**MANAJEMEN KOMUNITAS PEDULI LINGKUNGAN DAN PENGELOLAAN SAMPAH**

**DI PULAU MERAH BERBASIS WEB**

**LAPORAN PROYEK AKHIR**



# Oleh :

**ADE FIAN GALIH IMANDANI NIM. 361755401136**

**PROGRAM STUDI DIPLOMA III TEKNIK INFORMATIKA**

**MANAJEMEN KOMUNITAS PEDULI LINGKUNGAN DAN PENGELOLAAN SAMPAH**

**DI PULAU MERAH BERBASIS WEB**

**LAPORAN PROYEK AKHIR**



# Proyek Akhir Ini Dibuat dan Diajukan untuk Memenuhi Salah Satu Syarat Kelulusan Program Studi Diploma III Teknik Informatika dan Mencapai Gelar Ahli Madya (A.Md)

**Oleh :**

**ADE FIAN GALIH IMANDANI NIM. 361755401136**

**PROGRAM STUDI DIPLOMA III TEKNIK INFORMATIKA POLITEKNIK NEGERI BANYUWANGI 2020**

*-- Halaman ini sengaja dikosongkan --*

# PERSEMBAHAN

Alhamdulillah puji syukur kepada Allah SWT laporan proyek akhir ini dapat diselesaikan dalam waktu yang telah ditetapkan. Proyek akhir ini saya persembahkan untuk:

1. Bapak Son Kuswadi, Dr.Eng. selaku Direktur Politeknik Negeri Banyuwangi.
2. Bapak. Moh. Dimyati Ayatullah, S.T., M.Kom. selaku koordinator Program Studi Teknik Informatika.
3. Ibu Dianni Yusuf, S.Kom., M.Kom. dan Bapak Endi Sailul Haq, S.T., M.Kom. selaku pembimbing pertama dan kedua.
4. Bapak Subono, S.T., M.T. dan Moh. Nur Shodiq, S.T., M.T. selaku dosen penguji pertama dan kedua.
5. Ibu dan Ayah yang telah mendoakan serta memberikan dukungan dalam pengerjaan Tugas Akhir.
6. Saudara-saudaraku yang telah memberikan semangat dan dukungan serta doa dalam pengerjaan Tugas Akhir.
7. Sahabat-sahabat dan orang terbaik yang selalu memberikan semangat serta dukungan selama mengerjakan proyek akhir ini, serta teman-teman Teknik Informatika kelas 3E-TI 10 dan seluruh Mahasiswa Teknik Informatika di Politeknik Negeri Banyuwangi.

*-- Halaman ini sengaja dikosongkan --*

# MOTTO

**“** *Jadikan Masa Lalu Sebagai Pengalaman Bukan Penyesalan***,** *Masa Depan Sebagai Tujuan Bukan Angan-angan***”**

*- Ade Fian Galih Imandani -*

*-- Halaman ini sengaja dikosongkan --*

# PERNYATAAN

Saya yang bertanda tangan dibawah ini: Nama : Ade Fian Galih Imandani NIM : 361655401136

Menyatakan dengan sesungguhnya bahwa Proyek Akhir yang berjudul : **“MANAJEMEN KOMUNITAS PEDULI LINGKUNGAN DAN PENGELOLAAN SAMPAH DI PULAU MERAH BERBASIS WEB ”** adalah

benar-benar hasil karya sendiri, kecuali jika disebutkan sumbernya dan belum pernah diajukan pada institusi manapun, serta bukan karya plagiat. Saya bertanggung jawab atas keabsahan isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mandapat sanksi akademik jika ternyata dikemudian hari pernyataan ini tidak benar.

Banyuwangi, 12 Agustus 2020 Yang menyatakan,

Ade Fian Galih Imandani NIM. 361755401136

*-- Halaman ini sengaja dikosongkan --*

**MANAJEMEN KOMUNITAS PEDULI LINGKUNGAN DAN PENGELOLAAN SAMPAH**

**DI PULAU MERAH BERBASIS WEB**

# Proyek Akhir disusun untuk Memenuhi Salah Satu Syarat Mempe*role*h Gelar Ahli Madya (A.MD)

**Politeknik Negeri Banyuwangi**

**Oleh:**

**Ade Fian Galih Imandani NIM. 361755401136**

**Tanggal Ujian : .. September 2020 Menyetujui,**

|  |  |  |
| --- | --- | --- |
| **Pembimbing 1** | **: Dianni Yusuf, S.Kom., M.Kom.** | **(…………...…....)** |
| **Pembimbing 2** | **: Endi Sailul Haq, S.T., M.Kom.** | **(………………...)** |
| **Penguji 1** | **: Subono, S.T, M.T.** | **(……...………....)** |
| **Penguji 2** | **: Moh. Nur Shodiq, S.T., M.T.** | **(………………...)** |

|  |  |
| --- | --- |
| **Mengesahkan,** | **Mengetahui,** |
| **Ketua Jurusan Teknik Informatika** | **Koordinator Program Studi Teknik Informatika** |
| **Eka Mistiko Rini, S.Kom., M.Kom.** | **Moh. Dimyati Ayatullah, S.T., M.Kom.** |
| **NIP. 198310202014042001** | **NIK. 2008.36.004** |

*-- Halaman ini sengaja dikosongkan --*

# MANAJEMEN KOMUNITAS PEDULI LINGKUNGAN DAN PENGELOLAAN SAMPAH DI PULAU MERAH BERBASIS WEB

Nama : Ade Fian Galih Imandani

Nim 361755401136

Pembimbing : 1. Dianni Yusuf, S.Kom., M.Kom.

2. Endi Sailul Haq, S.T., M.Kom.

# ABSTRAK

Sampah menjadi masalah utama di negara besar khususnya Indonesia, karena masih belum ada penanganan mengenai limbah sampah yang terus bertambah khususnya di sektor pariwisata, contohnya pada pantai Pulau Merah yang ada di kabupaten Banyuwangi. Pantai Pulau Merah juga masih terus berbenah dengan bagaimana cara penangan tentang sampah agar seluruh lapisan masyarakat sadar dengan timbulnya sampah, tetapi masih menghadapi kendala utama yaitu rendahnya kesadaran masyarakat untuk memilah sampah. Untuk mengatasi kendala tersebut diadakan agenda tentang kegiatan bersih-bersih pada daerah wisata pantai Pulau Merah yang dilakukan oleh komunitas peduli lingkungan salah satunya adalah *EcoRanger*, komunitas ini juga membuat produk dari sampah plastik yang berguna bagi masyarakat dan berdampak baik bagi lingkungan yaitu *Ecobrick*. Dengan adanya komunitas *EcoRanger* ini ada beberapa komunitas baru yang terbentuk di beberapa tempat wisata Banyuwangi khususnya pantai, namun masih banyak masyarakat yang belum mengetahui adanya komunitas tersebut serta kegiatan yang diselenggarakan. Pemberitahuan tentang adanya agenda kegiatan dan informasi tentang *Ecobrick* masih terbilang konvensional, sehingga menyebabkan informasi yang diterima oleh masyarakat masih kurang maksimal. Oleh karena itu dengan adanya Aplikasi “Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah di Pulau Merah Berbasis Web” berguna untuk melakukan *monitoring* komunitas dan pengelolaan sampah serta pemberitahuan informasi tentang agenda kegiatan yang akan dilaksanakan juga informasi tentang *Ecobrick*, dalam aplikasi ini juga dapat melakukan *monitoring* perilaku masyarakat dalam membuang sampah dengan tepat atau masih belum bisa membedakan jenis-jenis sampah.

**Kata Kunci** : Sampah, Pulau Merah, *EcoRanger, Ecobrick, Monitoring*

*-- Halaman ini sengaja dikosongkan --*

## COMMUNITY MANAGEMENT CARES FOR THE ENVIRONMENT AND WASTE MANAGEMENT WEB BASED ON RED ISLAND

*By* : Ade Fian Galih Imandani

*Student Identity Number* : 361755401136

*Mentor* : 1. Dianni Yusuf, S.Kom., M.Kom.

2. Endi Sailul Haq, S.T., M.Kom.

***ABSTRACT***

*Garbage is an important problem in big countries, especially Indonesia, because there is still no handling of the increasing waste, especially in the tourism sector, for example at Red Island beach in Banyuwangi. Red Island Beach also continues to improve with how to manage waste so that all of society is aware of the emergence of garbage, but still faces the main problem, namely the low awareness of the community to sort waste. To solve that problem, There are beach clean-up activities in the Red Island which are carried out by people who care about the environment, one of which is EcoRanger, This community also makes products from plastic waste that are useful for the community and have a good impact on the environment, namely Ecobricks. With the EcoRanger community, there are several new communities formed in several tourist attractions in Banyuwangi, especially the beach, but there are still many people who do not know about the existence of this community and the activities it is holding. Notifications about the activity program and information about Ecobricks are still conventional, causing the information received by the public to be less than optimal. Therefore, with the application "Community Management Cares about the Environment and Waste Management on a* Web*-Based Red Island" is useful for community monitoring and waste management as well as the notification of information about the program of activities to be carried out and information about the Ecobricks, this application can also monitor community management in disposing of garbage properly or still can't identify between types of waste.*

*Keywords: Garbage, Red Island, EcoRanger, Ecobrick, Monitoring*

*-- Halaman ini sengaja dikosongkan --*

# KATA PENGANTAR

Puji syukur kehadirat Allah SWT atas segala rahmat dan karuniaNya sehingga penulis dapat menyelesaikan Proyek Akhir yang berjudul “Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah di Pulau Merah Berbasis Web”. Proyek akhir ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Diploma Tiga (D3) pada Program Studi Teknik Informatika Politeknik Negeri Banyuwangi.

Proyek akhir ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis ingin menyampaikan ucapan terima kasih kepada:

1. Bapak Son Kuswadi, Dr.Eng. selaku Direktur Politeknik Negeri Banyuwangi.
2. Bapak. Moh. Dimyati Ayatullah, S.T., M.Kom. selaku koordinator Program Studi Teknik Informatika.
3. Ibu Dianni Yusuf, S.Kom., M.Kom. dan Bapak Endi Sailul Haq, S.T., M.Kom. selaku pembimbing pertama dan kedua.
4. Bapak Subono, S.T., M.T. dan Moh. Nur Shodiq, S.T., M.Kom. selaku dosen penguji pertama dan kedua.
5. Ibu dan Ayah yang telah mendoakan serta memberikan dukungan dalam pengerjaan Tugas Akhir.
6. Saudara-saudaraku yang telah memberikan semangat dan dukungan serta doa dalam pengerjaan Tugas Akhir.
7. Sahabat-sahabat dan orang terbaik yang selalu memberikan semangat dan dukungan selama mengerjakan proyek akhir ini, serta teman-teman Teknik Informatika kelas 3E-TI 10 dan seluruh Mahasiswa Teknik Informatika di Politeknik Negeri Banyuwangi.

Penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan Proyek Akhir, dan penulis berharap Proyek Akhir yang telah dibuat dapat memberikan manfaat bagi penggunanya.

Banyuwangi, 27 Juli 2020

Ade Fian Galih Imandani NIM. 361755401136

*-- Halaman ini sengaja dikosongkan --*

# DAFTAR ISI

Hal

[HALAMAN JUDUL i](#_bookmark0)

[PERSEMBAHAN iii](#_bookmark1)

[MOTTO v](#_bookmark2)

[PERNYATAAN vii](#_bookmark3)

[LEMBAR PENGESAHAN ix](#_bookmark4)

[ABSTRAK xi](#_bookmark5)

[KATA PENGANTAR xv](#_bookmark6)

[DAFTAR ISI xvii](#_bookmark7)

[DAFTAR GAMBAR xxi](#_bookmark8)

[DAFTAR TABEL xxv](#_bookmark9)

[BAB 1 PENDAHULUAN 1](#_bookmark10)

* 1. [Latar Belakang 1](#_bookmark11)
  2. [Perumusan Masalah 3](#_bookmark12)
  3. [Tujuan 3](#_bookmark13)
  4. [Manfaat 3](#_bookmark14)
  5. [Batasan Masalah 4](#_bookmark15)

[BAB 2 TINJAUAN PUSTAKA 5](#_bookmark16)

* 1. [Landasan Teori 5](#_bookmark17)
     1. [Manajemen 5](#_bookmark18)
     2. [Pulau Merah 6](#_bookmark19)
     3. [EcoRanger 7](#_bookmark21)
     4. [Ecobrick 8](#_bookmark23)
     5. [Framework Laravel 10](#_bookmark25)
     6. [Bootstrap 12](#_bookmark26)
     7. [MySQL 12](#_bookmark27)
     8. [Node.JS 13](#_bookmark28)
     9. [Tawk.to 13](#_bookmark29)
     10. [MQTT 13](#_bookmark30)
     11. [UML (Unified Modelling Language) 15](#_bookmark32)
     12. [Metode RAD (Rapid Application Development) 18](#_bookmark35)
     13. [Black box testing 20](#_bookmark37)
  2. [Penelitian Terkait 21](#_bookmark38)

[BAB 3 METODE PENELITIAN 23](#_bookmark40)

* 1. [Waktu Pelaksanaan Penelitian 23](#_bookmark41)
  2. [Tempat Pelaksanaan Penelitian 23](#_bookmark42)
  3. [Jadwal Penelitian 23](#_bookmark43)
  4. [Metode Penelitian 23](#_bookmark45)
     1. [Analisa Kebutuhan 24](#_bookmark46)
     2. [Perancangan 24](#_bookmark47)
     3. [Desain Sistem 24](#_bookmark48)
     4. [Implementasi Desain 25](#_bookmark49)
     5. [Pengujian Sistem 25](#_bookmark50)
     6. [Implementasi Sistem 25](#_bookmark51)
     7. [Laporan Proyek Akhir 25](#_bookmark52)
  5. [Gambaran Umum Sistem 26](#_bookmark53)
     1. [Gambaran Umum Saat Ini 26](#_bookmark54)
     2. [Gambaran Umum Aplikasi yang diusulkan 27](#_bookmark56)
  6. [Pemodelan Sistem 28](#_bookmark59)
     1. [Use case Diagram 28](#_bookmark60)
     2. [Activity Diagram 34](#_bookmark77)
  7. [Spesifikasi Sistem 39](#_bookmark84)
     1. [Aturan Sistem 39](#_bookmark85)
  8. [Desain Aplikasi 40](#_bookmark87)
     1. [Struktur Menu 40](#_bookmark88)
     2. [Rancangan Tampilan Aplikasi 41](#_bookmark90)
     3. [Entity Relationship Diagram 50](#_bookmark110)

[BAB 4 HASIL DAN PEMBAHASAN 57](#_bookmark124)

* 1. [Hasil 57](#_bookmark125)
     1. [Halaman Utama 57](#_bookmark126)
     2. [Halaman Utama Ecobrick 59](#_bookmark132)
     3. [Halaman Login 61](#_bookmark136)
     4. [Halaman Daftar Anggota Komunitas 62](#_bookmark138)
     5. [Halaman Daftar Komunitas 62](#_bookmark140)
     6. [Halaman Beranda Pimpinan EcoRanger 63](#_bookmark142)
     7. [Monitoring Tempat Sampah 64](#_bookmark144)
     8. [Monitoring Komunitas 65](#_bookmark147)
     9. [Halaman Data User 66](#_bookmark151)
     10. [Ulasan Saran Ecobrick 68](#_bookmark156)
     11. [Riwayat Pembuangan Sampah 69](#_bookmark159)
     12. [Feedback 70](#_bookmark162)
  2. [Pembahasan 71](#_bookmark164)
     1. [Melakukan Validasi data komunitas 72](#_bookmark166)
     2. [Komunikasi data antara website dan hardware dengan MQTT 73](#_bookmark167)
  3. [Pengujian Sistem 75](#_bookmark169)
     1. [Pengujian Menu Daftar Anggota Komunitas 75](#_bookmark170)
     2. [Pengujian Halaman Login 77](#_bookmark175)
     3. [Pengujian Halaman Beranda *User* 78](#_bookmark179)
     4. [Pengujian Menu Daftar Komunitas 79](#_bookmark184)
     5. [Pengujian Menu Validasi Komunitas 81](#_bookmark188)
     6. [Pengujian Halaman Kelola Agenda Komunitas 82](#_bookmark192)
     7. [Pengujian Halaman Kelola Tempat Sampah 85](#_bookmark200)
     8. [Pengujian Menu Tambah Saran Pengaplikasian *Ecobrick* 86](#_bookmark205)
     9. [Pengujian Halaman Ulasan Saran Ecobrick 87](#_bookmark208)
     10. [Pengujian Menu Feedback 88](#_bookmark212)
     11. [Pengujian Menu Feedback dengan Live Chat Support 89](#_bookmark215)
     12. [Tabel Pengujian Black Box 90](#_bookmark218)

[BAB 5 PENUTUP 95](#_bookmark220)

* 1. [Kesimpulan 95](#_bookmark221)
  2. [Saran 95](#_bookmark222)

[DAFTAR PUSTAKA 97](#_bookmark223)

*-- Halaman ini sengaja dikosongkan --*

# DAFTAR GAMBAR

[**Gambar 2. 1** Pulau Merah 7](#_bookmark20)

[**Gambar 2. 2** Kegiatan EcoRanger 8](#_bookmark22)

[**Gambar 2. 3** Ecobrick 9](#_bookmark24)

[**Gambar 2. 4** Sistem Kerja *MQTT* 14](#_bookmark31)

[**Gambar 2.5** Alur Metode *RAD* 19](#_bookmark36)

[**Gambar 3. 1** Gambaran Umum Saat Ini 26](#_bookmark55)

[**Gambar 3. 2** Gambaran umum yang diusulkan secara kesuluruhan 27](#_bookmark57)

[**Gambar 3. 3** Gambaran umum yang diusulkan aplikasi web 27](#_bookmark58)

[**Gambar 3. 4** Use case Gabungan web dan Android 28](#_bookmark61)

[**Gambar 3. 5** Use case Web 29](#_bookmark62)

[**Gambar 3. 6** Activity Diagram Login 34](#_bookmark78)

[**Gambar 3. 7** Activity Diagram Monitoring Tempat Sampah Pintar 35](#_bookmark79)

[**Gambar 3. 8** *Activity* Diagram *Monitoring Komunitas* 36](#_bookmark80)

[**Gambar 3. 9** Activity Diagram Agenda 37](#_bookmark81)

[**Gambar 3. 10** Activity Diagram Saran Pengaplikasian Ecobrick 38](#_bookmark82)

[**Gambar 3. 11** Activity Diagram Notifikasi agenda yang mendesak 39](#_bookmark83)

[**Gambar 3. 12** Struktur Menu 41](#_bookmark89)

[**Gambar 3. 13** Rancangan Tampilan Halaman Utama 42](#_bookmark91)

[**Gambar 3. 14** Rancangan Tampilan Halaman Utama Untuk Notifikasi 42](#_bookmark92)

[**Gambar 3. 15** Rancangan Tampilan Halaman Utama dari Tempat Sampah 42](#_bookmark93)

[**Gambar 3. 16** Rancangan Tampilan Halaman Utama dari Komunitas 43](#_bookmark94)

[**Gambar 3. 17** Rancangan Tampilan Halaman Utama dari Ecobrick 43](#_bookmark95)

[**Gambar 3. 18** Rancangan Tampilan Register 44](#_bookmark96)

[**Gambar 3. 19** Rancangan Tampilan Login 44](#_bookmark97)

[**Gambar 3. 20** Rancangan Tampilan Daftakan Komunitas 45](#_bookmark98)

[**Gambar 3. 21** Rancangan Tampilan Beranda Pimpinan EcoRanger 45](#_bookmark99)

[**Gambar 3. 22** Rancangan Tampilan Beranda Petugas Lapangan 46](#_bookmark100)

[**Gambar 3. 23** Rancangan Tampilan Halaman Beranda Komunitas 46](#_bookmark101)

[**Gambar 3. 24** Rancangan Tampilan Halaman Indikasi Sampah Penuh 47](#_bookmark102)

[**Gambar 3. 25** Rancangan Tampilan Halaman Lokasi Tempat Sampah Pintar 47](#_bookmark103)

[**Gambar 3. 26** Rancangan Tampilan Halaman Kelola Agenda 48](#_bookmark104)

[**Gambar 3. 27** Rancangan Tampilan Halaman Lokasi Komunitas 48](#_bookmark105)

[**Gambar 3. 28** Rancangan Tampilan Halaman Daftar Komunitas 49](#_bookmark106)

[**Gambar 3. 29** Rancangan Tampilan Halaman Validasi 49](#_bookmark107)

[**Gambar 3. 30** Rancangan Tampilan Halaman Data Petugas Lapangan 50](#_bookmark108)

[**Gambar 3. 31** Rancangan Tampilan Halaman Ulasan Saran Ecobrick 50](#_bookmark109)

[**Gambar 3. 32** Relasi Antar Database 51](#_bookmark111)

[**Gambar 4. 1** Halaman Utama 57](#_bookmark127)

[**Gambar 4. 2** Tampilan Jadwal Agenda 58](#_bookmark128)

[**Gambar 4. 3** Tampilan Daftar Komunitas dan Tempat Sampah Pintar 58](#_bookmark129)

[**Gambar 4. 4** Tampilan *Feedback* 59](#_bookmark130)

