

**PENGEMBANGAN BACK-END SISTEM INFORMASI PENDATAAN
SEKOLAH DESA KOMUNITAS PENDAR FOUNDATION
YOGYAKARTA**



Disusun Oleh:

N a m a : Rahayu Pangestika
NIM : 16523024

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2021

HALAMAN PENGESAHAN DOSEN PEMBIMBING

PENGEMBANGAN BACK-END SISTEM INFORMASI PENDATAAN

SEKOLAH DESA KOMUNITAS PENDAR FOUNDATION

YOGYAKARTA

TUGAS AKHIR



N a m a : Rahayu Pangestika
NIM : 16523024



Yogyakarta, 13 Januari 2021

Pembimbing,

A handwritten signature in blue ink, which appears to be 'Raden Teduh Dirgahayu'.

(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PENGESAHAN DOSEN PENGUJI

**PENGEMBANGAN BACK-END SISTEM INFORMASI PENDATAAN
SEKOLAH DESA KOMUNITAS PENDAR FOUNDATION
YOGYAKARTA**

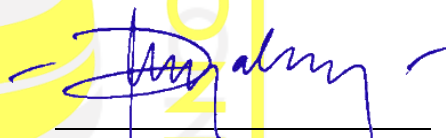
TUGAS AKHIR

Telah dipertahankan di depan sidang pengujian sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Teknik Informatika di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 13 Januari 2021

Tim Penguji

Dr. Raden Teduh Dirgahayu, S.T., M.Sc.



Anggota 1

Erika Ramadhani, S.T., M.Eng.



Anggota 2

Elyza Gustri Wahyuni, S.T., M.Cs.



Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Rahayu Pangestika
NIM : 16523024

Tugas akhir dengan judul:

PENGEMBANGAN BACK-END SISTEM INFORMASI PENDATAAN SEKOLAH DESA KOMUNITAS PENDAR FOUNDATION YOGYAKARTA

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 13 Januari 2021

METERAI
TEMPEL
TGL. 20
6E276AHF878187287
6000
ENAM RIBU RUPIAH
(Rahayu Pangestika)

HALAMAN PERSEMBAHAN

Tugas akhir ini saya persembahkan untu orang – orang terkasih:

- Kedua orang tua saya, Bapak **Kunhardadi** dan Ibu **Asnaini** yang selama ini tak henti – hentinya memberikan kasih sayang dan selalu mendoakan yang terbaik untuk saya.
- Kakak laki – laki saya, **Galih Nalendro** dan adik perempuan saya, **Ratih Arum Kinanthi** yang selalu memberikan dukungan dan bersedia mendengarkan keluh kesah saya.
- Tidak lupa pula, kepada diri saya sendiri **Rahayu Pangestika** yang sampai saat ini terus berusaha dan tidak menyerah dengan segala kondisi yang ada.



HALAMAN MOTO

“Maybe I made a mistake yesterday, but yesterday’s me is still me. I am who I am today, with all my faults. Tomorrow I might be a tiny bit wiser, and that’s me, too.”

(Kim Namjoon)

“Dengan teman-teman, duniamu akan berubah. Bagaimana aku bisa tahu? Soalnya aku ini bukti hidupnya.”

(Uzumaki Naruto, Boruto: Naruto Next Generation)

“Whatever happens tomorrow you must promise me one thing. That you will stay who you are. Not a perfect soldier, but a good man.”

(Abraham Erskine, Captain America: The First Avenger)

“Barang siapa yang menempuh jalan untuk mencari suatu ilmu. Niscaya Allah memudahkannya ke jalan menuju surga.”

(HR. Turmudzi)

UNIVERSITAS ISLAM INDONESIA
البعثة الإسلامية الأندلسية

KATA PENGANTAR

Assalamu'alaikum Wr. Wb

Alhamdulillahirobbil'alamin. Puji serta syukur kehadirat Allah *Subhanahu wa ta'ala*, atas segala nikmat, karunia dan hidayah-Nya sehingga Tugas Akhir yang berjudul “Pengembangan *Back-end* Sistem Informasi Pendataan Sekolah Desa Komunitas Pendar Foundation Yogyakarta” ini dapat selesai dengan baik. Laporan Tugas Akhir yang telah disusun ini sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata-1 (S1) Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.

Penulis menyadari akan adanya kesulitan – kesulitan yang dialami pada saat melakukan penelitian ini. Namun, berkat dorongan, dukungan, serta bimbingan dari berbagai pihak sehingga penulis dapat menyelesaikan tugas akhir dengan baik. Sehingga, penulis ingin mengucapkan banyak terima kasi kepada:

1. Fathul Wahid, S.T., M.Sc., Ph.D., selaku Rektor Universitas Islam Indonesia
2. Prof. Dr. Ir. Hari Purnomo, M.T., selaku Dekan Fakultas Teknologi Industri Universitas Islam Indonesia
3. Hendrik S.T., M.Eng selaku Ketua Jurusan Informatika Fakultas Teknologi Industri Universitas Islam Indonesia
4. Dr. Raden Teguh Dirgahayu S.T, M.Sc selaku dosen pembimbing yang telah dengan sabar membimbing saya dalam menyelesaikan tugas akhir ini
5. Segenap dosen Informatika Universitas Islam Indonesia yang telah memberikan ilmu yang bermanfaat selama masa studi
6. Kedua orang tua saya yang sangat saya sayangi Bapak Kunhardadi dan Ibu Asnaini, serta kakak laki – laki dan adik perempuan saya, Galih Nalendro dan Ratih Arum Kinanthi yang selalu memberikan dukungan dan semangat
7. Kedua sahabat saya dalam tim pengabdian masyarakat, Ashfa Fikriyya dan Raka Rizaldy yang selalu menemani penulis dalam setiap kondisi, menjadi tempat berkeluh kesah dan menghibur penulis untuk terus semangat mengerjakan tugas akhir ini
8. Sahabat saya dari awal masa perkuliahan, Berliana Putri Prasetyanti yang tiada henti memberikan dukungan dan semangat
9. Keluarga Besar Informatika Universitas Islam Indonesia angkatan 2016 “Hexadecima”

10. Serta semua pihak yang tidak bisa disebutkan satu persatu yang telah memberikan dorongan dan semangat kepada penulis

Penulis menyadari akan ketidaksempurnaan dalam penyusunan tugas akhir ini. Oleh karena itu, penulis berharap agar mendapatkan saran dan kritik yang membangun untuk bekal dimasa depan. Semoga penelitian ini dapat bermanfaat untuk berbagai pihak. Semoga semua pihak yang terkait dalam penyusunan tugas akhir ini mendapatkan pahala yang setimpal dari Allah swt.

Wassalamu'alaikum Wr. Wb.

Yogyakarta, 13 Januari 2021

(Rahayu Pangestika)



SARI

Komunitas Pendar Foundation adalah organisasi sosial yang bergerak di bidang filantropi dan pendidikan yang ditujukan untuk anak-anak petani dan buruh di daerah pedesaan Yogyakarta. Pendar Foundation menginginkan anak-anak pedesaan mendapatkan pengetahuan yang cukup dan tidak ada kesenjangan pengetahuan antara di desa dan di kota, sehingga mereka mengadakan kegiatan sekolah desa. Sekolah desa diadakan di beberapa desa dengan jumlah anak yang tidak sedikit dan dengan materi yang beragam, tetapi metode pencatatan data kegiatan yang digunakan sekarang masih dengan cara manual, yaitu menggunakan Microsoft Word.

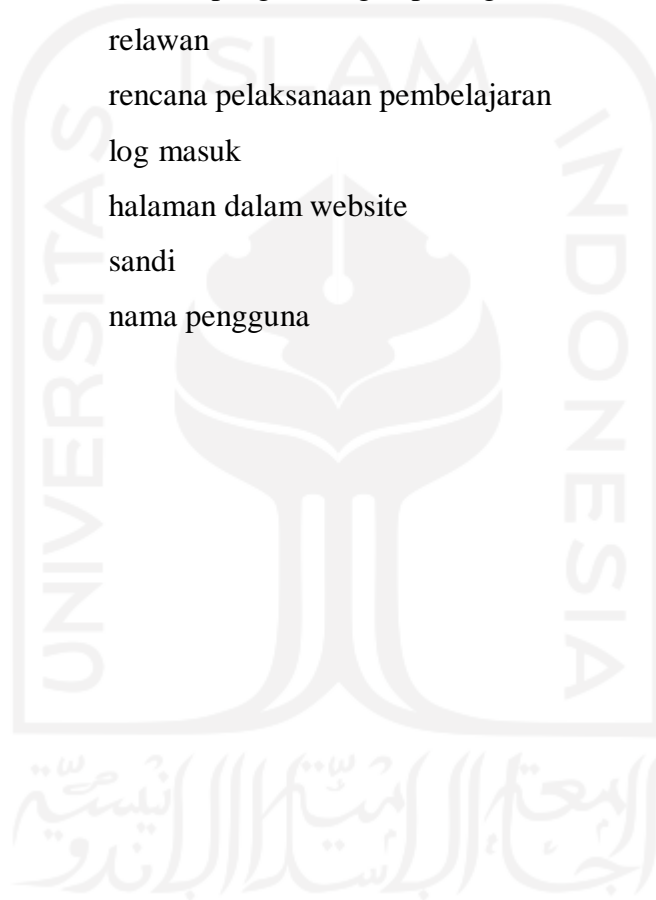
Komunitas Pendar Foundation merasa proses pencatatan data yang digunakan saat ini kurang efektif karena sering kali merasa kesulitan untuk melakukan sinkronisasi data. Untuk itu, komunitas Pendar Foundation membutuhkan sistem informasi yang dapat mencatat setiap kegiatan sekolah desa.

Tujuan dari pengembangan sistem informasi pendataan komunitas Pendar Foundation adalah mempermudah pencatatan dan pengelompokkan data kegiatan sekolah desa untuk komunitas Pendar Foundation. Selain itu, sistem juga dirancang agar dapat diakses oleh semua anggota komunitas sehingga semua anggota dapat melihat dan mengelola data secara detail. Pengembangan sistem informasi dilakukan menggunakan *framework* CodeIgniter dengan basis data MySQL. Hasil penelitian ini berupa implementasi basis data dan logika sistem.

Kata kunci: pendar foundation, filantropi, sekolah desa, framework, codeigniter, basis data, mysql

GLOSARIUM

<i>Activity Diagram</i>	diagram yang menggambarkan alur aktivitas
<i>Use case Diagram</i>	diagram kasus
<i>Framework</i>	sebuah perangkat lunak yang terdiri dari kelas dan fungsi untuk memudahkan programmer dalam mengembangkan sebuah sistem
<i>Back-end</i>	bagian belakang layar sebuah sistem
<i>Waterfall</i>	metode pengembangan perangkat lunak
<i>Volunteer</i>	relawan
<i>RPP</i>	rencana pelaksanaan pembelajaran
<i>Login</i>	log masuk
<i>Landing Page</i>	halaman dalam website
<i>Password</i>	sandi
<i>Username</i>	nama pengguna



DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING.....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI.....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO	vi
KATA PENGANTAR.....	vii
SARI.....	ix
GLOSARIUM.....	x
DAFTAR ISI	xi
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan Pengembangan.....	3
1.5 Manfaat Pengembangan.....	3
1.6 Metodologi Penelitian	3
1.7 Sistematika Penulisan	4
BAB II LANDASAN TEORI.....	6
2.1 Pengembangan <i>Back-End</i>	6
2.2 <i>Framework</i>	7
2.3 CodeIgniter.....	7
2.4 PHP	8
2.5 MySQL.....	8
2.6 Penelitian Terkait.....	9
BAB III METODOLOGI.....	14
3.1 Analisis Kebutuhan.....	14
3.1.1 Wawancara.....	14
3.1.2 Observasi Lapangan	15
3.2 Desain dan Perancangan.....	16

	xii
3.2.1 Rancangan <i>Use Case Diagram</i>	16
3.2.2 Rancangan <i>Activity Diagram</i>	18
3.2.3 Perancangan Basis Data.....	32
3.2.4 Rancangan Arsitektur HMVC.....	45
BAB IV HASIL DAN PEMBAHASAN	47
4.1 Implementasi Sistem	47
4.1.1 Halaman <i>Landing Page</i>	47
4.1.2 Halaman Login.....	48
4.1.3 Halaman Daftar Akun.....	48
4.1.4 Halaman Tentang Kami.....	49
4.1.5 Halaman Bantuan	50
4.1.6 Halaman Beranda	50
4.1.7 Halaman Hak Akses	51
4.1.8 Halaman Aktor.....	52
4.1.9 Halaman Pengguna.....	53
4.1.10 Halaman Sekolah Desa.....	53
4.1.11 Halaman Presensi.....	54
4.1.12 Halaman Data Siswa	55
4.1.13 Halaman RPP.....	55
4.1.14 Halaman Materi	56
4.1.15 Halaman Rapor	57
4.1.16 Halaman Galeri.....	58
4.1.17 Halaman Master Data Siswa	58
4.2 Pengujian Sistem.....	59
4.3 Sosialisasi dan Pelatihan Sistem.....	59
4.4 <i>Feedback</i> Pengguna	62
4.5 Pemeliharaan Sistem.....	63
BAB V KESIMPULAN DAN SARAN	64
5.1 Kesimpulan.....	64
5.2 Saran.....	64
DAFTAR PUSTAKA	66
LAMPIRAN	68

DAFTAR TABEL

Tabel 2. 1 Penelitian Terkait Pengembangan Sistem Informasi untuk Lembaga Sosial	12
Tabel 3. 1 Tabel village	33
Tabel 3. 2 Tabel users	33
Tabel 3. 3 Tabel theories	35
Tabel 3. 4 Tabel students	35
Tabel 3. 5 Tabel scores	36
Tabel 3. 6 Tabel rpp	37
Tabel 3. 7 Tabel reports	37
Tabel 3. 8 Tabel presences	38
Tabel 3. 9 Tabel pages	38
Tabel 3. 10 Tabel migrations	39
Tabel 3. 11 Tabel media	40
Tabel 3. 12 Tabel institutions	41
Tabel 3. 13 Tabel indicators	41
Tabel 3. 14 Tabel auth_users_permissions	42
Tabel 3. 15 Tabel auth_reset_attempts	42
Tabel 3. 16 Tabel auth_permissions	42
Tabel 3. 17 Tabel auth_logins	43
Tabel 3. 18 Tabel auth_groups_users	43
Tabel 3. 19 auth_groups_permissions	44
Tabel 3. 20 Tabel auth_groups	44
Tabel 3. 21 Tabel auth_activation_attempts	44

DAFTAR GAMBAR

Gambar 3. 1 Survei Kegiatan Belajar Sekolah Desa.....	16
Gambar 3. 2 Use Case Diagram Sistem Informasi Pendataan Sekolah Desa Komunitas Pendar Foundation Yogyakarta	17
Gambar 3. 3 <i>Activity Diagram</i> Kelola Aktor.....	18
Gambar 3. 4 <i>Activity Diagram</i> Kelola Pengguna.....	19
Gambar 3. 5 <i>Activity Diagram</i> Ubah Deskripsi Hak Akses	20
Gambar 3. 6 <i>Activity Diagram</i> Kelola Sekolah Desa.....	21
Gambar 3. 7 <i>Activity Diagram</i> Kelola Master Data Siswa.....	22
Gambar 3. 8 <i>Activity Diagram</i> Ubah Tentang Kami	23
Gambar 3. 9 <i>Activity Diagram</i> Ubah Bantuan.....	24
Gambar 3. 10 <i>Activity Diagram</i> Kelola Presensi.....	25
Gambar 3. 11 <i>Activity Diagram</i> Kelola Data Siswa	26
Gambar 3. 12 <i>Activity Diagram</i> Kelola RPP.....	27
Gambar 3. 13 <i>Activity Diagram</i> Kelola Materi	29
Gambar 3. 14 <i>Activity Diagram</i> Kelola Galeri	30
Gambar 3. 15 <i>Activity Diagram</i> Kelola Rapor	31
Gambar 3. 16 Relasi Basis Data Sistem Informasi Pendataan Sekolah Desa Komunitas Pendar Foundation Yogyakarta	32
Gambar 3. 17 Rancangan Arsitektur Model HMVC	45
Gambar 4.1 Halaman Landing Page	47
Gambar 4. 2 Halaman Login	48
Gambar 4. 3 Halaman Daftar Akun (1).....	48
Gambar 4. 4 Halaman Daftar Akun (2).....	49
Gambar 4. 5 Halaman Tentang Kami	49
Gambar 4. 6 Halaman Bantuan.....	50
Gambar 4. 7 Halaman Beranda.....	50
Gambar 4. 8 Sidebar Admin	51
Gambar 4. 9 Sidebar Staf dan <i>Volunteer</i>	51
Gambar 4. 10 Halaman Hak Akses.....	52
Gambar 4. 11 Halaman Aktor.....	52
Gambar 4. 12 Halaman Pengguna.....	53
Gambar 4. 13 Halaman Sekolah Desa.....	54

Gambar 4. 14 Halaman Presensi.....	54
Gambar 4. 15 Halaman Data Siswa	55
Gambar 4. 16 Halaman RPP.....	56
Gambar 4. 17 Halaman Materi	56
Gambar 4. 18 Halaman Rapor (1).....	57
Gambar 4. 19 Halaman Rapor (2).....	57
Gambar 4. 20 Halaman Galeri.....	58
Gambar 4. 21 Halaman Master Data Siswa	59
Gambar 4. 22 Sosialisasi dan Pelatihan di Bento Cafe, Klebengan.....	60
Gambar 4. 23 Sosialisasi dan Pelatihan di Upnormal Cafe, Seturan	61
Gambar 4. 24 Sosialisasi dan Pelatihan di Tempo Gelato, Taman Siswa.....	62
Gambar 4. 25 Sosialisasi dan Pelatihan di Indie.co, Sagan.....	62



BAB I PENDAHULUAN

1.1 Latar Belakang

Komunitas Pendar Foundation adalah organisasi sosial yang bergerak di bidang filantropi dan pendidikan yang ditujukan untuk anak-anak petani dan buruh di daerah pedesaan Yogyakarta. Pendar Foundation memiliki sebuah *project* kegiatan sosial yang dinamai sekolah desa. Sekolah desa merupakan kegiatan yang bertujuan untuk membantu anak-anak berlatar belakang perekonomian menengah ke bawah di daerah pedesaan Yogyakarta untuk bisa mendapatkan akses pendidikan informal sehingga tidak menimbulkan kesenjangan pengetahuan di desa dan di kota. Selain itu, sekolah desa juga berusaha membangun daerah pedesaan melalui pendidikan.

Pendar Foundation berdiri sejak tahun 2015 sebagai BaBe yang artinya Barang Bekas Berbagi. Kegiatan yang dilakukan BaBe adalah mengumpulkan barang-barang bekas layak pakai. Pada tahun 2016, BaBe membentuk sebuah kegiatan sosial bernama sekolah desa, saat itu dinamai sekolah desa babe. Kegiatan sekolah desa masih terus berjalan hingga saat ini, kegiatan sekolah desa berlangsung di salah satu desa yang terletak pada kaki Gunung Merbabu yaitu, Desa Sanden. Pendar Foundation berencana mendirikan lagi sekolah desa di Desa Kembaran yang terletak di daerah Madukismo.

Berdasarkan hasil wawancara, staf komunitas Pendar Foundation menyatakan bahwa banyaknya kegiatan dan jumlah murid sekolah desa membuat komunitas Pendar Foundation membutuhkan sistem informasi yang dapat mencatat data kegiatannya. Untuk saat ini, komunitas Pendar Foundation menggunakan Microsoft Word dan catatan tertulis untuk mencatat datanya. Cara tersebut kurang efektif karena belum adanya satu wadah khusus untuk mengelompokkan datanya. Dengan menggunakan sistem informasi, setiap anggota komunitas berharap dapat mengakses dan mengolah data kegiatan dengan baik dan tertata. Sistem informasi menjadi salah satu alat bantu yang sangat tepat sebagai solusi dari permasalahan yang dihadapi (Fajarianto & Wahyuni, 2016).

Tujuan dari pengembangan sistem informasi pendataan komunitas Pendar Foundation adalah untuk membantu pengelolaan dan pengelompokkan data kegiatan komunitas. Sistem dirancang agar dapat diakses oleh semua anggota komunitas Pendar Foundation, termasuk *volunteer* yang sedang mengikuti kegiatan komunitas. Hal ini dimaksudkan untuk

menciptakan transparansi antara anggota komunitas dan *volunteer* dalam sistem pendataan dan penilaian untuk anak didik di sekolah desa.

Sistem *development* pada aplikasi ini dilakukan dengan pembagian *development* yang terbagi menjadi 2 *sub-job*, yaitu bagian *front-end*, yang bertanggung jawab dalam keseluruhan *interface* pada aplikasi dan bagian *back-end*, yang berisi semua proses yang tidak berhubungan langsung dengan pengguna, seperti server, basis data dan data – data yang dibutuhkan oleh *front-end*. Penelitian ini bertujuan untuk memberikan solusi kepada komunitas Pendar Foundation dengan membangun sistem informasi berbasis *website* khususnya pada sisi *back-end*, pengembangan *front-end* dilakukan oleh peneliti lain.

Pengembangan *back-end* sistem informasi pendataan sekolah desa komunitas Pendar Foundation dibangun menggunakan arsitektur HMVC (*Hierarchical Model View Controller*), dimana proses pengembangan *back-end* berfokus pada bagian *model* dan *controller*. Penggunaan *framework* juga dimanfaatkan dalam pengembangan sistem. Pengembangan sistem menggunakan *Framework* CodeIgniter dengan basis data MySQL. Pemanfaatan *framework* dalam pengembangan sistem ini untuk memudahkan dalam segi *maintenance*. CodeIgniter dipilih dalam pengembangan sistem karena ukurannya yang kecil dibandingkan dengan *framework* lain, sehingga lebih hemat dalam segi *storage*.

12 Rumusan Masalah

Bagaimana mengembangkan *back-end* sebuah aplikasi berbasis web dengan *framework* CodeIgniter dan basis data MySQL?

13 Batasan Masalah

Adapun batasan masalah untuk pengembangan sistem informasi pendataan sekolah desa komunitas Pendar Foundation adalah sebagai berikut:

- a. Pengembangan hanya berfokus pada sisi *back-end*
- b. Aplikasi dibangun untuk digunakan oleh staf dan *volunteer* di komunitas Pendar Foundation saja
- c. Staf komunitas Pendar Foundation dapat membuat akun, mengelola data diri, mengelola desa, mengelola galeri, mengelola data anak, mengelola materi dan mengelola RPP
- d. *Volunteer* tidak dapat mengelola desa, mengelola galeri dan mengelola data anak, namun dapat membuat akun, mengelola data diri, mengelola materi, mengelola RPP, melihat desa, melihat galeri dan melihat data anak

14 Tujuan Pengembangan

Penelitian ini dilakukan untuk mengembangkan *back-end* sistem informasi pendataan kegiatan sekolah desa komunitas Pendar Foundation yang bertujuan untuk memudahkan staf dan *volunteer* dalam pencatatan data kegiatan. Data – data tersebut berupa data anak didik, foto – foto kegiatan sekolah desa, rencana pelaksanaan pembelajaran (RPP) dan materi pembelajaran.

