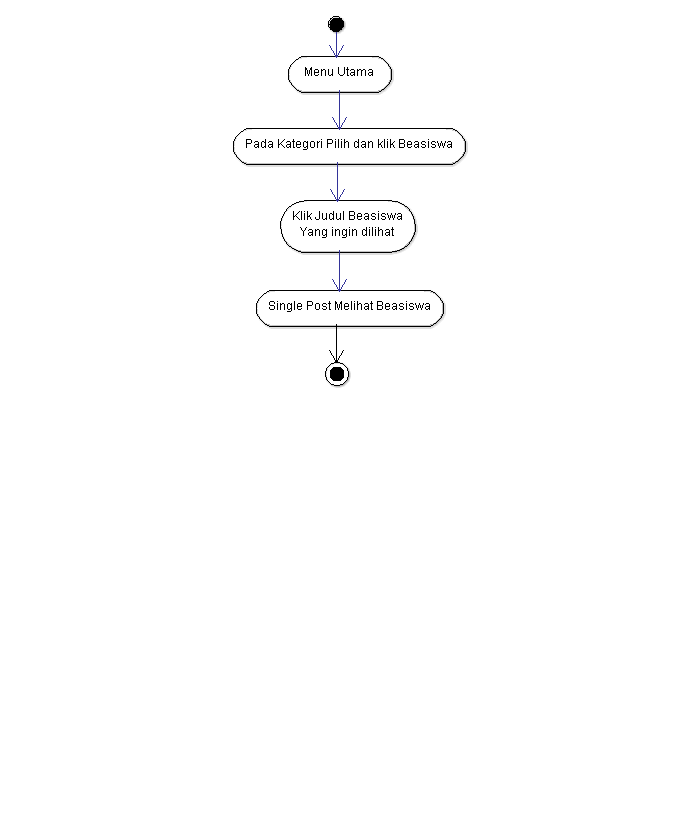
c*. Activity Diagram* Kelola Kategori



d. *Activity Diagram* Melihat Berita



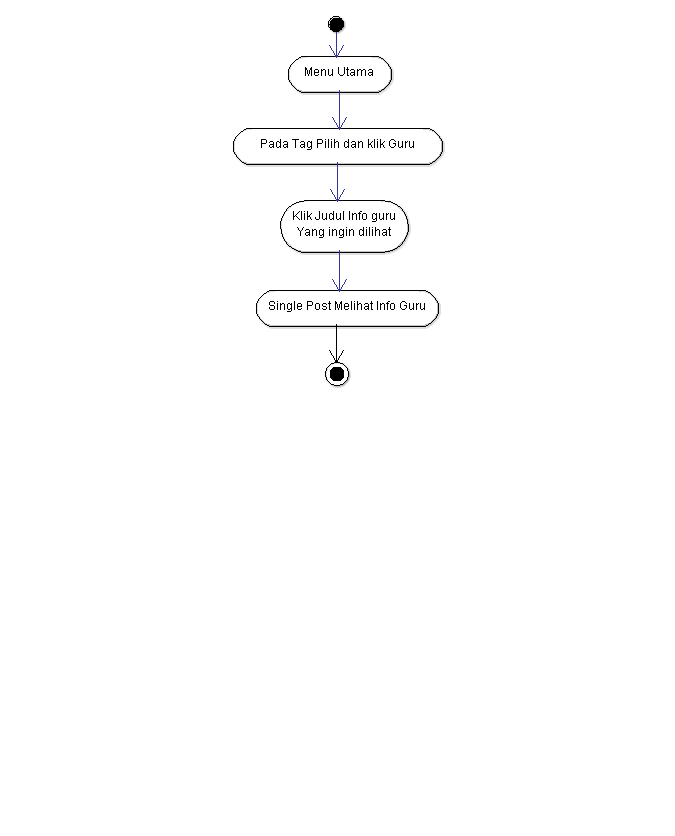
e*. Activity Diagram* *Melihat* Beasiswa