[**Gambar 4. 5** Tampilan *Feedback* Dalam Bentuk *Live Chat Support* 59](#_bookmark131)

[**Gambar 4. 6** Halaman *Ecobrick* 60](#_bookmark133)

[**Gambar 4. 7** Tampilan Tentang *Ecobrick* 60](#_bookmark134)

[**Gambar 4. 8** Tampilan Gambar *Ecobrick* 61](#_bookmark135)

[**Gambar 4. 9** Tampilan *Login* 61](#_bookmark137)

[**Gambar 4. 10** Halaman Daftar Anggota Komunitas 62](#_bookmark139)

[**Gambar 4. 11** Tampilan Daftar Komunitas 62](#_bookmark141)

[**Gambar 4. 12** Halaman *Beranda* Pimpinan *EcoRanger* 63](#_bookmark143)

[**Gambar 4. 13** Halaman Indikasi Sampah Penuh 64](#_bookmark145)

[**Gambar 4. 14** Halaman Lokasi Sampah 64](#_bookmark146)

[**Gambar 4. 15** Halaman Daftar Komunitas 65](#_bookmark148)

[**Gambar 4. 16** Halaman Kelola Agenda Komunitas 65](#_bookmark149)

[**Gambar 4. 17** Halaman Validasi Komunitas 66](#_bookmark150)

[**Gambar 4. 18** Halaman Data Pimpinan Komunitas 67](#_bookmark152)

[**Gambar 4. 19** Halaman Data Anggota Komunitas 67](#_bookmark153)

[**Gambar 4. 20** Halaman Data Petugas Lapangan 68](#_bookmark154)

[**Gambar 4. 21** Halaman Data Petugas Konten *Reward* 68](#_bookmark155)

[**Gambar 4. 22** Halaman Ulasan Saran *Ecobrick* 69](#_bookmark157)

[**Gambar 4. 23** Tampilan Kasih Saran *Ecobrick* 69](#_bookmark158)

[**Gambar 4. 24** Halaman Riwayat Pembuang Sampah 70](#_bookmark160)

[**Gambar 4. 25** Halaman Daftar Poin Terbesar 70](#_bookmark161)

[**Gambar 4. 26** Halaman *Feedback* 71](#_bookmark163)

**Gambar 4. 27** *Source code function index* WebValidasiController 72

**Gambar 4. 28** *Source code function update* WebValidasiController 73

**Gambar 4. 29** *Source Code Subscriber.js* 74

[**Gambar 4. 30** Tampilan *Subscriber.js* saat dijalankan 74](#_bookmark168)

[**Gambar 4. 31** Tampilan Jika *Form* Daftar Aggota Komunitas Kosong 75](#_bookmark171)

[**Gambar 4. 32** Tampilan Daftar Anggota Komunitas Berhasil 76](#_bookmark172)

[**Gambar 4. 33** Tampilan Daftar Anggota Komunitas Gagal 76](#_bookmark173)

[**Gambar 4. 34** Tampilan Jika *Password* tidak sama 76](#_bookmark174)

[**Gambar 4. 35** Tampilan Jika *Form* Login Kosong 77](#_bookmark176)

[**Gambar 4. 36** Tampilan *Login* Gagal 77](#_bookmark177)

[**Gambar 4. 37** Tampilan *Login* Berhasil 77](#_bookmark178)

[**Gambar 4. 38** Halaman Beranda Pimpinan *EcoRanger* 78](#_bookmark180)

[**Gambar 4. 39** Halaman Beranda Pimpinan Komunitas 78](#_bookmark181)

[**Gambar 4. 40** Halaman Beranda Anggota Komunitas 79](#_bookmark182)

[**Gambar 4. 41** Halaman Beranda Petugas Lapangan 79](#_bookmark183)

[**Gambar 4. 42** Tampilan Berhasil Mendaftarkan Komunitas 80](#_bookmark185)

[**Gambar 4. 43** Tampilan Jika Daftar Komunitas Kosong 80](#_bookmark186)

[**Gambar 4. 44** Tampilan Daftar Komunitas Gagal 80](#_bookmark187)

[**Gambar 4. 45** Tampilan Validasi Komunitas 81](#_bookmark189)

[**Gambar 4. 46** Tampilan Komunitas Telah Tervalidasi 81](#_bookmark190)

[**Gambar 4. 47** Tampilan Daftar Komunitas Telah Tervalidasi 81](#_bookmark191)

[**Gambar 4. 48** Tampilan Tambahkan Agenda Komunitas 82](#_bookmark193)

[**Gambar 4. 49** Tampilan Data Agenda Berhasil Ditambahkan 82](#_bookmark194)

[**Gambar 4. 50** Tampilan Notifikasi Agenda Komunitas Pada Aplikasi Android 83](#_bookmark195)

[**Gambar 4. 51** Tampilan Jika Jenis Agenda Mendesak 83](#_bookmark196)

[**Gambar 4. 52** Tampilan Jika Jenis Agenda Rutin 84](#_bookmark197)

[**Gambar 4. 53** Tampilan Ketika Berhasil Merubah Agenda Komunitas 84](#_bookmark198)

[**Gambar 4. 54** Tampilan Ketika Berhasil Menghapus Agenda Komunitas 84](#_bookmark199)

[**Gambar 4. 55** Tampilan Tambahkan Lokasi Tempat Sampah 85](#_bookmark201)

[**Gambar 4. 56** Tampilan Edit Data Lokasi Tempat Sampah 85](#_bookmark202)

[**Gambar 4. 57** Tampilan Berhasil Menghapus Data Lokasi Tempat Sampah 86](#_bookmark203)

[**Gambar 4. 58** Tampilan Berhasil Merubah Data Lokasi Tempat Sampah 86](#_bookmark204)

[**Gambar 4. 59** Tampilan Jika *Form* Saran *Ecobrick* Kosong 86](#_bookmark206)

[**Gambar 4. 60** Tampilan Berhasil Mengirim Saran *Ecobrick* 87](#_bookmark207)

[**Gambar 4. 61** Tampilan Daftar Saran *Ecobrick* 87](#_bookmark209)

[**Gambar 4. 62** Tampilan *Form* Edit Data Saran *Ecobrick* 88](#_bookmark210)

[**Gambar 4. 63** Tampilan Berhasil Menghapus Data Saran Ecobrick 88](#_bookmark211)

[**Gambar 4. 64** Tampilan Jika *Form Feedback* Kosong 88](#_bookmark213)

[**Gambar 4. 65** Tampilan Berhasil Menambahkan *Feedback* 89](#_bookmark214)

[**Gambar 4. 66** Tampilan Ketika Menambahkan *Feedback* dengan *Live Chat* 89](#_bookmark216)

[**Gambar 4. 67** Tampilan Dashboard *Tawk.To* 90](#_bookmark217)

*-- Halaman ini sengaja dikosongkan --*

# DAFTAR TABEL

[**Tabel 2. 1** Use case Diagram 16](#_bookmark33)

[**Tabel 2. 2** Activity Diagram 17](#_bookmark34)

[**Tabel 2. 3** Penelitian Terkait 21](#_bookmark39)

[**Tabel 3. 1** Jadwal penelitian tugas akhir 23](#_bookmark44)

[**Tabel 3. 2** Penjelasan *Aktor* 29](#_bookmark63)

[**Tabel 3. 3** Keterangan Use case Login 30](#_bookmark64)

[**Tabel 3. 4** Keterangan Use case Monitoring Tempat Sampah Pintar 30](#_bookmark65)

[**Tabel 3. 5** Keterangan Use case Menambahkan Agenda 30](#_bookmark66)

[**Tabel 3. 6** Melihat Agenda 31](#_bookmark67)

[**Tabel 3. 7** Mendaftar sebagai anggota komunitas 31](#_bookmark68)

[**Tabel 3. 8** Mendaftarkan Komunitas 32](#_bookmark69)

[**Tabel 3. 9** Validasi Pendaftaran Komunitas 32](#_bookmark70)

[**Tabel 3. 10** Melihat Hasil Validasi 32](#_bookmark71)

[**Tabel 3. 11** Melihat Lokasi Komunitas 32](#_bookmark72)

[**Tabel 3. 12** Saran Pengaplikasian Ecobrick 33](#_bookmark73)

[**Tabel 3. 13** Ulasan Saran Ecobrick 33](#_bookmark74)

[**Tabel 3. 14** Mengirim notifikasi agenda yang mendesak 33](#_bookmark75)

[**Tabel 3. 15** Menerima notifikasi agenda yang mendesak 34](#_bookmark76)

[**Tabel 3. 16** Aturan Sistem 39](#_bookmark86)

[**Tabel 3. 17** Tabel user 52](#_bookmark112)

[**Tabel 3. 18** Tabel pimpinan\_EcoRanger 52](#_bookmark113)

[**Tabel 3. 19** Tabel petugas\_lapangan 53](#_bookmark114)

[**Tabel 3. 20** Tabel pimpinan\_komunitas 53](#_bookmark115)

[**Tabel 3. 21** Tabel anggota\_komunitas 53](#_bookmark116)

[**Tabel 3. 22** Tabel komunitas 54](#_bookmark117)

[**Tabel 3. 23** Tabel agenda 54](#_bookmark118)

[**Tabel 3. 24** Tabel tempat\_sampah 55](#_bookmark119)

[**Tabel 3. 25** Tabel *Ecobrick* 55](#_bookmark120)

[**Tabel 3. 26** Tabel point 55](#_bookmark121)

[**Tabel 3. 27** Tabel *feedback* 56](#_bookmark122)

[**Tabel 3. 28** Tabel petugas\_konten*reward* 56](#_bookmark123)

[**Tabel 4. 1** Tabel User 71](#_bookmark165)

[**Tabel 4. 2** Tabel Pengujian *Blackbox* 90](#_bookmark219)

*-- Halaman ini sengaja dikosongkan --*

# BAB 1 PENDAHULUAN

# Latar Belakang

Tahun 2016 jumlah timbulan sampah di Indonesia mencapai 65.200.000 ton per tahun dengan penduduk sebanyak 261.115.456 orang. Proyeksi penduduk Indonesia menunjukkan angka penduduk yang terus bertambah dan tentunya akan meningkatkan jumlah timbulan sampah. (Safitri, *et al*, 2018, p.3) Sampah adalah sesuatu yang tidak digunakan, tidak dipakai, tidak disenangi atau sesuatu yang dibuang yang berasal dari kegiatan manusia dan tidak terjadi dengan sendirinya. Undang-Undang Pengelolaan Sampah Nomor 18 tahun 2008 menyatakan sampah adalah sisa kegiatan sehari-hari manusia dan/atau dari proses alam yang berbentuk padat. (Andriastuti, *et al*, 2019, p.56) Sampah menjadi masalah utama di negara besar khususnya Indonesia, karena masih belum ada penanganan mengenai limbah sampah yang terus bertambah khususnya di sektor pariwisata, contohnya pada kabupaten Banyuwangi.

Banyuwangi merupakan salah satu kabupaten di Jawa Timur yang memiliki tempat wisata yang beragam, salah satunya adalah pantai Pulau Merah yang bertempat di daerah Pesanggaran tepatnya di dusun Pancer, desa Sumberagung, kabupaten Banyuwangi. Sebagai salah satu tempat wisata yang sering dikunjungi wisatawan, pantai Pulau Merah juga masih terus berbenah dengan bagaimana cara penangan tentang sampah. Kegiatan pengurangan sampah bertujuan agar seluruh lapisan masyarakat melaksanakan kegiatan pembatasan timbulnya sampah, pendauran ulang dan pemanfaatan kembali sampah atau yang lebih dikenal dengan sebutan *Reduce*, *Reuse* dan *Recycle* (3R) dengan cerdas, efisien dan terprogram. Meskipun demikian, kegiatan 3R ini masih menghadapi kendala utama, yaitu rendahnya kesadaran masyarakat untuk memilah sampah. (Suryani, 2016, p. 360) Termasuk timbulan sampah plastik hanya 10-15% saja yang telah didaur ulang, 60- 70% ditimbun di tempat pembuangan akhir, dan 15-30% belum terkelola sampai terbuang ke lingkungan terutama perairan seperti sungai, danau, pantai, dan laut. Sampah plastik pada umumnya sulit untuk didegradasi dan menyebabkan pencemaran terhadap lingkungan. Sehingga perlu dilakukan upaya mengelola sampah plastik yaitu dengan *Ecobrick*. *Ecobrick* adalah teknik pengelolaan sampah

plastik yang terbuat dari botol-botol plastik bekas yang didalamnya telah diisi berbagai sampah plastik higga penuh kemudian dipadatkan sampai menjadi keras. (Andriastuti, *et al*, 2019, p.56)

Untuk mengurangi sampah yang dihasilkan dari sektor pariwisata khususnya Pantai Pulau Merah, maka perlu adanya kesadaran dari masyarakatnya dan wisatawan yang menikmati keindahan Pantai Pulau Merah. Tingkat kesadaran yang tinggi harus ditanamankan kepada semua kalangan, maka bisa saja dilakukan dengan adanya komunitas untuk menaungi itu, yaitu salah satu komunitas yang ada di Pulau Merah saat ini adalah *EcoRanger*. Banyak jenis sampah yang ada di pantai Pulau Merah saat ini antara lain adalah limbah padat plastik, plastik yang menjadi kendala utama dalam penganan saat ini. Akumulasi limbah padat plastic adalah salah satu kelemahan utama dari penggunaan sekali pakai. Masalah yang belum terselesaikan dalam penggunaan *bioreaktor* sekali pakai adalah deteksi bahan kimia yang timbul dari plastik berbagai jenis. (Nelles, *et al*, 2017)

*EcoRanger* sebagai salah satu komunitas peduli lingkungan yang bertempat di Pulau Merah Kabupaten Banyuwangi ini selalu memiliki agenda untuk melakukan kegiatan bersih-bersih pantai dan pemilahan sampah sesuai kategori, serta mengajak masyarakat sekitar maupun wisatawan sebagai relawan untuk ikut andil dalam kegiatan tersebut. Peran komunitas di wilayah pariwisata sangat membantu untuk kelestarian lingkungan serta memberi edukasi akan betapa pentingnya peduli tentang sampah. Dengan adanya komunitas *EcoRanger* ini ada beberapa komunitas baru yang terbentuk di beberapa tempat wisata Banyuwangi khususnya pantai, namun dengan beberapa komunitas yang sudah ada tetapi masih banyak masyarakat yang belum mengerti akan adanya komunitas tersebut serta kegiatan yang diselenggarakan dan belum terorganisir dengan baik.

Berdasarkan permasalahan yang telah diuraikan, maka dibuatlah sebuah manajemen komunitas peduli lingkungan dan pengelolaan sampah di Pulau Merah dengan tujuan untuk mengatasi kurangnya kesadaran masyarakat terhadap dampak yang ditimbulkan oleh sampah khususnya wisatawan di lingkungan wisata Banyuwangi yang terintegrasi dengan tempat sampah pintar. Dengan adanya peran serta komunitas akan membantu kebersihan, kelestarian lingkungan dan cara pengelolaan sampah terutama di tempat wisata yang ada di Banyuwangi.

# Perumusan Masalah

Perumusan masalah yang diangkat pada proyek akhir adalah sebagai berikut:

* + 1. Bagaimana merancang Aplikasi berbasis Web untuk manajemen komunitas peduli lingkungan dan pengelolaan sampah di Pulau Merah ?
    2. Bagaimana melakukan *monitoring* tempat sampah dan komunitas peduli lingkungan wisata pantai Pulau Merah Banyuwangi berbasis Web ?

# Tujuan

Adapun tujuan dari pembuatan manajemen komunitas peduli lingkungan dan pengelolaan sampah di Pulau Merah sebagai berikut:

* + 1. Membuat Aplikasi berbasis Web guna melakukan *monitoring* pengelolaan sampah dan komunitas peduli lingkungan wisata pantai Pulau Merah Banyuwangi.
    2. Memberikan notifikasi serta informasi tempat sampah yang penuh dan informasi tentang komunitas peduli lingkugan pantai Pulau Merah Banyuwangi.

# Manfaat

Adapun manfaat yang dipe*role*h adalah sebagai berikut :

* + 1. Bagi Pimpinan *EcoRanger*
       1. Dapat membantu mengatasi masalah sampah di daerah wisata pantai Pulau Merah Banyuwangi
       2. Mendapatkan dukungan dari berbagai pihak
       3. Lebih dikenal oleh masyarakat luas
    2. Bagi Petugas Lapangan
       1. Dapat membantu melakukan *monitoring* tempat sampah pintar di daerah wisata pantai Pulau Merah Banyuwangi
       2. Membantu pekerjaan petugas dalam kebersihan lingkungan
    3. Bagi Komunitas
       1. Membantu untuk menyebarkan informasi tentang agenda yang direncanakan oleh komunitas
       2. Membantu untuk menyebarkan informasi tentang agenda yang direncanakan oleh komunitas
    4. Bagi *Public User*
       1. Membantu dalam pencarian informasi tentang agenda yang dilaksanakan oleh komunitas peduli lingkungan Banyuwangi sehingga kegiatan tersebut dapat terlaksana lebih optimal
       2. Dapat memberikan informasi dan publikasi mengenai pemetaan lokasi tempat sampah pintar serta komunitas.

# Batasan Masalah

Agar pembuatan laporan proyek akhir lebih jelas dan terarah, maka perlu membatasi masalah yang akan dibahas, yaitu :

* + 1. Tempat Studi kasus yang digunakan di Pantai Pulau Merah Banyuwangi
    2. Melakukan *monitoring* tempat sampah pintar dan komunitas peduli lingkungan daerah wisata pantai Pulau Merah Banyuwangi
    3. Hak akses terdiri dari : Pimpinan *EcoRanger*, Petugas Lapangan, Pimpinan Komunitas, Anggota Komunitas.
    4. Menggunakan *Google maps* API Sebagai pertanda lokasi tempat sampah pintar dan komunitas
    5. Sistem Aplikasi menggunakan Bahasa pemrograman PHP (*Framework Laravel*) dan menggunakan *database* MySQL
    6. Komunitas Peduli Lingkungan di lingkungan wisata pantai Pulau Merah Banyuwangi
    7. Autentikasi Pengguna menggunakan *username* dan *password*

# BAB 2 TINJAUAN PUSTAKA

# Landasan Teori

# Manajemen

*Management* berasal dari kata *to manage* yang berarti mengatur. Dalam hal mengatur, akan timbul masalah, problem, proses dan pertanyaan tentang apa yang diatur, siapa yang mengatur, mengapa harus diatur dan apa tujuan pengaturan tersebut. Manajemen juga menganalisa, menetapkan tujuan/sasaran serta mendeterminasi tugas-tugas dan kewajiban-kewajiban secara baik, efektif dan efisien. Dari sudut pandang luas manajemen adalah proses memanfaatkan dan mengatur suatu sumber daya yang dimiliki oleh sebuah organisasi. Dengan kerjasama anggota organisasi untuk mencapai suatu tujuan secara efektif dan efisien.

Beberapa fungsi pokok manajemen adalah sebagai berikut :

* + - 1. Perencanaan (*planning*)

Perencanaan merupakan salah satu syarat mutlak untuk dapat melaksanakan manajemen yang baik. Sebelum melakukan tindakan-tindakan yang akan dikerjakan sebelumnya harus membuat suatu perencanaan yang matang. Perencanaan berfungsi untuk memperkecil resiko yang mungkin timbul, baik resiko kekeliruan maupun resiko kemungkinan kegagalan. Dengan perencanaan yang realistis tapi ekonomis maka tujuan yang telah ditetapkan mempunyai kemungkinan besar untuk dapat dicapai, tapi secara ekonomis dapat dipertanggung jawabkan.

* + - 1. Pengorganisasian (*organizing)*

Pengorganisasian merupakan fungsi manajemen yang kedua dan merupakan langkah strategis untuk mewujudkan suatu rencana organisasi. Pengorganisasian yang dilakukan secara efektif akan dapat menjelaskan siapa yang akan melakukan apa, siapa yang akan memimpin, menjelaskan saluran-saluran komunikasi dan memusatkan sumber-sumber data terhadap sasaran-sasaran.

* + - 1. Pengarahan *(directing)*

Pengarahan meliputi pemberian petunjuk/memberi gambaran tentang kegiatan-kegiatan yang akan dilakukan sehingga para manajer harus memotivasi

staf dan personil organisasi agar secara sukarela mau melakukan kegiatan dari rencana yang sudah dibuat.

* + - 1. Koordinasi

Koordinasi merupakan salah satu fungsi manajemen. Dalam organisasi keberadaan pengorganisasian sangat penting bagi terintegrasinya seluruh kegiatan organisasi untuk mencapai tujuan. Koordinasi dapat diartikan sebagai penyesuaian bagian yang berbeda-berbeda agar kegiatan dari bagian-bagian tesebut selesai secara maksimal.

* + - 1. Pengawasan *(controlling)*

Pengawasan merupakan proses pemantauan atau pengamatan terhadap pelaksanaan kegiatan organisasi untuk menjamin agar supaya semua pekerjaan yang sedang dilakukan berjalan sesuai rencana yang telah ditentukan. Strategi untuk menghindari penyimpangan-penyimpangan dari segi pendekatan rasional terhadap aktifitas seperti penjadwalan dan ketepatan pelaksanaan kegiatan organisasi adalah dengan adanya pengawasan yang dibuat dalam fungsi manajemen.