15 Manfaat Pengembangan

Manfaat dari pengembangan sistem ini, antara lain:

- a. Membantu staf komunitas Pendar Foundation untuk mengelola data anak didik sekolah desa
- b. Membantu staf dan *volunteer* komunitas Pendar Foundation untuk mengelola rencana rancangan pembelajaran sekolah desa
- c. Membantu staf dan *volunteer* komunitas Pendar Foundation untuk mengelola foto-foto kegiatan sekolah desa
- d. Memberikan sebuah *platform* khusus kepada Pendar Foundation sebagai arsip kegiatan sekolah desa
- e. Memudahkan kerja staf dan *volunteer* komunitas Pendar Foundation dalam pendataan kegiatan sekolah desa

16 Metodologi Penelitian

Pengembangan sistem informasi pendataan sekolah desa komunitas Pendar Foundation memerlukan adanya metodologi yang sesuai, tahap – tahapannya antara lain sebagai berikut:

- a. Tahap pengembangan perangkat lunak

Pengembangan perangkat lunak dilakukan dengan menggunakan metode *Waterfall*.

Tahapannya adalah sebagai berikut:

1. *Requirement* (analisis kebutuhan)

Pada tahap analisis kebutuhan, terjadi pertemuan antara tim pengembang dengan staf komunitas Pendar Foundation, dilakukan wawancara untuk pengumpulan data. Dibutuhkan keterlibatan dari staf komunitas untuk menentukan beberapa fitur yang akan tersedia dalam sistem, karena fitur tersebut disesuaikan dengan kebutuhan pengguna, yaitu staf dan *volunteer* komunitas Pendar Foundation.

2. Desain dan perancangan

Proses desain sistem dilakukan dengan mendesain antarmuka dan fitur – fitur dalam sistem yang telah disetujui oleh pengguna, yaitu komunitas Pendar Foundation. Kemudian dilanjutkan dengan perancangan logika sistem dan basis data sesuai dengan kebutuhan sistem.

3. Implementasi

Tahap selanjutnya dilakukan implementasi pengembangan *back-end* menggunakan basis data MySQL pada PHPMyAdmin dan penerapan kode – kode program dengan memanfaatkan *framework* CodeIgniter.

4. Pengujian sistem

Tahap pengujian sistem dilakukan dengan menggunakan metode *Black-box Testing*. Pengujian berfokus pada sudut pandang fungsional sistem. Pada tahap ini sistem akan diuji untuk mengetahui apakah sistem sudah dapat berjalan dengan baik dan memberikan hasil sesuai dengan yang diharapkan.

5. *Maintenance* (pemeliharaan)

Perangkat lunak yang telah disampaikan kepada pengguna akan dilakukan pemeliharaan untuk melakukan pembaruan apabila ada perubahan, kesalahan atau permintaan dari pengguna.

b. Sosialisasi dan pelatihan penggunaan sistem

Kegiatan pelatihan merupakan tahap pengenalan sistem informasi pendataan sekolah desa, dengan tujuan memberikan edukasi kepada anggota komunitas Pendar Koundation mengenai sistem yang telah dibangun.

c. *Feedback* pengguna

Feedback pengguna didapatkan dari hasil wawancara dan diskusi yang dilakukan pada saat kegiatan sosialisasi dan pelatihan penggunaan sistem bersama dengan anggota komunitas Pendar Foundation. *Feedback* dari pengguna dibutuhkan untuk mengetahui kepuasan pengguna terhadap sistem yang telah dibangun.

17 Sistematika Penulisan

Adapun sistematika penulisan dalam penulisan laporan tugas akhir adalah sebagai berikut:

BAB I PENDAHULUAN

Berisi latar belakang, rumusan masalah, batasan masalah, tujuan pengembangan, manfaat pengembangan, metodologi pengembangan, dan sistematika penulisan.

BAB II LANDASAN TEORI

Berisi penelitian terkait pengembangan back-end sistem informasi pendataan sekolah desa komunitas Pendar Foundation, bahasa pemrograman PHP, *framework* yang digunakan dalam pengembangan sistem, yaitu *framework* CodeIgniter, dan penjelasan mengenai *back-end* dalam pengembangan sistem.

BAB II METODOLOGI

Berisi uraian terkait metode *Waterfall* yang digunakan dalam pengembangan sistem informasi pendataan sekolah desa komunitas Pendar Foundation, tahapannya meliputi tahap *requirement* (analisis kebutuhan), tahap desain dan perancangan, tahap implementasi, tahap pengujian sistem dan tahap *maintenance* (pemeliharaan).

BAB IV HASIL DAN PEMBAHASAN

Berisi tulisan dan gambar yang menjelaskan hasil dari tahap implementasi sistem informasi pendataan sekolah desa komunitas Pendar Foundation serta hasil dari tahap pengujian sistem.

BAB V KESIMPULAN DAN SARAN

Berisi kesimpulan dari hasil dan saran terhadap pengembangan yang telah dilakukan, sebagai rangkuman terhadap hasil pengembangan sistem informasi pendataan sekolah desa komunitas Pendar Foundation serta demi kepentingan dokumentasi atau pengembangan di masa yang akan datang.

BAB II LANDASAN TEORI

2.1 Pengembangan *Back-End*

Pengembangan *back-end* sistem informasi pendataan sekolah desa komunitas Pendar Foundation, menggunakan arsitektur model *Hierarchical Model View Controller* (HMVC). Model arsitektur HMVC merupakan versi pengembangan dari model arsitektur MVC (*Model View Controller*), atau bisa dikatakan sebagai versi MVC yang diimplementasikan secara hirarkis (Muzakir, 2014). MVC adalah metode yang memisahkan aplikasi menjadi *view*, *model* dan *controller*. Sedangkan HMVC adalah kumpulan dari MVC yang saling berhubungan.

Pembagian tugas pada *model*, *view* dan *controller*:

- a. *Model*, sebagai pengelola data pada domain aplikasi, melakukan tanggapan terhadap permintaan informasi dan memproses intruksi untuk merubah suatu kondisi tertentu. Contohnya fungsi – fungsi yang berhubungan dengan basis data, seperti *create*, *read*, *update* dan *delete*.
- b. *View*, biasanya berupa antarmuka yang melakukan interaksi secara langsung dengan pengguna. *View* menerjemahkan informasi dari *model* ke dalam bentuk yang sesuai untuk berinteraksi dengan *user*.
- c. *Controller*, mengandung proses *input* dari pengguna, kemudian mengirim perintah kepada *model* dan *view* agar melakukan proses sesuai dengan masukan. Singkatnya, *controller* adalah pemetaan aksi pengguna terhadap respons sistem.

Front-end developer bertugas membuat bagian *view*, yaitu membuat antarmuka sistem yang berhubungan langsung dengan pengguna. Sedangkan *back-end developer* bertugas untuk membuat *model* dan *controller*. *Back-end* adalah tempat di mana proses pada suatu sistem informasi atau aplikasi berjalan, tempat memproses ketika data ditambah, diubah dan dihapus. *Back-end* mengurus segala jenis proses yang tidak berhubungan langsung dengan pengguna, seperti server dan basis data. *Back-end* dibutuhkan dalam pengembangan sistem dan manajemen data pada sistem.

22 Framework

Framework merupakan kerangka kerja yang berisikan sekumpulan fungsi dan kelas yang siap digunakan untuk tujuan pengembangan aplikasi. Dengan demikian *framework* dapat memudahkan pekerjaan *programmer*, yaitu *programmer* tidak perlu membuat semuanya dari awal. *Framework* membuat pengerjaan lebih terstruktur dan lebih tersusun dengan menggunakan fungsi, prosedur dan kelas yang sudah disediakan.

Beberapa contoh *framework* yang populer digunakan dalam pengembangan sistem informasi berbasis web, diantaranya:

- a. CodeIgniter, merupakan salah satu *framework* PHP yang bersifat *open-source*. Kerangka kerja CodeIgniter menyediakan sebuah *library* yang dapat digunakan sesuai kebutuhan, sehingga menjadi sangat membantu bagi *developer* dalam membangun aplikasi.
- b. Laravel, tidak berbeda dengan CodeIgniter, Laravel juga merupakan *framework* PHP yang bersifat *open-source*. Aplikasi web yang menggunakan *framework* Laravel cenderung menghasilkan *page size* yang lebih besar dan berubah – ubah karena *framework* Laravel me-load lebih banyak *library* (Erinton, Negara, & Sanjoyo, 2017).
- c. Symfony, *framework* Symfony mengelompokkan aturan – aturan bisnis aplikasi, logika *serve* dan tampilan presentasi untuk mempersingkat durasi pengembangan sebuah sistem yang kompleks. Kerangka kerja Symfony menyediakan beragam alat dan kelas yang dapat dimanfaatkan.

23 CodeIgniter

Pengembangan sistem informasi pendataan sekolah desa komunitas Pendar Foundation dilakukan dengan menggunakan CodeIgniter 4. CodeIgniter merupakan *framework* PHP yang bersifat *open-source*. Salah satu keunggulan dalam menggunakan *framework* CodeIgniter adalah ukurannya yang kecil dibandingkan dengan *framework* lain yang membutuhkan *resource* besar untuk dioperasikan. Kerangka kerja CodeIgniter menyediakan sebuah *library* yang dapat digunakan sesuai dengan kebutuhan, sehingga menjadi sangat membantu bagi *developer* dalam membangun aplikasi. *Library* berisikan sekumpulan fungsi dan kelas yang dapat digunakan dalam membangun aplikasi. Adapun *library* tersebut, antara lain:

- a. *Database, library database* atau basis data digunakan untuk mengakses dan mengelola data yang terdapat di dalam basis data
- b. *Input, library input* digunakan untuk memproses data yang didapatkan dari *form*

- c. *File uploading, library file uploading* digunakan untuk membangun sebuah web yang memerlukan proses untuk mengunggah berkas
- d. *Session, library session* digunakan untuk memelihara data
- e. *URI class, library URI class* digunakan untuk mendapatkan informasi mengenai alamat web

Pada Codeigniter 4 sudah terdapat fitur *composer* sehingga tidak perlu menggunakan *package* yang telah dibuat oleh *developer* lain. *Package* bertujuan untuk mempercepat proses pengembangan aplikasi. Adapun *package* ini bersifat modular, sehingga *package* dapat dipakai dan dihapus sesuai kebutuhan. Adapun *package* yang digunakan dalam pengembangan ini adalah sebagai berikut:

- a. *Myth-auth, package* ini digunakan untuk keperluan *authentication*, seperti pembuatan fitur *login, register, logout* dan untuk memberikan hak akses tertentu dalam sebuah sistem informasi
- b. *Carbon, package* memberikan kemudahan dalam pembuatan dan pengelolaan data tanggal atau hari atau biasa disebut dengan format *datetime*
- c. *Eloquent, package* yang berhubungan dengan pengelolaan basis data. *Package* ini digunakan untuk memudahkan *developer* dalam proses penarikan data
- d. *Php-enum, package* yang digunakan untuk memudahkan pengguna *enum class*

24 PHP

PHP (PHP : Hypertext Processor) merupakan bahasa pemrograman yang dapat berjalan di sistem operasi yang berbeda (Windows, Linux dan Mac). Bahasa pemrograman PHP secara umum digunakan untuk membangun dan mengembangkan sebuah situs web. PHP dapat menampilkan konten yang dinamis dari sebuah website, karena PHP dapat berinteraksi langsung dengan basisdata, file dan folder, berbeda dengan HTML yang hanya dapat menampilkan konten statis (Yuliano, 2009).

25 MySQL

MySQL merupakan implementasi dari sistem manajemen basis data (DBMS) dibawah lisensi *General Public License* (Anhar, 2010). MySQL dapat dioperasikan melalui *Command Line* dan aplikasi *PhpMyAdmin*, untuk dijalankan pada *Command Line*, perlu menggunakan

sintaks *query* yang digunakan dalam pemrograman MySQL. Sedangkan pengoperasian dengan aplikasi PhpMyAdmin, tidak memerlukan sintaks *query*.

Bahasa pemrograman yang digunakan dalam manajemen basis data disebut SQL (*Structured Query Language*). Dalam SQL terdapat 3 sub bahasa, yaitu *Data Definition Language* (DDL), *Data Manipulation Language* (DML), dan *Data Control Language* (DCL). DDL berisikan perintah untuk membangun isi dari basis data, bertugas untuk membuat objek SQL dan menyimpan definisinya ke dalam tabel. DML berfungsi untuk menampilkan, mengubah, menambah, dan menghapus baris dalam tabel. DCL bertugas untuk mengontrol keamanan basis data dengan membuat hak akses tertentu bagi setiap pengguna (Anhar, 2010). Keunggulan yang dimiliki MySQL, yaitu: dapat berjalan dengan stabil pada beberapa sistem operasi, *open-source*, *multi user*, dan memiliki keamanan yang baik. Keamanan yang dimiliki SQL berupa izin akses pengguna, *password* terenkripsi, pengenalan nama host dan *subnet mask* (Anhar, 2010).

26 Penelitian Terkait

Berdasarkan hasil kajian pustaka, terdapat beberapa komunitas, organisasi, yayasan dan lembaga sosial yang merancang sistem informasi sebagai alat bantu untuk menunjang eksistensinya. Sistem informasi tersebut dikembangkan dengan menggunakan teknik pengembangan *back-end* yang beragam, berikut merupakan beberapa penelitian terkait pengembangan sistem informasi pada komunitas, organisasi dan lembaga sosial.

Penelitian pertama adalah penelitian yang dilakukan oleh Nur Afifah, Denny Sagita Rusdianto dan Aditya Rachmadi pada tahun 2020, tentang pembangunan website pengelolaan kegiatan komunitas, studi kasus Komunitas Generasi Baru Indonesia Malang (GenBI). Penelitian ini dilatar belakangi oleh sistem penyimpanan dokumen kegiatan komunitas yang tidak terpusat, sehingga mengakibatkan proses penilaian kegiatan komunitas tidak berjalan secara efektif. Berdasarkan permasalahan tersebut, peneliti mengembangkan sistem informasi berbasis website yang dapat mengelola seluruh informasi anggota dan kegiatan komunitas. Sistem diuji pada berbagai *web server* dengan tujuan untuk memastikan sistem dapat diakses dengan mudah dan dapat memfasilitasi pelaksanaan proses pengelolaan kegiatan demi mencapai pengelolaan yang efektif dan *traceable*. Pengembangan sistem dilakukan dengan metode pengembangan *Software Development Life Cycle* (SDLC) *Waterfall*, sedangkan implementasi kode program dengan bahasa pemrograman PHP dilakukan menggunakan *framework* CodeIgniter yang mengadopsi model arsitektur MVC dan basis data MySQL.

Tahap pengujian sistem dilakukan dengan 3 tahap pengujian, yaitu tahap pengujian *basis path testing*, *black-box testing* untuk mendapatkan hasil pengujian validasi agar dapat mengetahui hasil implementasi kebutuhan fungsional sistem sesuai dengan kebutuhan dan pengujian *compatibility* yang memberikan hasil berupa 100% konten dan fungsionalitas sistem dapat berfungsi dengan baik (Afifah, Rusdianto, & Rachmadi, 2020).

Penelitian kedua adalah penelitian yang dilakukan oleh Gusti Rachman Nico, Satria Agung Wicaksono dan Fajar Pradana pada tahun 2018, tentang pengembangan sistem informasi manajemen donasi dan relawan bagi komunitas sosial, studi kasus komunitas Turun Tangan Malang. Tujuan dari penelitian ini adalah untuk memberikan solusi yang ada dengan cara mengembangkan sebuah sistem informasi manajemen dengan menerapkan model pengembangan perangkat lunak *Evolutionary Prototyping* sehingga sistem yang dibangun dapat memenuhi kebutuhan pengguna. Metode pengembangan sistem *Evolutionary Prototyping* dipilih karena dapat meminimalkan kurangnya analisis terhadap kebutuhan sistem yang akan dikembangkan dan juga memastikan sistem tersebut sesuai dengan permintaan dan kebutuhan pengguna. Pengembangan sistem dilakukan dengan menerapkan *framework* CodeIgniter dan rancangan basis data *Physical Data Model* (PDM), sedangkan pengujian sistem dilakukan menggunakan 3 metode, yaitu *black-box testing*, pengujian kompatibilitas dan pengujian UAT (*User Acceptance Test*). Kesimpulan dari hasil pengujian menunjukkan bahwa sistem telah berjalan dengan baik dan memenuhi kebutuhan pengguna (Nico, Wicaksono, & Pradana, 2018).

Penelitian ketiga adalah penelitian yang dilakukan oleh Yuris Socio Perestroika, Adam Hendra Brata dan Yusi Tyroni Mursityo pada tahun 2018, tentang pengembangan sistem informasi penggalangan donasi pada yayasan Gerakan Nurani Orang Tua Asuh (GN-OTA) Kabupaten Lumajang. Penelitian ini dilatar belakangi oleh penurunan jumlah donasi. Dari permasalahan tersebut, peneliti memberikan solusi berupa pengembangan sistem informasi yang didasari oleh maraknya fenomena *e-charity*, *online fundrising* atau donasi yang dilakukan melalui internet yang dapat dikatakan cukup sukses meningkatkan kepekaan sosial masyarakat. Sistem dikembangkan dengan memanfaatkan model pengembangan iteratif dan pendekatan berorientasi objek. Pengembangan sistem dilakukan dengan menerapkan konsep arsitektur MVC (*Model View Controller*), sedangkan proses implementasi pengkodean program dikembangkan menggunakan bahasa pemrograman PHP dengan *framework* CodeIgniter. Penulis menerapkan 2 metode testing pada pengembangan sistem ini, yaitu *black-box testing* sebagai pengujian non fungsional dan *white-box testing* sebagai pengujian

fungsional. Kesimpulan yang didapat dari hasil pengujian yaitu keseluruhan fitur sudah dapat berjalan baik sesuai harapan namun perlu adanya perbaikan dari sisi *minimum HTTP request* (Perestrioka, Brata, & Mursityo, 2018).

Penelitian keempat adalah penelitian yang dilakukan oleh Crystiana May Sari, Delfin Jefriansyah, Maya Norriastini dan Zuriati pada tahun 2019, tentang aplikasi penggalangan dana berbasis web, studi kasus Komunitas Jumat Jaman Now (JJN). Tujuan dari penelitian ini adalah untuk mengembangkan sebuah aplikasi berbasis web yang dapat mengelola penggalangan dana dengan cara sumbangan dari masyarakat secara *online*, termasuk penyebaran informasi dan pendataan kegiatan komunitas. sehingga masyarakat dapat melakukan donasi dengan cepat dan mudah, Dana yang telah terkumpul nantinya akan disalurkan untuk membantu fakir miskin di jalanan dan anak – anak yatim. Sistem dikembangkan dengan menggunakan metode pengembangan *Rapid Application Development* (RAD) dan implementasi pengkodean sistem dengan *framework* Laravel. Tahap pengujian sistem dilakukan dengan metode *black-box testing* dengan teknik *Ecuivalence ClassPartitioning*, yaitu pengujian dengan proses masukan yang sama akan menghasilkan respon yang sama pula dan teknik *Boundary Value Analysis* (BVA), merupakan teknik pengujian yang berfokus pada nilai yang dimasukkan sesuai dengan batasan yang ditentukan pada nilai awal dan batas pada nilai akhir. Dari tahap pengujian didapatkan hasil bahwa adanya fungsi yang hilang atau tidak sesuai, kesalahan antarmuka dan kesalahan dalam struktur data atau akses basis data eksternal (Sari, Jefriansyah, Norriastini, & Zuriati, 2019).

Penelitian kelima adalah penelitian yang dilakukan oleh Ayu Fitri Wulandari pada tahun 2019, tentang rancang bangun aplikasi SIMPADI (Sistem Informasi *Monitoring* Pengasuhan Anak Disabilitas) berbasis web, studi kasus *Moslem Daycare* ABK Bina Anak Surabaya. Tujuan dari penelitian ini untuk mengembangkan sebuah sistem informasi yang dapat menggantikan proses *monitoring* perkembangan pengasuhan anak disabilitas yang sebelumnya masih dilakukan secara manual, yaitu dilakukan dengan cara datang langsung ke lembaga. Aplikasi berbasis web SIMPADI dikembangkan dengan metode pengembangan *Software Development Life Cycle* (SDLC) *Waterfall* yang memiliki tahapan analisis, desain, implementasi, pengujian dan pemeliharaan. Implementasi kode program memanfaatkan *framework* Laravel dengan basis data MySQL dan tahap pengujian sistem dilakukan dengan menggunakan metode *Black-box testing*. Pada tahap pengujian dengan *Black-box testing* sesuai dengan tabel kebutuhan fungsionalitas, didapatkan hasil bahwa semua fungsi berjalan sesuai dengan pernyataan pada tabel fungsionalitas, sehingga disimpulkan aplikasi SIMPADI

berhasil dikembangkan dan diimplementasikan sesuai dengan kebutuhan pengguna (Wulandari, 2019).

Berikut merupakan penelitian terkait yang dijadikan sumber referensi apabila disajikan dalam bentuk tabel.

Tabel 2. 1 Penelitian Terkait Pengembangan Sistem Informasi untuk Lembaga Sosial

No	Literatur		Metode Pengembangan	Framerowk	Pengujian
	Sumber	Sistem			
1	(Afifah, Rusdianto, & Rachmadi, 2020)	Website Pengelolaan Kegiatan Komunitas	<i>Software Development Life Cycle (SDLC) Waterfall</i>	CodeIgniter	<i>Basis Path Testing, Black-box Testing dan Compatibility Testing</i>
2	(Nico, Wicaksono, & Pradana, 2018)	Sistem Informasi Manajemen Donasi dan Relawan Bagi Komunitas Sosial	<i>Evolutionary Prototyping</i>	CodeIgniter	<i>Black-box Testing, Compatibility Testing, dan User Acceptance Test (UAT).</i>
3	(Perestroika, Brata, & Mursityo 2018)	Sistem Informasi Penggalangan Donasi Yayasan Nurani Orang Tua Asuh (GN-OTA) Kabupaten Lumajang	Sistem dikembangkan dengan model pengembangan iterative dan pendekatan berorientasi objek	CodeIgniter	<i>Black-box Testing dan White-box Testing</i>
4	(Sari, Jefriansya,	Aplikasi Penggalangan	<i>Rapid Application</i>	Laravel	<i>Black-box Testing dengan</i>

	Norriastini, & Zuriati 2019)	Dana Komunitas Jumat Jaman Now	<i>Development</i> (RAD)		teknik <i>Boundary Value</i> <i>Analysis</i> (BVA) dan teknik pengujian berfokus pada nilai
5	(Wulandari, 2019)	SIMPADI (Sistem Informasi Pengasuhan Anak Disabilitas)	<i>Software</i> <i>Development</i> <i>Life Cycle</i> (SDLC) <i>Waterfall</i>	Laravel	<i>Black-box</i> <i>Testing</i>

Berdasarkan penelitian terkait yang dijadikan sumber referensi diketahui bahwa, pengembangan *back-end* sistem informasi pendataan sekolah desa komunitas Pendar Foundation menggunakan metode pengembangan perangkat lunak *Software Development Life Cycle* (SDLC) *Waterfall* sama seperti penelitian pertama dan kelima. Menggunakan kerangka kerja CodeIgniter, seperti yang digunakan pada penelitian terkait pertama, kedua dan ketiga. Sedangkan, pengujian sistem dilakukan menggunakan teknik pengujian *Black-box Testing*, seperti yang digunakan oleh kelima penelitian terkait.