f. *Activity Diagram* Melihat Info Siswa

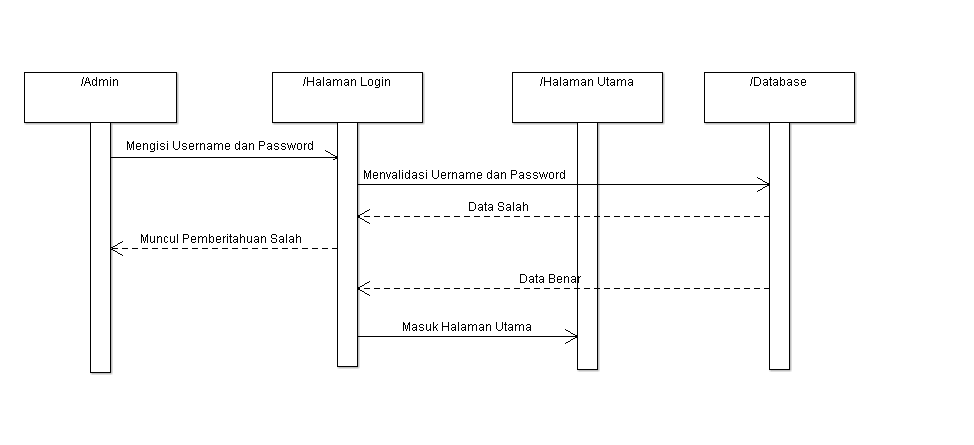


g. *Activity Diagram* Melihat Info Guru

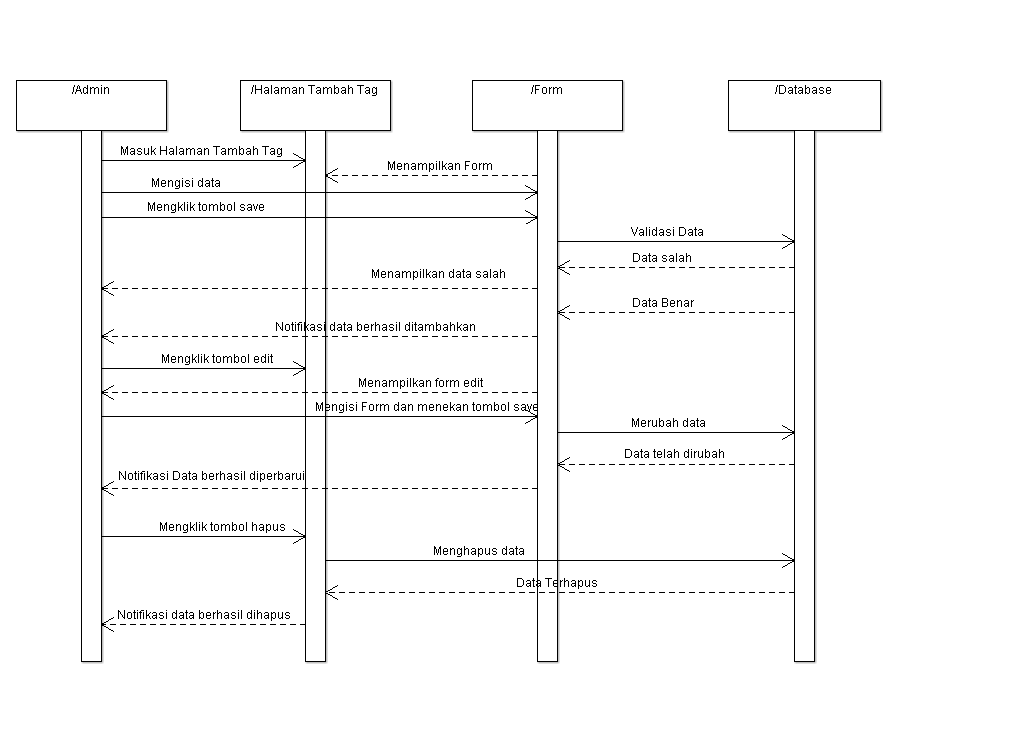


**3. Sequence Diagram**

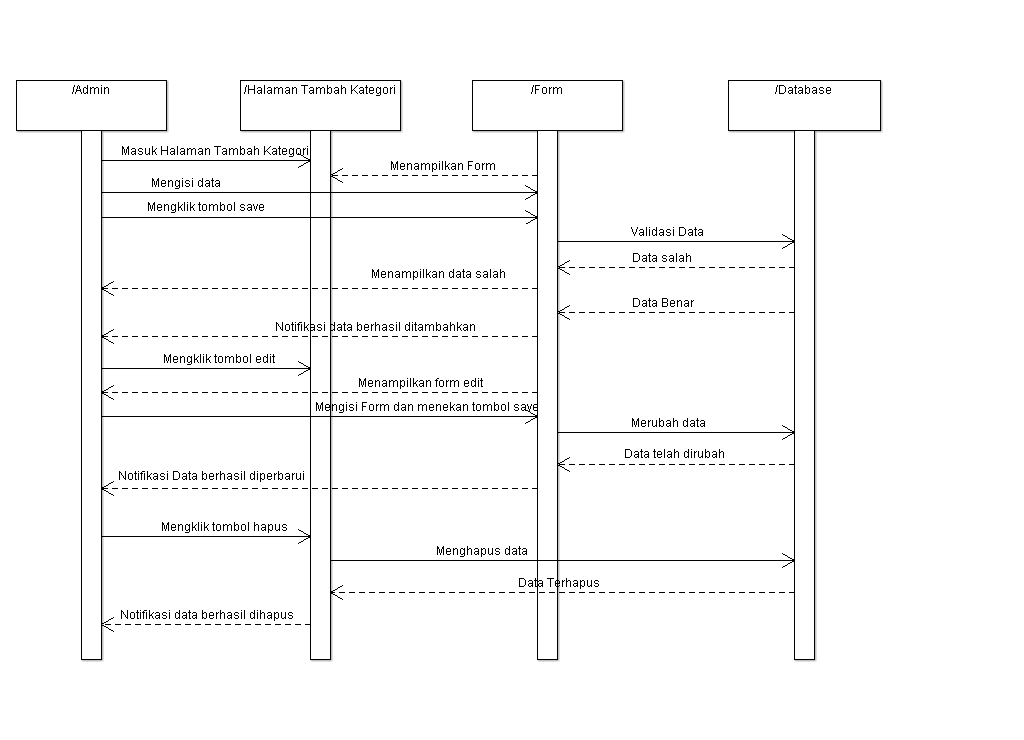