Berkaitan dengan tujuan yang telah diuraikan, sebenarnya pengawasan merupakan proses yang mana terdiri dari 3 langkah universal yaitu mengukur perbuatan, membandingkan perbuatan dengan standar yang telah ditetapkan dan memperbaiki penyimpangan dengan tindakan pembetulan. (Wijaya & Rifa'i, 2016).

# Pulau Merah

Pulau Merah adalah tempat wisata alam yang terletak di Desa Sumberagung, Kecamatan Pesanggarang, Kabupaten Banyuwangi. Pulau ini dikenal karena adanya bukit hijau kecil bertanah merah yang terletak didekat bibir pantai. Kabupaten Banyuwangi merupakan kabupaten yang paling timur di Pulau Jawa, dengan julukan “*Sunrise of Java*” yang memiliki arti bahwa Banyuwangi adalah Kabupaten pertama yang disinari matahari di Pulau Jawa. Pulau Merah juga memiliki daya tarik yang lain yaitu pemandangan lautnya dan bukit – bukit di pinggir pantai yang indah, dan juga banyak wisatawan menunggu sore hari untuk menikmami *sunset*. Kemajuan wisata alam Pulau Merah berdampak positif terhadap masyarakat sekitar, terutama disektor perkonomian yang berkembang sangat pesat. Jumlah wisatawan yang datang ke Pulau Merah dari tahun ketahun selalu meningkat. Hal ini sesuai dengan pernyataan dari Bapak Hendro selaku

sekretaris Pulau Merah yang menyatakan bahwa jumlah pengunjung pada tahun 2014 sejumlah 200 ribu pengunjung dan pada tahun 2015 lebih dari 300 ribu pengunjung. (Harianik, *et al*, 2016) Jumlah wisatawan yang selalu meningkat setiap tahunnya menyebabkan perekonomian dan pariwisata di Pulau Merah lebih maju, setiap wisatawan pasti mengeluarkan biaya untuk menikmati wisata alam di Pulau Merah ditunjukan pada Gambar 2.1.



**Gambar 2. 1** Pulau Merah (sumber : [https://www.jejakpiknik.com/wp-](https://www.jejakpiknik.com/wp-content/uploads/2017/05/8-matajatim-com-630x380.jpg) [content/uploads/2017/05/8-matajatim-com-630x380.jpg](https://www.jejakpiknik.com/wp-content/uploads/2017/05/8-matajatim-com-630x380.jpg))

## EcoRanger

*EcoRanger* adalah tim keren di titik destinasi pariwisata Indonesia penjaga kepedulian untuk negeri bersih indah lestari. Dengan berbagi pengetahuan, wirausaha, memandu, berorganisasi, dan membangun keberdayaan. Lima kompetensi ini untuk mengurangi masalah sampah di destinasi wisata, terutama 10 Kawasan Pariwisata Nasional (KSPN). Ingin memastikan destinasi fokus selalu tertata bersih dan indah untuk dikunjungi.

*EcoRanger* ingin memastikan destinasi fokus selalu tertata bersih dan indah untuk dikunjungi. *EcoRanger* memiliki kontribusi bagi Sustainable Developments Goals yaitu diantaranya adalah :

1. Quality Education

*EcoRanger* meningkatkan kualitas pendidikan melalui berbagai Training dan Workshop.

1. Industry, Innovation, and Infrastructure

Menjadi program yang turut membangun infrastuktur dan inovasi industri.

1. Sustainable Cities and Communities

Program ini fokus pada pengembangan kapasitas masyarakat berkelanjutan, tingkah laku dan gaya hidup.

1. Responsible Consumption and Production

Konsep awal dari program ini yaitu untuk mengimplementasikan tanggungjawab atas produksi dan konsumsi.

1. Climate Action

Salah satu misi dari program ini yaitu untuk mengurangi perubahan iklim.

1. Life Below Water

*EcoRanger* akan memberikan dampak langsung terhadap perlindungan perikanan. *EcoRanger* yang ada di Banyuwangi bertempat di tempat wisata Pulau Merah, desa Sumberagung, Kecamatan Pesanggaran, Kabupaten Banyuwangi.

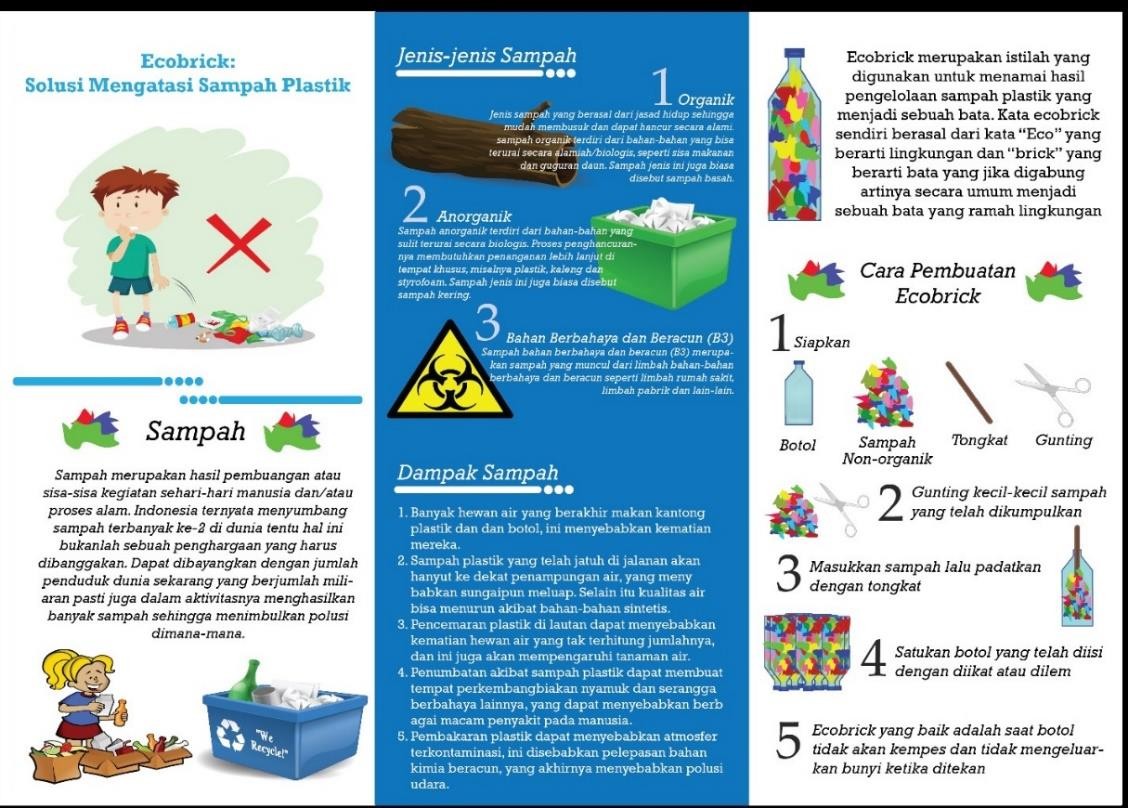
Berikut adalah gambar kegiatan *EcoRanger*, ditunjukan pada Gambar 2.2.

**Gambar 2. 2** Kegiatan *EcoRanger (*sumber: [https://www.*EcoRanger*.id/wp-](https://www.ecoranger.id/wp-content/uploads/2019/10/Cleanup-ER-1-1140x570.jpg) [content/uploads/2019/10/Cleanup-ER-1-1140x570.jpg](https://www.ecoranger.id/wp-content/uploads/2019/10/Cleanup-ER-1-1140x570.jpg)*)*

## Ecobrick

*Ecobrick* adalah cara lain utilisasi sampah-sampah nonbiological selain mengirimnya ke *landfill* (pembuangan akhir). Sampah-sampah tersebut diubah menjadi bagian-bagian kecil (*brick*) lalu dimasukkan ke dalam botol plastik. Lalu dengan bantuan kayu, brick tersebut dimampatkan agar tidak ada ruang kosong pada botol tersebut. Dengan *Ecobrick*, sampah-sampah plastik akan tersimpan dan terjaga di dalam botol sehingga tidak perlu dibakar, menggunung dan tertimbun. Fungsi dari *Ecobrick* bukan untuk menghancurkan sampah plastik, melainkan untuk memperpanjang usia plastik-plastik tersebut dan mengolahnya menjadi sesuatu yang berguna, yang bisa dipergunakan bagi kepentingan manusia pada umumnya. *Ecobrick* dapat dipergunakan sebagai furniture rumah tangga seperti meja, kursi, bahkan dinding pembatas. Hal ini sesuai dengan fungsi *Ecobrick* sebagai penggaanti bata yang tentunya lebih efisien karena ramah lingkungan serta

dapat mengurangi sampah plastik yang ada. Selain itu, *Ecobrick* juga dapat menjadi salah satu solusi untuk mengurangi dampak racun (Bisphenol-A) yang menyebar dan merusak kehidupan mahluk hidup yang ditunjukan pada Gambar 2.3.



**Gambar 2. 3** *Ecobrick* (sumber: https://kkp.go.id/djprl/bkkpnkupang/artikel/ 11508-pengelolaan-sampah-plastik-yang-mudah-dan-murah-melalui-*Ecobrick)*

Pembuatan *Ecobrick*s tidak membutuhkan skill khusus, dan tanpa biaya, karena berangkat dari bekas konsumsi sehari-hari, bisa dilakukan kapan saja, dan bisa juga dikerjakan bersama-sama maupun sendiri sambil melakukan kegiatan sehari-hari lainnya, sembari mengisi waktu. Membuat *Ecobrick* tidak sulit, hanya memerlukan ketelatenan dan sedikit usaha. Secara umum langkah-langkah membuat *Ecobrick* adalah sebagai berikut :

* + - 1. Mengumpulkan botol-botol plastik bekas, seperti botol bekas kemasan minuman (misalnya air mineral), botol bekas kemasan minyak goreng dan lain sebagainya. Kemudian mencucinya hingga bersih, lalu dikeringkan.
      2. Mengumpulkan berbagai macam kemasan plastik, seperti kemasan mie instan, minuman-minuman instan, plastik pembungkus, tas plastik dan sebagainya. Harus dipastikan plastik-plastik tersebut bebas dari segala jenis makanan (yang tersisa didalamnya), dalam keadaan kering dan tidak tercampur oleh bahan lain (klip, benang, kertas dan sebagainya).
      3. Memasukkan segala jenis plastik yang ada di poin ke 2 ke dalam botol-botol plastik pada poin ke 1.
      4. Tidak boleh tercampur dengan kertas, kaca, logam, benda-benda yang tajam dan bahan-bahan lain selain plastik.
      5. Bahan-bahan plastik yang dimasukkan ke dalam botol plastik harus dimampatkan hingga sangat padat dan mengisi seluruh ruangan dalam botol plastiknya.
      6. Cara memadatkannya bisa dengan menggunakan alat yang terbuat dari bamboo atau kayu (seperti tongkat bambu atau kayu).
      7. Jika ingin membuat sesuatu dengan hasil *Ecobrick* ini, misalnya membuat meja, kursi, atau benda-benda lain, maka bisa menggunakan botol-botol yang berukuran sama, atau bahkan dari jenis dan merk yang sama, sehingga memudahkan penyusunan.
      8. Jika menginginkan hasil yang berwarna-warni, maka plastik-plastik kemasan yang disusun didalamnya bisa diatur sedemikian rupa sehingga menghasilkan warna sesuai yang diinginkan. Bisa juga dengan cara membungkus botol plastik dengan cellophone/pita perekat yang berwarna.
      9. Setelah semua botol plastik diisi dengan kemasan-kemasan plastik hingga padat, maka botol-botol plastik tersebut siap disusun dan digabungkan menjadi benda lain, seperti meja, kursi, bahkan dinding dan atau lantai panggung, pembatas ruangan dan banyak lagi lainnya. (Santi, 2018)

## Framework Laravel

Laravel adalah sebuah *framework* PHP yang dirilis dibawah lisensi MIT, dibangun dengan konsep MVC (model view controller). Laravel adalah pengembangan *website* berbasis MVP yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan menghemat waktu.

Kelebihan *Framework* ini juga terdapat pada fiturnya antara lain *Modularity* yaitu penyedia berbagai macam modul dan *library*. Kedua fitur ini dapat membantu meningkatkan dan menyempurnakan aplikasi web. Seluruh modul yang tersedia di

*Laravel* terintegrasi dengan *Composer dependency* manager yang dapat membantu mempermudah proses pembaruan. *Testability* yaitu pemelihara *source code* sesuai dengan kebutuhan pengguna. *Schema Builder* dengan mengelola berbagai jenis definisi *database* dan skema di dalam kode PHP*. Eloquent* ORM membantu untuk meng-query *database* menggunakan berbagai macam rangkaian metode sederhana.

Kekurangan dari *Framework Laravel* ini yaitu memiliki ukuran file yang cukup besar yang mengakibatkan kinerjanya sedikit lebih berat dan membutuhkan koneksi internet untuk installasi dan mendownload library.

Struktur dari *framework* laravel ini berisikan antara lain :

* + - 1. App, yaitu kumpulan logika dan alur sistem yang akan dibuat.
      2. *Bootstrap***,** direktori ini berisi beberapa file kerangka *framework* laravel termasuk autoload guna mengoptimasi kinerja sistem yang dihasilkan.
      3. Config, mencakup seluruh konfigurasi *framework* mulai dari *database*, app, mail, dan lain sebagainya.
      4. *Database***,** sebagai folder penampung file migrations dan seeds yang berhubungan langsung ke pengolahan data dalam *database*.
      5. *Public***,** sebagai folder yang akan diakses oleh *public*/*user*s nantinya. Folder ini juga berisi file-file assets (css/js/images/dll).
      6. Resource, folder ini berisi semua resource untuk bagian frontend.
      7. Storage**,** berisi compiled blade templates, session, cache, logs dan file lainnya yang digenerate otomatis oleh *framework*.
      8. Tests**,** berisi semua test yang kita buat untuk aplikasi.
      9. Vendor**,** berisi seluruh library-library yang digunakan dalam *framework* laravel maupun yang diinstall melalui composer.

MVC adalah sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi. MVC memisahkan aplikasi berdasarkan komponen- komponen aplikasi, seperti : manipulasi data, controller, dan *user* interface.

1. Model, Model mewakili struktur data. Biasanya model berisi fungsi-fungsi yang membantu seseorang dalam pengelolaan basis data seperti memasukkan data ke basis data, pembaruan data dan lain-lain.
2. View, View adalah bagian yang mengatur tampilan ke pengguna. Bisa dikatakan berupa halaman web.
3. Controller, Controller merupakan bagian yang menjembatani model dan view. (Luthfi, 2016).

## Bootstrap

*Bootstrap* adalah *framework* CSS yang dikhususkan untuk pengembangan *front-end website*. *Framework* ini mempunyai nama asli *Twitter Blueprint*. Ada kata ‘*Twitter’* karena pada awalnya dikembangkan untuk sosial media *Twitter* yang sangat terkenal saat ini dengan pengguna hampir 326 juta orang. Sebelum *Bootstrap* muncul, sudah ada *framework* lain yang tersedia. Hanya saja tingkat konsistensi dalam proses pengembangan buruk. Selain itu juga butuh biaya perawatan mahal. Berdasarkan permasalahan ini, akhirnya para pengembang menemukan *Bootstrap*. Berbeda dengan *framework* sebelumnya, *Bootstrap* dikenal konsisten dan lebih sederhana. *Bootstrap* juga memungkinkan *developer* dapat mengembangkan *website* dengan mudah dan cepat. *Developer* hanya perlu memanggil *class* tertentu untuk membuat tombol, panel, tabel, pesan peringatan, dan lain sebagainya. (Yasin, 2019)

## MySQL

Menurut (Sucipto, 2017) *Database* adalah kumpulan data yang dihubungkan secara bersama-sama, dan gambaran dari data yang dirancang untuk memenuhi kebutuhan informasi dari suatau organisasi. Secara umum *database* dapat diartikan sebagai sebuah tempat pemyimpanan data sebagai pengganti dari sistem konvesional yang berupa dokumen file. Perancangan *database* sendiri terdapat tiga fase utama yaitu perancangan *database* konseptual yang merupakan proses membangun model dari data yang digunakan dalam sebuah organisasi dan tidak tergantung pada petimbangan fisik, perancangan *database* logical, merupakan proses membangun model dari informasi yang digunakan dalam perusahaan berdasarkan model data spesifikasi, dan terbebas dari DBMS (*Database Management System*) tententu dan pertimbangan fisik lainnya.

Basis data merupakan suatu kesatuan dari kumpulan informasi yang secara utuh disimpan di dalam perangkat keras (komputer) secara sistematis sehingga dapat diolah menggunakan perangkat lunak. Dapat diartikan bahwa basis data adalah kumpulan dari catatan-catatan, atau potongan dari pengetahuan. Sistem basis data merupakan suatu sistem yang menyusun dan mengelola *record* menggunakan

komputer untuk menyimpan atau merekam serta memelihara data operasional lengkap. Sehingga membantu suatu perusahaan/organisasi mampu menyediakan informasi yang diperlukan pemakai untuk proses mengambil keputusan. (Qalbi, 2014).

# Node.JS

Node.js adalah plat*form* buatan Ryan Dahl untuk menjalankan aplikasi web berbasis [*Javascript*](https://www.niagahoster.co.id/blog/belajar-javascript/) yang dikenalkan pada tahun 2009. Dengan plat*form* ini, Anda dapat menjalankan *Javascript* dari sisi server. Untuk mendukung kemampuan tersebut, Node.js dibangun dengan engine *Javascript* V8 milik Google. Di samping itu, Node.js juga memiliki pustaka server sendiri sehingga Anda tidak perlu menggunakan program server web seperti Nginx dan [Apache](https://www.niagahoster.co.id/blog/apache-adalah/). Dengan model event- driven dan non-blocking I/O-nya, Node.js lebih mampu menangani banyak proses secara bersamaan daripada plat*form* bersifat thread-based networking. Node.js menyediakan beragam library *javascript* jadi kita lebih mudah untuk mengembangkan aplikasi. (Tua, 2018).

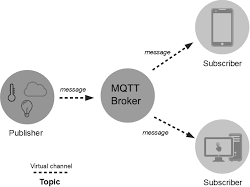
## Tawk.to

[*Tawk.to*](http://tawk.to/) merupakan perusahaan besar yang memberikan layanan aplikasi berupa fitur chat secara langsung menggunakan pihak ketiga untuk layanan *website* dan pemakaiannya tidak berbayar, layanan yang diberikan sangat lengkap untuk kelas penyedia layanan gratis dan bisa memenuhi semua yang dibutuhkan untuk dukungan pelanggan lebih cepat dan handal. Pada layanan ini bisa digunakan untuk mengirim teks dan file berupa gambar dan sejenisnya. Untuk memakai layanan ini harus daftar menggunakan email pada plat*form website tawk.to*. (Astina, 2016).

## MQTT

*Message Queuing Telemetry Transport (MQTT)* protokol merupakan sebuah protokol yang berjalan diatas *stack* TCP/IP dan dirancang khusus untuk *machine to machine* yang tidak memiliki alamat khusus. Maksud dari kata tidak memiliki alamat khusus ini seperti halnya sebuah *arduino, raspi* atau *device* lain yang tidak memiliki alamat khusus. Sistem kerja *MQTT* menerapkan *Publish* dan *Subscribe* data. Dan pada penerapannya, *device* akan terhubung pada sebuah *Broker* dan mempunyai suatu *Topic* tertentu.

Berikut penjelasan bagian-bagian dari *MQTT*:

1. ***Broker*** pada *MQTT* berfungsi untuk menghandle data *publish* dan *subscribe* dari berbagai *device*, bisa diibaratkan sebagai server yang memiliki alamat IP khusus. Beberapa contoh dari *Broker* yang ada seperti Mosquitto, HiveMQ dan Mosca.
2. ***Publish*** merupakan cara suatu *device* untuk mengirimkan datanya ke *subscriber*s. Biasanya pada *publish*er ini adalah sebuah *device* yang terhubung dengan sensor tertentu.
3. ***Subscribe*** merupakan cara suatu *device* untuk menerima berbagai macam data dari *publish*er. *Subscriber* dapat berupa aplikasi *monitoring* sensor dan sebagainya, *subscriber* ini yang nantinya akan meminta data dari *publish*er.
4. ***Topic*** seperti halnya pengelompokan data disuatu kategori tertentu. Pada sistem kerja *MQTT*, *topic* bersifat wajib hukumnya. Pada setiap transaksi data antara *Publish*er dan *Subscriber* harus memiliki suatu *topic* tertentu.