BAB III METODOLOGI

3.1 Analisis Kebutuhan

Proses analisis kebutuhan sistem didapatkan dari hasil pengumpulan data yang dilakukan dengan kegiatan wawancara dan observasi lapangan. Dari kegiatan tersebut didapatkan data yang dibutuhkan untuk kepentingan pengembangan sistem.

3.1.1 Wawancara

Kegiatan wawancara pertama dilakukan bersama dengan *founder* dan *project manager* komunitas Pendar Foundation. Pada wawancara pertama, informasi yang didapatkan lebih banyak mengarah pada identitas komunitas Pendar Foundation sebagai tahap pengenalan antara pihak komunitas dan tim pengembang. Informasi yang didapatkan dari wawancara pertama adalah sebagai berikut:

- a. Pendar Foundation adalah organisasi sosial yang berfokus di bidang filantropi dan pendidikan yang ditujukan untuk anak – anak petani dan buruh di daerah pedesaan Yogyakarta. Pendar Foundation memutuskan untuk mengabdikan kepada masyarakat kecil di pedesaan dikarenakan tekad dan niat baik untuk memberikan pengetahuan dan pendidikan yang cukup kepada anak – anak di pedesaan agar tidak adanya kesenjangan antara pengetahuan anak – anak di desa dan di kota
- b. Kegiatan utama komunitas Pendar Foundation adalah kegiatan sekolah desa. Kegiatan sekolah desa adalah kegiatan belajar mengajar yang dilakukan di beberapa desa tempat pengabdian dari komunitas Pendar Foundation.
- c. Komunitas Pendar Foundation dibantu oleh *volunteer* untuk pelaksanaan kegiatan sekolah desa
- d. Proses pendataan untuk kegiatan sekolah desa menarapkan cara manual, yaitu dicatat pada Microsoft Word, kemudian dikelompokkan dalam satu *file* berdasarkan kelompok data yang sama
- e. Komunitas Pendar Foundation mengharapkan sebuah wadah khusus yang mampu membantu mereka dalam pencatatan semua data kegiatan sekolah desa

Berdasarkan hasil dari kegiatan wawancara pertama, diketahui bahwa komunitas Pendar Foundation membutuhkan sistem informasi yang dapat membantu proses pencatatan seluruh

data kegiatan sekolah desa. Maka dari itu, dilakukan wawancara kedua bersama ketua divisi pendidikan komunitas Pendar Foundation untuk membahas kebutuhan dan fitur sistem. Hasil wawancara kedua adalah sebagai berikut:

- a. Sistem akan digunakan oleh staf Pendar Foundation dan *volunteer*
- b. Sistem dapat mencatat data anak didik sekolah desa
- c. Sistem dapat mencatat presensi anak didik dalam kegiatan sekolah desa
- d. Sistem dapat menyimpan rancangan pembelajaran (RPP)
- e. Sistem dapat menyimpan materi yang akan diajarkan dalam sekolah desa
- f. Sistem dapat mencatat rapor anak didik sekolah desa
- g. Sistem dapat menyimpan foto – foto kegiatan sekolah desa
- h. Staf dapat melihat dan mengelola semua data kegiatan sekolah desa
- i. *Volunteer* dapat melihat semua data kegiatan sekolah desa, namun hanya dapat mengelola data RPP dan materi saja

3.1.2 Observasi Lapangan

Kegiatan observasi atau survei dilakukan untuk melihat langsung proses kegiatan belajar mengajar yang dilaksanakan komunitas Pendar Foundation. Survei dilaksanakan di salah satu sekolah desa yang terletak di Desa Sanden. Data yang diperoleh dari hasil observasi adalah sebagai berikut:

- a. Kegiatan belajar mengajar dilakukan di salah satu rumah warga
- b. Setiap kegiatan belajar diisi oleh 2-4 orang pengajar yang terdiri dari staf pengajar dan *volunteer*
- c. Media belajar yang digunakan adalah papan tulis dan kertas
- d. Presensi dilakukan oleh pengajar dengan memanggil nama anak didik, kemudian dicatat dalam selembar kertas. Nama – nama tersebut nantinya akan dipindahkan ke dalam Microsoft Word
- e. Diakhir sesi belajar anak didik akan diberikan selembar kertas berisi latihan soal, hasil dari latihan soal tersebut akan dijadikan evaluasi untuk penilaian rapor anak didik

Dokumentasi kegiatan observasi atau survei pada saat pelaksanaan sekolah desa di Desa Sanden dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Survei Kegiatan Belajar Sekolah Desa

32 Desain dan Perancangan

Desain dan perancangan berisikan rancangan *use case diagram*, *activity diagram*, rancangan basis data beserta deskripsi tabel basis data dan juga desain antar muka sistem yang akan dibangun.

3.2.1 Rancangan *Use Case Diagram*

Berdasarkan hasil analisis kebutuhan sistem, tersedia *use case diagram* yang mendeskripsikan fungsi – fungsi yang dapat dilakukan oleh setiap aktor dalam sistem yang akan dibuat.

Pada sistem informasi pendataan sekolah desa komunitas Pendar Foundation yang akan dibangun, terdapat 3 aktor yang dapat mengoperasikan sistem. Aktor – aktor tersebut ialah admin, staf dan *volunteer*. Admin untuk mengatur seluruh pengelolaan data dalam sistem, yaitu kelola aktor, kelola pengguna, kelola sekolah desa dan kelola master data siswa. Admin juga dapat mengubah informasi pada deksripsi hak akses, halaman tentang kami dan halaman bantuan. Staf adalah aktor yang dapat mengelola data sekolah desa, master data siswa. Staf dapat melihat halaman tentang kami dan halaman bantuan. Akun untuk aktor staf akan dibuat oleh admin, sedangkan *volunteer* dapat membuat akun sendiri.

Aktor *volunteer* dapat melihat seluruh data sekolah desa dan master data siswa, namun hanya dapat mengelola dua fitur dalam menu sekolah desa, yaitu kelola RPP dan kelola materi. Aktor *volunteer* diberikan batasan, yaitu hanya dapat mengelola RPP dan materi sekolah desa berdasarkan permintaan dari pihak komunitas Pendar Foundation. Hal ini bertujuan agar *volunteer* tidak dapat mengubah data – data penting dalam sistem, karena *volunteer* bukanlah anggota tetap komunitas Pendar Foundation.

Pada penelitian ini, pengembangan sistem berfokus pada sisi *back-end*, dimana sisi *back-end* membahas logika sistem, seperti kelola aktor, kelola pengguna, kelola sekolah desa dan kelola master data siswa. Pada Gambar 3.2, dapat dilihat *use case diagram* sistem informasi pendataan sekolah desa komunitas Pendar Foundation.



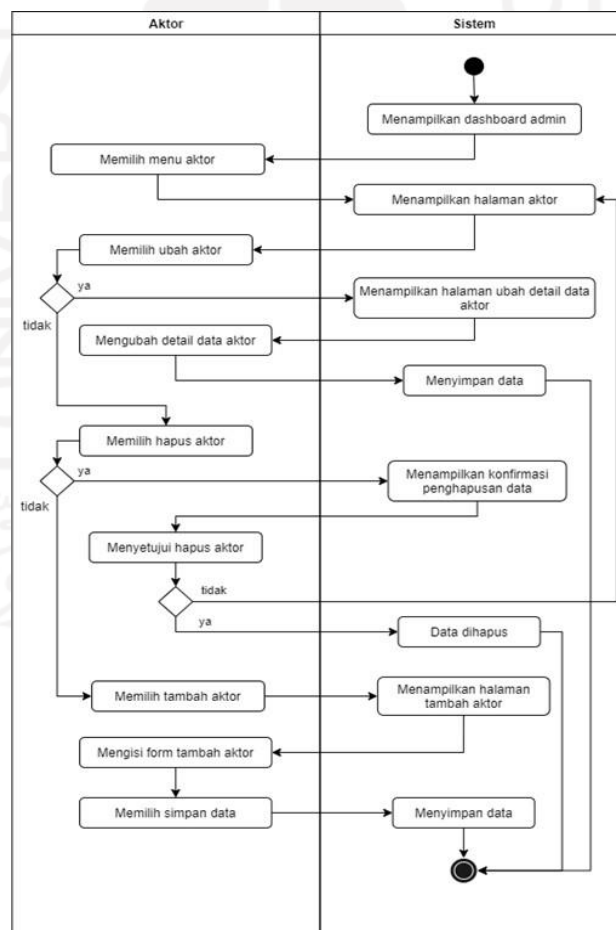
Gambar 3. 2 Use Case Diagram Sistem Informasi Pendataan Sekolah Desa Komunitas Pendar Foundation Yogyakarta

3.2.2 Rancangan Activity Diagram

Dari *use case diagram* pada gambar 3.1, tiap – tiap *use case* akan digambarkan alurnya menggunakan *activity diagram*. *Activity diagram* mendeskripsikan aliran fungsionalitas dalam suatu sistem. Singkatnya, *activity diagram* menggambarkan di mana *workflow* dimulai dan berhenti, serta urutan kejadian dari aktivitas tersebut (Dewi, Indahyanti, & S, 2012). Berikut adalah penjabaran setiap *use case* menggunakan *activity diagram*.

a. Kelola Aktor

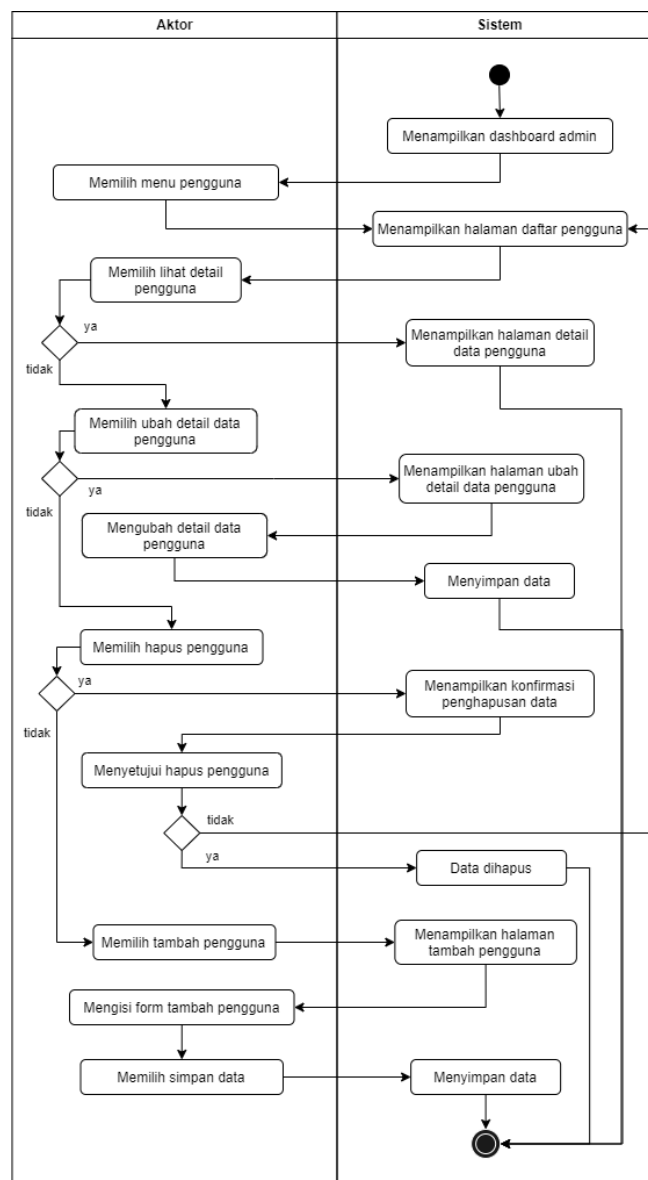
Fungsi kelola aktor hanya dimiliki oleh admin. Untuk melakukan kelola aktor, admin harus memilih menu aktor. Kemudian sistem akan menampilkan halaman aktor yang berisi daftar tiap – tiap aktor. Pada fungsi kelola aktor, admin dapat memilih ubah, hapus atau tambah aktor. Admin memilih salah satu aktor dan memilih aksi yang perlu dilakukan pada aktor tersebut. *Activity diagram* kelola aktor dapat dilihat pada Gambar 3.3



Gambar 3. 3 Activity Diagram Kelola Aktor

b. Kelola Pengguna

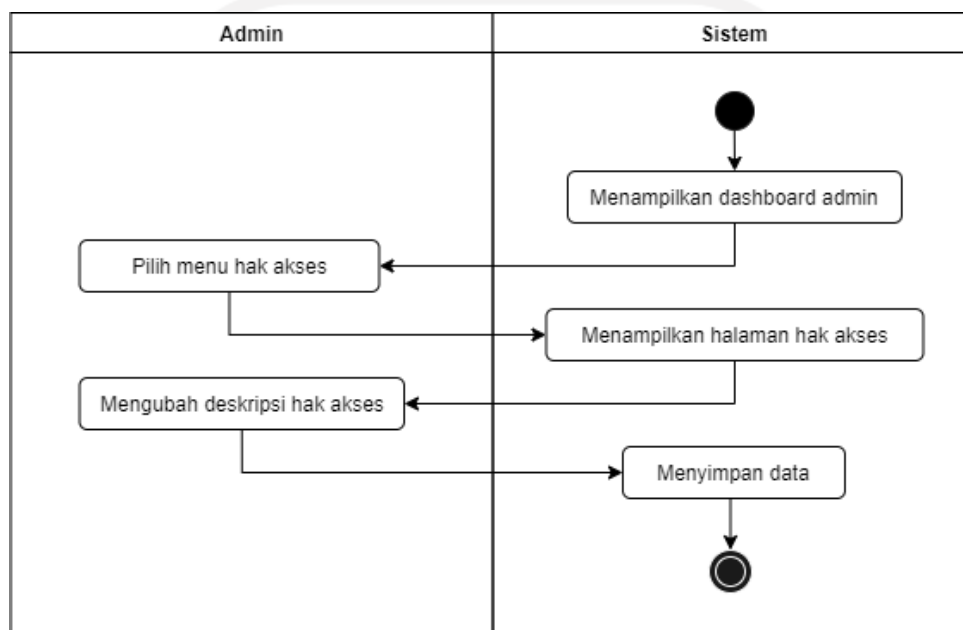
Fungsi kelola pengguna hanya dimiliki oleh Admin. Untuk melakukan kelola pengguna, admin harus memilih menu pengguna, kemudian sistem akan menampilkan halaman pengguna yang berisi daftar seluruh pengguna dalam sistem. Pada fungsi kelola pengguna, admin akan memilih aksi yang diperlukan, yaitu ubah, hapus atau tambah pengguna, khususnya akun pengguna staf. Staf akan dibuatkan akun oleh admin. *Activity diagram* kelola pengguna dapat dilihat pada Gambar 3.4.



Gambar 3. 4 *Activity Diagram* Kelola Pengguna

c. Ubah Deskripsi Hak Akses

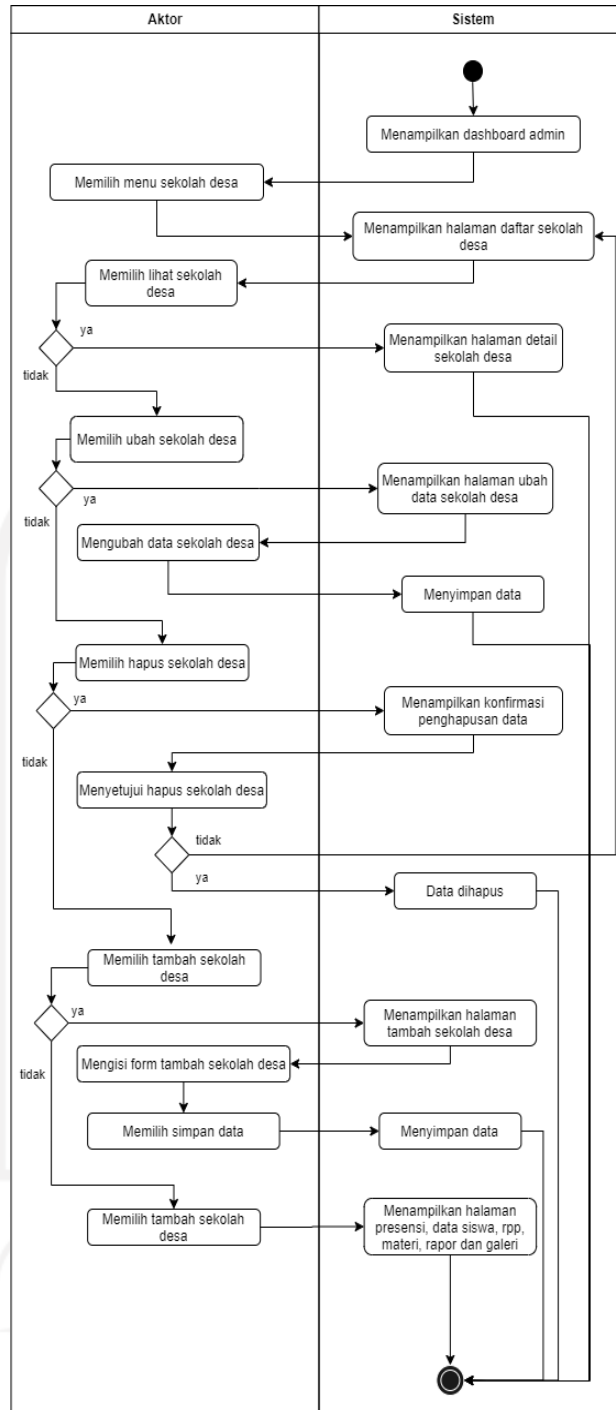
Fungsi ubah deskripsi hak akses hanya dimiliki oleh Admin. Untuk mengubah deskripsi hak akses, admin harus memilih menu hak akses, kemudian sistem akan menampilkan halaman hak akses yang berupa daftar nama – nama hak akses yang ada, admin akan mengubah deskripsi hak akses, setelah itu data yang diubah akan disimpan oleh sistem. Deskripsi hak akses berfungsi untuk menjelaskan fungsi dari setiap hak akses yang ada agar mempermudah admin saat ingin menentukan hak akses untuk tiap – tiap aktor. *Activity diagram* ubah deskripsi hak akses dapat dilihat pada Gambar 3.5.



Gambar 3. 5 *Activity Diagram* Ubah Deskripsi Hak Akses

d. Kelola Sekolah Desa

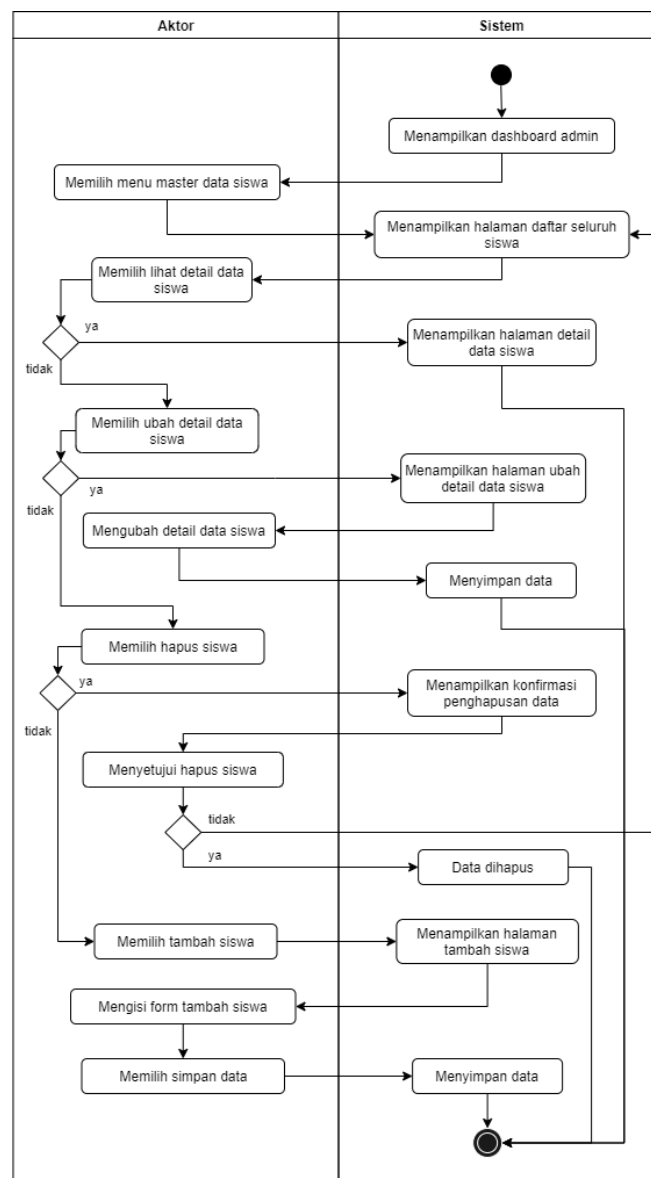
Fungsi kelola sekolah desa dimiliki oleh admin dan staf. Untuk melakukan kelola sekolah desa, staf atau admin harus memilih menu sekolah desa, kemudian sistem akan menampilkan halaman detail sekolah desa. Pada fungsi kelola sekolah desa, admin atau staf memilih aksi yang diperlukan, yaitu ubah, hapus atau tambah sekolah desa. *Activity diagram* kelola sekolah desa dapat dilihat pada Gambar 3.6.



Gambar 3. 6 Activity Diagram Kelola Sekolah Desa

e. Kelola Master Data Siswa

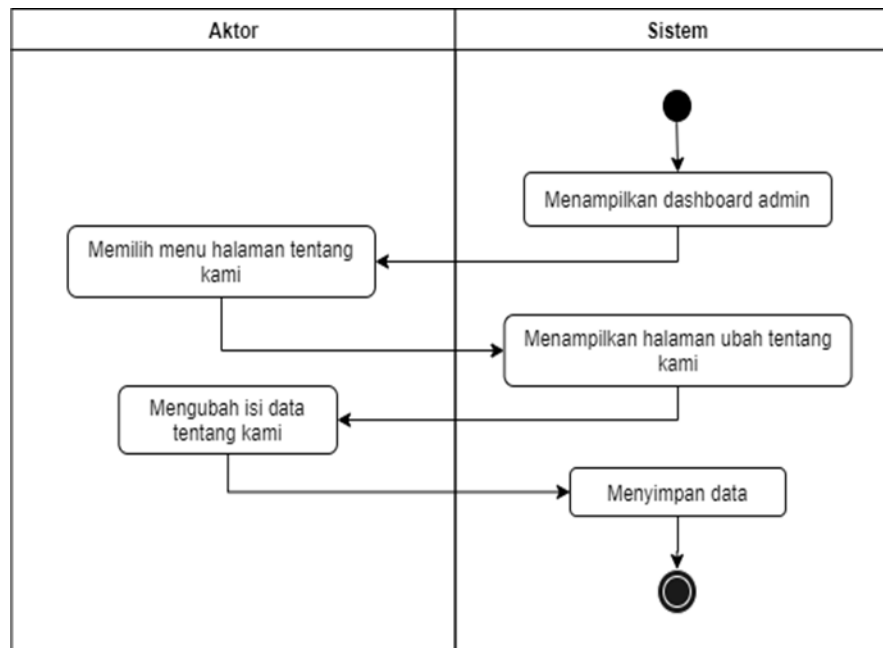
Fungsi kelola master data siswa dimiliki oleh admin dan staf. Untuk mengelola master data siswa, admin atau staf harus memilih menu master data siswa, kemudian sistem akan menampilkan halaman master data siswa yang berisi daftar seluruh siswa sekolah desa. Pada fungsi kelola master data siswa, admin atau staf memilih aksi yang diperlukan, yaitu ubah, hapus atau tambah siswa baru. *Activity diagram* kelola master data siswa dapat dilihat pada Gambar 3.7.