1. *Sequence Diagram Login*



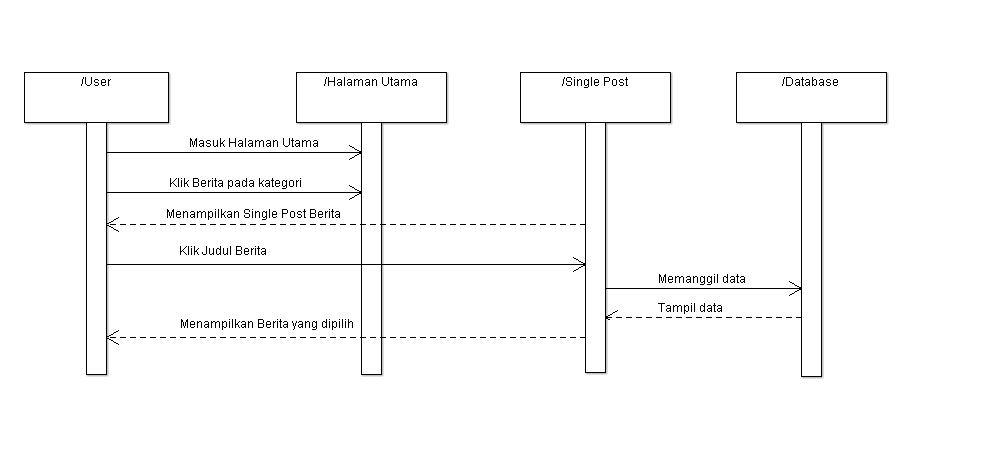
1. *Sequence Diagram Kelola Tag*



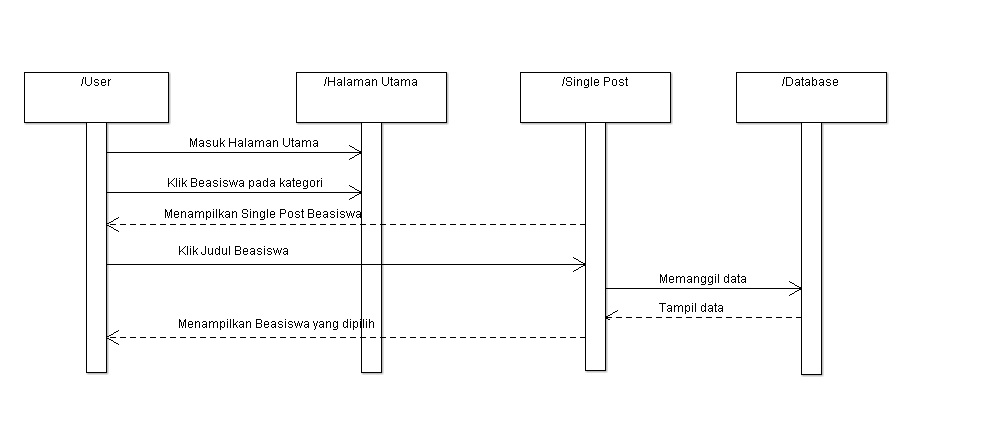
1. *Sequence Diagram* Kelola Kategori