**Gambar 2. 4** Sistem Kerja *MQTT*

*MQTT* bersifat lightweight message artinya *MQTT* berkomunikasi dengan mengirimkan data pesan yang memiliki header berukuran kecil yaitu hanya sebesar 2bytes untuk setiap jenis data, sehingga dapat bekerja di dalam lingkungan yang terbatas sumber dayanya seperti kecilnya bandwidth dan terbatasnya sumber daya listrik, selain itu protokol ini juga menjamin terkiriminya semua pesan walaupun koneksi terputus sementara, protokol *MQTT* menggunakan metode *publish*/*subscribe* untuk metode komunikasinya. *Publish*/*subscribe* sendiri adalah sebuah pola pertukaran pesan di dalam komuunikasi jaringan dimana pengirim data disebut *publish*er dan penerima data disebut dengan *subscriber*. (Rochman, *et al*, 2017)

## UML (Unified Modelling Language)

*Unified Modelling Language (UML)* adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. *UML* menawarkan sebuah standar untuk merancang model sebuah sistem. *UML* mendefinisikan notasi dan *syntax/semantic* seperti bahasa-bahasa lainnya. Membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi dapat berjalan pada perangkat keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun yaitu dengan menggunakan *UML*. *UML* mempunyai beberapa diagram diantaranya adalah *Use case* Diagram dan *Activity* Diagram. (Rembulan, 2015).

1. *Use case* Diagram

Dalam sebuah sistem *use case* diagram diharapkan dapat menggambarkan fungsionalitas sebuah sistem. Yang mana menekankan “apa” yang dikerjakan sistem, dan bukan “bagaimana”. Jadi dalam sebuah *use case* itu merepresentasikan sebuah interaksi antara aktor dengan sistem. Dapat diartikan *use case* adalah sebuah pekerjaan seperti *Login* ke sistem, membuat sebuah daftar belanja dan lain-lain. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. Jadi *use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat.

Kelebihan dari *use case* diagram sendiri dapat membantu mengkomunikasikan rancangan dengan klien, menyusun *requirement* sebuah sistem, dan merancang *test case* untuk semua *feature* yang ada pada sistem. Sebuah *use case* dapat meng-*include* fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya. Secara umum diasumsikan bahwa *use case* yang di-*include* akan dipanggil setiap kali *use case* yang meng-*include* dieksekusi secara normal. Sebuah *use case* dapat di-*include* oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*. Sebuah *use case* juga dapat meng-*extend use case* lain dengan *behaviour*- nya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case* yang satu merupakan spesialisasi dari yang lain.

Pada *Use case* Diagram terdapat komponen/simbol-simbol yang ditunjukkan pada Tabel 2.1. (Yusmiarti, 2016).

**Tabel 2. 1** *Use case* Diagram

|  |  |  |
| --- | --- | --- |
| **No.** | **Simbol** | **Deskripsi** |
| 1. | Nama *use* | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal frase nama *.* |
| 2. | Aktor/*actor* | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor. |
| 3. | Asosiasi/*association* | Komunikasi antara aktor dan yang berpartisilpasi pada *use case* atau *use case* memiliki interaksi dengan aktor. |
| 4. | Ekstensi/*extend* | Relasi tambahan kesebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan itu, mirip dengan prinsip inheritance pada pemrograman berorientasi objek, biasanya *use case* tambahan memiliki nama depan yang sama dengan *use case* yang ditambahkan. |
| 5. | Generalisasi/*generalization* | Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya. |
| 66. | Menggunakan / *include* | Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan memerlukan *use case* untuk menjalankan fungsinya atau sebagai syarat dijalankan *use case* ini. |

1. *Activity* Diagram

*Activity* diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang. Struktur diagram mirip seperti flowchart atau Data Flow

Diagram pada perancangan terstruktur. Sangat bermanfaat apabila kita membuat diagram ini terlebih dahulu dalam memodelkan sebuah proses untuk membantu memahami proses secara keseluruhan. *Activity* diagram berfungsi untuk menggambarkan *workflow* / aliran kerja dari suatu proses bisnis. Suatu aliran kerja bisa saja dituangkan dalam bentuk narasi / teks, akan tetapi jika aliran kerjanya sudah kompleks maka kita akan kesulitan untuk membayangkan bagaimana proses itu terjadi. Oleh karena itu, dibuatlah *Activity* diagram sebagai salah satu cara untuk menggambarkan aliran kerja tersebut. (Harisantyo, *et al*,2015).

Menurut (Rembulan, 2015) Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

Pada *Activity* diagram terdapat komponen/simbol-simbol yang ditunjukkan pada Tabel 2.2. (Yusmiarti, 2016)

**Tabel 2. 2** *Activity* Diagram

|  |  |  |
| --- | --- | --- |
| No. | Simbol | Deskripsi |
| 1. | Status awal | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal |
| 2. | Aktivitas  aktivitas | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja. |
| 3. | Percabangan / *Decision* | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu |

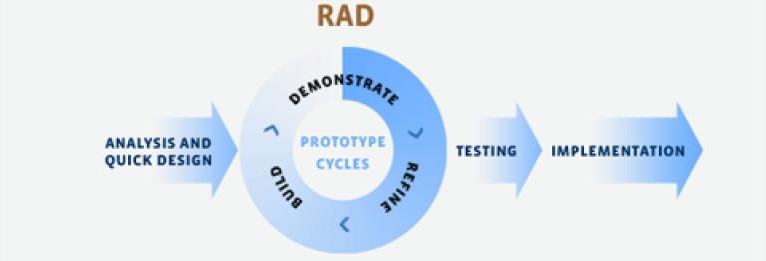
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4. | Penggabungan/ JoinAsosiasi | | | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu. |
| 5. | Status akhir | | | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir. |
| 6. | Swimlane | | | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. |
|  | Nama swimlane |  |
|  |

## Metode RAD (Rapid Application Development)

Metode *RAD (Rapid Application Development)* sebagai salah satu *alternative* dari metode *SDLC (System Development Life Cycle).* Metode *SDLC* banyak diterapkan untuk mengatasi keterlambatan yang terjadi apabila menggunakan metode konvensional. Metode *RAD* merupakan metode yang sesuai digunakan untuk menghasilkan sistem perangkat lunak dengan kebutuhan yang mendesak dan waktu yang singkat dalam penyelesaiannya. Metode *RAD* merupakan pendekatan berorientasi objek untuk menghasilkan sebuah sistem dengan sasaran utama mempersingkat waktu pengerjaan aplikasi dan proses agar sesegera mungkin memberdayakan sistem perangkat lunak tersebut secara tepat dan cepat. Metode *RAD (Rapid Application Development)* menggunakan metode *iteratif* (berulang) dalam mengembangkan sistem dimana model bekerja sistem dikonstruksikan di awal tahap pengembangan dengan tujuan menetapkan kebutuhan pengguna. (Kosasi & Yuliani, 2015)

Dalam pengembangan sistem informasi normal, memerlukan waktu minimal 180 hari, namun dengan menggunakan metode *RAD*, sistem dapat

diselesaikan dalam waktu 30-90 hari. Dapat dilihat gambar alur metode *RAD* pada Gambar 2.5.



**Gambar 2.5** Alur Metode *RAD*

Model RAD memiliki 3 tahapan sebagai berikut.

* + - 1. Rencana Kebutuhan (*Requirement Planning*): *User* dan *analyst* melakukan pertemuan untuk mengidentifikasi tujuan dari sistem dan kebutuhan informasi untuk mencapai tujuan. Pada tahap ini merupakan hal terpenting yaitu adanya keterlibatan dari kedua belah pihak.
      2. Proses Desain Sistem (*Design Sistem*): Pada tahap ini keaktifan *user* yang terlibat menentukan untuk mencapai tujuan karena pada proses ini melakukan proses desain dan melakukan perbaikan-perbaikan apabila masih terdapat ketidaksesuaian desain antara *user* dan *analyst*. Seorang *user* dapat langsung memberikan komentar apabila terdapat ketidaksesuaian pada desain, merancang sistem dengan mengacu pada dokumentasi kebutuhan *user* yang dibuat pada tahap sebelumnya. Keluaran dari tahapan ini adalah spesifikasi *software* yang meliputi organisasi sistem secara umum, struktur data dan yang lain.
      3. Implementasi (*Implementation*): Tahapan ini adalah tahapan *programmer* yang mengembangkan desain suatu program yang telah disetujui oleh *user* dan *analyst.* Sebelum diaplikasikan pada suatu organisasi terlebih dahulu dilakukan proses pengujian terhadap program tersebut apakah ada kesalahan atau tidak. Pada tahap ini *user* biasa memberikan tanggapan akan sistem yang sudah dibuat serta mendapat persetujuan mengenai sistem tersebut.

Beberapa Keunggulan Model *RAD* :

1. Model RAD dapat mempersingkat waktu pengerjaan sampai kurang dari 3 bulan dan progresnya bisa diukur.
2. Lebih fleksibel karena dapat melakukan proses desain ulang pada saat yang bersamaan.
3. Tahapan yang dilakukan pengembang memiliki kemampuan untuk menggunakan kembali komponen yang ada (*reusable object*) sehingga pengembang tidak perlu membuat dari awal lagi.

Kelemahan Model *RAD* :

1. Proyek yang besar dan berskala, *RAD* memerlukan sumber daya manusia yang memadai untuk menciptakan jumlah tim yang baik.
2. *RAD* menuntut pengembang dan pelanggan memiliki komitmen dalam aktivitas *rapid fire* yang diperlukan untuk melengkapi sebuah sistem dalam waktu yang singkat. Jika komitmen tersebut tidak ada maka proyek *RAD* akan gagal. (Rukmana & Desiyani, 2017)

## Black box testing

*Black box testing* merupakan teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak. Keuntungan dalam penggunaan metode *Blackbox Testing* adalah penguji tidak perlu memiliki pengetahuan tentang bahasa pemrograman tertentu, pengujian yang dilakukan dari sudut pandang pengguna, ini membantu untuk mengungkapkan ambiguitas atau inkonsistensi dalam spesifikasi persyaratan, dan *programmer* dan *tester* keduanya saling bergantung sama lain. Sedangkan kekurangan dari metode ini adalah uji kasus sulit didesain tanpa spesifikasi yang jelas, kemungkinan memiliki pengulangan tes yang sudah dilakukan oleh *programmer*, dan beberapa bagian *back end* tidak diuji sama sekali. (Jaya, 2018)

*Black box testing* bukanlah solusi alternatif dari *White Box Testing* tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh *White Box Testing*. (Mustaqbal, *et al*, 2015) *Black box testing* cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*).
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan per*form*ansi (*performance errors*).
5. Kesalahan inisialisasi dan terminasi.

Kelebihan dari *Black box testing ini* tidak perlu melihat *source code* secara detail, mendeteksi kesalahan pengetikan, mendeteksi kesalahan design sebuah *website*, menampilkan asumsi yang tidak sesuai dengan kenyataan untuk di analisa dan diperbaiki. Seorang Tester tidak harus Programmer.

Sedangkan kekurangannya masih tergantung pada dokumen dan design software tersebut. Tidak sampai level code, sehingga tester tidak mengetahui level security dari software tersebut.

# Penelitian Terkait

Aplikasi Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah di Pulau Merah Berbasis Web ini dibuat dengan referensi dari penelitian- penelitian terkait yang hampir sejenis dengan aplikasi yang dibangun.

Untuk lebih detail dijelaskan pada Tabel 2.3 Penelitian Terkait.

**Tabel 2. 3** Penelitian Terkait

|  |  |  |  |
| --- | --- | --- | --- |
| *Comparison*  *Elements* | (Wulandari, 2018) | (Ramadhan & Latifah,  2018) | (Nugroho, 2017) |
| *Objects* | Dinas Kebersihan dan  Pertamanan Kota Pekanbaru | Umum | Kota Yogyakarta |
| *Tittles* | Sistem Informasi Geografis Pemetaan Titik Lokasi Pembuangan Sampah Kota Pekanbaru | Perancangan Komunitas Pencinta Kucing Dengan Metode Waterfall Berbasis Web | Komunitas Muda Urban Mengelola Sampah: Kajian Partisipatoris Gerakan Peduli Sampah Nasional di Kota  Yogyakarta |
| *Features* | Merancang Sistem Informasi Pemetaan Titik Lokasi Pembuangan Sampah Kota Pekanbaru dan fitur laporan untuk mengelola pengangkutan sampah akan memudahkan petugas | Forum komunitas kucing berbasis web ini dapat mempermudah para pecinta kucing menambah wawasan seputar kucing. Dapat berinteraksi sesama pecinta kucing. Dapat memberikan informasi mengenai kucing  dengan adanya artikel. | Beberapa komunitas muda urban di Yogyakarta berkolaborasi dalam Aliansi Komunitas Yogyakarta menginisiasi suatu gerakan lingkungan dengan nama Peduli Sampah Nasional (PESAN) 2017 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | dalampembuatan laporan angkut sampah untuk dapat dijadikan evaluasi kinerja pengangkutan dan *monitoring* keberadaan  TPS atau sampah. | Desain forum komunitas kucing ini terlihat *user* friendly. |  |
| *Technology* | Web (*Framework CodeIgniter)* | Web (Metode *Waterfall*) | Poster Kegiatan Peduli Sampah Nasional  (PESAN) 2017 |
|  |  |  | Tiap komunitas muda |
|  | salah satu bentuk media |  | urban berhak atas klaim |
|  | informasi dan publikasi |  | bagaimana berkontribusi |
|  | mengenai pemetaan | Forum komunitas | dalam pelestarian |
|  | lokasi TPS kepada | pencinta kucing ini | lingkungan. Maka |
|  | masyarakat dan | dapat berinteraksi | menjadi luar biasa |
|  | *stakeholder*. adanya | sesama pencinta kucing | ketika komunitas yang |
|  | pemetaan TPS | tanpa harus bertatap | tergabung dalam Aliansi |
| *Advantages* | memudahkan dalam | muka atau bertemu | Komunitas Yogyakarta |
|  | pemeliharaan data, | langsung dan tanpa | ini mengesampingkan |
|  | mengamati, mengevaluasi | harus tercampur dengan | ego komunitas masing- |
|  | dan *monitoring* | forum komunitas lain | masing dan bersama |
|  | persebaran dan | atau social media | menginisiasi suatu |
|  | pengelolaan TPS atau | lainnya. | gerakan. Suatu sikap |
|  | sampah di Pekanbaru. |  | dewasa yang ditunjukan |
|  |  |  | oleh para komunitas |
|  |  |  | muda urban |
|  |  |  | Kegiatan bersih sampah |
|  |  |  | ini masih menginfokan |
|  | Pada aplikasi ini hanya | Pada Aplikasi ini hanya | lewat poster tanpa ada |
|  | digunakan untuk pemetaan | sebatas untuk bertukar | teknologi lain yang |
| *Deficiencies* | lokasi TPS dan berfokus | informasi, aplikasi ini | dapat mengoptimalkan |
|  | pada melakukan *monitoring* | masih perlu | dalam memberitahukan |
|  | sampah | dikembangkan kembali | komunitas maupun |
|  |  |  | masyarakat akan adanya |
|  |  |  | kegiatan tersebut |

# BAB 3 METODE PENELITIAN

# Waktu Pelaksanaan Penelitian

Penelitian ini dijadwalkan akan dilaksanakan selama kurang lebih 6 bulan (enam bulan) terhitung mulai bulan Februari s/d Juli 2020.

# Tempat Pelaksanaan Penelitian

Tempat penelitian Proyek Akhir ini akan dilakukan di Pantai Pulau Merah di dusun Pancer, Sumberagung, Pesanggaran, Kabupaten Banyuwangi, Jawa Timur. Pada Komunitas *EcoRanger*.

# Jadwal Penelitian

Berikut adalah jadwal penelitian yang akan dilakukan sesuai waktu pelaksanaan dan mencakup tahap perencanaan, desain sistem, pengujian, evaluasi dan penyusunan laporan, berikut rinciannya ditunjukkan pada Tabel 3.1.

**Tabel 3. 1** Jadwal penelitian tugas akhir

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Kegiatan | 2020 Bulan Ke- | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | | | | 3 | | | | 4 | | | | 5 | | | | 6 | | | | 7 | | | |
| 1 | Analisa Kebutuhan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Perancangan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Desain Sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Implementasi Desain |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Pengujian Sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Implementasi Sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | Laporan Proyek  Akhir |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# Metode Penelitian

Metode yang digunakan dalam pengerjaan proyek akhir ini yaitu metode *Rapid Application Development* (RAD). Metode RAD menggunakan metode berulang dalam mengembangkan sistem. RAD menekankan pada siklus pembangunan pendek, singkat, dan cepat. Jadi metode RAD sesuai untuk menghasilkan sistem perangkat lunak dengan kebutuhan yang mendesak dan menyingkat waktu dalam penyelesaiannya. RAD juga mempunyai kemampuan untuk menggunakan kembali komponen yang ada (*reusable object*) sehingga pengembang tidak perlu membuat dari awal lagi dan waktu lebih singkat.

# Analisa Kebutuhan

Tahapan ini merupakan pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh pengguna. Adapun data yang dibutuhkan untuk pembuatan Aplikasi Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah di Pulau Merah Berbasis Web ini antara lain :

1. Data Lokasi

Data lokasi ini untuk menentukan letak dari tempat sampah pintar dan letak dari komunitas agar dapat diketahui saat ditampilkan dalam *website*. Data lokasi ini ditampilkan dalam *website* bentuk *google maps* dan untuk menandai lokasi tersebut menggunakan *marker*. *Marker* ini membutuhkan dua hal penting yaitu posisi untuk koordinat *latitude* dan *longitude marker* pada peta serta map objek dari peta (*Google maps*).

1. Data Komunitas

Untuk data komunitas sendiri sekarang masih lingkup daerah Pantai Pulau Merah di Banyuwangi yaitu komunitas *EcoRanger*. Data komunitas ini bisa ditambahkan oleh pihak daerah pantai langsung melalui *website* yang akan dibuat.

# Perancangan

Hasil dari perencanaan sistem, menggunakan sistem berbasis web menggunakan *Framework laravel* yang dapat diakses dan sebagai media *monitoring* dan informasi yang *real time* dan *up to date*, dengan menggunakan sistem manajemen pada pengguna membedakan hak akses dan dirancang sesuai dengan kebutuhan pengguna.

# Desain Sistem

Pada tahap ini, dilakukan implementasi hasil dari desain Aplikasi Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah di Pulau Merah Berbasis Web yang akan diimplementasi dengan menggunakan Bahasa pemrograman PHP (*Framework laravel*) dan MySQL Server 5.7 dapat digunakan untuk membuat dan mengelola *database* beserta isinya. *Framework laravel* Memiliki banyak fitur yang dibutuhkan serta termasuk *Framework* yang baru dan modern.

# Implementasi Desain

Pada tahap ini, dilakukan implementasi hasil dari desain Aplikasi Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah berbasis Web yang akan diimplementasi dengan Tempat Sampah Pintar menggunakan Bahasa pemrograman *PHP* (*Framework laravel*) dan MySQL dapat digunakan untuk membuat dan mengelola *database* beserta isinya.

# Pengujian Sistem

Setelah sistem terbentuk, dilakukan pengujian untuk mengetahui kekurangan dan kelemahan sistem yang kemudian dilakukan pengkajian ulang dan perbaikan terhadap aplikasi menjadi lebih baik. Pengujian dilakukan juga untuk mengetahui seberapa kuat aplikasi yang telah dibangun. Pengujian aplikasi dilakukan dengan metode *black box testing* yang merupakan pengujian yang melihat aplikasi hanya dari luar saja tanpa tau yang terjadi di dalam aplikasi.

# Implementasi Sistem

Implementasi merupakan suatu proses untuk menerapkan Aplikasi Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah di Pulau Merah Berbasis Web yang terintegrasi dengan tempat sampah pintar dan mobile untuk melakukan *monitoring* yang telah dibuat. Pada tahap ini, implementasi dilakukan dengan beberapa proses, yaitu:

* + - 1. Melakukan *monitoring* tempat sampah pintar untuk kebersihan lingkungan wisata pantai.
      2. Memperkenalkan aplikasi kepada Masyarakat, Wisatawan lokal atau asing yang sedang berada di tempat wisata pantai Pulau Merah Banyuwangi.
      3. Melatih kesadaran masyarakat tentang kepedulian sampah yang ada di lingkungan wisata dengan bantuan Aplikasi yang dibuat.

# Laporan Proyek Akhir

Setelah melakukan implementasi sistem, tahap terakhir adalah membuat laporan proyek akhir sebagai bentuk hasil proyek akhir yang sudah dikerjakan secara tertulis. Dalam pembuatan laporan proyek akhir diisi dengan seluruh kegiatan sudah dijadwalkan sebelumnya dan dilengkapi dokumentasi. Tidak hanya itu perlu juga memberikan bukti bahwa proyek akhir sudah dilaksanakan.