Gambar 3.7 Activity Diagram Kelola Master Data Siswa

f. Ubah Tentang Kami

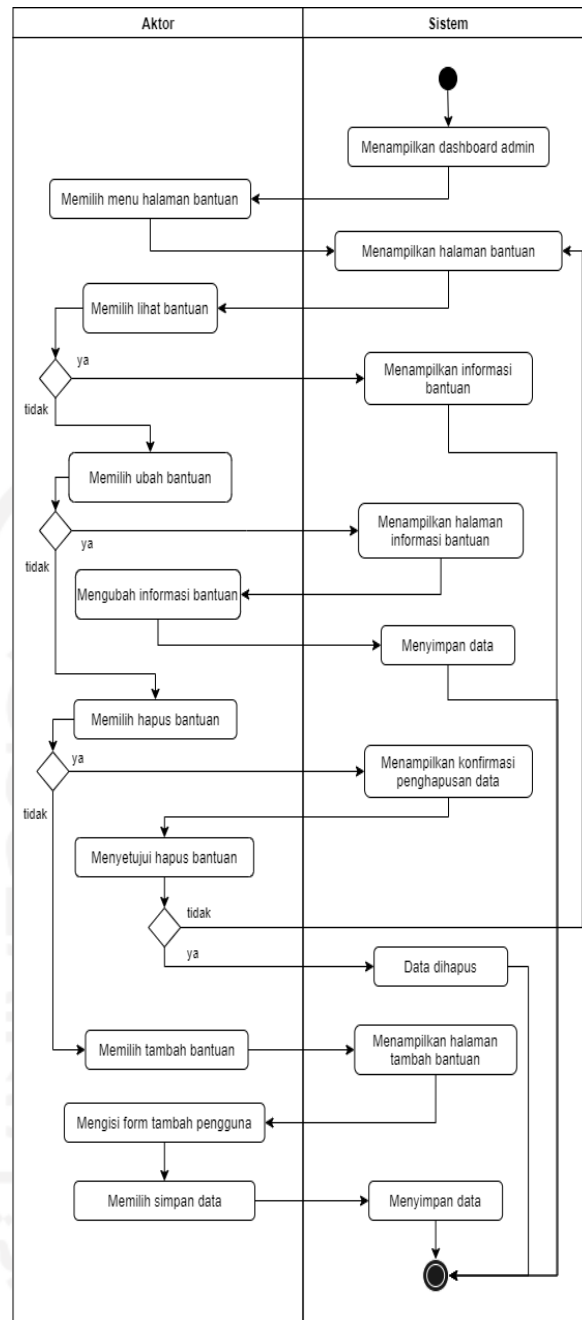
Fungsi ubah tentang kami hanya dimiliki oleh admin. Untuk mengubah data di halaman tentang kami, admin harus memilih menu tentang kami, kemudian sistem akan menampilkan halaman ubah tentang kami. Admin mengubah data tentang kami dan sistem akan menyimpan data yang diubah. *Activity diagram* ubah tentang kami dapat dilihat pada Gambar 3.8.



Gambar 3. 8 *Activity Diagram* Ubah Tentang Kami

g. Kelola Bantuan

Fungsi kelola bantuan hanya dimiliki oleh admin. Untuk mengelola informasi pada halaman bantuan, admin harus memilih menu bantuan, kemudian sistem akan menampilkan halaman ubah bantuan. Admin memilih aksi yang diperlukan, yaitu lihat, ubah, hapus atau tambah bantuan. *Activity diagram* kelola bantuan dapat dilihat pada Gambar 3.9.

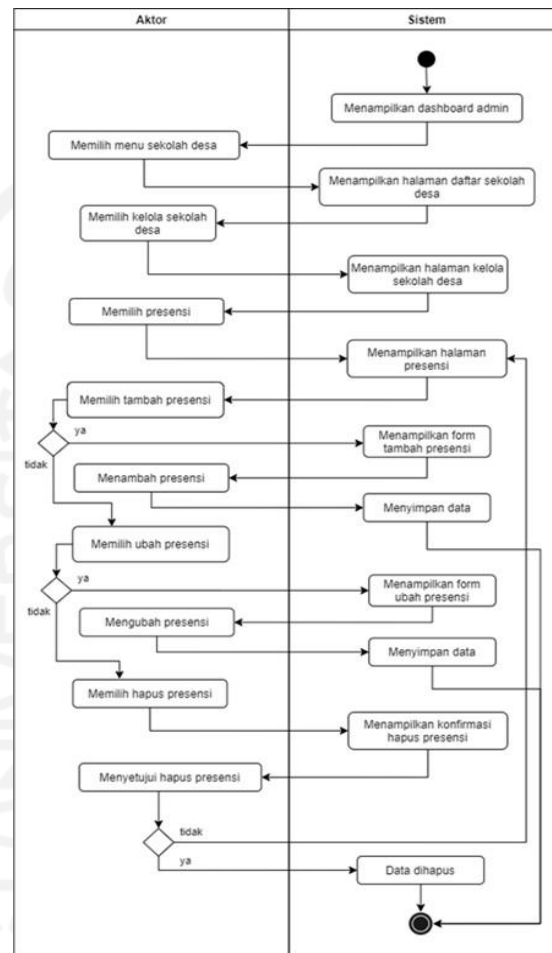


Gambar 3. 9 Activity Diagram Ubah Bantuan

h. Kelola Presensi

Fungsi kelola presensi dimiliki oleh admin dan staf. Kelola presensi merupakan fungsi yang terdapat di dalam menu sekolah desa. Untuk melakukan kelola presensi, admin atau staf harus memilih menu sekolah desa, kemudian sistem akan menampilkan daftar sekolah desa. Admin memilih salah satu sekolah desa dan memilih aksi kelola, kemudian

sistem akan menampilkan halaman sekolah desa yang terdiri dari presensi, data siswa, RPP, materi, rapor dan galeri. Admin atau staf memilih presensi, kemudian sistem akan menampilkan halaman presensi. Pada fungsi kelola presensi, admin atau staf memilih aksi yang diperlukan, yaitu ubah, hapus atau tambah presensi. *Activity diagram* kelola presensi dapat dilihat pada Gambar 3.10.

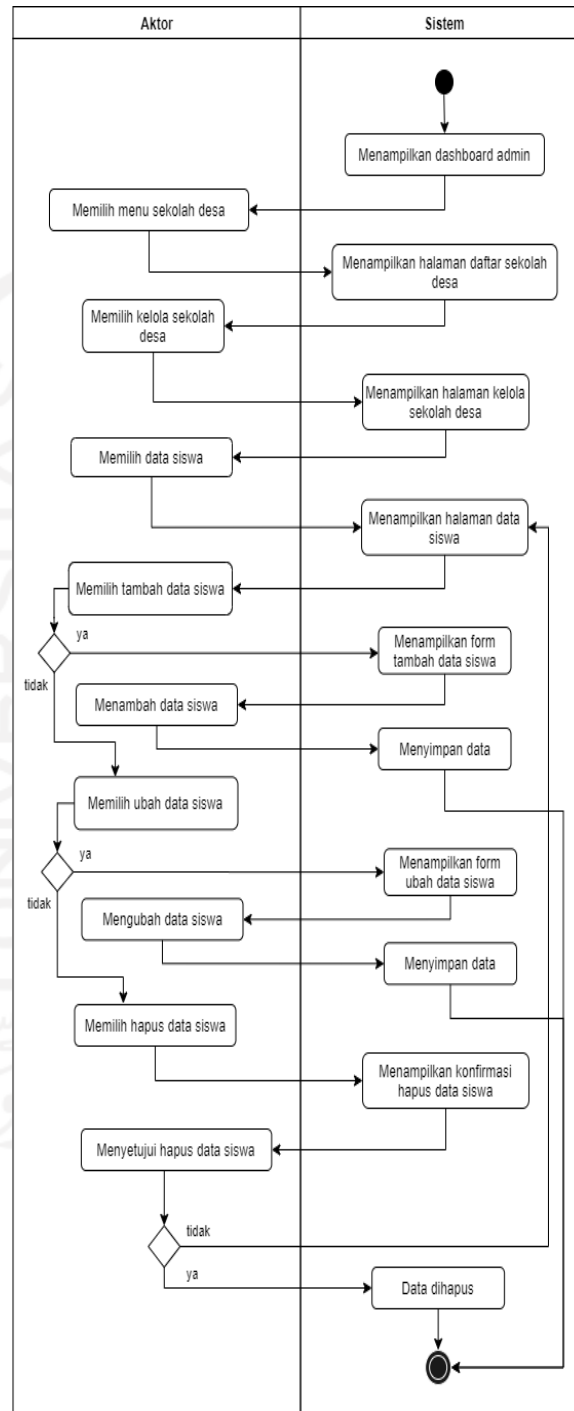


Gambar 3. 10 *Activity Diagram* Kelola Presensi

i. Kelola Data Siswa

Fungsi kelola data siswa dimiliki oleh admin dan staf. Kelola data siswa merupakan fungsi yang terdapat di dalam menu sekolah desa. Untuk mengelola data siswa admin atau staf harus memilih menu sekolah desa, kemudian sistem akan menampilkan daftar sekolah desa. Admin memilih salah satu sekolah desa dan memilih aksi kelola, kemudian sistem akan menampilkan halaman sekolah desa yang terdiri dari presensi, data siswa, RPP, materi, rapor dan galeri. Admin atau staf memilih data siswa, kemudian sistem akan

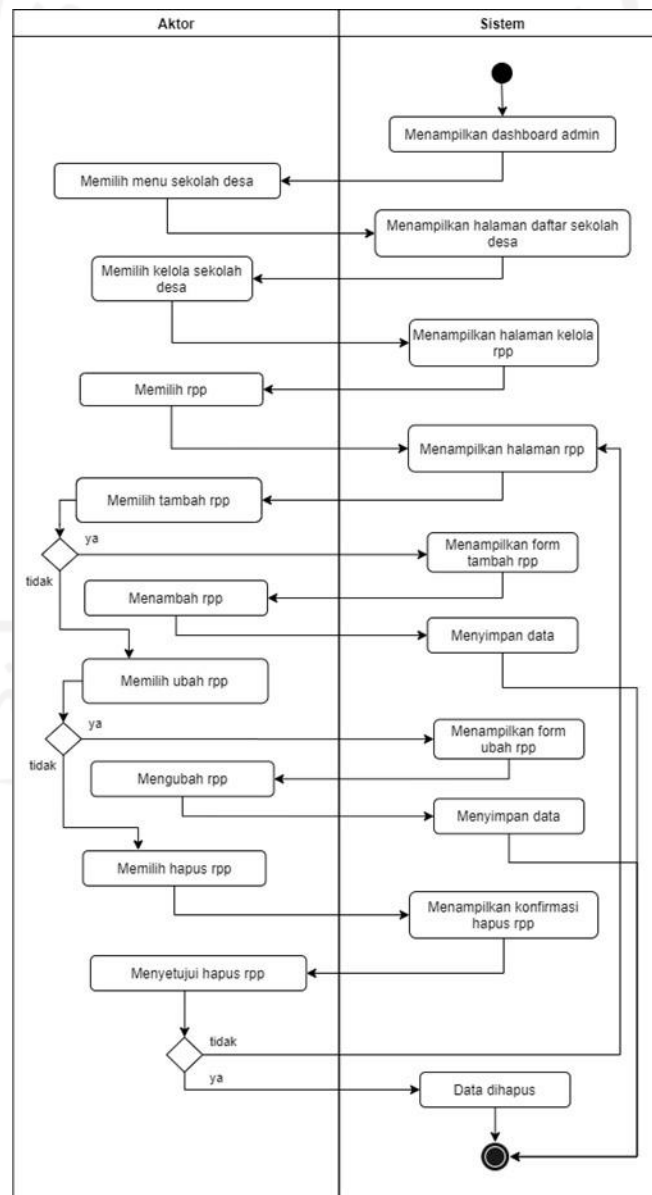
menampilkan halaman data siswa. Pada fungsi kelola data siswa, admin atau staf memilih aksi yang diperlukan yaitu ubah, hapus atau tambah siswa. *Activity diagram* kelola data siswa dapat dilihat pada Gambar 3.11.



Gambar 3. 11 *Activity Diagram* Kelola Data Siswa

j. Kelola RPP

Fungsi kelola RPP dimiliki oleh admin, staf dan *volunteer*. Kelola RPP merupakan fungsi yang terdapat di dalam menu sekolah desa. Untuk mengelola RPP, admin, staf atau *volunteer* harus memilih menu sekolah desa, kemudian sistem akan menampilkan daftar sekolah desa. Admin, staf atau *volunteer* memilih salah satu sekolah desa dan memilih aksi kelola, kemudian sistem akan menampilkan halaman sekolah desa yang terdiri dari presensi, data siswa, RPP, materi, rapor dan galeri. Admin, staf atau *volunteer* memilih RPP, kemudian sistem akan menampilkan halaman RPP. Pada fungsi kelola RPP, admin, staf atau *volunteer* memilih aksi yang diperlukan, yaitu ubah, hapus atau tambah RPP. *Activity diagram* kelola RPP dapat dilihat pada Gambar 3.12.

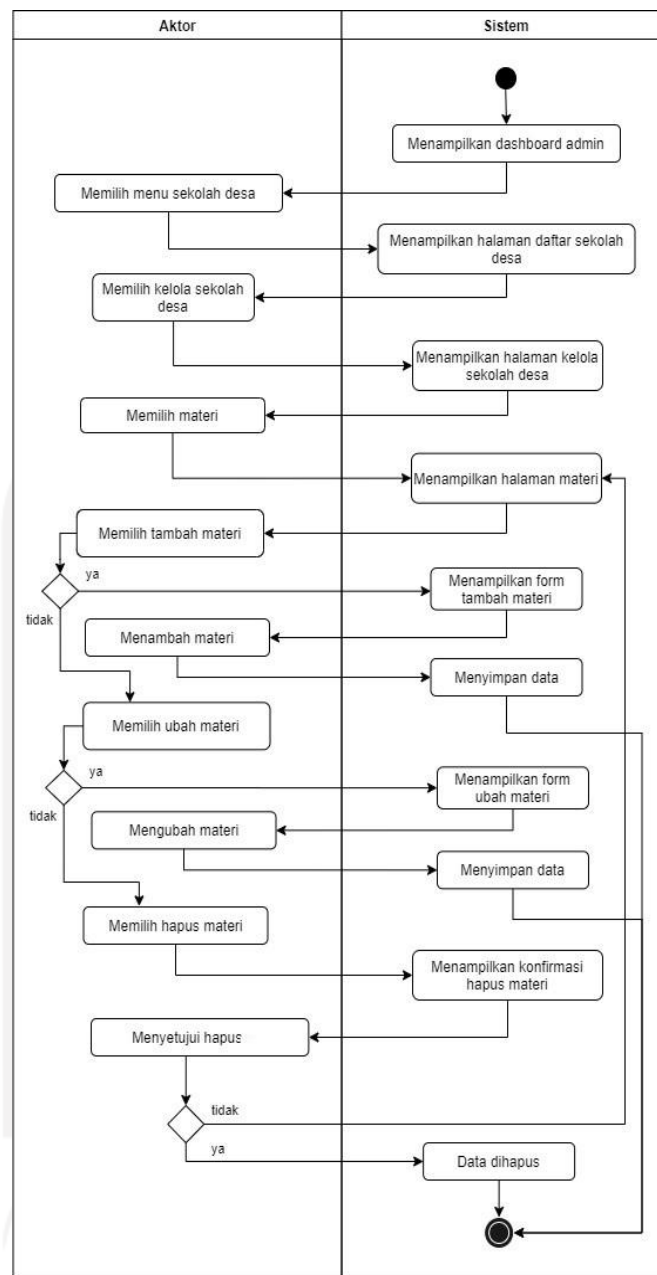


Gambar 3. 12 *Activity Diagram* Kelola RPP

k. Kelola Materi

Fungsi kelola materi dimiliki oleh admin, staf dan *volunteer*. Kelola materi merupakan fungsi yang terdapat di dalam menu sekolah desa. Untuk mengelola materi, admin, staf atau *volunteer* harus memilih menu sekolah desa, kemudian sistem akan menampilkan daftar sekolah desa. Admin, staf atau *volunteer* memilih salah satu sekolah desa dan memilih aksi kelola, kemudian sistem akan menampilkan halaman sekolah desa yang terdiri dari presensi, data siswa, RPP, materi, rapor dan galeri. Admin, staf atau *volunteer* memilih materi, kemudian sistem akan menampilkan halaman materi. Pada fungsi kelola materi, admin, staf atau *volunteer* memilih aksi yang diperlukan, yaitu ubah, hapus atau tambah materi. *Activity diagram* kelola materi dapat dilihat pada Gambar 3.13.



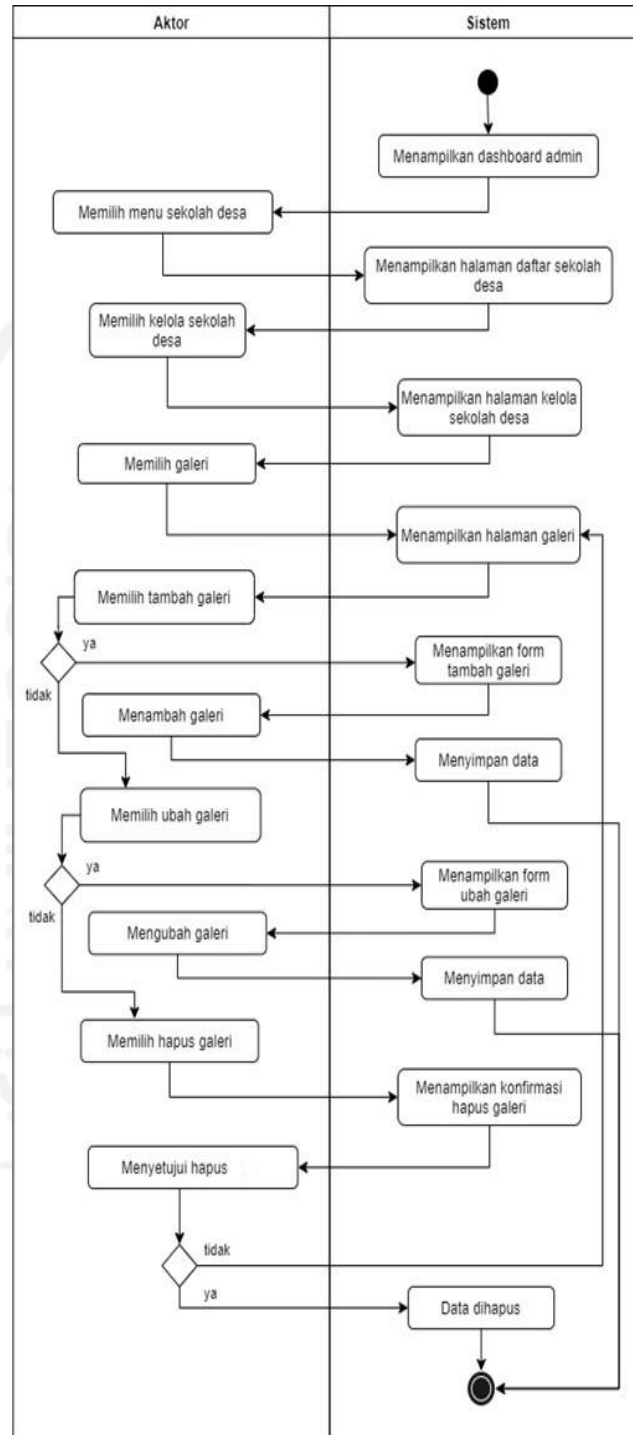


Gambar 3. 13 Activity Diagram Kelola Materi

I. Kelola Galeri

Fungsi kelola galeri dimiliki oleh admin dan staf. Kelola galeri merupakan fungsi yang terdapat di dalam menu sekolah desa. Untuk mengelola galeri, admin atau staf harus memilih menu sekolah desa, kemudian sistem akan menampilkan daftar sekolah desa. Admin atau staf memilih salah satu sekolah desa dan memilih aksi kelola, kemudian sistem akan menampilkan halaman sekolah desa yang terdiri dari presensi, data siswa, RPP, materi, rapor dan galeri. Admin atau staf memilih galeri dan sistem akan

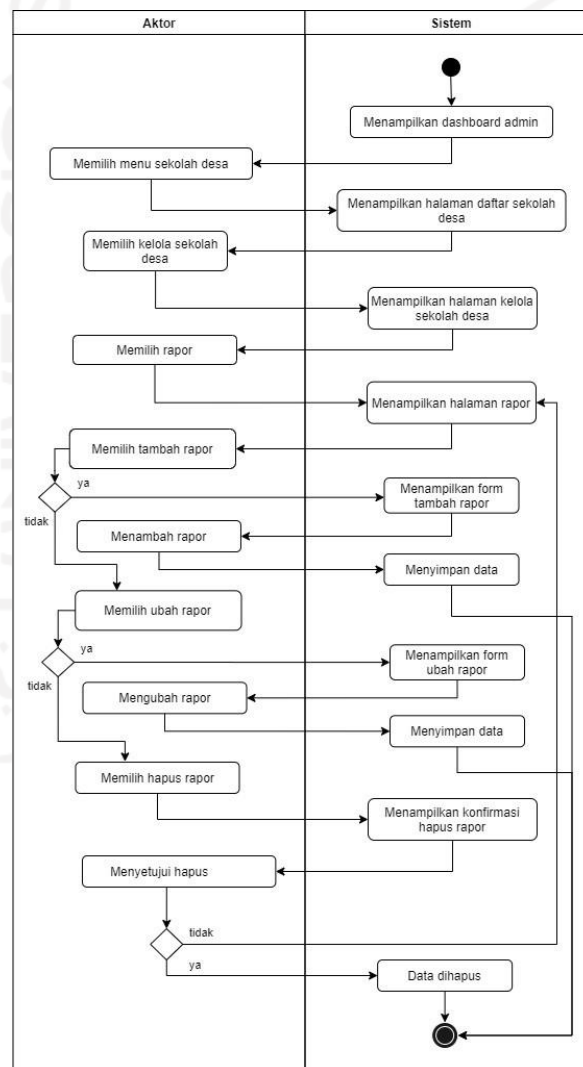
menampilkan halaman galeri. Pada kelola galeri, admin atau staf memilih aksi yang diperlukan, yaitu ubah, hapus atau tambah galeri. *Activity diagram* kelola galeri dapat dilihat pada Gambar 3.14.