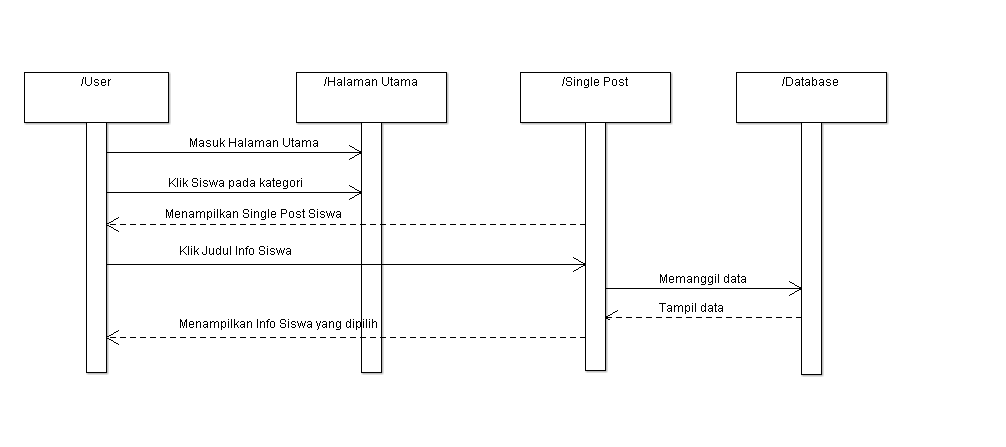
1. *Sequence Diagram* Melihat Berita



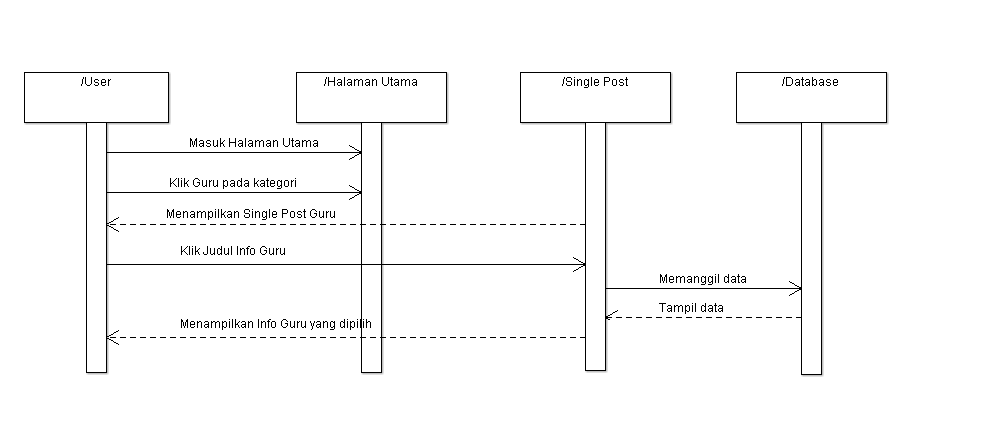
e*. Sequence Diagram* Melihat Beasiswa



f. *Sequence Diagram* Melihat Info Siswa

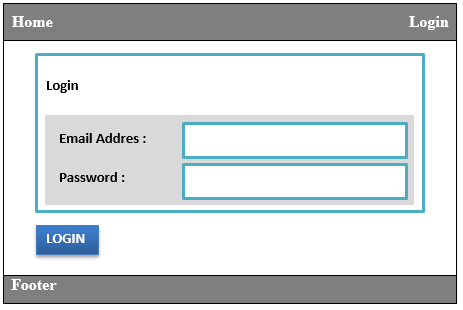