# Gambaran Umum Sistem

# Gambaran Umum Saat Ini

**Gambar 3. 1** Gambaran Umum Saat Ini

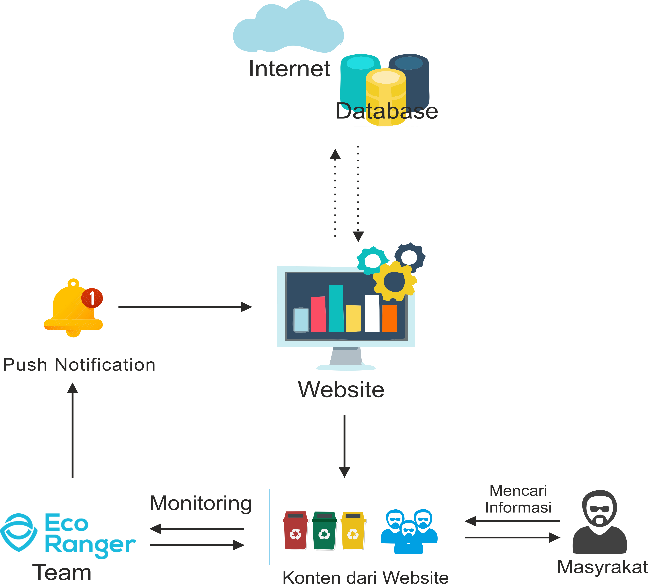
Pada Gambar 3.1 terdapat gambaran umum saat ini yang menjelaskan tentang keadaan yang ada pada Pantai Pulau Merah mengenai lingkungan yang mana prosesnya masih dilakukan secara konvensional. Untuk memperjelas alurnya akan dijelaskan sebagai berikut: daerah tempat wisata pantai sebagian besar pasti sudah menyediakan tempat sampah baik berbentuk tempat sampah biasa maupun yg sudah dibedakan berdasarkan warna dan tulisan sesuai kategori dari sampah tersebut. Dengan pembagian berdasarkan kategori tersebut tetap masih banyak orang membuangnya ke tempat sampah yang salah, jadi masih banyak sampah yang tercampur pada satu wadah kategori sampah. Meskipun sudah disediakan tempat sampah oleh pihak pengelola wisata tetapi masih ada sampah berserakan dan menjadikan tempat tersebut menjadi kotor. Ada juga sampah yang terbawa oleh ombak laut dan berserakan di tepi pantai yang membuat pantai tersebut kelihatan kotor, untuk menangani hal tersebut munculah komunitas *EcoRanger* Pantai Pulau Merah yang mengadakan agenda bersih pantai dari sampah setiap minggunya. Kegiatan tersebut hanya diinfokan melalui pemberitahuan penjaga pantai dan juga melalui media social itupun hanya sebatas *story whatsapp* juga *Instagram* yang mengakibatkan kegiatan tersebut kurang optimal. Wisatawan pun juga masih kurang adanya kesadaran atas lingkungan tetapi sesudah ada kegiatan tersebut ada

sebagian warga yang ikut andil meskipun masih sedikit. Komunitas yang peduli terhadap lingkungan di Pantai Pulau Merah salah satunya adalah komunitas *EcoRanger* yang memiliki agenda kegiatan seperti *clean beach up* atau pemilahan sampah di sentra kelola sampah (Sekolah) yang selanjutnya sampah tersebut di kelola menjadi bahan-bahan yang berguna contohnya *Ecobrick*. Masyarakat / wisatawan juga bisa menukarkan sampahnya di bank sampah di Pantai Pulau Merah yang bekerjasama dengan pegadaian dan akan ditukarkan dengan emas.

# Gambaran Umum Aplikasi yang diusulkan

**Gambar 3. 2** Gambaran umum yang diusulkan secara kesuluruhan

Pada Gambar 3.2 dijelaskan bahwa gambaran umum yang diusulkan secara keseluruhan dengan membuat Hardware, Aplikasi Android dan *Website*. Untuk aplikasi *website* yang diusulkan dapat dijelaskan pada Gambar 3.3.



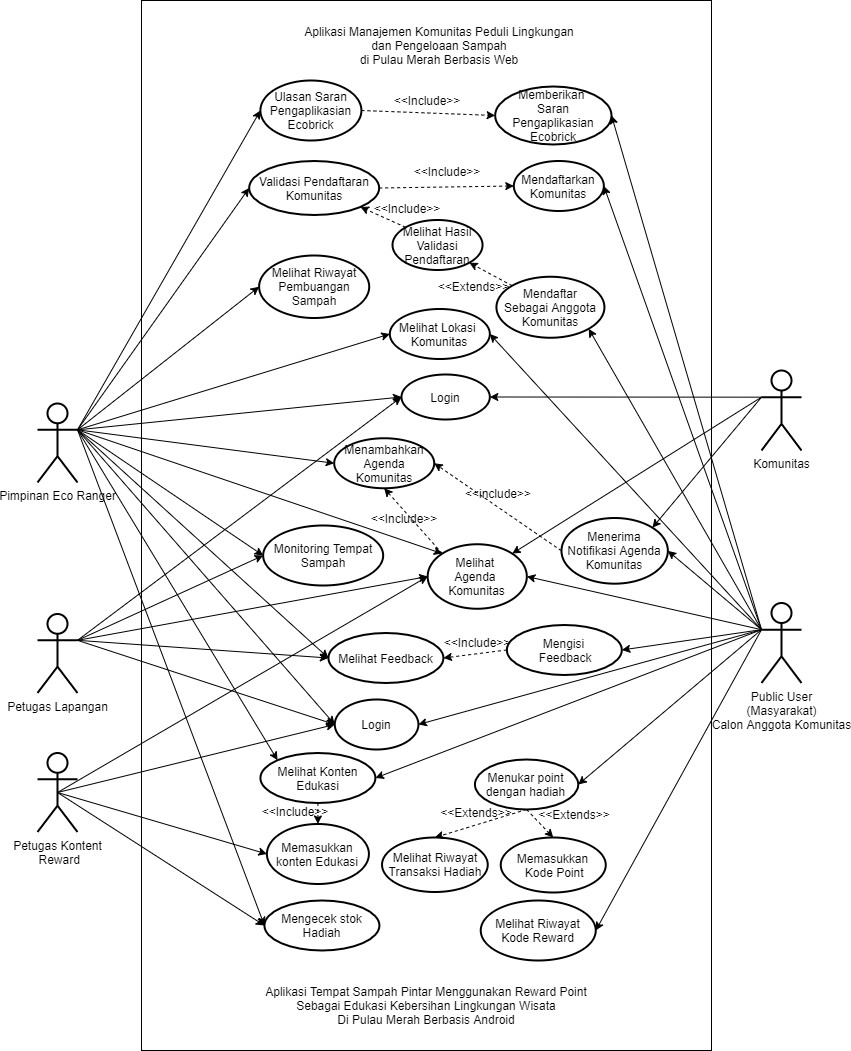
**Gambar 3. 3** Gambaran umum yang diusulkan aplikasi web

Pada Gambar 3.3 dijelaskan bahwa gambaran umum yang diusulkan Aplikasi *Website* yaitu Aplikasi Komunitas Peduli Lingkungan dan Pengelolaan

Sampah di Pulau Merah Berbasis Web yang akan membantu melakukan *monitoring* adanya tempat sampah pintar dan memanajemen komunitas sektor pariwisata pantai yang ada di kabupaten Banyuwangi. Untuk web sendiri akan menjadi web server dari project ini dan menjadi pla*form* khusus untuk manajemen komunitas serta pengelolaan sampah di pulau merah Banyuwangi dengan *EcoRanger* sebagai pilot project dengan hasil pengelolaan sampah yang sudah berjalan saat ini yaitu *Ecobrick*.

# Pemodelan Sistem

* + 1. ***Use case* Diagram**

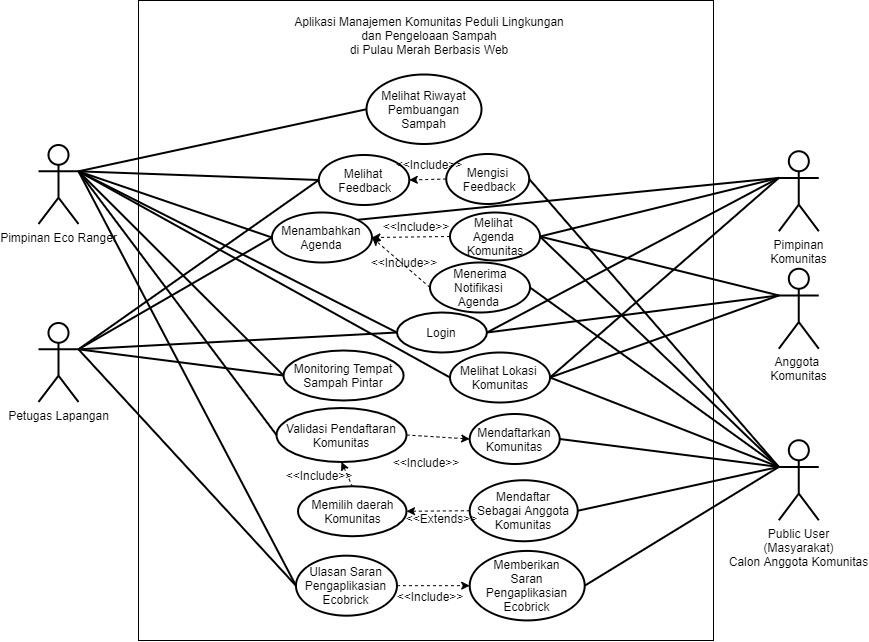


Android

Web

|  |
| --- |
|  |
|  |
|  |

**Gambar 3. 4** *Use case* Gabungan web dan Android



**Gambar 3. 5** *Use case* Web

Pada Gambar 3.4 menunjukkan pemodelan *Use case* gabungan aplikasi Android serta *Website* dari sistem yang diusulkan dan Gambar 3.5 menunjukkan pemodelan *Use case* Aplikasi Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah di Pulau Merah Banyuwangi Berbasis Web.. Terdapat 5 aktor dalam sistem ini untuk lebih detail dijelaskan pada Table 3.2 dan Penjelasan tiap *use case* ditunjukkan pada Tabel 3.3, Tabel 3.4, Tabel 3.5, Tabel 3.6, Tabel 3.7,

Tabel 3.8, Tabel 3.9, Tabel 3.10, Tabel 3.11, Tabel 3.12, Tabel 3.13, Tabel 3.14 dan

Tabel 3.15.

**Tabel 3. 2** Penjelasan *Aktor*

|  |  |  |
| --- | --- | --- |
| No. | Aktor | Deskripsi |
| 1. | Pimpinan  *EcoRanger* | Pimpinan *EcoRanger* merupakan bagian yang dapat melakukan *monitoring* tempat sampah dan komunitas. Bagian ini juga dapat menambahkan agenda dan menambahkan lokasi komunitas. Bagian ini  juga dapat melakukan validasi pendaftaran komunitas. |
| 2. | Petugas Lapangan | Petugas Lapangan bertugas untuk mengelola data dan informasi tekait  Memanajemen, melakukan *monitoring* dan mengelola *feedback* yang ada pada aplikasi terkecuali pada validasi pendaftaran komunitas. |
| 3. | Pimpinan  Komunitas | Pimpinan Komunitas bertugas untuk mengelola agenda komunitas  masing-masing dan semua anggota yang berada pada komunitas tersebut. |

|  |  |  |
| --- | --- | --- |
| 4. | Anggota Komunitas | Bagian Anggota Komunitas merupakan bagian dari anggota komunitas- komunitas peduli lingkungan wisata pantai Banyuwangi yang terbagi  berdasarkan daerahnya. |
| 5. | *Public User* | *Public User* adalah bagian pengguna sistem seperti masyarakat atau stakeholder yang memiliki hak akses terbatas dengan hanya dapat melihat tampilan utama *website*. Dapat melakukan pencarian informasi pemetaan titik lokasi sampah, lokasi komunitas peduli lingkungan wisata pantai Banyuwangi dan agenda-agenda yang akan dilakukan oleh  komunitas. |

**Tabel 3. 3** Keterangan *Use case Login*

|  |  |
| --- | --- |
| *Use case Name* | *Login* |
| *Actor* | Pimpinan *EcoRanger*, Petugas Lapangan, dan Komunitas |
| *Description* | Validasi untuk masuk ke sistem |
| *Normal Course* | 1. Sistem menampilkan *form Login* 2. Memasukkan *Username* dan *Password* yang sudah terdaftar pada sistem |
| *Alternative Course* | *Login* tidak berhasil karena *Username* dan  *password* tidak terdaftar pada *database* sistem |
| *Pre-Condition* | *Login* |
| *Post-Condition* | Masuk halaman *Beranda* |

**Tabel 3. 4** Keterangan *Use case Monitoring* Tempat Sampah Pintar

|  |  |
| --- | --- |
| *Use case Name* | *Monitoring* Tempat Sampah Pintar |
| *Actor* | Pimpinan *EcoRanger*, dan Petugas Lapangan |
| *Description* | Menambahkan titik lokasi dan melakukan  *monitoring* tempat sampah pintar |
| *Normal Course* | 1. Menambahkan titik lokasi tempat sampah pintar 2. Sistem menampilkan titik lokasi tempat sampah pintar pada Maps 3. Sistem memberikan notifikasi jika tempat sampah dalam keadaan penuh |
| *Alternative Course* | *-* |
| *Pre-Condition* | *Login* sebagai Pimpinan *EcoRanger* dan Petugas Lapangan |
| *Post-Condition* | Melakukan *monitoring* tempat sampah pintar |

**Tabel 3. 5** Keterangan *Use case* Menambahkan Agenda

|  |  |
| --- | --- |
| *Use case Name* | Menambahkan Agenda |
| *Actor* | Pimpinan *EcoRanger*, dan Pimpinan Komunitas |

|  |  |  |
| --- | --- | --- |
| *Description* | Menambahkan agenda komunitas sesuai kebutuhan | |
| *Normal Course* | 1. Sistem menampilkan agenda yang akan dilaksanakan 2. Menambahkan Agenda komunitas jika dirasa agenda bisa dilaksanakan | |
| *Alternative Course* | *-* |  |
| *Pre-Condition* | a. *Login* sebagai Pimpinan *EcoRanger* atau Pimpinan Komunitas | |
| *Post-Condition* | Memeriksa dan mengunggah agenda yang akan dilaksanakan | |

**Tabel 3. 6** Melihat Agenda

|  |  |
| --- | --- |
| *Use case Name* | Melihat Agenda |
| *Actor* | Pimpinan *EcoRanger*, Komunitas, dan *Public User* |
| *Description* | Melihat Agenda yang akan dilaksanakan oleh komunitas serta deskripsi kegiatan |
| *Normal Course* | Sistem menampilkan agenda yang akan dilaksanakan oleh komunitas |
| *Alternative Course* | *-* |
| *Pre-Condition* | Membuka *Website* |
| *Post-Condition* | Melihat Agenda |

**Tabel 3. 7** Mendaftar sebagai anggota komunitas

|  |  |
| --- | --- |
| *Use case Name* | Mendaftar sebagai anggota komunitas |
| *Actor* | *Public User* |
| *Description* | Mendaftarkan sebagai anggota komunitas peduli lingkungan wisata pantai Banyuwangi  berdasarkan wilayah, jika daerah belum tersedia maka mendaftarkan daerahnya |
| *Normal Course* | 1. Sistem menampilkan *form* perndaftaran anggota komunitas 2. Memasukkan data serta memilih sebagai anggota komunitas berdasarkan daerah 3. Jika daerah tidak tersedia maka mendaftarkan daerahnya beserta titik koordinat lokasi 4. Simpan data dan berkas yang telah ditambahkan |
| *Alternative Course* | - |
| *Pre-Condition* | *Register* sebagai anggota komunitas |
| *Post-Condition* | Masuk ke halaman *Beranda* komunitas |

**Tabel 3. 8** Mendaftarkan Komunitas

|  |  |
| --- | --- |
| *Use case Name* | Mendaftarkan Komunitas |
| *Actor* | *Public User* |
| *Description* | Mendaftarkan komunitas peduli lingkungan wisata pantai Banyuwangi berdasarkan wilayah |
| *Normal Course* | 1. Sistem menampilkan *form* perndaftaran komunitas 2. Memasukkan data serta titik koordinat longitude dan latitude 3. Simpan data dan berkas yang telah ditambahkan |
| *Alternative Course* | - |
| *Pre-Condition* | *Register* sebagai komunitas |
| *Post-Condition* | Menunggu hasil validasi |

**Tabel 3. 9** Validasi Pendaftaran Komunitas

|  |  |
| --- | --- |
| *Use case Name* | Validasi Pendaftaran Komunitas |
| *Actor* | Pimpinan *EcoRanger* |
| *Description* | Validasi pendaftaran komunitas untuk memastikan bahwa data yang diajukan memang  benar |
| *Normal Course* | 1. Sistem menampilkan data yang diajukan 2. Melakukan validasi |
| *Alternative Course* | *-* |
| *Pre-Condition* | 1. *Login* sebagai Pimpinan *EcoRanger* 2. Persetujuan pendaftaran komunitas |
| *Post-Condition* | Menampilkan hasil validasi pendaftaran komunitas |

**Tabel 3. 10** Melihat Hasil Validasi

|  |  |
| --- | --- |
| *Use case Name* | Melihat Hasil Validasi |
| *Actor* | Komunitas |
| *Description* | Anggota komunitas dapat melihat hasil validasi |
| *Normal Course* | Sistem menampilkan nama komunitas beserta daerah hasil validasi |
| *Alternative Course* | *-* |
| *Pre-Condition* | 1. *Login* sebagai Komunitas 2. *Actor* Pimpinan *EcoRanger* sudah memvalidasi komunitas |
| *Post-Condition* | Melihat hasil validasi |

**Tabel 3. 11** Melihat Lokasi Komunitas

|  |  |
| --- | --- |
| *Use case Name* | Melihat Lokasi Komunitas |
| *Actor* | Pimpinan*EcoRanger* dan *Public User* |

|  |  |
| --- | --- |
| *Description* | Melihat lokasi komunitas peduli lingkungan wisata pantai Banyuwangi |
| *Normal Course* | 1. Masuk dalam web 2. Sistem menampilkan titik lokasi komunitas pada Maps |
| *Alternative Course* | *-* |
| *Pre-Condition* | Membuka *website* |
| *Post-Condition* | Melihat lokasi komunitas |

**Tabel 3. 12** Saran Pengaplikasian *Ecobrick*

|  |  |
| --- | --- |
| *Use case Name* | Saran Pengaplikasian *Ecobrick* |
| *Actor* | *Public User* |
| *Description* | Mengajukan saran untuk mengaplikasikan  *Ecobrick* tersebut akan dibuat seperti apa |
| *Normal Course* | 1. Sistem menampilkan *form* saran pengaplikasian *Ecobrick* 2. Mengisi data dan berkas yang diajukan 3. Mengirim data dan berkas yang sudah terisi |
| *Alternative Course* | *-* |
| *Pre-Condition* | Mempersiapkan saran |
| *Post-Condition* | Saran terkirim |

**Tabel 3. 13** Ulasan Saran *Ecobrick*

|  |  |
| --- | --- |
| *Use case Name* | Ulasan Saran *Ecobrick* |
| *Actor* | Pimpinan *EcoRanger* dan Petugas Lapangan |
| *Description* | Ulasan hasil saran |
| *Normal Course* | 1. Melihat hasil saran 2. Saran yang baik akan ditampilkan dalam sistem dan akan di aplikasikan 3. Sistem menampilkan hasil saran |
| *Alternative Course* | *-* |
| *Pre-Condition* | *Login* sebagai Pimpinan *EcoRanger* dan Petugas Lapangan |
| *Post-Condition* | Saran sudah ditampilkan |

**Tabel 3. 14** Mengirim notifikasi agenda yang mendesak

|  |  |
| --- | --- |
| *Use case Name* | Mengirim notifikasi agenda yang mendesak |
| *Actor* | Pimpinan *EcoRanger* dan Petugas Lapangan |
| *Description* | Mengirim notifikasi |
| *Normal Course* | 1. Melihat kondisi pantai 2. Menambahkan agenda yang mendesak untuk langsung dibersihkan 3. Sistem mengirimkan notifikasi ke *public user* dan komunitas |
| *Alternative Course* | *-* |

|  |  |
| --- | --- |
| *Pre-Condition* | *Login* sebagai Pimpinan *EcoRanger* atau Petugas Lapangan |
| *Post-Condition* | Mengirimkan notifikasi |

**Tabel 3. 15** Menerima notifikasi agenda yang mendesak

|  |  |
| --- | --- |
| *Use case Name* | Menerima notifikasi agenda yang mendesak |
| *Actor* | Komunitas dan *Public User* |
| *Description* | menerima notifikasi |
| *Normal Course* | 1. Masuk ke halaman *website* 2. Sistem mengirimkan notifikasi ke *public user* dan komunitas 3. *Public User* menerima notifikasi bahwa ada agenda yang mendesak 4. Sistem menampilkan notifikasi |
| *Alternative Course* | *-* |
| *Pre-Condition* | Membuka *website* |
| *Post-Condition* | Menerima notifikasi |

* + 1. ***Activity* Diagram**

Dalam sistem ini terdapat 4 *Activity* diagram, diantaranya *Activity* diagram *Login, Monitoring* tempat sampah pintar dan komunitas, pengusulan agenda dan penambahan agenda, Saran pengaplikasian *Ecobrick*.