Gambar 3. 14 *Activity Diagram* Kelola Galeri

m. Kelola Rapor

Fungsi kelola rapor dimiliki oleh admin dan staf. Kelola rapor merupakan fungsi yang terdapat di dalam menu sekolah desa. Untuk mengelola rapor, admin atau staf harus memilih menu sekolah desa, kemudian sistem akan menampilkan daftar sekolah desa. Admin atau staf memilih salah satu sekolah desa dan memilih aksi kelola, kemudian sistem akan menampilkan halaman sekolah desa yang terdiri dari presensi, data siswa, RPP, materi, rapor dan galeri. Admin atau staf memilih rapor dan sistem akan menampilkan halaman rapor. Pada fungsi kelola rapor, admin atau staf memilih aksi yang diperlukan, yaitu ubah, hapus atau tambah rapor. *Activity diagram* kelola rapor dapat dilihat pada Gambar 3.15.

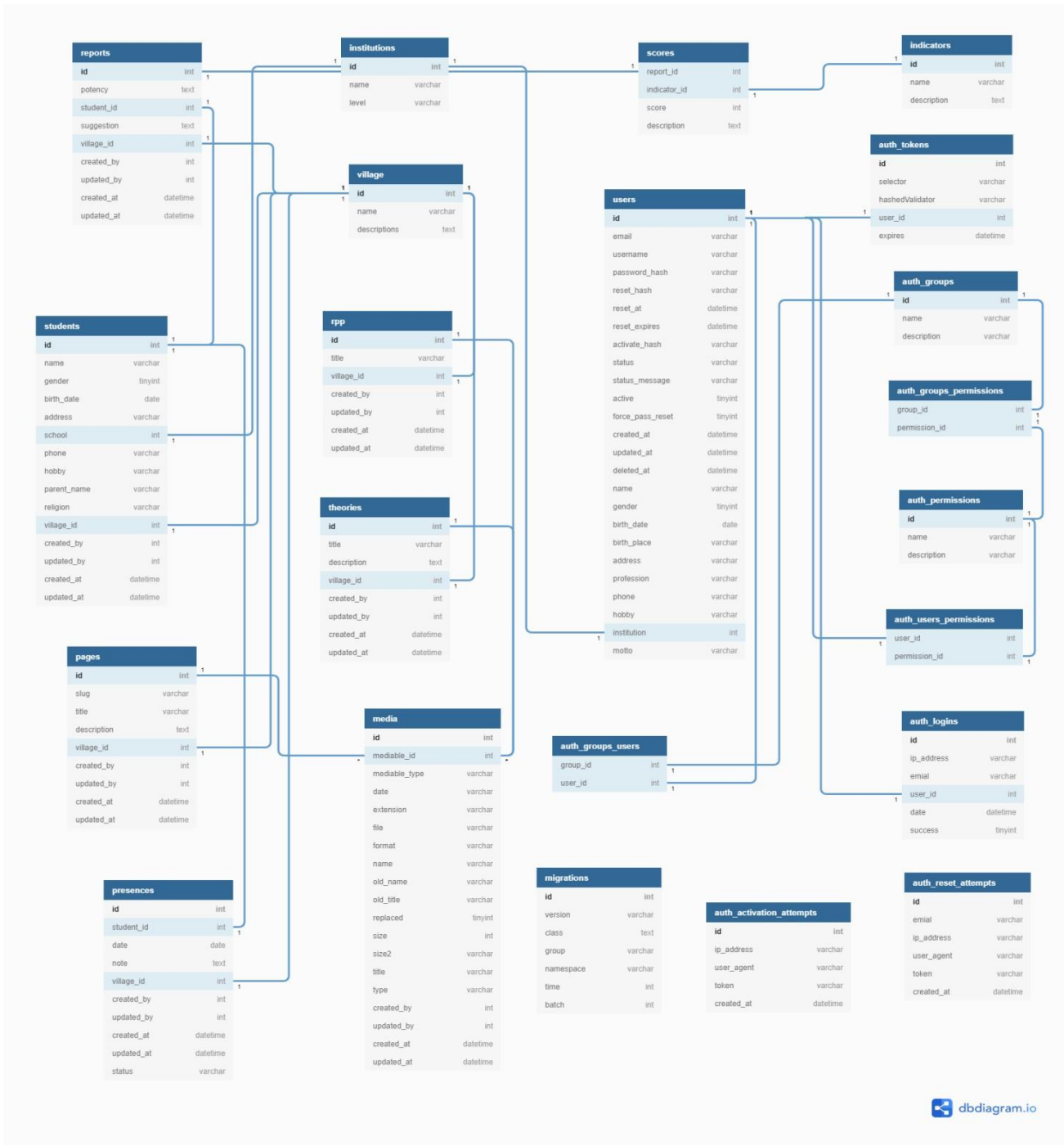


Gambar 3. 15 *Activity Diagram* Kelola Rapor

3.2.3 Perancangan Basis Data

3.2.3.1 Relasi Basis Data

Diagram relasi tabel basis data pada pengembangan sistem informasi pendataan Sekolah Desa Komunitas Pendar Foundation Yogyakarta dapat dilihat pada Gambar 3.16.



Gambar 3. 16 Relasi Basis Data Sistem Informasi Pendataan Sekolah Desa Komunitas Pendar Foundation Yogyakarta

3.2.3.2 Deskripsi Tabel Basis Data

Struktur tabel adalah rancangan tabel yang menjelaskan tipe, ukuran dan keterangan dari *field*. struktur tabel basis data sistem, antara lain:

1. Nama tabel: *villages*

Tabel *villages* merupakan tabel yang memuat data desa pada sekolah desa Pendar Foundation (lihat tabel 3.1).

Tabel 3. 1 Tabel *villages*

No	Nama Field	Tipe Data	<i>Data Length</i>	Keterangan
1	id (pk)	<i>int</i>	11	id desa
2	name	<i>varchar</i>	100	nama desa
3	descriptions	<i>text</i>		deskripsi tentang desa

2. Nama tabel: *users*

Tabel *users* merupakan tabel yang memuat data pengguna yang dapat mengakses sistem. Data pengguna meliputi data admin, data staf dan data *volunteer*. Tabel *users* memiliki *institution* yang berfungsi sebagai *foreign key* dan berelasi dengan tabel *institution* (lihat tabel 3.2).

Tabel 3. 2 Tabel *users*

No	Nama Field	Tipe Data	<i>Data Length</i>	Keterangan
1	id (pk)	<i>int</i>	11	id pengguna
2	email (unique)	<i>varchar</i>	255	email pengguna
3	username (unique)	<i>varchar</i>	30	nama pengguna dalam sistem
4	password_hash	<i>varchar</i>	255	password pengguna yang terenkripsi
5	reset_hash	<i>varchar</i>	255	Password yang direset
6	reset_at	<i>datetime</i>		tanggal password

				direset
7	reset_expires	<i>datetime</i>		Masa aktif reset password
8	activate_hash	<i>varchar</i>	255	Kode verifikasi untuk reset akun
9	status	<i>varchar</i>	255	status pengguna
10	status_message	<i>varchar</i>	255	pesan status pengguna
11	active	<i>tinyint</i>	1	Status verifikasi pengguna
12	Force_pass_reset	<i>tinyint</i>	1	Mewajibkan pengguna ganti password ketika login
13	created_at	<i>datetime</i>		tanggal data dibuat
14	updated_at	<i>datetime</i>		tanggal data diupdate
15	deleted_at	<i>datetime</i>		tanggal data dihapus
16	name	<i>varchar</i>	100	nama pengguna
17	gender	<i>tinyint</i>	1	jenis kelamin
18	birth_date	<i>date</i>		tanggal lahir
19	birth_place	<i>varchar</i>	100	tempat lahir
20	address	<i>varchar</i>	100	alamat
21	profession	<i>varchar</i>	100	profesi
22	phone	<i>varchar</i>	100	nomor telepon
23	hobby	<i>varchar</i>	100	hobi
24	institution (fk)	<i>int</i>	11	institusi
25	motto	<i>varchar</i>	100	moto hidup

3. Nama tabel: *theories*

Tabel *theories* merupakan tabel yang memuat data materi pelajaran sekolah desa. Tabel *theories* berelasi dengan tabel *villages* dan tabel *users* (lihat tabel 3.3).

Tabel 3. 3 Tabel *theories*

No	Nama Field	Tipe Data	Data Length	Keterangan
1	id (pk)	<i>int</i>	11	id materi
2	title	<i>varchar</i>	100	judul materi
3	description	<i>text</i>		deskripsi materi
4	village_id (fk)	<i>int</i>	11	id desa
5	created_by (fk)	<i>int</i>	11	data dibuat oleh
6	updated_by (fk)	<i>int</i>	11	data diubah oleh
7	created_at	<i>datetime</i>		tanggal data dibuat
8	updated_at	<i>datetime</i>		tanggal data diupdate

4. Nama tabel: *students*

Tabel *students* merupakan tabel yang memuat data siswa didik sekolah desa komunitas Pendar Foundation. Tabel *students* berelasi dengan tabel *villages*, tabel *institutions* dan tabel *users* (lihat tabel 3.4).

Tabel 3. 4 Tabel *students*

No	Nama Field	Tipe Data	Data Length	Keterangan
1	id (pk)	<i>int</i>	11	id siswa
2	name	<i>varchar</i>	100	nama siswa
3	gender	<i>tinyint</i>	1	jenis kelamin
4	birth_date	<i>date</i>		tanggal lahir
5	birth_place	<i>varchar</i>	100	tempat lahir
6	address	<i>varchar</i>	100	alamat siswa
7	school (fk)	<i>int</i>	11	asal sekolah
8	phone	<i>varchar</i>	100	nomor telepon

9	hobby	<i>varchar</i>	100	hobi siswa
10	parent_name	<i>varchar</i>	100	nama orang tua siswa
11	religion	<i>varchar</i>	100	agama
12	village_id (fk)	<i>int</i>	11	id desa
13	created_by (fk)	<i>int</i>	11	data dibuat oleh
14	updated_by (fk)	<i>int</i>	11	data diupdate oleh
15	created_at	<i>datetime</i>		tanggal data dibuat
16	updated_at	<i>datetime</i>		tanggal data diupdate

5. Nama tabel: *scores*

Tabel *scores* merupakan tabel yang berfungsi sebagai penghubung antara tabel *reports* dan tabel *indicators*. Tabel *scores* memiliki relasi dengan tabel *reports* dan tabel *indicators* (lihat tabel 3.5).

Tabel 3. 5 Tabel *scores*

No	Nama Field	Tipe Data	<i>Data Length</i>	Keterangan
1	report_id (fk)	<i>int</i>	11	id rapor
2	indicator_id (fk)	<i>int</i>	11	id indikator
3	score	<i>int</i>	3	skor
4	description	<i>text</i>		deskripsi

6. Nama tabel: *rpp*

Tabel *rpp* merupakan tabel yang memuat data RPP (Rencana Pelaksanaan Pembelajaran) sekolah desa. Tabel *rpp* berelasi dengan tabel *villages* dan tabel *users* (lihat tabel 3.6).

Tabel 3. 6 Tabel rpp

No	Nama Field	Tipe Data	Data Length	Keterangan
1	id (pk)	<i>int</i>	11	id rpp
2	title	<i>varchar</i>	100	judul rpp
3	village_id (fk)	<i>int</i>	11	id desa
4	created_by (fk)	<i>int</i>	11	data dibuat oleh
5	updated_by (fk)	<i>int</i>	11	data diupdate oleh
6	created_at	<i>datetime</i>		tanggal data dibuat
7	updated_at	<i>datetime</i>		tanggal data diupdate

7. Nama tabel: *reports*

Tabel *reports* merupakan tabel yang memuat data rapot siswa sekolah desa. Tabel *reports* berelasi dengan tabel *students*, tabel *villages* dan tabel *users* (lihat tabel 3.7).

Tabel 3. 7 Tabel *reports*

No	Nama Field	Tipe Data	Data Length	Keterangan
1	id (pk)	<i>int</i>	11	id rapor
2	potency	<i>text</i>		potensi siswa
3	student_id (fk)	<i>int</i>	11	id siswa
4	suggestion	<i>text</i>		saran
5	village_id (fk)	<i>int</i>	11	id desa
6	created_by (fk)	<i>int</i>	11	data dibuat oleh
7	updated_by (fk)	<i>int</i>	11	data diupdate oleh
8	created_at	<i>datetime</i>		tanggal data dibuat
9	updated_by	<i>datetime</i>		tanggal data diupdate

8. Nama tabel: *presences*

Tabel *presences* merupakan tabel yang memuat data presensi siswa sekolah desa. Tabel *presences* memiliki relasi dengan tabel *students*, tabel *villages* dan tabel *users* (lihat tabel 3.8).

Tabel 3. 8 Tabel *presences*

No	Nama Field	Tipe Data	Data Length	Keterangan
1	id (pk)	<i>int</i>	11	id presensi
2	student_id (fk)	<i>int</i>	11	id siswa
3	date	<i>date</i>		tanggal
4	note	<i>text</i>		catatan
5	village_id (fk)	<i>int</i>	11	id desa
6	created_by (fk)	<i>int</i>	11	data dibuat oleh
7	updated_by (fk)	<i>int</i>	11	data diupdate oleh
8	created_at	<i>datetime</i>		tanggal data dibuat
9	updated_at	<i>datetime</i>		tanggal data diupdate
10	status	<i>varchar</i>	255	status kehadiran

9. Nama tabel: *pages*

Tabel *pages* merupakan tabel yang memuat data yang memberikan penjelasan mengenai tentang kami (Pendar Foundation) dan halaman bantuan yang berisi panduan penggunaan sistem bagi pengguna. Tabel *pages* berelasi dengan tabel *villages* dan tabel *users* (lihat tabel 3.9).

Tabel 3. 9 Tabel *pages*

No	Nama Field	Tipe Data	Data Length	Keterangan
1	id (pk)	<i>int</i>	11	id halaman
2	slug	<i>varchar</i>	100	penamaan untuk url
3	title	<i>varchar</i>	100	judul halaman
4	description	<i>text</i>		deskripsi halaman

5	village_id (fk)	<i>int</i>	11	id desa
6	created_by (fk)	<i>int</i>	11	data dibuat oleh
7	updated_by (fk)	<i>int</i>	11	data diupdate oleh
8	created_at	<i>datetime</i>		tanggal data dibuat
9	updated_at	<i>datetime</i>		tanggal data diupdate

10. Nama tabel: *migrations*

Tabel *migrations* merupakan tabel yang memuat data migrasi yang ada pada sistem. Tabel *migrations* adalah tabel bawaan dari sistem (lihat tabel 3.10).

Tabel 3. 10 Tabel *migrations*

No	Nama Field	Tipe Data	<i>Data Length</i>	Keterangan
1	id (pk)	<i>bigint</i>	20	id migrations
2	version	<i>varchar</i>	255	tanggal data dibuat
3	class	<i>text</i>		nama file yang telah dimigrasi
4	group	<i>varchar</i>	255	pengelompokkan migrasi
5	namespace	<i>varchar</i>	255	nama lokasi file yang telah dimigrasi
6	time	<i>int</i>	11	waktu migrasi
7	batch	<i>int</i>	11	tahapan migrasi

11. Nama tabel: media

Tabel media merupakan tabel yang memuat semua data yang diunggah kedalam sistem. Tabel media berelasi dengan tabel *user*. Tabel media adalah tabel bawaan dari sistem (lihat tabel 3.11).

Tabel 3. 11 Tabel media

No	Nama Field	Tipe Data	Data Length	Keterangan
1	id (pk)	<i>int</i>	11	id media
2	mediable_id	<i>int</i>	11	id mediable
3	mediable_type	<i>varchar</i>	255	tipe mediable
4	date	<i>varchar</i>	255	waktu file tersimpan
5	extension	<i>varchar</i>	255	jenis file
6	file	<i>varchar</i>	255	nama lokasi file disimpan
7	format	<i>varchar</i>	255	format file
8	name	<i>varchar</i>	255	nama file
9	old_name	<i>varchar</i>	255	nama file sebelumnya
10	old_title	<i>varchar</i>	255	judul file sebelumnya
11	replaced	<i>tinyint</i>	1	jumlah mengganti file
12	size	<i>int</i>	255	ukuran file
13	size2	<i>varchar</i>	255	ukuran file setelah didalam sistem
14	title	<i>varchar</i>	255	judul file
15	type	<i>varchar</i>	255	tipe file
16	created_by (fk)	<i>int</i>	11	data dibuat oleh
17	updated_by (fk)	<i>int</i>	11	data diubah oleh
18	created_at	<i>datetime</i>		tanggal data dibuat
19	updated_at	<i>datetime</i>		tanggal data diupdate

12. Nama tabel: *institutions*

Tabel *institutions* merupakan tabel yang memuat tingkatan siswa di sekolah. Contohnya siswa berasal dari sekolah dasar, sekolah menengah pertama atau sekolah menengah atas. Tabel ini memiliki id yang berfungsi sebagai *primary key* (lihat tabel 3.12).

Tabel 3. 12 Tabel *institutions*

No	Nama Field	Tipe Data	Data Length	Keterangan
1	id (pk)	<i>int</i>	11	id institusi
2	name	<i>varchar</i>	100	nama institusi
3	level	<i>varchar</i>	100	tingkatan

13. Nama tabel: *indicators*

Tabel *indicators* merupakan tabel yang memuat nilai indikator yang dibutuhkan dalam rapor siswa (lihat tabel 3.13).

Tabel 3. 13 Tabel *indicators*

No	Nama Field	Tipe Data	Data Length	Keterangan
1	id (pk)	<i>int</i>	11	id indikator
2	name	<i>varchar</i>	100	nama indikator
3	description	<i>text</i>		deskripsi indikator

14. Nama tabel: *auth_users_permissions*

Tabel *auth_users_permissions* merupakan tabel relasi antara tabel *users* dan tabel *auth_permissions*. Tabel *auth_users_permissions* merupakan tabel bawaan dari sistem (lihat tabel 3.14).

Tabel 3. 14 Tabel *auth_users_permissions*

No	Nama Field	Tipe Data	<i>Data Length</i>	Keterangan
1	user_id (pk)	<i>int</i>	11	id pengguna
2	permission_id (fk)	<i>int</i>	11	id permission

15. Nama tabel: *auth_reset_attempts*

Tabel *auth_reset_attempts* merupakan tabel yang memuat data pengguna yang melakukan *reset password*. Tabel *auth_reset_attempts* merupakan tabel bawaan dari sistem (lihat table 3.15).

Tabel 3. 15 Tabel *auth_reset_attempts*

No	Nama Field	Tipe Data	<i>Data Length</i>	Keterangan
1	id (pk)	<i>int</i>	11	id <i>auth_reset_attempts</i>
2	email	<i>varchar</i>	255	email pengguna
3	ip_address	<i>varchar</i>	255	ip address pengguna
4	user_agent	<i>varchar</i>	255	nama user
5	token	<i>varchar</i>	255	token
6	created_at	<i>datetime</i>		tanggal data dibuat

16. Nama tabel: *auth_permissions*

Tabel *auth_permissions* merupakan tabel yang memuat data hak akses aktor. Tabel *auth_permissions* merupakan tabel bawaan dari sistem (lihat tabel 3.16).

Tabel 3. 16 Tabel *auth_permissions*

No	Nama Field	Tipe Data	<i>Data Length</i>	Keterangan
1	id (pk)	<i>int</i>	11	id <i>auth_permissions</i>

2	name	<i>varchar</i>	255	nama hak akses
3	description	<i>varchar</i>	255	deskripsi hak akses

17. Nama tabel: *auth_logins*

Tabel *auth_logins* merupakan tabel yang memuat data *login* pengguna. Tabel *auth_logins* merupakan tabel bawaan dari sistem yang berelasi dengan tabel *users* (lihat tabel 3.17).

Tabel 3. 17 Tabel *auth_logins*

No	Nama Field	Tipe Data	<i>Data Length</i>	Keterangan
1	id (pk)	<i>int</i>	11	id <i>auth_logins</i>
2	ip_address	<i>varchar</i>	255	ip address pengguna
3	email (unique)	<i>varchar</i>	255	email pengguna
4	user_id (fk)	<i>int</i>	11	id pengguna
5	date	<i>datetime</i>		waktu login
6	success	<i>tinyint</i>	1	status login

18. Nama tabel: *auth_groups_users*

Tabel *auth_groups_users* merupakan tabel relasi antara tabel *auth_groups* dan tabel *users*. Tabel ini merupakan bawaan dari sistem (lihat tabel 3.18).

Tabel 3. 18 Tabel *auth_groups_users*

No	Nama Field	Tipe Data	<i>Data Length</i>	Keterangan
1	group_id	<i>int</i>	11	id <i>auth_groups</i>
2	user_id	<i>int</i>	11	id pengguna

19. Nama tabel: *auth_groups_permissions*

Tabel *auth_groups_permissions* merupakan tabel relasi antara tabel *auth_groups* dan tabel *auth_permissions*. Tabel ini merupakan tabel bawaan dari sistem (lihat tabel 3.19).

Tabel 3. 19 Tabel *auth_groups_permissions*

No	Nama Field	Tipe Data	Data Length	Keterangan
1	group_id	<i>int</i>	11	id auth_groups
2	permission_id	<i>int</i>	11	id auth_permissions

20. Nama tabel: *auth_groups*

Tabel *auth_groups* merupakan tabel yang memuat data aktor. Tabel ini merupakan tabel bawaan dari sistem (lihat tabel 3.20).

Tabel 3. 20 Tabel *auth_groups*

No	Nama Field	Tipe Data	Data Length	Keterangan
1	id (pk)	<i>int</i>	11	id auth_groups
2	name	<i>varchar</i>	255	nama aktor
3	description	<i>varchar</i>	255	deskripsi

21. Nama tabel: *auth_activation_attempts*

Tabel *auth_activation_attempts* merupakan tabel yang memuat data aktivasi akun. Tabel *auth_activation_attempts* merupakan tabel bawaan dari sistem (lihat tabel 3.21).