g. *Sequence Diagram* Melihat Info Guru



4. Rancangan Layout

a. Halaman *Login*

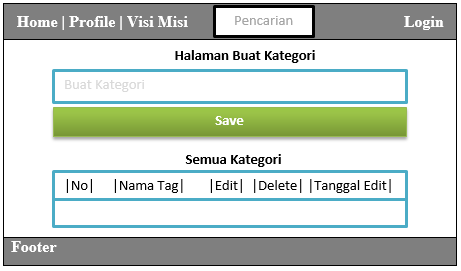


b. Halaman Kelola Tag



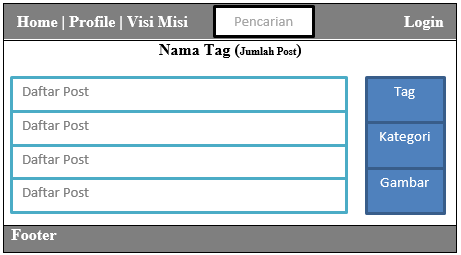
**Sejarah**

c. Halaman Tambah Kategori



**Sejarah**

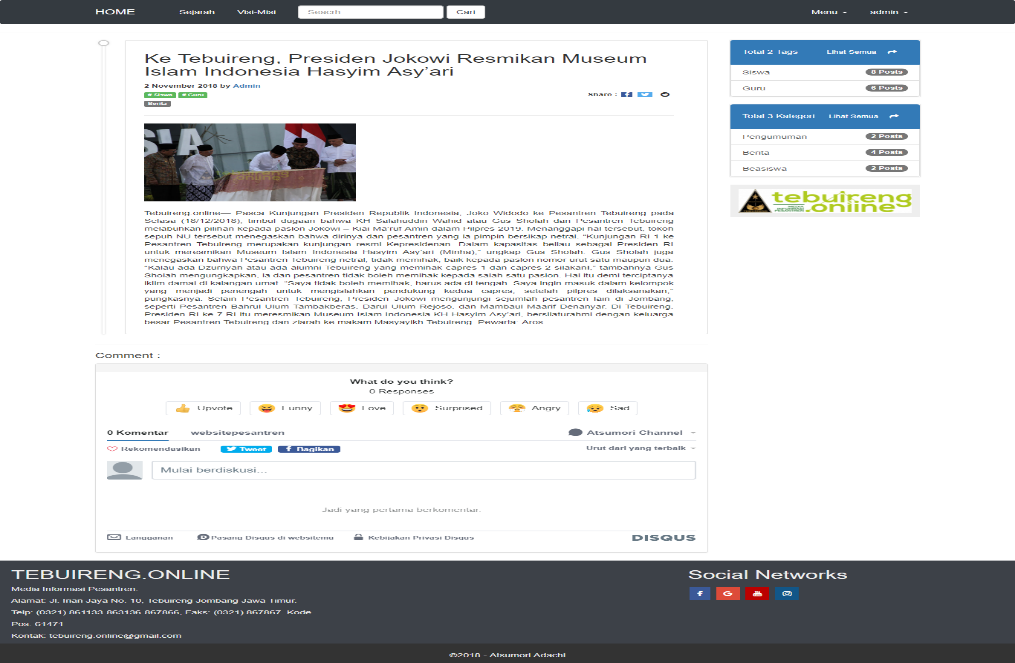
d. Halaman Single Tag



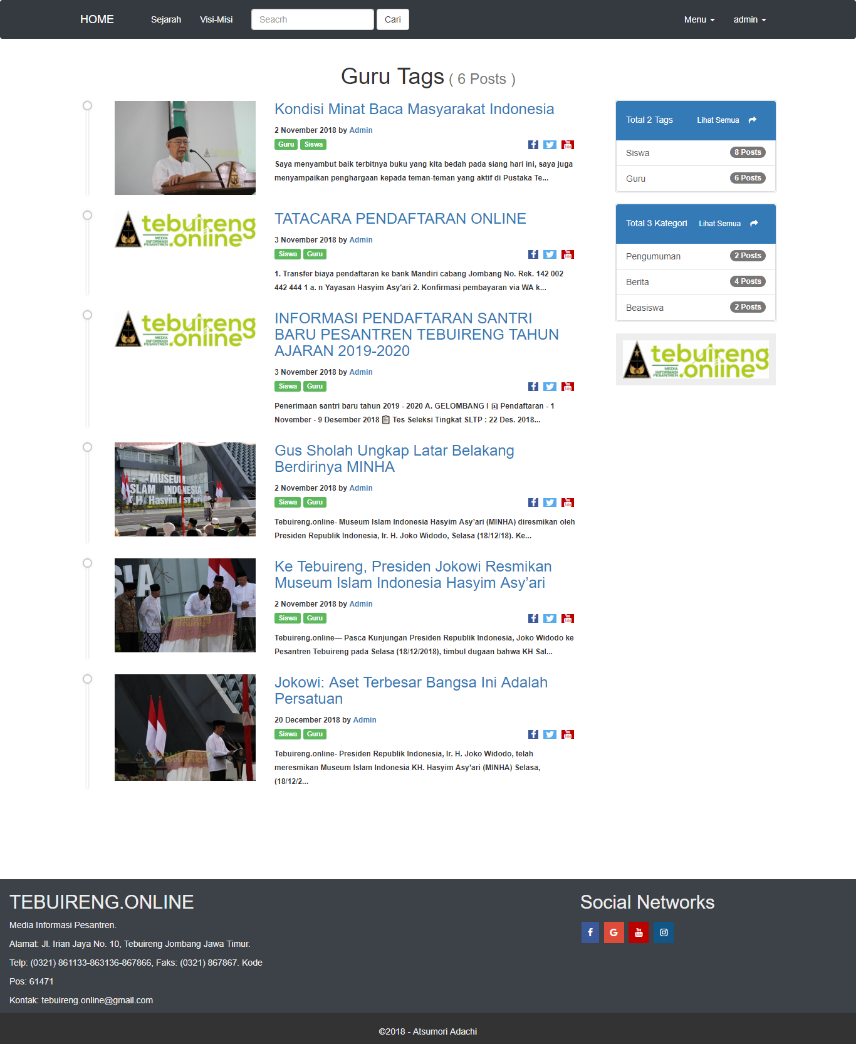
**Sejarah**

5. Tampilan Aplikasi Versi Beta

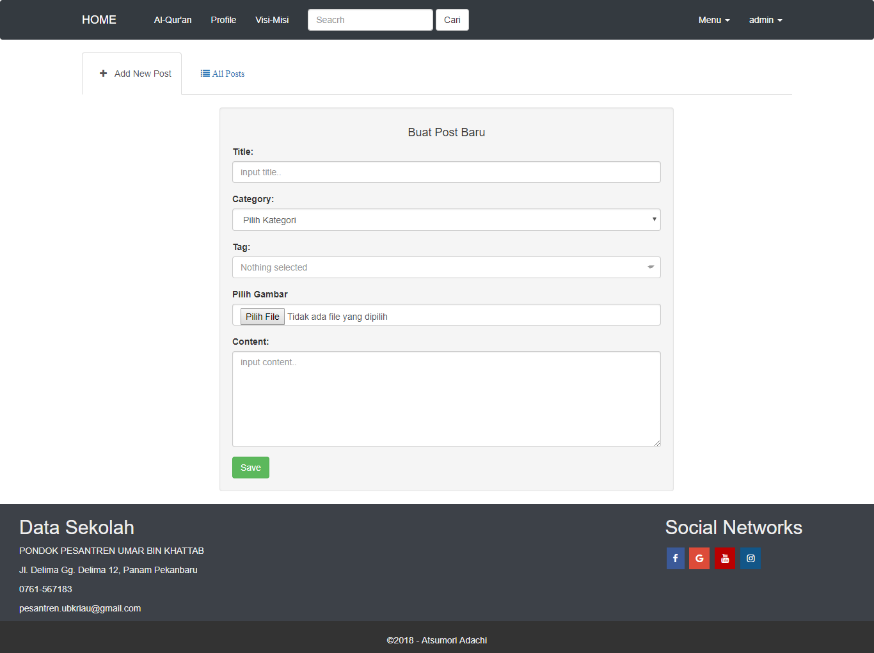