**Gambar 3. 6** *Activity* Diagram *Login*

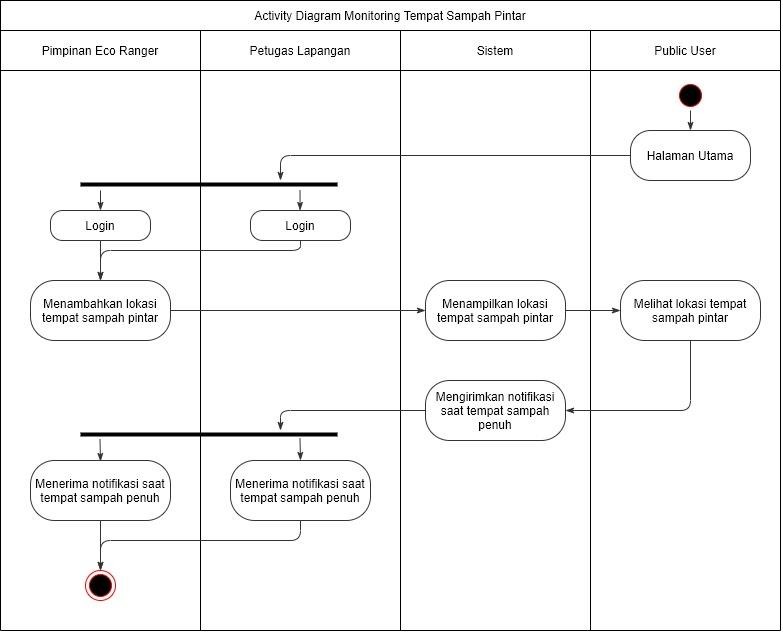
Untuk *Activity Login* ditunjukan pada Gambar 3.6 , berikut penjelasanya :

* + - 1. Memasukkan *user*name dan password.
      2. Saat melakukan *Login*, jika proses berhasil maka akan tampil pesan

*Login* berhasil dan masuk pada halaman aplikasi sesuai hak akses.

* + - 1. Apabila salah memasukkan *user*name dan password maka sistem akan menampilkan pesan *user*name dan password salah dan akan kembali pada halaman memasukkan *user*name dan password.
      2. Proses selesai.

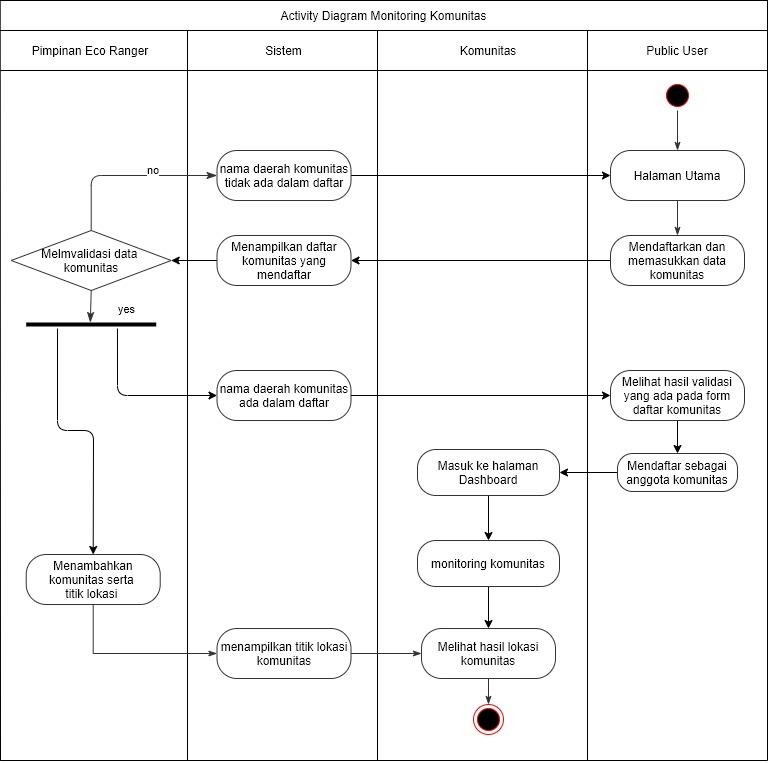
Kemudian *Activity* diagram *monitoring* tempat sampah pintar dan komunitas ditunjukkan pada Gambar 3.7 dan Gambar 3.8.



**Gambar 3. 7** *Activity* Diagram *Monitoring* Tempat Sampah Pintar Berikut penjelasan *Actvity* Diagram *monitoring* tempat sampah pintar dari

Gambar 3.7 :

1. Aktor *Public User* masuk ke halaman utama *website* untuk dapat melihat lokasi tempat sampah pintar.
2. Untuk menambahkan lokasi tempat sampah pintar harus *Login* terlebih dahulu dan yang hanya memiliki hak akses tersebut adalah Pimpinan *EcoRanger* dan Petugas Lapangan.
3. Setelah masuk kedalam akses Pimpinan *EcoRanger* dan Petugas Lapangan bisa menambahkan tempat sampah pintar.
4. Sistem akan menampilkan lokasi tempat sampah pintar.
5. *Public User* bisa melihat titik lokasi sampah pintar.
6. Sistem akan mengirimkan notifikasi saat tempat sampah
7. Aktor Pimpinan *EcoRanger* dan Petugas Lapangan menerima notifikasi yang dikirim oleh sistem saat tempat sampah penuh.



**Gambar 3. 8** *Activity* Diagram *Monitoring Komunitas*

Kemudian penjelasan dari *Activity* Diagram *Monitoring* Komunitas yang ditunjukan pada Gambar 3.8. Berikut penjelasanya :

1. *Public user* masuk dalam halaman utama *website*.
2. Mendaftarkan komunitas jika tidak ada daerah yang diinginkan untuk menjadi anggota komunitas tersebut.
3. Sistem akan menampilkan daftar komunitas yang mendaftar pada Pimpinan

*EcoRanger*.

1. Pimpinan *EcoRanger* memvalidasi data.
2. Jika tidak tervalidasi maka nama daerah komunitas tidak ditampilkan dalam sistem.
3. Jika tervalidasi maka nama daerah komunitas tersebut akan tersedia dalam

*form* daftar anggota komunitas.

1. Dan juga menambahkan komunitas dalam sistem serta titik lokasi agar terlihat oleh semua *user*.
2. Sistem menampilkan titik lokasi yang dapat dilihat oleh semua *user*.
3. *Public user* mendaftar sebagai anggota komunitas daerah yang dipilih.
4. Masuk ke halaman *Beranda* komunitas.
5. *Monitoring* komunitas.
6. Lokasi komunitas akan ditampilkan dan dapat dilihat.

**Gambar 3. 9** *Activity* Diagram Agenda

Berikut penjelasan *Activity* Diagram Agenda dari Gambar 3.9 :

1. *Public User* masuk pada halaman utama *website*.
2. Untuk melihat agenda dari komunitas harus *Login* terlebih dahulu.
3. *Login* pada *User* Anggota Komunitas untuk melihat agenda yang akan dilaksanakan.
4. Pimpinan *EcoRanger* dan Pimpinan Komunitas bisa menambahkan agenda dan melihat Laporan agenda yang .
5. Sistem Menampilkan agenda yang selanjutnya bisa dilihat di halaman utama

*website*.

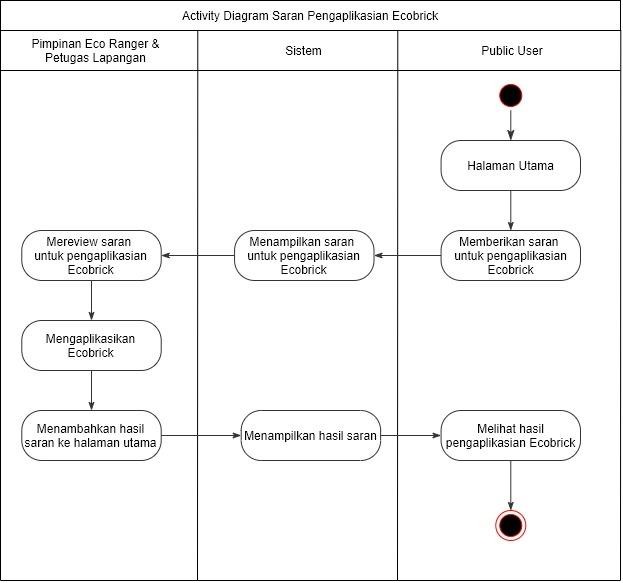
Selanjutnya penjelasan *Activity* Diagram Saran Pengaplikasian *Ecobrick*

dari Gambar 3.10 :

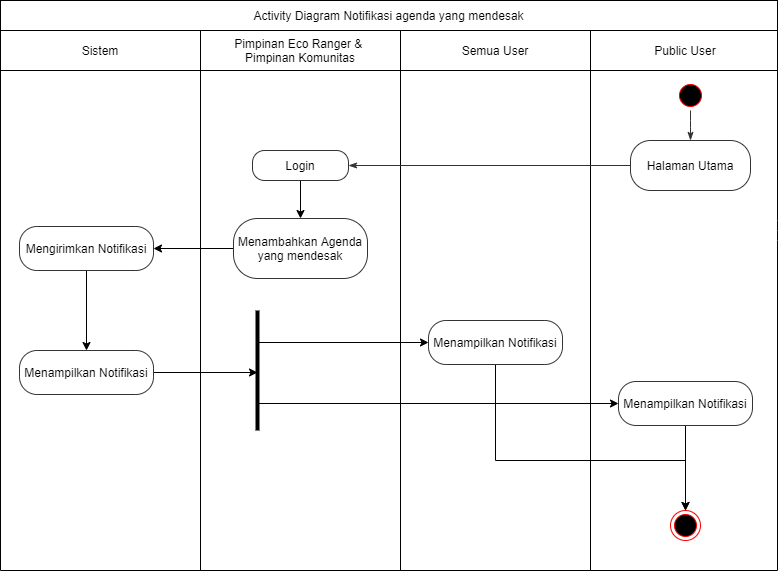
1. *Public User* masuk pada halaman utama *website*.
2. Terdapat *form* untuk memberikan saran pengaplikasian *Ecobrick*.
3. Sistem menerima dan menampilkan saran untuk pengaplikasian *Ecobrick*

ke halaman Ulasan saran.

1. Pimpinan *EcoRanger* dan Petugas Lapangan meUlasan saran untuk mengaplikasikan *Ecobrick*.
2. *Ecobrick* di aplikasikan jika sudah dianggap memang dibutuhkan.
3. Menambahkan hasil saran ke halaman utama beserta foto hasil pengaplikasian *Ecobrick* yang telah direalisasikan.
4. Sistem menampilkan hasil ke halaman utama dan bisa dilihat di halaman utama *website*.



**Gambar 3. 10** *Activity* Diagram Saran Pengaplikasian *Ecobrick*



**Gambar 3. 11** *Activity* Diagram Notifikasi agenda yang mendesak Berikut penjelasan *Activity* Diagram Notifikasi agenda yang mendesak dari

Gambar 3.11 :

* 1. *Public User* membuka aplikasi dan masuk pada halaman utama
  2. Pimpinan *EcoRanger* atau petugas lapangan *Login* terlebih dahulu untuk menambahkan agenda yang mendesak
  3. Sistem mengirimkan notifikasi
  4. Sistem menampilkan notifikasi ke aktor komunitas dan *public user*
  5. Semua Akrot menerima notifikasi
  6. Melihat notifikasi

# Spesifikasi Sistem

# Aturan Sistem

Adapun aturan sistem untuk pengguna aplikasi yang dijelaskan pada Tabel 3.16

**Tabel 3. 16** Aturan Sistem

|  |  |  |
| --- | --- | --- |
| NO | KETERANGAN | PENGGUNA |
| 1 | Data pengelolaan sampah dan komunitas hanya dapat dikelola penuh baik dalam menambah, mengubah, menghapus, melihat data dan memvalidasi data  pendaftaran komunitas. | Pimpinan *EcoRanger* |

|  |  |  |
| --- | --- | --- |
| 2 | Data pengelolaan sampah dan komunitas dapat dikelola penuh baik dalam menambah, mengubah, menghapus, dan melihat data kecuali memvalidasi data pendaftaran  komunitas. | Petugas Lapangan |
| 3 | Mengelola titik lokasi tempat sampah pintar dan komunitas baik menambah, mengubah, menghapus dan melihat data. Menerima notifikasi tempat sampah pintar dalam kondisi  penuh. | Pimpinan *EcoRanger* |
| 4 | Mengelola titik lokasi tempat sampah pintar dan komunitas baik menambah, mengubah, menghapus dan melihat data. Menerima notifikasi tempat sampah pintar dalam kondisi  penuh. | Petugas Lapangan |
| 5 | Mendaftarkan dan melihat hasil validasi komunitas peduli lingkungan wisata pantai Banyuwangi, Menambahan  Agenda yang akan dilaksanakan oleh komunitas | Pimpinan Komunitas |
| 6 | Melihat Agenda yang akan dilaksanakan oleh komunitas. | Anggota Komunitas |
| 7 | Melihat agenda dan lokasi tempat sampah pintar serta  lokasi komunitas. | *Public User* |
| 9 | Mengirimkan saran pengaplikasian *Ecobrick*. | *Public User* |
| 10 | Ulasan Saran pengaplikasian *Ecobrick*. | Pimpinan *EcoRanger* |
| 11 | Ulasan Saran pengaplikasian *Ecobrick*. | Petugas Lapangan |
| 12 | Menerima notifikasi saat ada agenda yang mendesak | *Public User* |
| 13 | Menerima notifikasi saat ada agenda yang mendesak | Semua Aktor |

# Desain Aplikasi

# Struktur Menu

Struktur menu pada Gambar 3.12 dimana pengguna yang harus *Login* terlebih dahulu yaitu Pimpinan *EcoRanger*, Petugas Lapangan, Pimpinan Komunitas dan Anggota Komunitas sedangkan *Public User* tidak memerlukan *Login* terlebih dahulu. *User* yang telah *Login* akan masuk ke dalam beranda masing- masing sesuai hak aksesnya. Ada 5 halaman pengguna yang berbeda dalam mengakses data yang ada yaitu ada halaman untuk Pimpinan *EcoRanger*, Petugas Lapangan, Pimpinan Komunitas, Anggota Komunitas dan *Public User*. Pimpinan *EcoRanger* merupakan bagian yang dapat mengakses dan mengola data tanpa ada batasan bisa dikatakan sebagai hak akses tertinggi dalam sistem. Petugas lapangan

sebagai aktor yang mengatur penuh untuk melakukan *monitoring* tempat sampah pintar mulai dari menerima notifikasi saat tempat sampah pintar penuh maupun menambahkan titik lokasi tempat sampah. Komunitas sebagai aktor yang mengatur segala sesuatu yang ada dalam komunitas kecuali pada validasi data komunitas yang hanya bisa dilakukan oleh Pimpinan *EcoRanger*. *Public user* sebagai aktor yang tidak memiliki hak akses yang mana dalam hal ini adalah masyarakat yang bisa mendaftarkan diri sebagai anggota komunitas dan mendaftarkan daerah komunitas.



**Gambar 3. 12** Struktur Menu

# Rancangan Tampilan Aplikasi

Rancangan tampilan aplikasi merupakan rancangan halaman web yang akan dibuat dan nantinya terdapat pada Aplikasi Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah di Pulau Merah Berbasis Web. Rancangan tampilan aplikasi dibuat dengan sederhana. Berikut rancangan tampilan beserta penjelasannya.

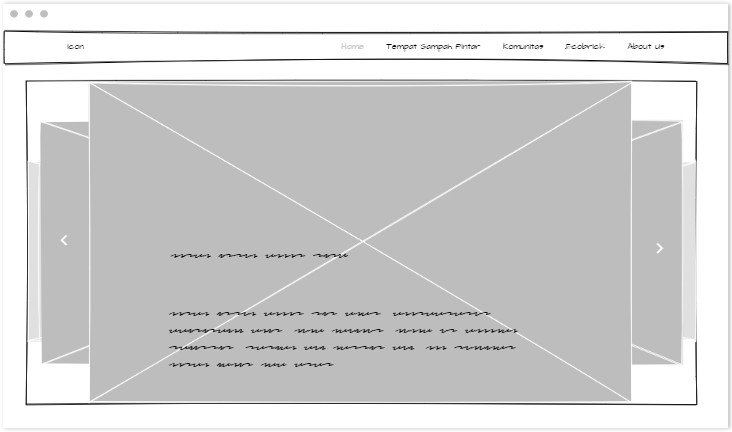
1. Halaman Utama

Pada Gambar 3.13 merupakan Halaman Utama dari *Website* yang bisa diakses oleh semua *user* meskipun yang tidak memiliki hak akses. Halaman ini menampilkan beberapa gambar dan sebagai intro dari *website* juga dilengkapi dengan kata-kata yang membangun.



**Gambar 3. 13** Rancangan Tampilan Halaman Utama

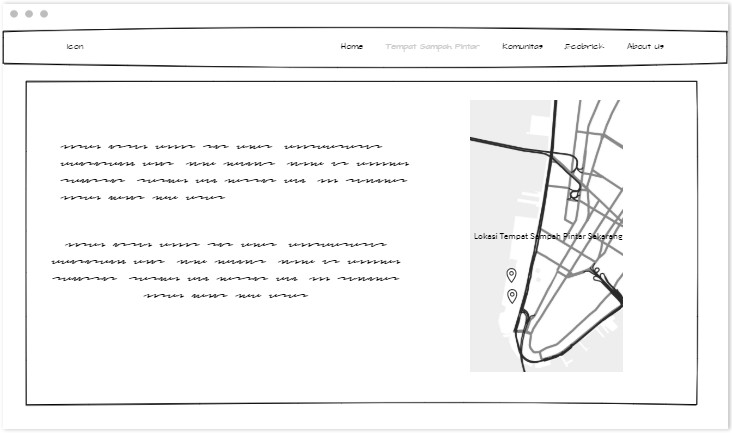
1. Halaman Utama Untuk Notifikasi



**Gambar 3. 14** Rancangan Tampilan Halaman Utama Untuk Notifikasi

Pada Gambar 3.14 memperlihatkan notifikasi untuk agenda yang mendesak berupa tulisan untuk mengajak dan disertai gambar.

1. Halaman Utama dari Tempat Sampah Pintar



**Gambar 3. 15** Rancangan Tampilan Halaman Utama dari Tempat Sampah

Pada Gambar 3.15 memperlihatkan peta lokasi yang mempunyai tanda dari lokasi Tempat Sampah Pintar serta penjelasan singkat dari Tempat sampah tersebut.

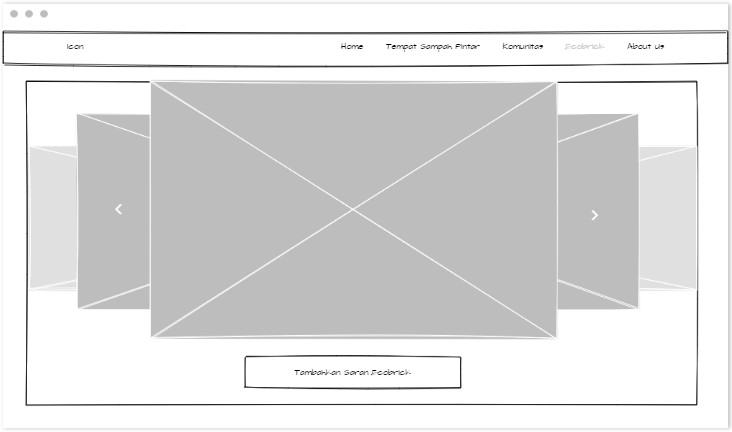
1. Halaman Utama dari Komunitas



**Gambar 3. 16** Rancangan Tampilan Halaman Utama dari Komunitas

Pada Gambar 3.16 terlihat rancangan tampilan halaman utama dari bagian komunitas yang menampilkan peta serta titik lokasi komunitas peduli lingkungan daerah wisata pantai Pulau Merah Banyuwangi dan memperlihatkan Kalender untuk memberitahukan pada waktu kapan saja agenda yang akan dilaksanakan oleh komunitas.

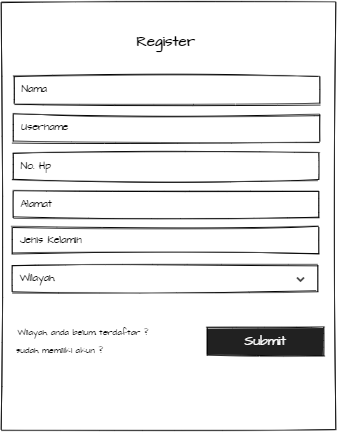
1. Halaman Utama dari *Ecobrick*



**Gambar 3. 17** Rancangan Tampilan Halaman Utama dari *Ecobrick*

Pada Gambar 3.17 Halaman Utama dari bagian *Ecobrick* menampilkan gambar bergerak yang berisi konten dari *Ecobrick* yang telah terealisasikan ataupun juga yang akan direalisasikan. Juga ada tombol untuk menambahkan saran pengaplikasian dari *Ecobrick*. Dimana *form* tersebut ditampilkan bentuk *pop-up* yang dapat diisi oleh *Public User*.

1. Halaman *Register*

Pada Gambar 3.18 menampilkan *form Register* dimana *form Register* ini dapat digunakan untuk mendaftar sebagai anggota komunitas dari wilayah komunitas yang sudah terdaftar dan tervalidasi oleh sistem. Jika wilayah yang dipilih tidak tersedia maka dapat mendaftarkan komunitas sesuai dengan wilayahnya terlebih dahulu yang terdapat pada Gambar 3.20.