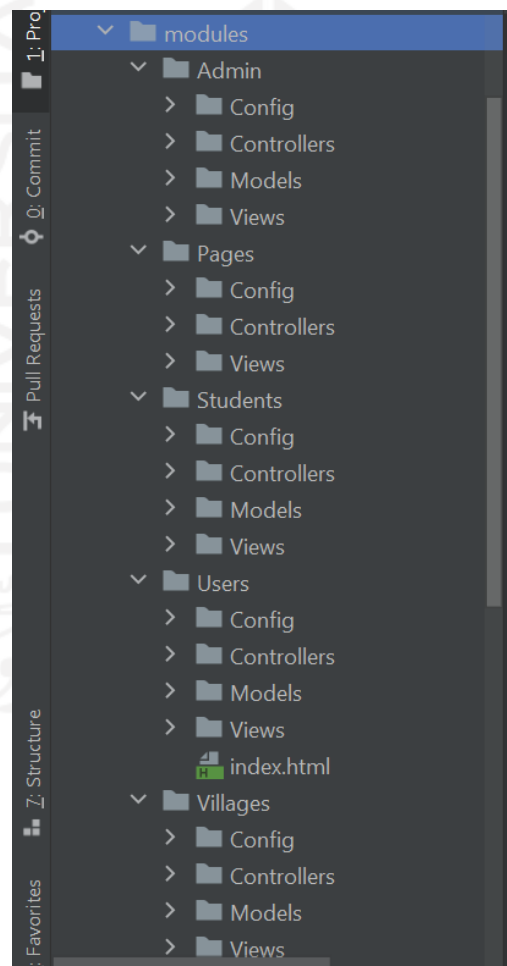
Tabel 3. 21 Tabel *auth_activation_attempts*

No	Nama Field	Tipe Data	Data Length	Keterangan
1	id (pk)	<i>int</i>	11	id auth_activation_attempts
2	ip_address	<i>varchar</i>	255	ip address pengguna
3	user_agent	<i>varchar</i>	255	nama pengguna
4	token	<i>varchar</i>	255	token
5	created_at	<i>datetime</i>		tanggal data dibuat

3.2.4 Rancangan Arsitektur HMVC

Pengembangan *back-end* sistem informasi pendataan sekolah desa komunitas Pendar Foundation menggunakan arsitektur model *Hierarchical Model View Controller* (HMVC). HMVC memungkinkan pengembang untuk mendesain *modules* secara terpisah sebagai kerangka dari direktori MVC (*Model-View-Controller*) dalam base path masing-masing. Teknik arsitektur HMVC cocok digunakan untuk pengerjaan *project* yang memerlukan kerja sama tim, dikarenakan kelebihan fungsinya yang dapat memisahkan pengerjaan *project* menjadi beberapa *modules*, sehingga struktur file lebih tertata rapi dan memberikan kemudahan dalam menyelesaikan tiap – tiap *modules* tanpa mengganggu *modules* lainnya.

Adapun rancangan model arsitektur HMVC pengembangan sistem dapat dilihat pada Gambar 3.17.

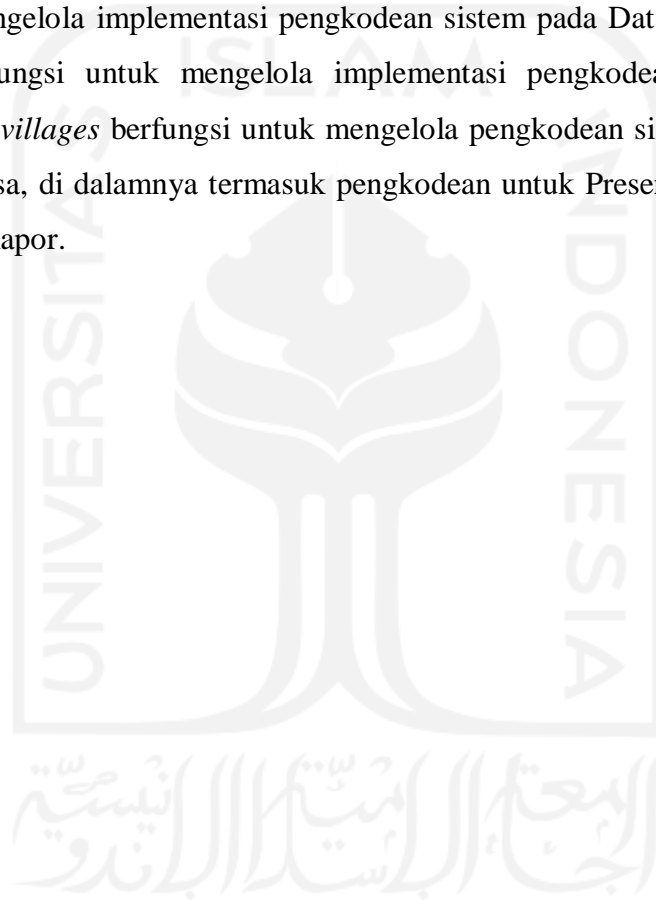


Gambar 3. 17 Rancangan Arsitektur Model HMVC

Pada Gambar 3.17 dapat dilihat bahwa perancangan HMVC digunakan untuk memisahkan tiap – tiap *modules*. Terdapat 5 (lima) buah *modules*, dimana masing – masing

modules memiliki *file MVC (Model-View-Controller)*. Pada pengembangan *back-end*, pengerjaan berfokus pada folder *controller* dan *model*, dimana pada folder *controller* berisi *file controller* yang digunakan untuk mengatur alur proses aliran data atau fungsi yang dibuat. Sedangkan folder *model* berisikan *file* berektensi *php* yang memuat bait program untuk pengelolaan data ke *database*.

Modules admin berfungsi untuk mengelola implementasi pengkodean sistem pada proses kerja Admin. *Modules pages* berfungsi untuk mengelola implementasi pengkodean sistem pada proses kerja menu Tentang Kami, Bantuan dan Artikel. *Modules students* berfungsi untuk mengelola implementasi pengkodean sistem pada Data Siswa sekolah desa. *Modules user* berfungsi untuk mengelola implementasi pengkodean sistem pada data pengguna. *Modules villages* berfungsi untuk mengelola pengkodean sistem pada data – data kegiatan sekolah desa, di dalamnya termasuk pengkodean untuk Presensi, Data Siswa, RPP, Materi, Galeri dan Rapor.



BAB IV

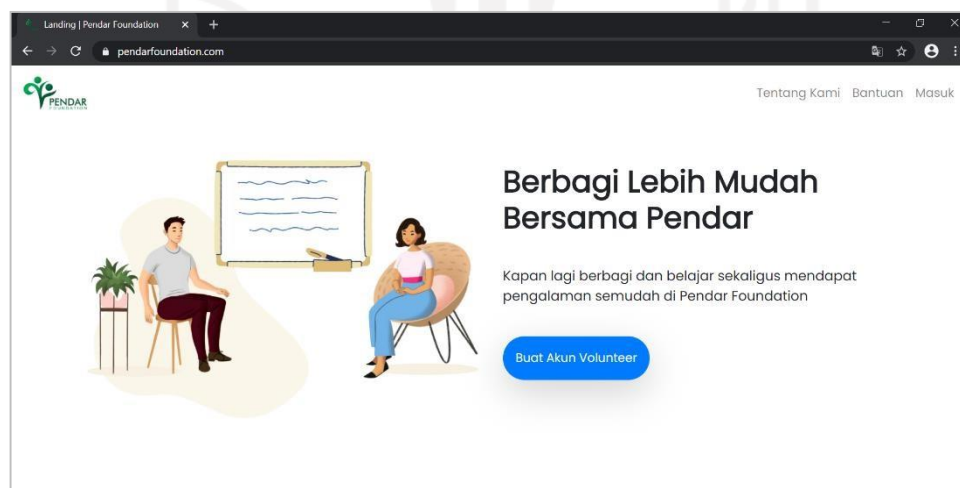
HASIL DAN PEMBAHASAN

41 Implementasi Sistem

Tahap implementasi sistem merupakan tahap dimana sistem telah dibangun sesuai dengan tahap perancangan yang telah dilakukan. Hasil dari sistem yang telah dibangun ditunjukkan melalui *screenshot* dari sistem itu sendiri. Berikut adalah hasil implementasi dari sistem yang telah dibangun.

4.1.1 Halaman *Landing Page*

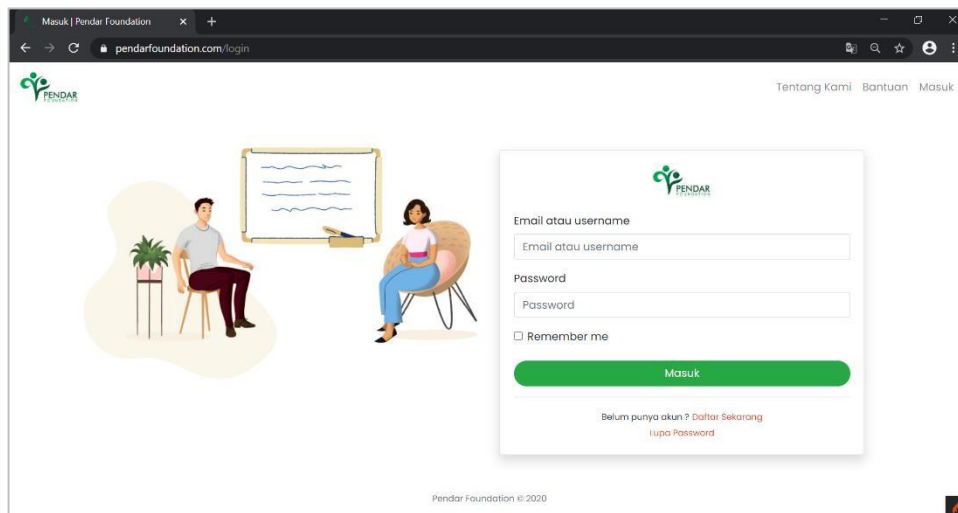
Halaman *landing page* merupakan halaman awal yang akan dilihat oleh pengguna saat mengakses sistem. Jika belum memiliki akun, *volunteer* dapat membuat akun menggunakan *button* yang tersedia. *Volunteer* dan staf juga dapat *login* jika memiliki akun yang sudah terdaftar pada sistem. Selain itu, pada halaman *landing page* pengguna dapat memilih untuk melihat halaman tentang kami dan halaman bantuan. Halaman *landing page* ditunjukkan pada Gambar 4.1



Gambar 4.1 Halaman *Landing Page*

4.1.2 Halaman Login

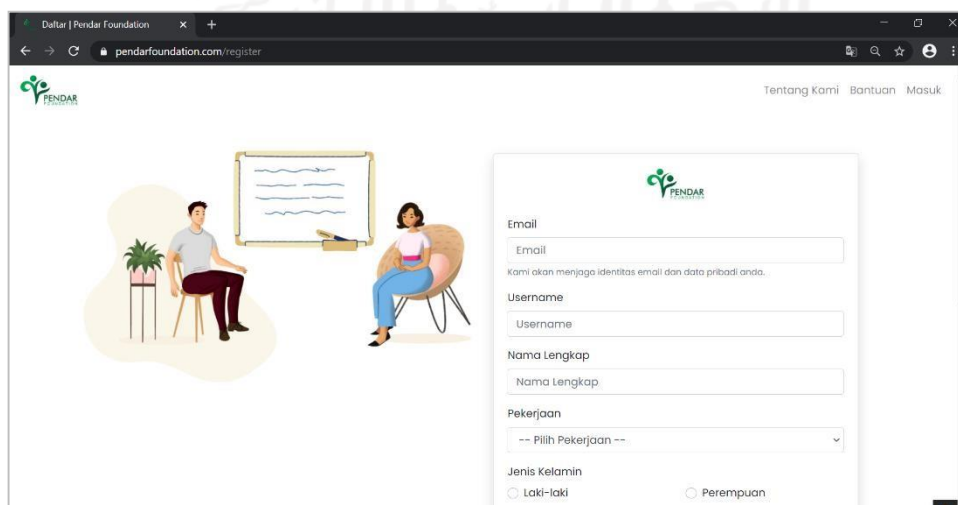
Halaman *login* menampilkan *form* dengan memasukkan berupa email atau *username* dan kata sandi yang harus diisikan oleh pengguna untuk melakukan *login*. Pada halaman *login* juga tersedia pilihan untuk mendaftar akun bagi pengguna yang belum memiliki akun dan pilihan lupa *password*. Halaman *login* ditunjukkan pada Gambar 4.2.



Gambar 4. 2 Halaman *Login*

4.1.3 Halaman Daftar Akun

Halaman daftar akun menampilkan *form* yang harus diisikan oleh pengguna untuk mendaftarkan akunnya sebagai *volunteer*. Daftar akun hanya berlaku bagi *volunteer*, sedangkan akun untuk staf akan didaftarkan oleh admin dengan alasan keamanan. Halaman daftar akun ditunjukkan pada Gambar 4.3 dan Gambar 4.4.

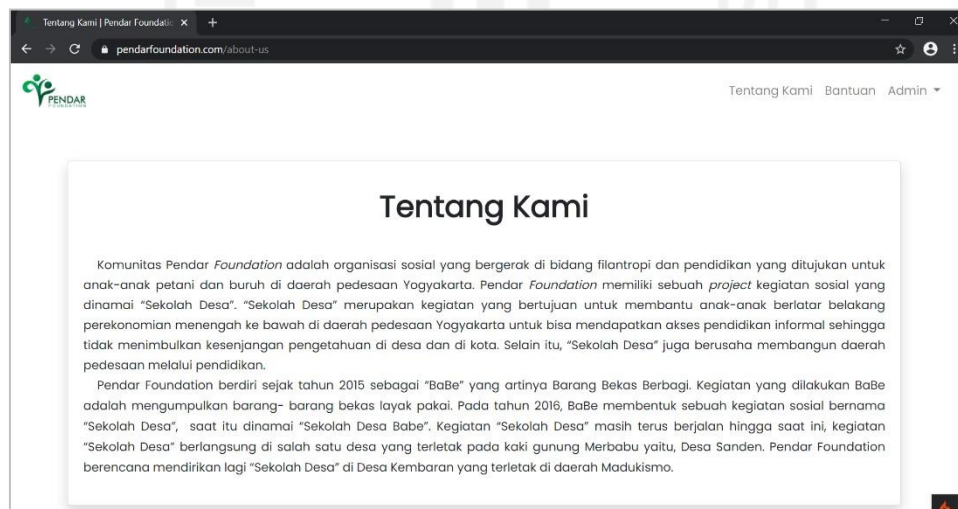


Gambar 4. 3 Halaman Daftar Akun (1)

Gambar 4. 4 Halaman Daftar Akun (2)

4.1.4 Halaman Tentang Kami

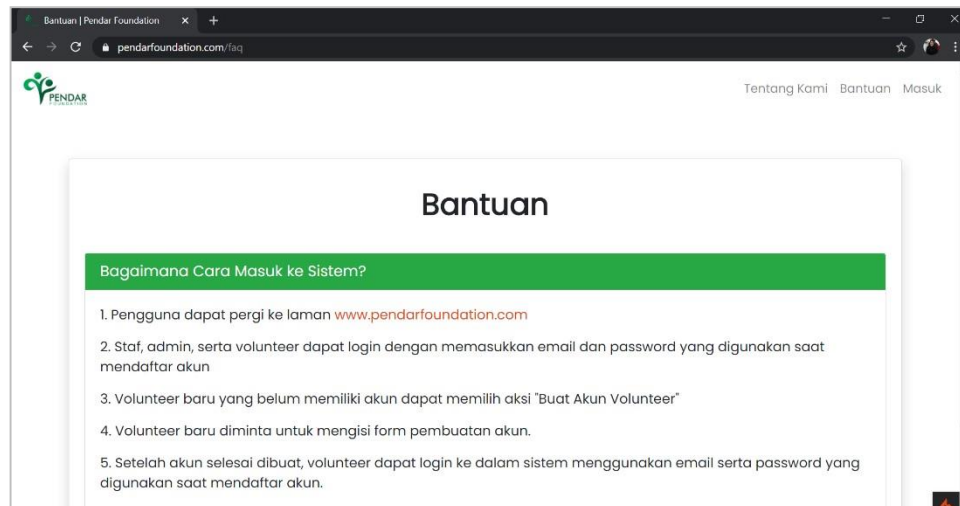
Halaman tentang kami menampilkan informasi yang menjelaskan informasi tentang komunitas Pendar Foundation Yogyakarta. Halaman tentang kami ditunjukkan pada Gambar 4.5.



Gambar 4. 5 Halaman Tentang Kami

4.1.5 Halaman Bantuan

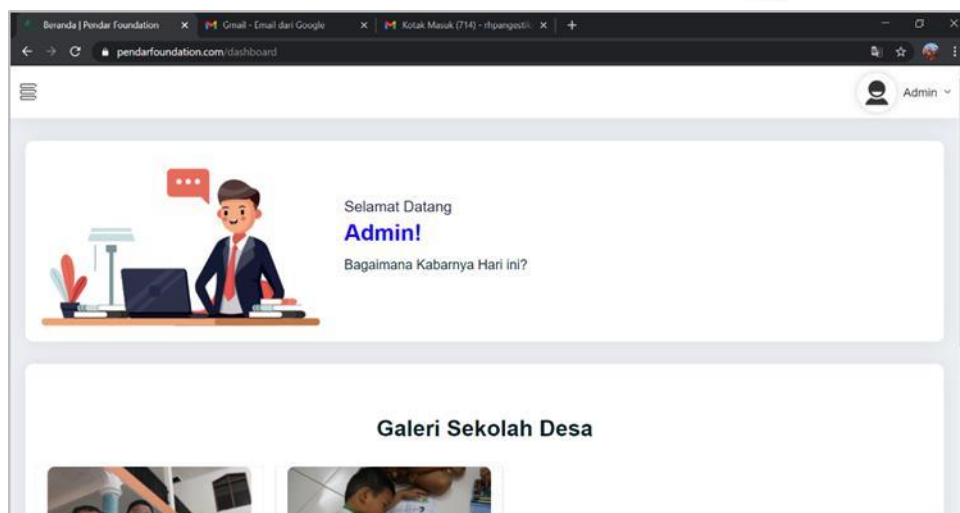
Halaman bantuan menampilkan informasi yang menjelaskan dasar – dasar penggunaan sistem. Halaman bantuan diharapkan dapat membantu pengguna ketika mengalami kebingungan dalam mengoperasikan sistem. Halaman bantuan ditunjukkan pada Gambar 4.6.



Gambar 4. 6 Halaman Bantuan

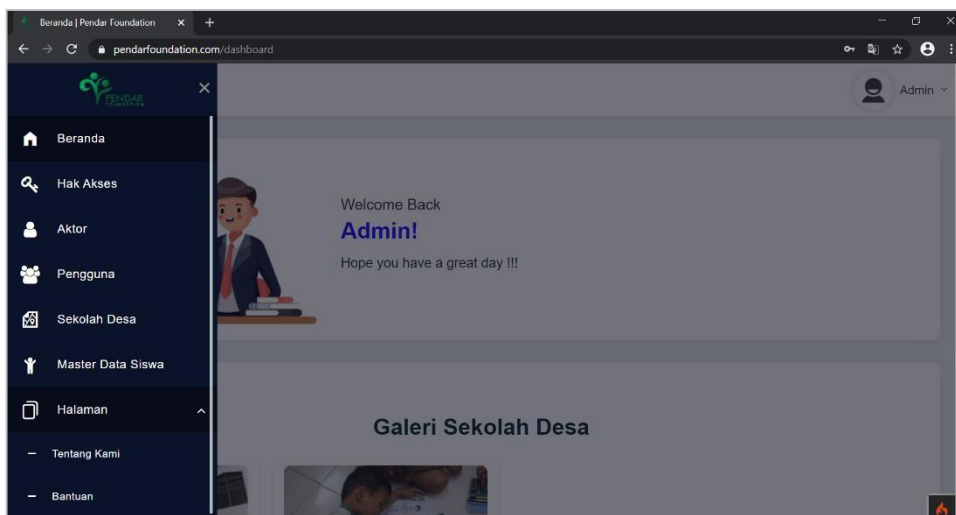
4.1.6 Halaman Beranda

Halaman beranda merupakan halaman yang tampil setelah pengguna melakukan proses *login*. Pada pojok kanan atas terdapat nama pengguna, ketika nama tersebut *ditap*, maka akan menampilkan pilihan *profile*, *help* dan *logout*. Halaman beranda ditunjukkan pada Gambar 4.7.

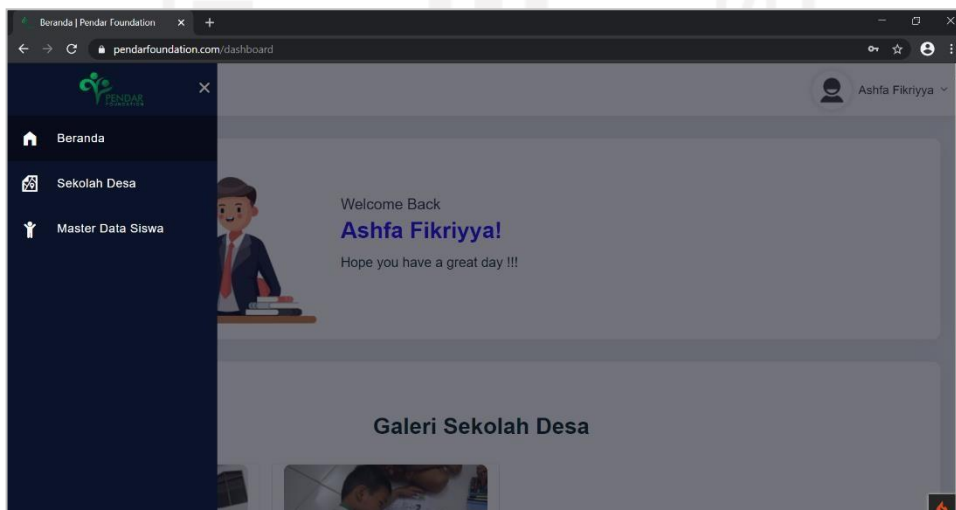


Gambar 4. 7 Halaman Beranda

Selain itu, pada halaman beranda terdapat tiga garis dibagian pojok kiri halaman yang berfungsi untuk menampilkan *sidebar*. Terdapat perbedaan antara *sidebar* admin dengan *sidebar* staf dan *volunteer*. Adapun menu yang terdapat pada *sidebar* admin yaitu, beranda, hak akses, aktor, pengguna, sekolah desa, master data siswa dan halaman yang berisi menu tentang kami dan bantuan. Sedangkan *sidebar* staf dan *volunteer* hanya memuat beranda, sekolah desa dan master data siswa. *Sidebar* admin ditunjukkan pada Gambar 4.8. *Sidebar* staf dan *volunteer* ditunjukkan pada Gambar 4.9.