a. Tampilan Halaman *Single* *Post*



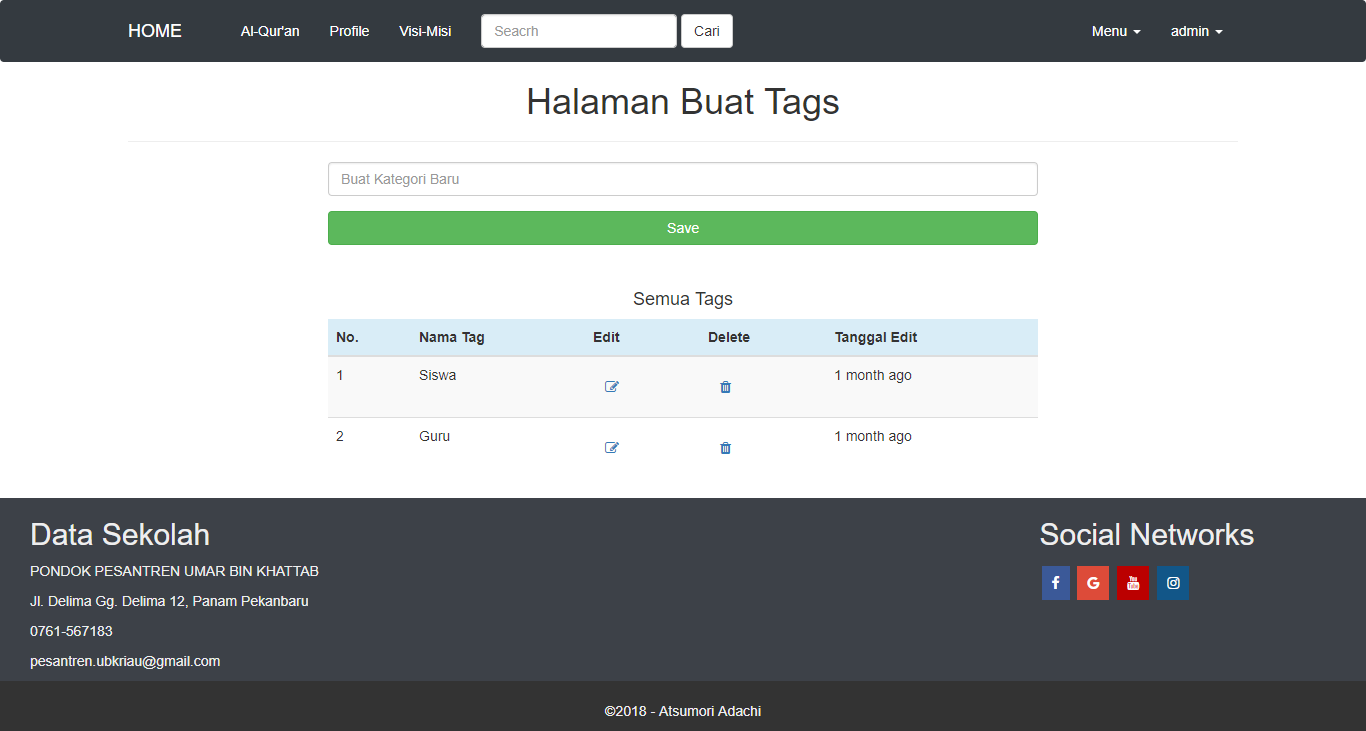
1. Tampilan Halaman *Single* Tag



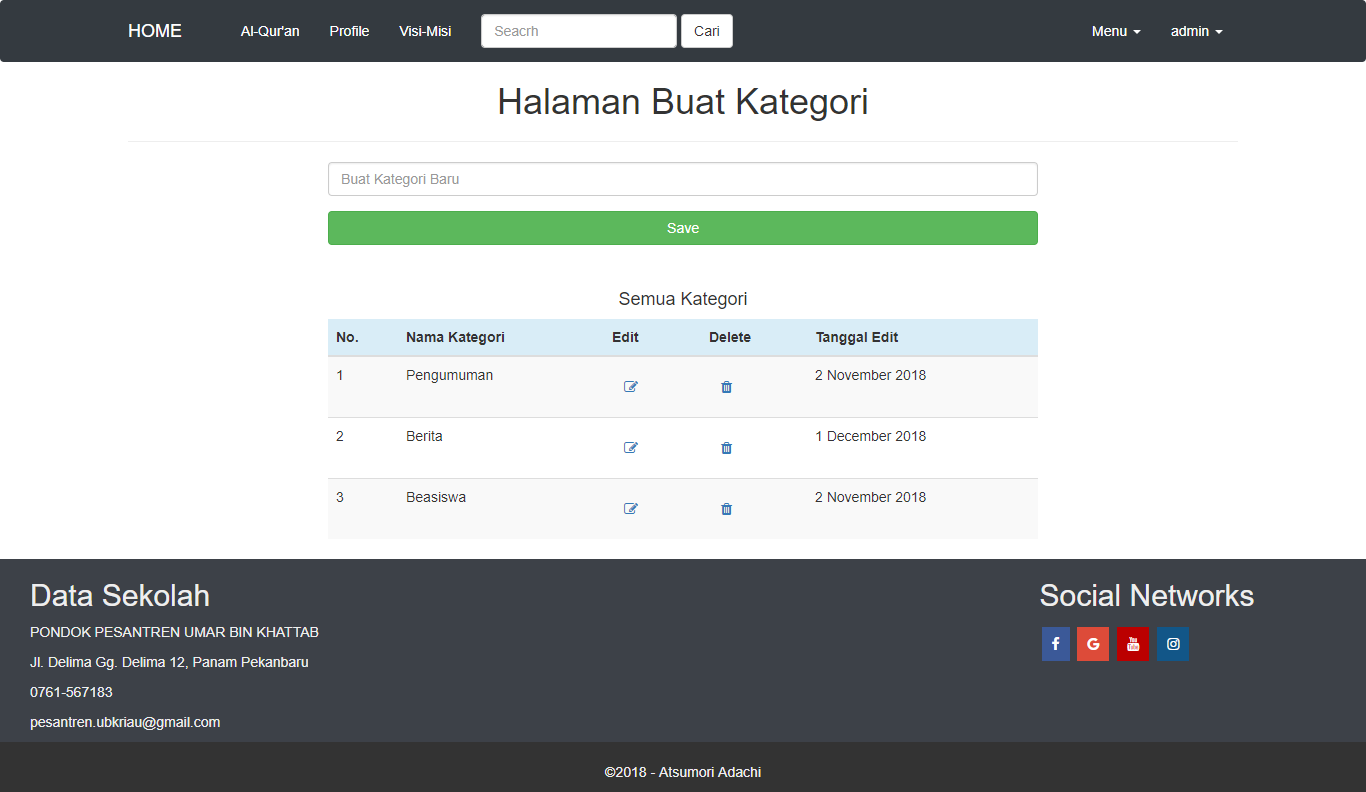
1. Tampilan Halaman Tambah Post