**Gambar 3. 18** Rancangan Tampilan *Register*

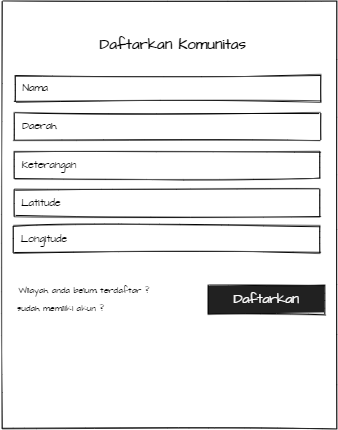
1. Halaman *Login*



**Gambar 3. 19** Rancangan Tampilan *Login*

Pada Gambar 3.19 Rancangan Tampilan *Login* sebagai menu *Login* yang berlaku bagi semua *user* sesuai dengan *username* dan *passwordnya.*

1. Halaman Mendaftarkan Komunitas



**Gambar 3. 20** Rancangan Tampilan Daftakan Komunitas

Pada Gambar 3.20 terlihat rancangan tampilan daftarkan komunitas yang berlanjut dari *form Register* Gambar 3.18 untuk mendaftarkan komunitas yang belum terdaftar pada sistem.

1. Halaman *Beranda* Admin Pimpinan *EcoRanger*

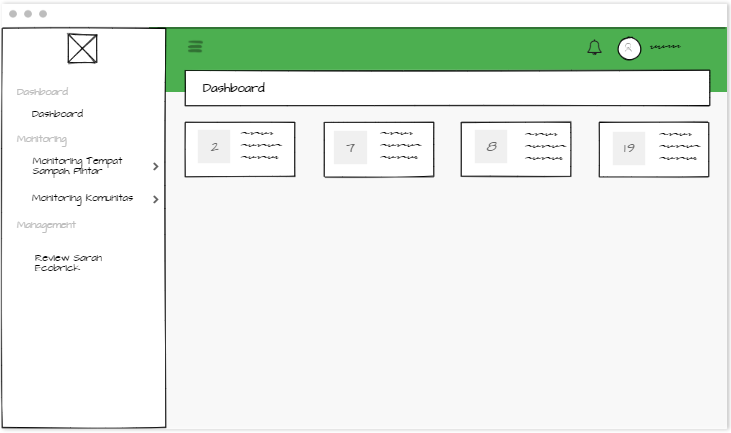


**Gambar 3. 21** Rancangan Tampilan *Beranda* Pimpinan *EcoRanger*

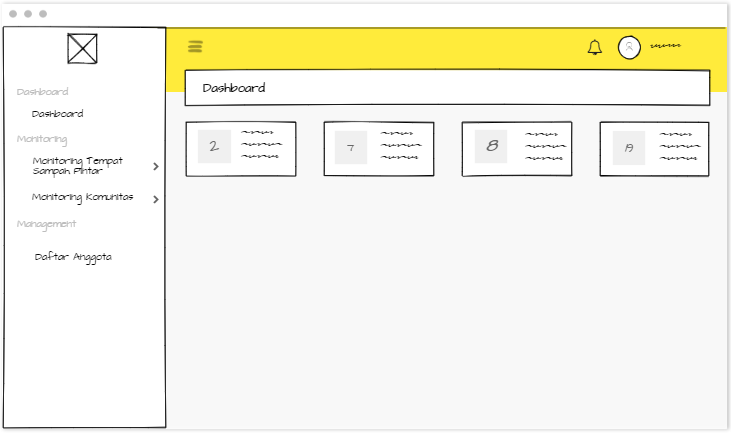
Pada Gambar 3.21 menampilkan *Beranda* dari Pimpinan *EcoRanger* menampilkan beberapa data yang dibutuhkan seperti data berapa banyak komunitas yang terdaftar dalam sistem, tempat sampah yang sudah ada dalam map serta data

orang yang membuang sampah di tempat sampah pintar yang dilengkapi dengan grafik agar lebih mudah dalam melakukan Analisa perilaku masyarakat yang membuang sampah .

1. Halaman *Beranda* Admin Petugas Lapangan



**Gambar 3. 22** Rancangan Tampilan *Beranda* Petugas Lapangan

1. Halaman *Beranda* Admin Komunitas

**Gambar 3. 23** Rancangan Tampilan Halaman *Beranda* Komunitas Gambar 3.23 dan Gambar 24 merupakan rancangan dari halaman *Beranda*

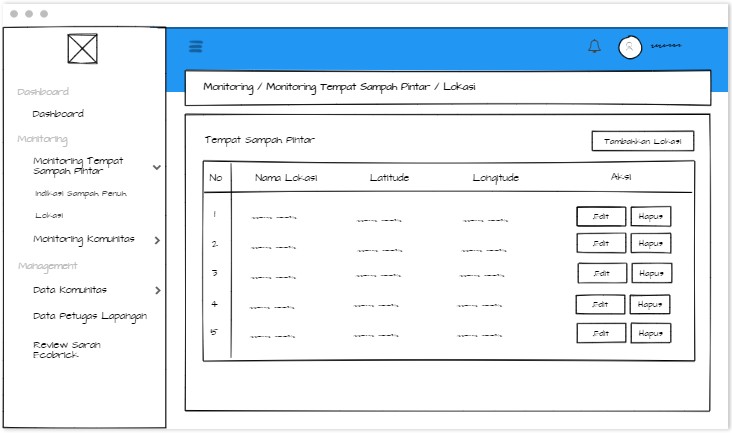
komunitas dengan warna kuning dan halaman *Beranda* petugas lapangan dengan

warna hijau yang dapat diakses setelah *Login* dengan hak akses masing-masing.

1. Halaman *Monitoring* Tempat Sampah Pintar



**Gambar 3. 24** Rancangan Tampilan Halaman Indikasi Sampah Penuh



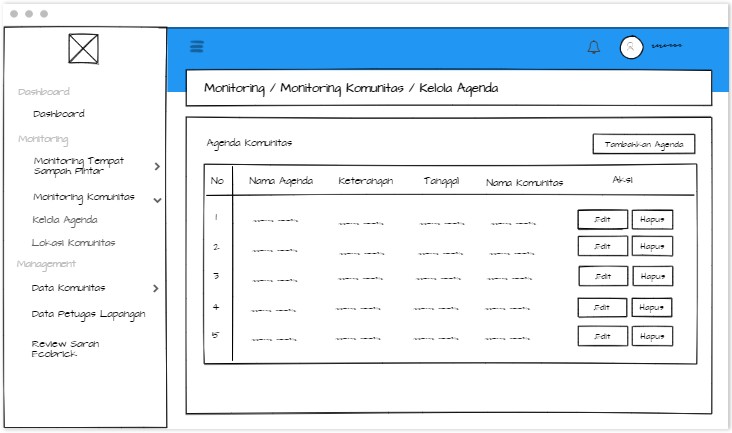
**Gambar 3. 25** Rancangan Tampilan Halaman Lokasi Tempat Sampah Pintar Pada Gambar 3.24 merupakan Halaman Indikasi Sampah Penuh yang berisi

peta lokasi yang menampilkan titik lokasi dari tempat sampah pintar dan sebagai

pemberi notifikasi saat tempat sampah pintar tersebut penuh. Terlihat bahwa ada warna merah, kuning dan hijau pada gambar yang menandakan jika warna merah tersebut terdapat pada halaman ini maka dapat dipastikan bahwa ada tempat sampah yang penuh disalah satu tempat sampah. Warna hijau untuk menandakan tempat

sampah pintar tersebut masih belum penuh serta alat tersebut masih bekerja dengan baik dan warna kuning menandakan bahwa tempat sampah tersebut dalam fase perbaikan atau dalam proses pengosongan isi tempat sampah pintar. Pada Gambar

3.25 merupakan halaman untuk menambahkan, mengedit ataupun menghapus lokasi tempat sampah pintar yang dapat diganti sewaktu-waktu.

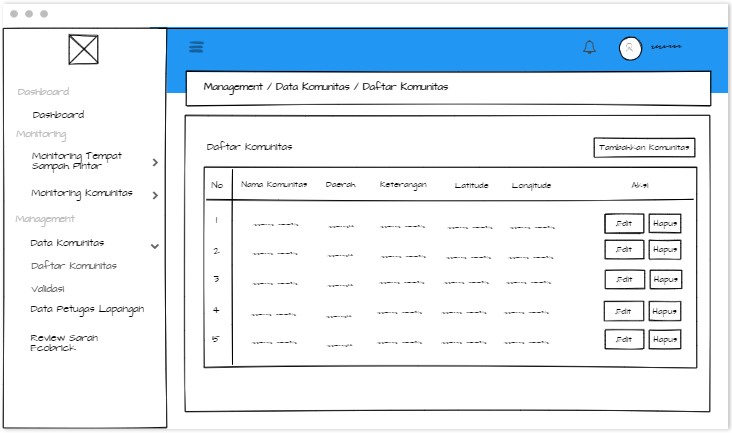
1. Halaman *Monitoring* Komunitas

**Gambar 3. 26** Rancangan Tampilan Halaman Kelola Agenda

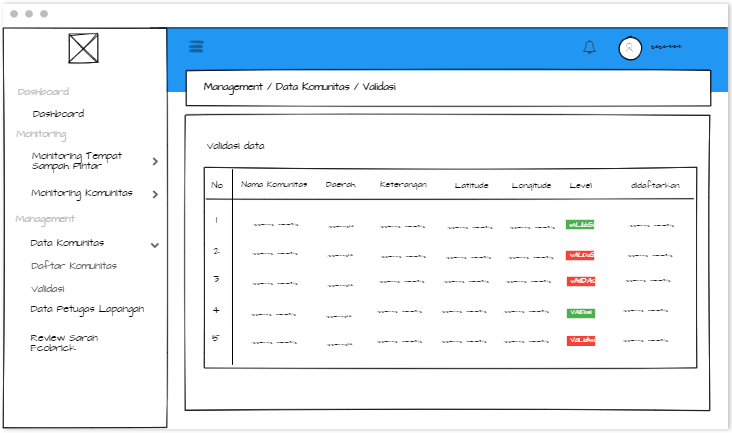


**Gambar 3. 27** Rancangan Tampilan Halaman Lokasi Komunitas

1. Halaman Data Komunitas



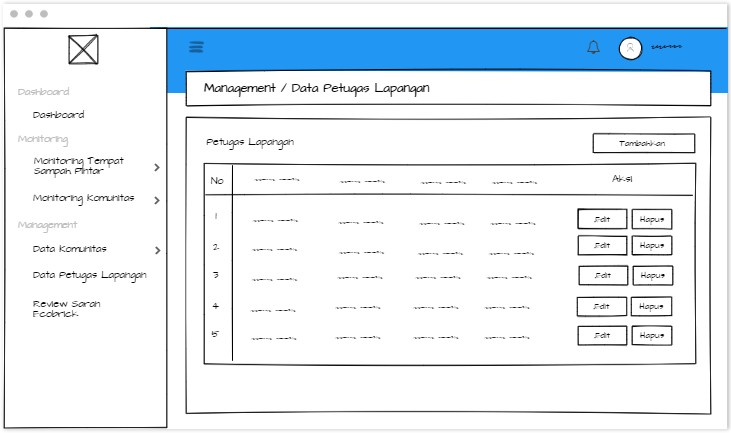
**Gambar 3. 28** Rancangan Tampilan Halaman Daftar Komunitas



**Gambar 3. 29** Rancangan Tampilan Halaman Validasi

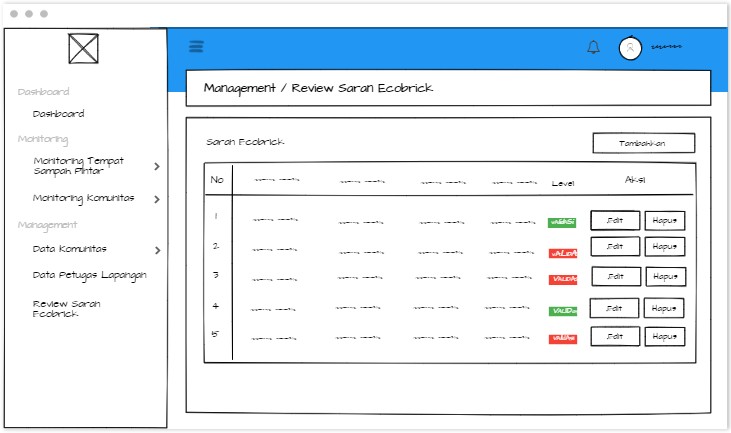
Pada Gambar 3.29 merupakan Tampilan Halaman Validasi yang hanya ada pada *user* tim *EcoRanger* dimana dapat memvalidasi komunitas yang mendaftar pada sistem. Untuk warna merah menandakan bahwa komunitas tersebut belum tervalidasi sedangkan warna hijau sudah tervalidasi dan akan tertera dalam map serta sudah tampil pada field wilayah saat mendaftar sebagai anggota komunitas dan dapat memilih sesuai wilayah yang sudah terdaftar.

1. Halaman Data Petugas Lapangan



**Gambar 3. 30** Rancangan Tampilan Halaman Data Petugas Lapangan

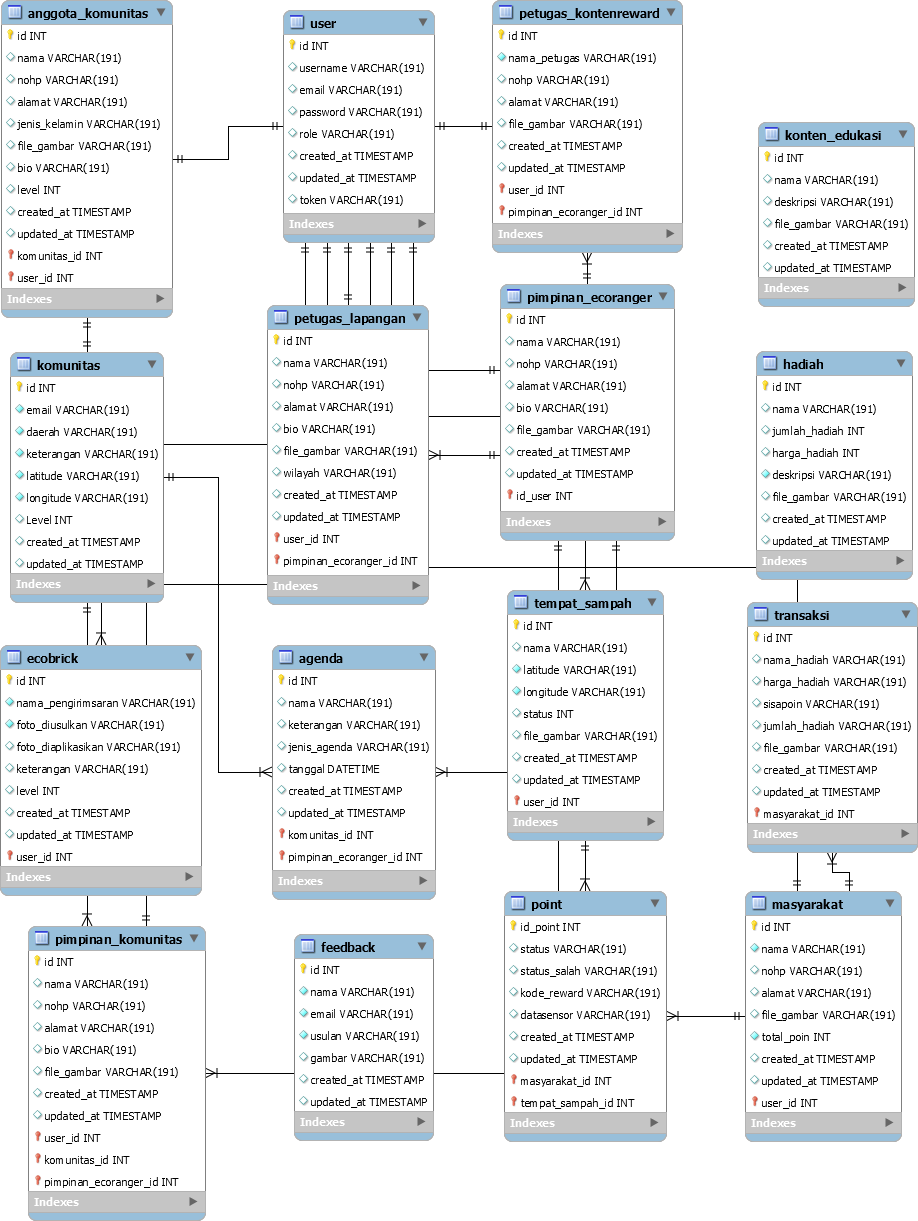
1. Halaman Ulasan Saran *Ecobrick*



**Gambar 3. 31** Rancangan Tampilan Halaman Ulasan Saran *Ecobrick*

# Entity Relationship Diagram

Pada Gambar 3.32 Relasi Antar *Database* yang akan dibuat terdapat beberapa tabel yang digunakan untuk menyimpan data yang berhubungan dengan sistem. Dengan ada garis putus-putus berwarna biru dan merah menjelaskan bahwa tabel untuk *website* dan tabel untuk android terpisah tetapi ada 4 tabel yang digunakan sama. *Website* menggunakan tabel yang berada di dalam garis warna biru sedangkan android berada di dalam garis warna merah.



Android

Website

**Gambar 3. 32** Relasi Antar *Database*

Tabel yang digunakan antara lain tabel *user* yang digunakan saat *Login* dimana semua *user* yang menggunakan *user*name dan password digunakan dalam *website* terdapat pada tabel *user*. Dari total 16 tabel dari server ada 12 tabel yang digunakan oleh sistem *website* yaitu tabel pimpinan\_*EcoRanger*, tabel *user*, tabel agenda, tabel petugas\_lapangan, tabel tempat\_sampah, tabel point, tabel

petugas\_konten*reward*, tabel petugas\_konten*reward*, tabel anggota\_komunitas, tabel *Ecobrick*, tabel pimpinan\_komunitas dan tabel *feedback*. Dimana tabel pimpinan\_*EcoRanger* akan menjadi hak akses tertinggi dalam sistem, tabel petugas\_lapangan yang berelasi dengan tabel *user* serta tabel tempatsampah yang akan berpengaruh pada saat notifikasi tempat sampah pintar penuh serta lokasi. Pada tahap ini akan membuat sebuah pemodelan dengan menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antar tabel tersebut. Berikut pada Gambar 3.32 merupakan rancangan relasi *database* yang akan dibuat. Penjelasan setiap tabel dapat dilihat pada Tabel 3.17, Tabel 3.18, Tabel 3.19, Tabel 3.20, Tabel 3.21, Tabel 3.22, Tabel 3.23, Tabel 3.24, Tabel 3.25, Tabel 3.26, Tabel 3.27 dan

Tabel 3.28.