Gambar 4. 8 *Sidebar* Admin

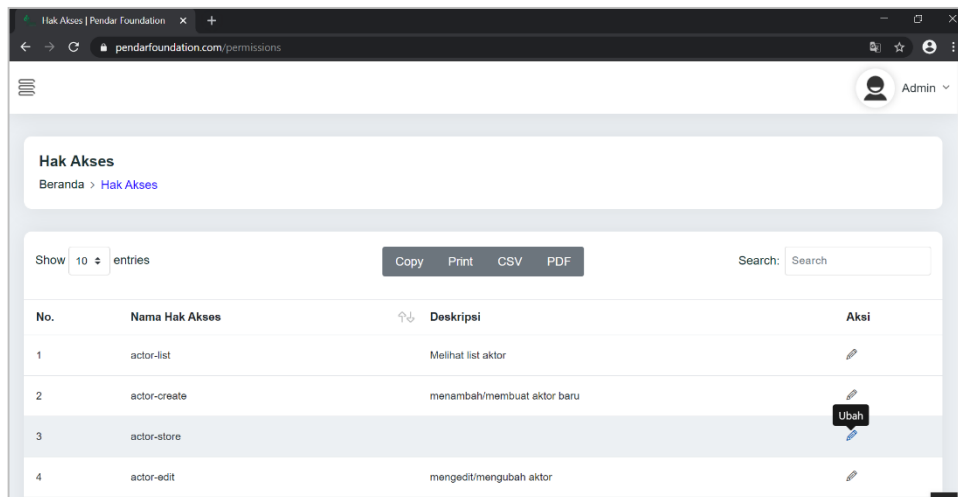


Gambar 4. 9 *Sidebar* Staf dan *Volunteer*

4.1.7 Halaman Hak Akses

Halaman hak akses hanya tampil pada admin. Halaman hak akses menampilkan daftar aksi yang dapat diakses oleh tiap – tiap aktor. Halaman hak akses berfungsi untuk

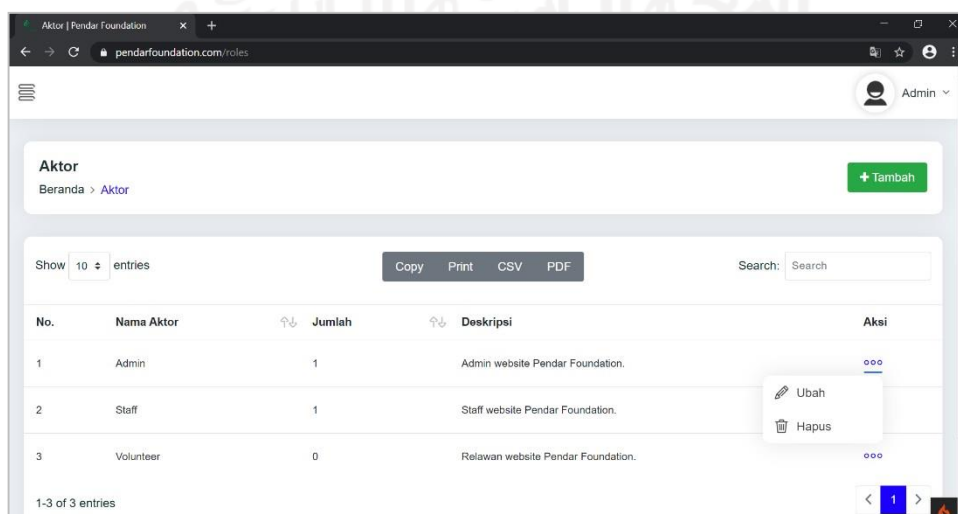
memberikan deskripsi dan penjelasan pada masing – masing hak akses. Tujuannya agar admin dapat mengetahui fungsi dari semua hak akses yang tersedia dan dapat menentukan hak akses yang diperlukan untuk tiap – tiap aktor. Pada halaman hak akses admin dapat menambahkan dan mengubah deskripsi hak akses menggunakan *button* yang tersedia. Halaman hak akses ditunjukkan pada Gambar 4.10.



Gambar 4. 10 Halaman Hak Akses

4.1.8 Halaman Aktor

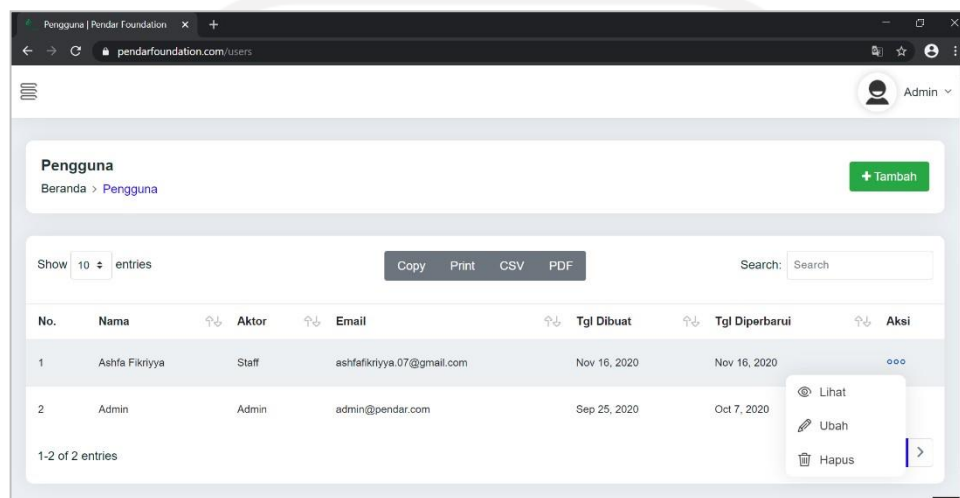
Halaman aktor hanya tampil pada admin. Halaman aktor menampilkan daftar aktor yang mengakses sistem beserta jumlahnya. Pada halaman aktor, admin dapat melakukan ubah data dan hapus data menggunakan *button* aksi. Selain itu, admin juga dapat menambahkan aktor baru jika suatu saat diperlukan, dengan menggunakan *button* tambah pada bagian kanan atas. Halaman Aktor ditunjukkan pada Gambar 4.11.



Gambar 4. 11 Halaman Aktor

4.1.9 Halaman Pengguna

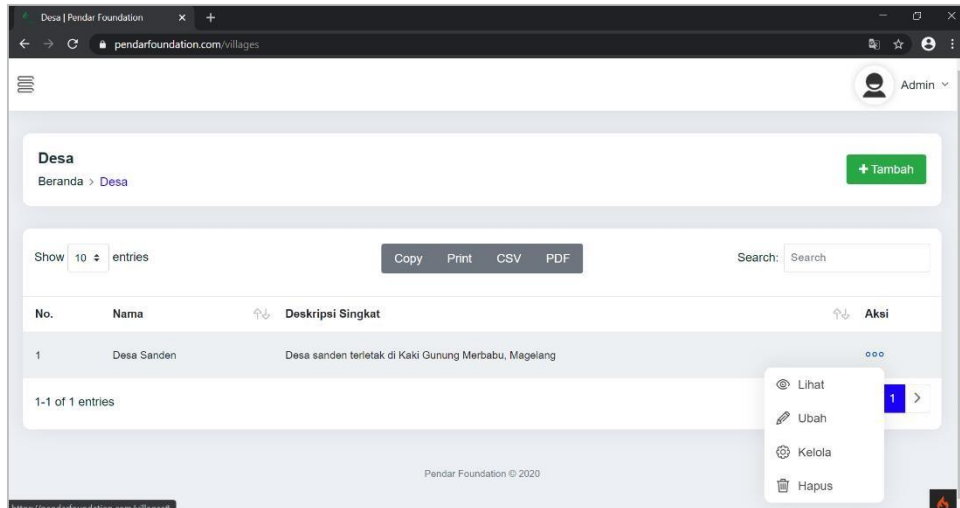
Halaman pengguna hanya tampil pada admin. Halaman pengguna menampilkan daftar pengguna yang mengakses sistem beserta informasi dari pengguna tersebut. Pada halaman pengguna, admin dapat melihat data pengguna, mengubah data pengguna dan menghapus pengguna menggunakan *button* aksi. Selain itu, admin juga dapat menambahkan akun untuk pengguna baru, khususnya akun untuk staf. Halaman pengguna ditunjukkan pada Gambar 4.12.



Gambar 4. 12 Halaman Pengguna

4.1.10 Halaman Sekolah Desa

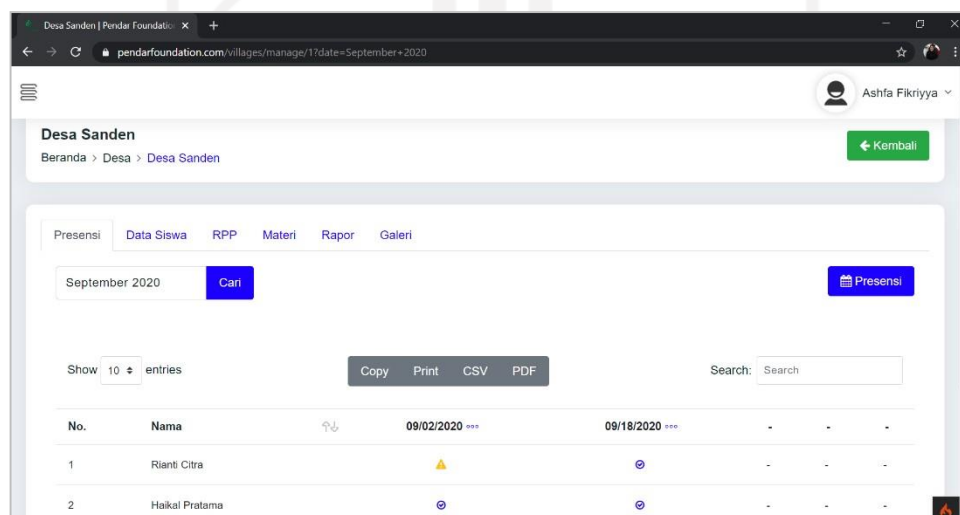
Halaman sekolah desa tampil pada admin, staf dan *volunteer*. Halaman sekolah desa menampilkan daftar sekolah desa beserta deskripsinya. Pada halaman sekolah desa, admin dan staf dapat melihat, mengubah, mengelola dan menghapus sekolah desa menggunakan *button* aksi. Apabila pengguna memilih aksi 'kelola', maka sistem akan menampilkan menu data sekolah desa seperti presensi, data siswa, RPP, materi, rapor dan galeri. Selain itu admin dan staf juga dapat menambahkan sekolah desa baru menggunakan *button* tambah pada bagian kanan atas. Halaman sekolah desa ditunjukkan pada Gambar 4.13.



Gambar 4. 13 Halaman Sekolah Desa

4.1.11 Halaman Presensi

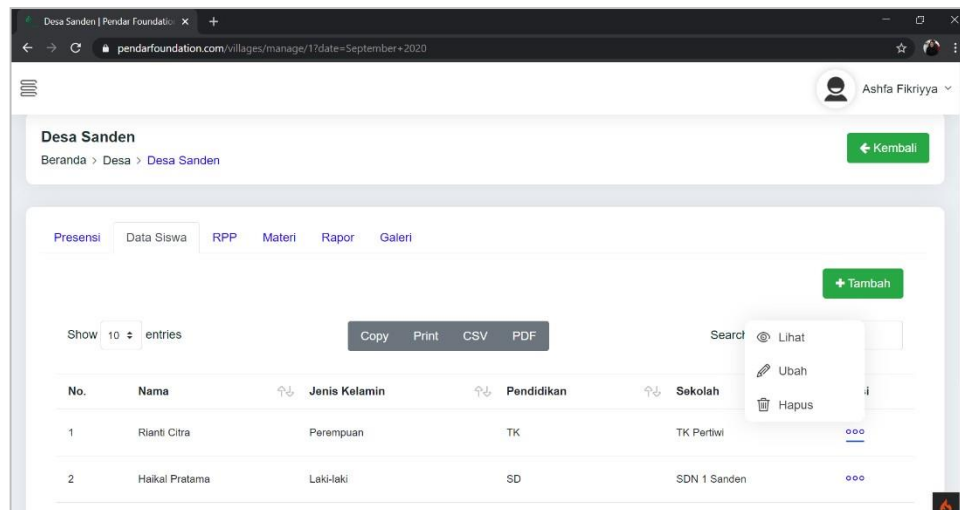
Halaman presensi tampil pada admin, staf dan *volunteer*. Halaman presensi menampilkan daftar nama siswa, tanggal pelaksanaan kegiatan sekolah desa beserta keterangan kehadiran siswa. Admin dan staf dapat mengelola data presensi, sedangkan *volunteer* hanya dapat melihat data presensi. Pada halaman presensi, pengelolaan dilakukan dengan menggunakan *button* presensi yang terdapat di sebelah kanan atas halaman. Halaman presensi ditunjukkan pada Gambar 4.14.



Gambar 4. 14 Halaman Presensi

4.1.12 Halaman Data Siswa

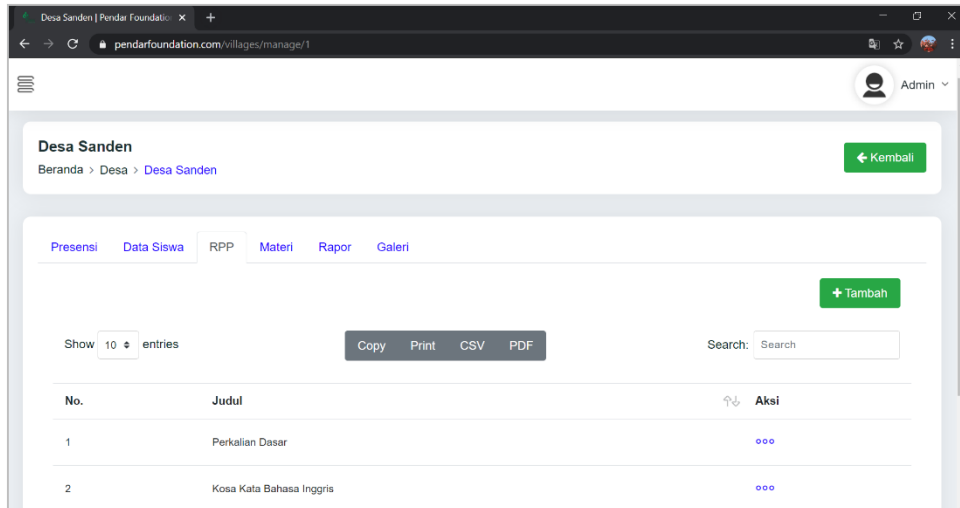
Halaman data siswa tampil pada admin, staf dan *volunteer*. Halaman data siswa menampilkan daftar nama siswa beserta informasi data pribadi siswa. Admin dan staf dapat melihat, mengubah data dan menghapus siswa menggunakan *button* aksi. Selain itu, admin dan staf juga dapat menambah siswa baru menggunakan *button* tambah yang terdapat di sebelah kanan atas halaman. Halaman data siswa ditunjukkan pada Gambar 4.15.



Gambar 4. 15 Halaman Data Siswa

4.1.13 Halaman RPP

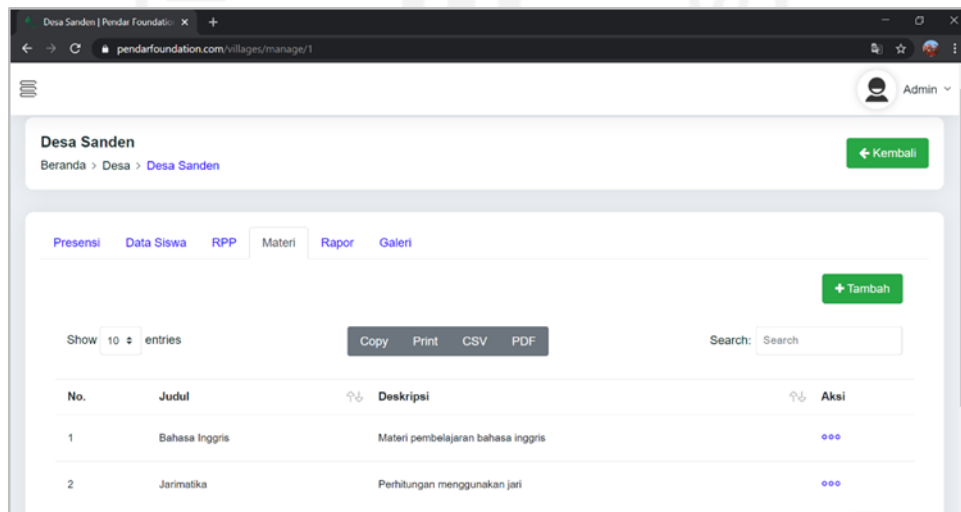
Halaman RPP tampil pada admin, staf dan *volunteer*. Halaman RPP menampilkan daftar judul rencana pelaksanaan pembelajaran, didalamnya terdapat dokumen RPP yang diunggah oleh pengguna. RPP dapat diunggah atau ditambahkan menggunakan *button* tambah yang tersedia di sebelah kanan atas halaman. Selain itu, pengguna juga dapat melihat, mengubah dan menghapus RPP menggunakan *button* aksi. Halaman RPP ditunjukkan pada Gambar 4.16



Gambar 4. 16 Halaman RPP

4.1.14 Halaman Materi

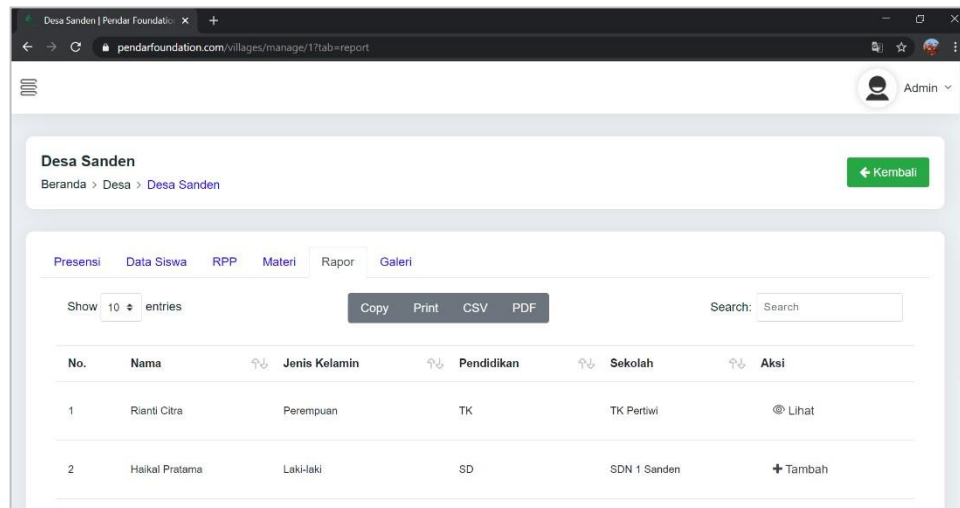
Halaman materi tampil pada admin, staf dan *volunteer*. Halaman materi menampilkan daftar judul materi pembelajaran, didalamnya terdapat dokumen yang diunggah oleh pengguna. Materi dapat diunggah atau ditambahkan menggunakan *button* tambah yang tersedia di sebelah kanan atas halaman. Selain itu, pengguna juga dapat melihat, mengubah dan menghapus RPP menggunakan *button* aksi. Halaman materi ditunjukkan pada Gambar 4.17.



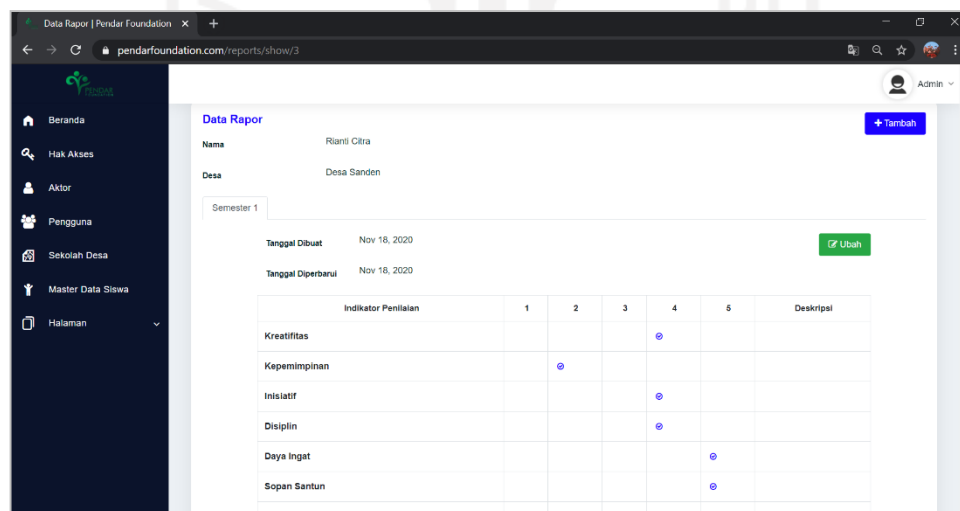
Gambar 4. 17 Halaman Materi

4.1.15 Halaman Rapor

Halaman rapot tampil pada admin, staf dan *volunteer*. Halaman rapor menampilkan daftar nama siswa beserta informasi pribadi siswa seperti jenis kelamin, pendidikan dan asal sekolah. Pada halaman rapor, pengguna dapat melihat rapot siswa, menambahkan rapor baru, menghapus dan mengubah data rapor. Namun aksi ini hanya dapat dilakukan oleh admin dan staf. Halaman rapor ditunjukkan pada Gambar 4.18 dan Gambar 4.19.



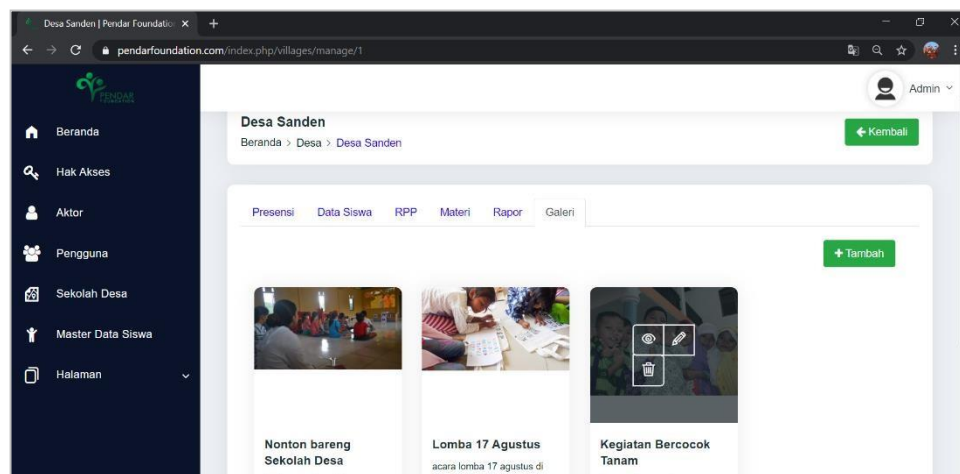
Gambar 4. 18 Halaman Rapor (1)



Gambar 4. 19 Halaman Rapor (2)

4.1.16 Halaman Galeri

Halaman galeri tampil pada admin, staf dan *volunteer*. Halaman galeri menampilkan folder galeri beserta judul dan deskripsi singkat mengenai foto – foto kegiatan sekolah desa. Pada halaman galeri, admin dan staf dapat menambah folder baru atau foto menggunakan *button* tambah yang tersedia di sebelah kanan atas halaman. Selain itu, admin atau staf juga dapat mengubah dan menghapus foto atau folder pada halaman halaman galeri. Halaman galeri ditunjukkan pada Gambar 2.20.