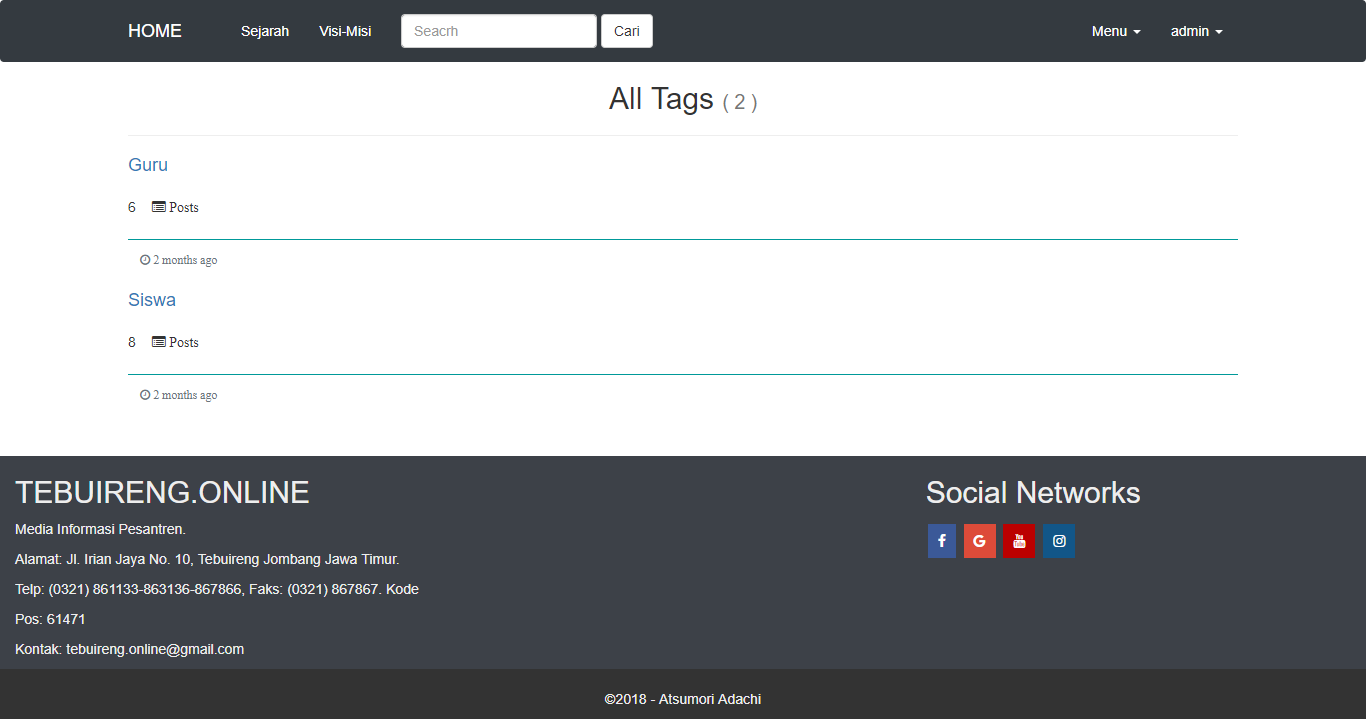
1. Tampilan Halaman Tambah Tag



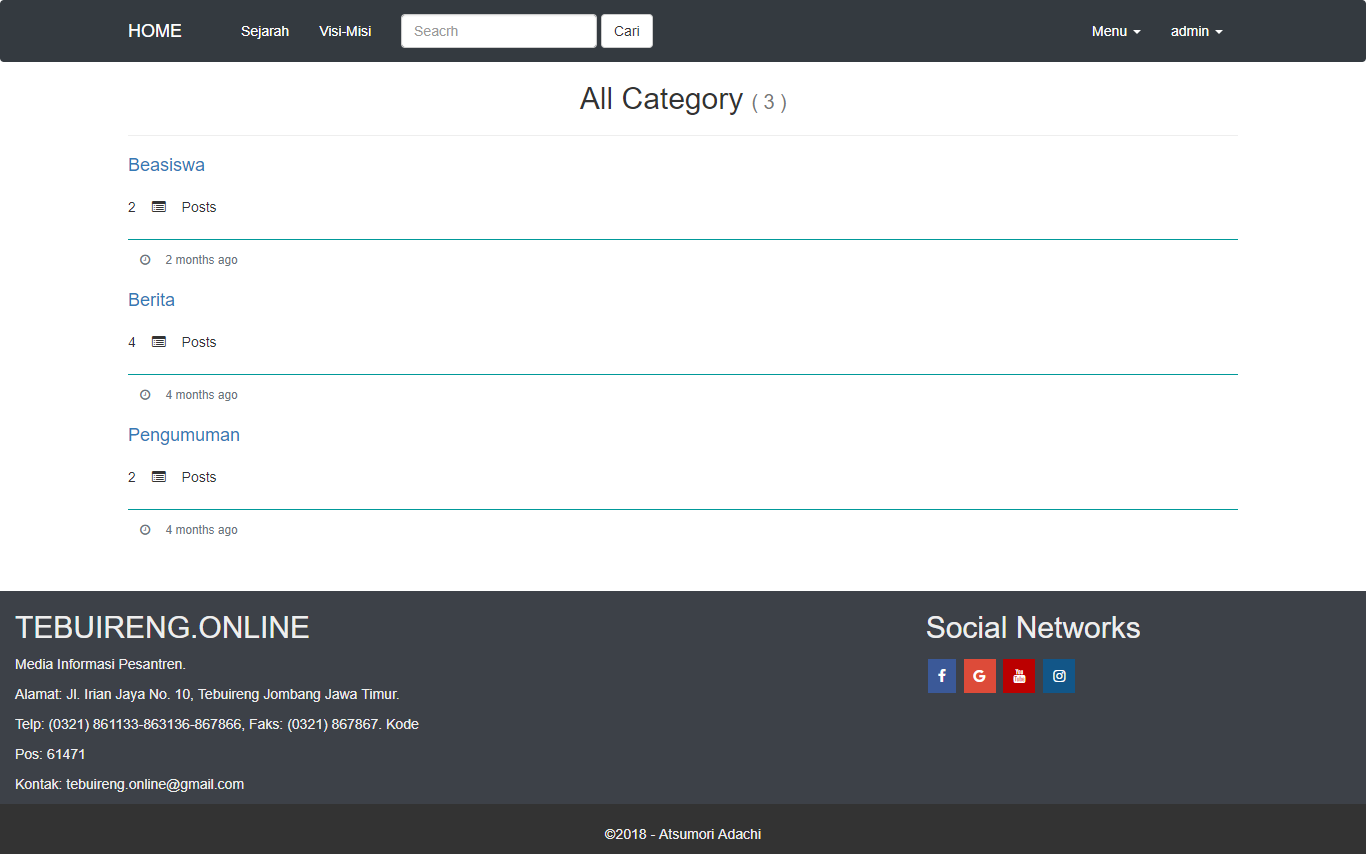
1. Tampilan Halaman Tambah Kategori



f. Gambar *All Tags*



g. Gambar *All* Kategori



h. Tampilan Edit *Post*