Gambar 4. 20 Halaman Galeri

4.1.17 Halaman Master Data Siswa

Halaman master data siswa tampil pada admin, staf dan *volunteer*. Halaman master data siswa menampilkan daftar nama siswa dari semua sekolah desa beserta informasi singkat siswa seperti jenis kelamin, pendidikan dan asal sekolah. Pada halaman master data siswa, pengguna dapat mencari nama siswa menggunakan kotak *search* yang tersedia di sebelah kanan bagian atas halaman. Selain itu admin dan staf juga dapat menambah siswa baru menggunakan *button* tambah yang ada di bawah kotak *search*. Mengubah data siswa dan menghapus dapat dilakukan menggunakan *button* aksi. Halaman master data siswa ditunjukkan pada Gambar 4.21.

No.	Nama	Desa	Jenis Kelamin	Pendidikan	Sekolah	Aksi
1	Bimala Renjani	Desa Sanden	Perempuan	SD	SDN 1 Sanden	Lihat, Ubah, Hapus
2	Haikal Pratama	Desa Sanden	Laki-laki	SD	SDN 1 Sar	
3	Hamdan Saputra	Desa Sanden	Laki-laki	SD	SDN 1 Sar	
4	Rianti Citra	Desa Sanden	Perempuan	TK	TK Perliwi	

Gambar 4. 21 Halaman Master Data Siswa

42 Pengujian Sistem

Tahap pengujian sistem dilakukan menggunakan teknik pengujian *black box*. Pengujian berfokus pada sudut pandang fungsional sistem. Pada tahap ini sistem akan diuji untuk mengetahui apakah sistem sudah dapat berjalan dengan baik dan memberikan hasil sesuai dengan yang diharapkan.

Pengujian *black box* dilakukan terhadap semua fitur yang tersedia pada sistem informasi pendataan sekolah desa komunitas pendar foundation. Dari setiap fitur, dilakukan beberapa aktivitas kasus uji. Tabel hasil pengujian *black box* dapat dilihat di halaman lampiran.

43 Sosialisasi dan Pelatihan Sistem

Sosialisasi dan pelatihan penggunaan sistem dilakukan bersama anggota komunitas pendar foundation. Kegiatan pelatihan merupakan tahap pengenalan sistem informasi pendataan sekolah desa, dengan tujuan memberikan edukasi kepada anggota komunitas pendar foundation mengenai sistem yang telah dikembangkan. Pada tahap ini dilakukan demo sistem kepada pengguna dan pengguna juga diminta untuk mencoba menggunakan sistem untuk membiasakan pengguna dalam memahami sistem yang telah dibangun.

Sosialisasi dan pelatihan penggunaan sistem pertama dilakukan pada Minggu, 22 November 2020, bertempat di Bento Cafe, Klebengan. Berikut ini adalah anggota komunitas pendar foundation yang ikut hadir dalam sosialisasi dan pelatihan penggunaan sistem informasi pendataan sekolah desa komunitas Pendar Foundation:

1. Ibar Furqonul Akbar sebagai Project Manger Pendar Foundation

2. Aqina Madhania P.R sebagai Staf Divisi Pendidikan Pendar Foundation
3. Prilliza Monicha Ayessy sebagai Staf Divisi Pendidikan

Dokumentasi kegiatan sosialisasi dan pelatihan penggunaan sistem pada hari Minggu, 22 November 2020, bertempat di Bento Cafe, Klebengan dapat dilihat pada Gambar 4.22.



Gambar 4. 22 Sosialisasi dan Pelatihan di Bento Cafe,
Klebengan

Sosialisasi dan pelatihan penggunaan sistem kedua dilakukan pada Rabu, 25 November 2020, bertempat di Upnormal Cafe, Seturan. Anggota komunitas Pendar Foundation yang ikut hadir pada pelatihan kedua antara lain:

1. Retno Kusumawati Putri sebagai Kepala Divisi Pendidikan Pendar Foundation

Dokumentasi kegiatan sosialisasi dan pelatihan penggunaan sistem pada hari Rabu, 25 November 2020, bertempat di Upnormal Cafe, Seturan dapat dilihat pada Gambar 4.23.



Gambar 4. 23 Sosialisasi dan Pelatihan di Upnormal Cafe,
Seturan

Kegiatan sosialisasi dan pelatihan penggunaan sistem ketiga dilaksanakan pada Jumat, 27 November 2020. Kegiatan pelatihan dilakukan di dua tempat yang berbeda, yaitu di Tempo Gelato, Taman Siswa dan Indie.co, Sagan. Adapun anggota Pendar Foundation yang ikut pada sosialisasi dan pelatihan ketiga adalah:

1. Iffah Khoiriyah sebagai Koordinator Sekolah Desa Pendar Foundation
2. Indah Gitaningrum sebagai Founder Pendar Foundation
3. Erbin Zalukhu sebagai Staf Divisi Pendidikan

Dokumentasi kegiatan sosialisasi dan pelatihan penggunaan sistem pada hari Jumat, 27 November 2020, bertempat di Tempo Gelato, Taman Siswa dan Indie.co, Sagan dapat dilihat pada Gambar 4.24 dan Gambar 4.25.



Gambar 4. 25 Sosialisasi dan Pelatihan di Tempo Gelato,
Taman Siswa



Gambar 4. 24 Sosialisasi dan Pelatihan di Indie.co, Sagan

44 Feedback Pengguna

Feedback pengguna didapatkan dari hasil wawancara dan diskusi yang dilakukan pada saat kegiatan sosialisasi dan pelatihan penggunaan sistem bersama dengan anggota komunitas Pendar Foundation. *Feedback* dari pengguna dibutuhkan untuk mengetahui kepuasan pengguna terhadap sistem yang telah dibangun.

Dari hasil wawancara dan diskusi bersama anggota beserta founder komunitas Pendar Foundation, didapatkan umpan balik beserta kesan yang dirasakan oleh pengguna saat

mencoba menggunakan sistem informasi pendataan sekolah desa komunitas Pendar Foundation. Sebagian besar pengguna menyampaikan bahwa alur sistem mudah dimengerti dan fitur – fitur yang terdapat dalam sistem sudah sesuai dengan kebutuhan pendataan sekolah desa. Sebagian lainnya menyatakan bahwa pengguna hanya membutuhkan waktu untuk terbiasa dalam menggunakan sistem dan dapat memahami keseluruhan alur sistem, tidak didapatkan kesulitan yang berarti pada saat mengoperasikan sistem. Komunitas Pendar Foundation yakin bahwa sistem yang telah dibangun dapat membantu meningkatkan kinerja anggotanya dalam proses pendataan sekolah desa, serta memberikan dampak positif berskala panjang untuk keberlangsungan kegiatan sekolah desa.

45 Pemeliharaan Sistem

Pemeliharaan sistem informasi merupakan tahapan yang bertujuan untuk memperbaiki dan menjaga sistem yang telah dikembangkan. Pemeliharaan yang dilakukan pada sistem informasi pendataan sekolah desa komunitas Pendar Foundation, yaitu pemeliharaan korektif. Pemeliharaan korektif adalah bagian pemeliharaan sistem yang mengoreksi kesalahan-kesalahan yang ditemukan pada saat sistem sedang berjalan.

Pada proses pemeliharaan sistem, sempat terjadi penambahan fungsi pada sistem. Pada halaman RPP dan materi, komunitas Pendar Foundation mengharapkan agar sistem dapat menampilkan oleh siapa dokumen RPP dan materi diunggah. Sehingga ditambahkan fungsi pada sistem, yaitu fungsi *updated by*. Selain itu, komunitas Pendar Foundation juga mengharapkan agar rapor anak didik sekolah desa dapat diunduh.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pengembangan back-end sistem informasi pendataan sekolah desa komunitas pendar foundation Yogyakarta, maka diperoleh kesimpulan yaitu:

- a. Berdasarkan hasil pengujian *black-box testing*, dimana semua fitur untuk kebutuhan pendataan kegiatan sekolah desa komunitas Pendar Foundation sudah dapat berjalan dengan semestinya, maka penelitian ini telah berhasil mengembangkan sebuah sistem informasi pendataan sekolah desa komunitas Pendar Foundation Yogyakarta dengan menggunakan bahasa pemrograman PHP, Javascript dan HTML, serta MySQL sebagai basis data sistem.
- b. Sistem diimplementasikan dengan melakukan uji coba kepada anggota komunitas pendar foundation dengan tahapan sebagai berikut:
 1. Admin mendaftarkan akun untuk staf komunitas pendar foundation
 2. Staf mengelola informasi sekolah desa, antara lain menambahkan nama – nama siswa sekolah desa, menambahkan presensi sekolah desa, menambahkan RPP, menambahkan materi dan menambahkan foto – foto pada galeri
 3. *Volunteer* mengelola RPP dan materi
 4. *Volunteer* dan staf lainnya mendapatkan informasi berdasarkan data sekolah desa yang telah diunggah
- c. Pengujian sistem dilakukan menggunakan metode *black box* untuk mengetahui fungsionalitas sistem. Dari tahap pengujian diketahui bahwa fungsionalitas sistem sudah dapat berjalan sesuai kebutuhan pengguna.
- d. Berdasarkan umpan balik dari pengguna yaitu anggota komunitas Pendar Foundation, pengguna merasa sistem sudah dapat memenuhi kebutuhan dan sesuai dengan yang diharapkan.

5.2 Saran

Implementasi sistem informasi pendataan sekolah desa komunitas pendar foundation yang telah dilakukan masih memiliki kekurangan dan kelemahan. Berdasarkan umpan balik

dari komunitas pendar foundation, berikut adalah saran yang diberikan untuk sistem informasi pendataan sekolah desa komunitas Pendar Foundation Yogyakarta:

- a Dalam mengoperasikan sistem, pengguna merasa terlalu banyak melakukan klik. Sehingga dibutuhkan pengembangan lanjutan untuk memperbaiki tampilan sistem
- b Penambahan fitur untuk mengunduh rapor siswa
- c Penambahan fitur keterangan *updated by* pada setiap dokumen yang diunggah ke dalam sistem



DAFTAR PUSTAKA

- Afifah, N., Rusdianto, D. S., & Rachmadi, A. (2020). Pengembangan Website Pengelolaan Kegiatan Komunitas Menggunakan Framework CodeIgniter. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 1344-1353.
- Anhar. (2010). *Panduan Menguasai PHP & MySQL Secara Otodidak*. Jakarta: Mediakita.
- Dewi, L. P., Indahyanti, U., & S, Y. H. (2012). Pemodelan Proses Bisnis Menggunakan Activity Diagram UML dan BPMN.
- Erinton, R., Negara, R. M., & Sanjoyo, D. D. (2017). Analisis Performasi Framework CodeIgniter dan Laravel menggunakan Web Server Apache. *e-Proceeding of Engineering*.
- Fajarianto, O., & Wahyuni, Y. (2016). Rancangan Sistem Informasi Pendataan Keluarga Miskin Desa Klebet Kec. Kemiri. *Sisfotek Global*, 6, 7.
- Hendrik, Anjomshooa, A., & Tjoa, A. M. (2014). Towards Semantic Mashup Tools For Big Data Analysis. *Proceeding of the Information & Communication Technology-EurAsia Conference 2014*, (pp. 100-145). Bali.
- Muzakir, A. (2014). Implementasi Manajemen Perpustakaan menggunakan Framework Codeigniter (CI) dengan Teknik Hierarchical Model View Controller (HMVC). *Seminar Nasional Sains dan Teknologi Informasi (SeNASTI)*.
- Nico, G. R., Wicaksono, S. A., & Pradana, F. (2018). Pengembangan Sistem Informasi Manajemen Donasi dan Relawan Bagi Komunitas Sosial. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 2869-2875.
- Perestrioka, Y. S., Brata, A. H., & Mursityo, Y. T. (2018). Pengembangan Sistem Informasi Penggalangan Donasi Pada Yayasan Nurani Orang Tua Asuh (GN-OTA) Kabupaten Lumajang. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 1833-1840.
- Sari, C., Jefriansyah, D., Norriastini, M., & Zurianti. (2019). Aplikasi Penggalangan Dana Komunitas Jumat Jaman Now Berbasis Web Menggunakan Framework Laravel.
- Setiawan, A. M. (2013). *Integrated Framework For Business Process Complexity Analysis*. Retrieved from ECIS 2013 Completed Research: http://aisel.aisnet.org/ecis2013_cr/49
- Taufiq, H. (2015). *Argumentasi dan Validitas*. Yogyakarta: Darqin.

- Wahid, F. (2014). The Antecedents And Impacts of a Green Eprocurement Infrastructure: Evidence From The Indonesian Public Sector. *International Journal of internet Protocol Technology*, 7(4), 210-218.
- Wulandari, A. F. (2019). Rancang Bangun Aplikasi SIMPADI (Sistem Informasi Monitoring Pengasuhan Anak Disabilitas) Berbasis Web. *Jurnal Manajemen Informatika*, 18-27.
- Yuliano, T. (2009). *Ilmu Komputer*. Retrieved from <https://ilmukomputer.org>
- Zukhri, Z. (2014). *Algoritma Genetika: Metode Komputasi Evolusioner untuk Menyelesaikan Masalah Optimasi*. Yogyakarta: Andi Publisher.



LAMPIRAN

A. Tabel hasil pengujian *black box*

Fitur	Skenario	Kode	Aktivitas	Hasil	Kesimpulan
<i>Login volunteer</i>	Menampilkan halaman <i>login</i>	BX-01	Memasukkan <i>username</i> dan kata sandi yang valid	Menampilkan halaman beranda	Berhasil
		BX-02	Memasukkan <i>username</i> dan kata sandi yang tidak terdaftar	Menampilkan pesan “tidak dapat melakukan <i>login</i> , mohon periksa kembali <i>username</i> dan kata sandi”	Berhasil
		BX-03	Tidak memasukkan <i>username</i> dan/atau kata sandi	Menampilkan pesan “ <i>username</i> dan kata sandi harus diisi”	Berhasil
Daftar akun <i>volunteer</i>	Menampilkan halaman daftar akun	BX-04	Mengisi form daftar akun dengan data yang valid	Berhasil mendaftar akun, menampilkan halaman <i>login</i>	Berhasil
		BX-05	Tidak mengisi form daftar akun	Menampilkan pesan untuk melengkapi data	Berhasil
Kelola data diri <i>volunteer</i>	Mengubah <i>username</i>	BX-06	Memasukkan <i>username</i> yang belum terdaftar	Menampilkan pesan “data berhasil diperbarui”	Berhasil
		BX-07	Memasukkan <i>username</i> yang sudah terdaftar	Menampilkan pesan “ <i>username</i> sudah terdaftar, mohon masukkan <i>username</i> lain”	Berhasil
Lihat	Menampilkan	BX-08	Mengklik aksi	Menampilkan	Berhasil

sekolah desa	halaman sekolah desa		lihat	deskripsi sekolah desa	
Kelola sekolah desa	Menampilkan halaman sekolah desa	BX-09	Mengklik aksi kelola	Menampilkan halaman presensi, data siswa, RPP, materi, rapor dan galeri	Berhasil
Lihat presensi sekolah desa	Menampilkan halaman presensi	BX-10	Melihat presensi sekolah desa di bulan September	Menampilkan presensi sekolah desa pada bulan September	Berhasil
Lihat data diri siswa sekolah desa	Menampilkan halaman data siswa	BX-11	Mengklik aksi lihat pada salah satu nama siswa	Menampilkan data diri siswa	Berhasil
Kelola RPP	Menambah RPP	BX-12	Mengunggah file RPP	Menampilkan pesan "RPP berhasil ditambahkan"	Berhasil
	Mengubah RPP	BX-13	Mengubah judul folder RPP	Menampilkan pesan "RPP berhasil diperbarui"	Berhasil
	Menghapus RPP	BX-14	Mengklik aksi hapus	Menampilkan pesan "RPP berhasil dihapus"	Berhasil
	Mengunduh RPP	BX-15	Mengklik unduh	Mengunduh file RPP	Berhasil
Kelola materi sekolah desa	Menambah materi	BX-16	Mengunggah file materi	Menampilkan pesan "materi berhasil ditambahkan"	Berhasil
	Mengubah materi	BX-17	Mengubah judul folder materi	Menampilkan pesan "materi berhasil diperbarui"	Berhasil
	Menghapus materi	BX-18	Mengklik aksi hapus	Menampilkan pesan "materi berhasil"	Berhasil

				dihapus”	
	Mengunduh materi	BX-19	Mengklik unduh	Mengunduh file materi	Berhasil
Lihat rapor rapor sekolah desa	Menampilkan halaman rapor	BX-20	Mengklik aksi lihat pada salah satu siswa	Menampilkan rapor siswa	Berhasil
Lihat galeri sekolah desa	Menampilkan halaman galeri	BX-21	Mengklik aksi lihat pada salah satu folder galeri	Menampilkan foto – foto dalam folder galeri	Berhasil
Lihat <i>master data</i> siswa	Menampilkan halaman <i>master data</i> siswa	BX-22	Mengklik menu <i>master data</i> siswa	Menampilkan daftar seluruh siswa sekolah desa	Berhasil
<i>Login staf</i>	Menampilkan halaman <i>login</i>	BX-23	Memasukkan <i>username</i> dan kata sandi yang valid	Menampilkan halaman beranda	Berhasil
		BX-24	Memasukkan <i>username</i> dan kata sandi yang tidak terdaftar	Menampilkan pesan “tidak dapat melakukan <i>login</i> , mohon periksa kembali <i>username</i> dan kata sandi”	Berhasil
		BX-25	Tidak memasukkan <i>username</i> dan/atau kata sandi	Menampilkan pesan “ <i>username</i> dan kata sandi harus diisi”	Berhasil
Kelola sekolah desa	Melihat sekolah desa	BX-26	Mengklik aksi lihat	Menampilkan deskripsi sekolah desa	Berhasil
	Mengubah sekolah desa	BX-27	Mengubah nama sekolah desa	Menampilkan pesan “data berhasil diubah”	Berhasil
	Mengelola sekolah desa	BX-28	Mengklik aksi kelola	Menampilkan halaman presensi, data siswa, RPP,	Berhasil

				materi, rapor dan galeri	
	Menghapus sekolah desa	BX-29	Mengklik aksi hapus	Menampilkan pesan “data berhasil dihapus”	Berhasil
Kelola presensi	Menambah tanggal pelaksanaan sekolah desa	BX-30	Menambah jadwal pelaksanaan pada bulan September	Menampilkan halaman presensi bulan September	Berhasil
	Mengisi presensi	BX-31	Mengisi kehadiran siswa pada bulan September	Menampilkan informasi kehadiran siswa	Berhasil
	Mengubah presensi	BX-32	Mengubah kehadiran siswa pada bulan September	Menampilkan informasi kehadiran yang telah diubah	Berhasil
	Menghapus jadwal pelaksanaan	BX-33	Mengklik aksi hapus pada salah satu tanggal pelaksanaan	Menampilkan pesan “presensi berhasil dihapus”	Berhasil
Kelola data siswa	Menambah siswa	BX-34	Mengisi data diri siswa baru	Menampilkan pesan “siswa berhasil ditambahkan”	Berhasil
	Melihat data siswa	BX-35	Mengklik aksi lihat	Menampilkan data lengkap siswa	Berhasil
	Mengubah data siswa	BX-36	Mengubah alamat siswa	Menampilkan pesan “data berhasil diperbarui”	Berhasil
	Menghapus siswa	BX-37	Mengklik aksi hapus	Menampilkan pesan “berhasil menghapus data siswa”	Berhasil
Kelola rapor	Melihat rapor	BX-38	Mengklik aksi lihat	Menampilkan halaman data rapor siswa	Berhasil

	Menambah rapor	BX-39	Mengisi nilai pada rapor siswa	Menampilkan pesan “rapor berhasil ditambahkan”	Berhasil
	Mengubah rapor	BX-40	Mengubah nilai pada rapor siswa	Menampilkan pesan “rapor berhasil diperbarui”	Berhasil
Kelola galeri	Melihat galeri	BX-41	Mengklik aksi lihat	Menampilkan foto – foto dalam folder galeri	Berhasil
	Menambah galeri	BX-42	Menambah folder baru dalam galeri	Menampilkan pesan “galeri berhasil ditambahkan”	Berhasil
	Mengubah galeri	BX-43	Menghapus salah satu foto yang terdapat dalam folder galeri	Menampilkan pesan “galeri berhasil diperbarui”	Berhasil
	Menghapus galeri	BX-44	Mengklik aksi hapus	Menampilkan pesan “galeri berhasil dihapus”	Berhasil
Kelola master data siswa	Menambah siswa	BX-45	Memasukkan data diri siswa baru	Menampilkan pesan “siswa berhasil ditambahkan”	Berhasil
	Melihat data siswa	BX-46	Mengklik aksi lihat	Menampilkan data lengkap siswa	Berhasil
	Mengubah data siswa	BX-47	Mengubah alamat siswa	Menampilkan pesan “data siswa berhasil diperbarui”	Berhasil
	Menghapus siswa	BX-48	Mengklik aksi hapus	Menampilkan pesan “berhasil menghapus data siswa”	Berhasil
<i>Login admin</i>	Menampilkan halaman <i>login</i>	BX-49	Memasukkan <i>username</i> dan	Menampilkan halaman	Berhasil

			kata sandi yang valid	beranda	
		BX-50	Memasukkan <i>username</i> dan kata sandi yang tidak valid	Menampilkan pesan “tidak dapat melakukan <i>login</i> , mohon periksa kembali <i>username</i> dan <i>password</i> ”	Berhasil
		BX-51	Tidak memasukkan <i>username</i> dan <i>password</i>	Menampilkan pesan “ <i>username</i> dan <i>password</i> harus diisi”	Berhasil
Ubah hak akses	Menampilkan halaman hak akses	BX-52	Mengubah deskripsi hak akses	Menampilkan pesan “hak akses berhasil diperbarui”	Berhasil
Kelola aktor	Menambah aktor	BX-53	Menambah aktor baru	Menampilkan pesan “aktor berhasil ditambahkan”	Berhasil
	Mengubah aktor	BX-54	Mengubah nama aktor	Menampilkan pesan “aktor berhasil diperbarui”	Berhasil
	Menghapus aktor	BX-55	Menghapus salah satu aktor	Menampilkan pesan “berhasil menghapus aktor”	Berhasil
Kelola pengguna	Melihat pengguna	BX-56	Mengklik aksi lihat	Menampilkan detail data pengguna	Berhasil
	Menambah pengguna	BX-57	Menambah pengguna baru sebagai staf	Menampilkan pesan “Pengguna berhasil ditambahkan”	Berhasil
	Mengubah pengguna	BX-58	Mengubah <i>role</i> pengguna	Menampilkan pesan “Pengguna	Berhasil

				berhasil diperbarui”	
	Menghapus pengguna	BX-59	Menghapus salah satu pengguna	Menampilkan pesan “Pengguna berhasil dihapus”	Berhasil
Ubah tentang kami	Menampilkan halaman ubah tentang kami	BX-60	Mengubah informasi tentang kami	Menampilkan pesan “tentang kami berhasil diperbarui”	Berhasil
Kelola bantuan	Menampilkan halaman bantuan	BX-61	Menambah informasi bantuan	Menampilkan pesan “bantuan berhasil ditambahkan”	Berhasil
		BX-62	Mengubah informasi bantuan	Menampilkan pesan “bantuan berhasil diperbarui”	Berhasil