1. Tabel *user*

**Tabel 3. 17** Tabel *user*

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Atribut** | **Tipe Data** | **Keterangan** |
| 1. | id | INT | *Primary Key , Not Null , Auto Increment* |
| 2. | *user*name | Varchar(191) | *Not Null* |
| 3. | email | Varchar(191) | *Not Null* |
| 4. | password | Varchar(191) | *Not Null* |
| 5. | *role* | Varchar(191) | *Not Null* |
| 6. | created\_at | TIMESTAMP |  |
| 7. | updated\_at | TIMESTAMP |  |
| 8. | token | Varchar(191) |  |

1. Tabel tim\_*EcoRanger*

**Tabel 3. 18** Tabel pimpinan\_*EcoRanger*

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Atribut** | **Tipe Data** | **Keterangan** |
| 1. | id | INT | *Primary Key , Not Null , Auto Increment* |
| 2. | nama | Varchar(191) | *Not Null* |
| 3. | nohp | Varchar(191) | *Not Null* |
| 4. | alamat | Varchar(191) | *Not Null* |
| 5. | bio | Varchar(191) |  |
| 6. | file\_gambar | Varchar(191) |  |
| 7. | created\_at | TIMESTAMP |  |
| 8. | updated\_at | TIMESTAMP |  |
| 9. | *user*\_id | INT | *Foreign key* dari Tabel *user* |

1. Tabel petugas\_lapangan

**Tabel 3. 19** Tabel petugas\_lapangan

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Atribut** | **Tipe Data** | **Keterangan** |
| 1. | id | INT | *Primary Key , Not Null , Auto Increment* |
| 2. | nama | Varchar(191) | *Not Null* |
| 3. | nohp | Varchar(191) | *Not Null* |
| 4. | alamat | Varchar(191) | *Not Null* |
| 5. | bio | Varchar(191) |  |
| 6. | file\_gambar | Varchar(191) |  |
| 7. | wilayah | Varchar(191) |  |
| 8. | created\_at | TIMESTAMP |  |
| 9. | updated\_at | TIMESTAMP |  |
| 10. | *user*\_id | INT | *Foreign key* dari Tabel *user* |
| 11. | pimpinan\_*EcoRanger*\_id | INT | *Foreign key* dari Tabel  pimpinan\_*EcoRanger* |

1. Tabel pimpinan\_komunitas

**Tabel 3. 20** Tabel pimpinan\_komunitas

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Atribut** | **Tipe Data** | **Keterangan** |
| 1. | id | INT | *Primary Key , Not Null , Auto Increment* |
| 2. | nama | Varchar(191) | *Not Null* |
| 3. | nohp | Varchar(191) | *Not Null* |
| 4. | alamat | Varchar(191) | *Not Null* |
| 5. | bio | Varchar(191) | *Not Null* |
| 6. | file\_gambar | Varchar(191) | *Not Null* |
| 7. | created\_at | TIMESTAMP |  |
| 8. | updated\_at | TIMESTAMP |  |
| 9. | *user*\_id | INT | *Foreign key* dari Tabel *user* |
| 10. | komunitas\_id | INT | *Foreign key* dari Tabel komunitas |
| 11. | pimpinan\_*EcoRanger*\_id | INT | *Foreign key* dari Tabel  pimpinan\_komunitas |

1. Tabel anggota\_komunitas

**Tabel 3. 21** Tabel anggota\_komunitas

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Atribut** | **Tipe Data** | **Keterangan** |
| 1. | id | INT | *Primary Key , Not Null , Auto Increment* |
| 2. | nama | Varchar(191) | *Not Null* |
| 3. | nohp | Varchar(191) | *Not Null* |

|  |  |  |  |
| --- | --- | --- | --- |
| 4. | alamat | Varchar(191) |  |
| 5. | jenis\_kelamin | Varchar(191) |  |
| 6. | file\_gambar | Varchar(191) |  |
| 7. | bio | Varchar(191) |  |
| 7. | created\_at | TIMESTAMP |  |
| 8. | updated\_at | TIMESTAMP |  |
| 9. | *user*\_id | INT | *Foreign key* dari Tabel *user* |
| 10. | komunitas\_id | INT | *Foreign key* dari Tabel komunitas |

1. Tabel komunitas

**Tabel 3. 22** Tabel komunitas

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Atribut** | **Tipe Data** | **Keterangan** |
| 1. | id | INT | *Primary Key , Not Null , Auto Increment* |
| 2. | email | Varchar(191) | *Not Null* |
| 3. | daerah | Varchar(191) | *Not Null* |
| 4. | keterangan | Varchar(191) |  |
| 5. | latitude | Varchar(191) |  |
| 6. | longitude | Varchar(191) |  |
| 7. | level | INT |  |
| 7. | created\_at | TIMESTAMP |  |
| 8. | updated\_at | TIMESTAMP |  |

1. Tabel agenda

**Tabel 3. 23** Tabel agenda

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Atribut** | **Tipe Data** | **Keterangan** |
| 1. | id | INT | *Primary Key , Not Null , Auto Increment* |
| 2. | nama | Varchar(191) | *Not Null* |
| 3. | keterangan | Varchar(191) | *Not Null* |
| 4. | jenis agenda | Varchar(191) | *Not Null* |
| 5. | tanggal | DATETIME | *Not Null* |
| 6. | created\_at | TIMESTAMP |  |
| 7. | updated\_at | TIMESTAMP |  |
| 8. | komunitas\_id | INT | *Foreign key* dari tabel komunitas |
| 9. | pimpinan\_*EcoRanger*\_id | INT | *Foreign key* dari tabel  pimpinan\_*EcoRanger* |

1. Tabel tempat\_sampah

**Tabel 3. 24** Tabel tempat\_sampah

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Atribut** | **Tipe Data** | **Keterangan** |
| 1. | id | INT | *Primary Key , Not Null , Auto Increment* |
| 2. | nama | Varchar(191) | *Not Null* |
| 3. | latitude | Varchar(191) | *Not Null* |
| 4. | longitude | Varchar(191) | *Not Null* |
| 5. | status | Varchar(191) |  |
| 6. | file\_gambar | Varchar(191) |  |
| 6. | created\_at | TIMESTAMP |  |
| 7. | update\_at | TIMESTAMP |  |
| 8. | *user*\_id | INT | *Foreign key dari* Tabel *user* |

1. Tabel *Ecobrick*

**Tabel 3. 25** Tabel *Ecobrick*

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Atribut** | **Tipe Data** | **Keterangan** |
| 1. | id | INT | *Primary Key , Not Null , Auto Increment* |
| 2. | nama | Varchar(191) | *Not Null* |
| 3. | foto\_diusulkan | Varchar(191) | *Not Null* |
| 4. | foto\_diaplikasikan | Varchar(191) |  |
| 5. | keterangan | Varchar(191) |  |
| 6. | level | INT |  |
| 5. | created\_at | TIMESTAMP |  |
| 6. | updated\_at | TIMESTAMP |  |
| 7. | *user*\_id | INT | *Foreign key dari* Tabel *user* |

1. Tabel point

**Tabel 3. 26** Tabel point

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Atribut** | **Tipe Data** | **Keterangan** |
| 1. | id | INT | *Primary Key , Not Null , Auto Increment* |
| 2. | status | Varchar(191) |  |
| 3. | datasensor | Varchar(191) |  |
| 4. | kode\_*reward* | Varchar(191) |  |
| 5. | created\_at | TIMESTAMP |  |
| 6. | updated\_at | TIMRSTAMP |  |
| 7. | masyarakat\_id | INT | *Foreign\_key* dari Tabel masyarakat |
| 8. | tempat\_sampah\_id | INT | *Foreign key* dari Tabel tempat\_sampah |

1. Tabel *feedback*

**Tabel 3. 27** Tabel *feedback*

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Atribut** | **Tipe Data** | **Keterangan** |
| 1. | id | INT | *Primary Key , Not Null , Auto Increment* |
| 2. | nama | Varchar(191) | *Not Null* |
| 3. | email | Varchar(191) | *Not Null* |
| 4. | usulan | Varchar(191) | *Not Null* |
| 5. | gambar | Varchar(191) |  |
| 6. | created\_at | TIMESTAMP |  |
| 7. | updated\_at | TIMESTAMP |  |

1. Tabel petugas\_konten*reward*

**Tabel 3. 28** Tabel petugas\_konten*reward*

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Atribut** | **Tipe Data** | **Keterangan** |
| 1. | id | INT | *Primary Key , Not Null , Auto Increment* |
| 2. | nama | Varchar(191) | *Not Null* |
| 3. | nohp | Varchar(191) | *Not Null* |
| 4. | alamat | Varchar(191) |  |
| 5. | file\_gambar | Varchar(191) |  |
| 6. | created\_at | TIMESTAMP |  |
| 7. | updated\_at | TIMESTAMP |  |
| 8. | *user*\_id | INT | *Foreign\_key* dari Tabel *user* |
| 9. | pimpinan\_*EcoRanger*\_id | INT | *Foreign\_key* dari Tabel  pimpinan\_*EcoRanger* |

# BAB 4

**HASIL DAN PEMBAHASAN**

# Hasil

Aplikasi Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah di Pulau Merah merupakan aplikasi yang berguna untuk melakukan *monitoring* komunitas dengan menampilkan posisi komunitas, agenda yang akan dilaksanakan oleh komunitas serta melakukan *monitoring* tempat sampah dengan menampilkan posisi tempat sampah dan status tempat sampah dalam keadaan penuh atau masih kosong. Terdapat lima hak akses *user* yaitu pimpinan *EcoRanger*, pimpinan komunitas, anggota komunitas, petugas lapangan dan *public user* (masyarakat). Dalam aplikasi ini juga dibuat untuk menginformasikan tentang *Ecobrick* sebagai produk yang dihasilkan dari sampah plastik dan bermanfaat bagi lingkungan.

# Halaman Utama

Pada Gambar 4. 1 merupakan Halaman utama yang menampilkan bagian depan dari *website* yang menampilkan informasi singkat dari Aplikasi dan dilengkapi dengan :

* + - 1. Bagian *navbar* sebagai penunjuk tampilan apasaja yang ada pada halaman utama.
      2. terdapat tombol mulai dan donasi yang langsung dihubungkan pada halaman yang sudah ditentukan.
      3. Tampilan w*idget* dari *tawk.to* untuk memulai pesan secara langsung.



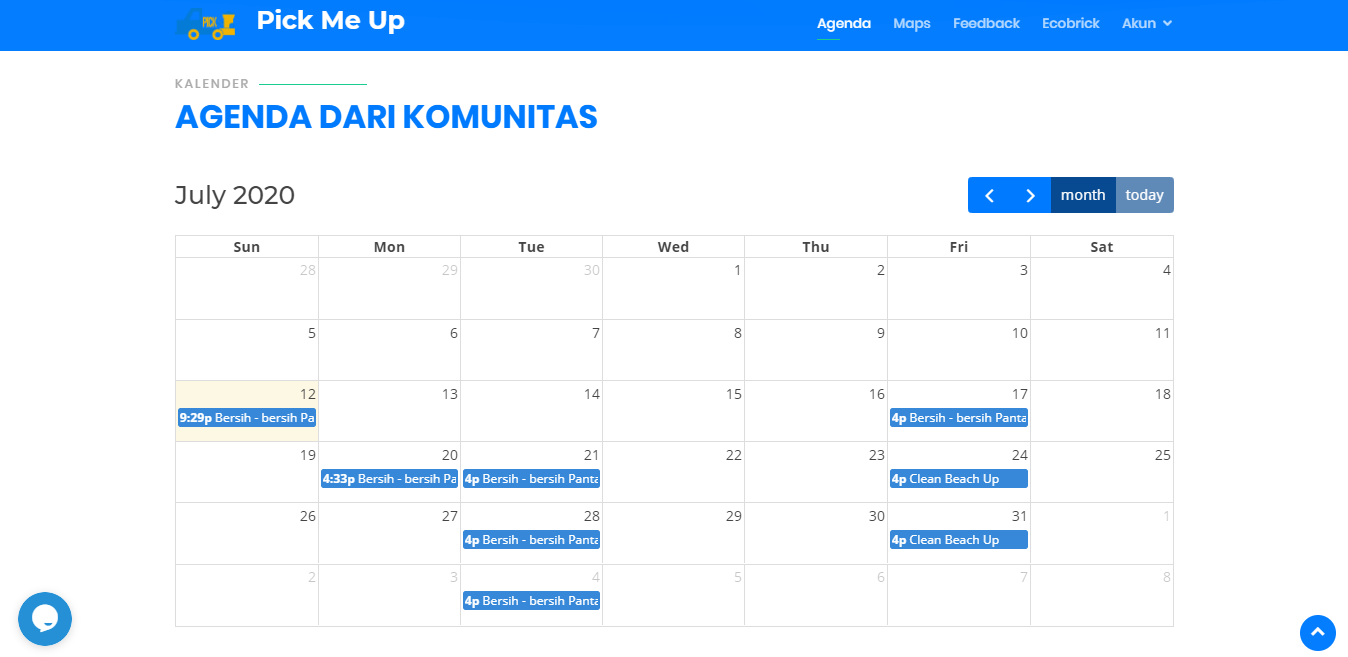
1

2

3

**Gambar 4. 1** Halaman Utama

Pada Gambar 4. 2 tampilan jadwal agenda komunitas yang menampilkan semua agenda yang akan dilaksanakan sesuai dengan waktu yang telah ditentukan ditunjukkan pada gambar angka 1.

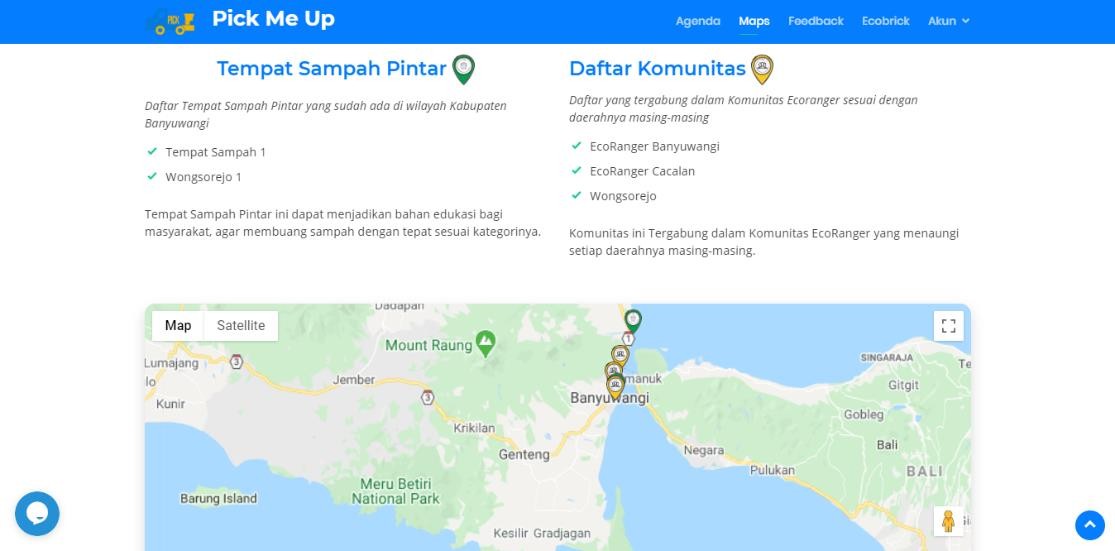


1

**Gambar 4. 2** Tampilan Jadwal Agenda

Pada Gambar 4. 3 terdapat tampilan untuk melihat daftar komunitas serta daftar tempat sampah yang sudah tersedia juga terdapat peta untuk memperjelas lokasi komunitas maupun tempat sampah. Untuk penjelasan lebihnya sebagai berikut :

* 1. Daftar komunitas dan tempat sampah yang dapat dilihat dengan tanda centang hijau dan gambar marker untuk menandakan posisi dalam peta.
  2. Tampilan peta dengan marker sesuai kategorinya.
  3. Marker yang jika ditekan maka akan muncul tampilan rincian dari komunitas atau tempat sampah.



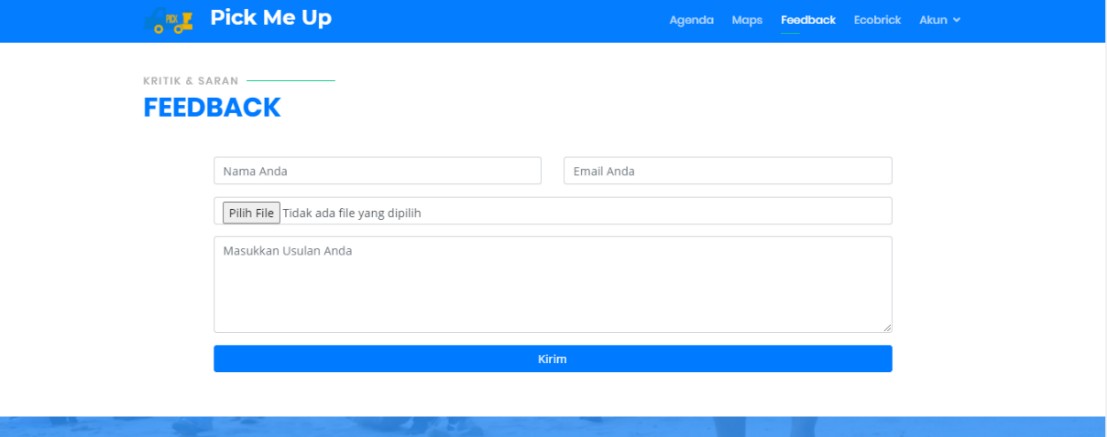
1

2

3

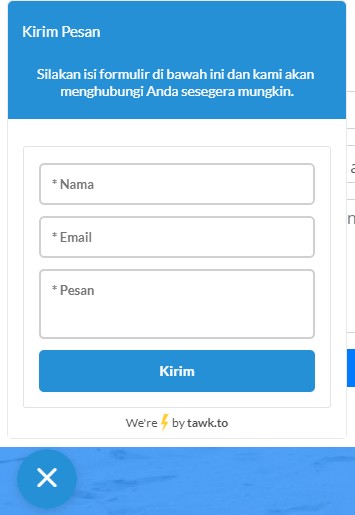
**Gambar 4. 3** Tampilan Daftar Komunitas dan Tempat Sampah Pintar

Pada Gambar 4. 4 merupakan tampilan *feedback* yang menyediakan layanan kritik ataupun saran dari masyarakat yang menyertakan nama dan email serta keterangan dan sebagai opsional bisa menambahkan gambar yang bisa dilihat pada angka 1.



1

**Gambar 4. 4** Tampilan *Feedback*

Dalam *feedback* juga terdapat fitur *live chat support* untuk menunjang layanan dari Aplikasi ini dengan pihak ketiga yaitu *tawk.to* dengan tampilan *interface* yang dapat dilihat pada Gambar 4. 5. Pada tampilan *interface* masyarakat dapat langsung memberikan *feedback* kepada admin/petugas lapangan yang memiliki hak akses untuk membalas *feedback* dari masyarakat.

**Gambar 4. 5** Tampilan *Feedback* Dalam Bentuk *Live Chat Support*

* + 1. **Halaman Utama *Ecobrick***

Halaman Utama *Ecobrick* merupakan halaman yang menyediakan semua informasi yang berisi video maupun pengertian tentang *Ecobrick*, juga berisi gambar-gambar yang sudah direalisasikan di lapangan serta *navbar* untuk

menunjukkan tampilan apa saja yang terdapat halaman utama *Ecobrick* terdapat pada Gambar 4. 6. penjelasan tambahan :

1. Dropdown untuk bagian akun yang menampilkan link untuk ke halaman

*login* atau daftar sebagai anggota komunitas.

1. Bagian *navbar* sebagai penunjuk tampilan apasaja yang ada pada halaman

*Ecobrick*.

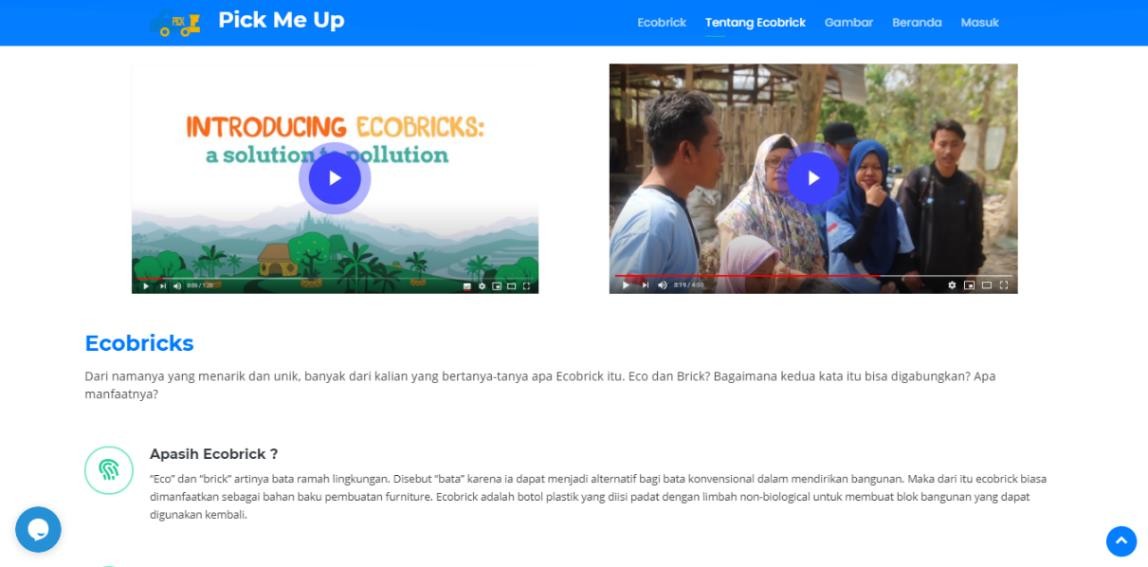


2

1

**Gambar 4. 6** Halaman *Ecobrick*

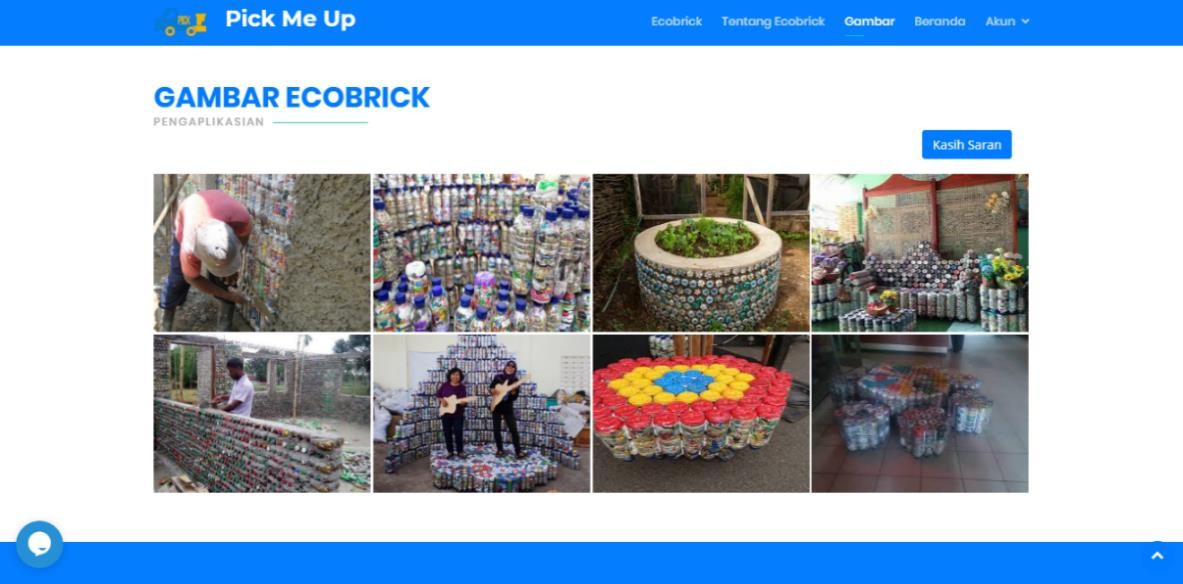
Gambar 4. 7 merupakan tampilan tentang *Ecobrick* yang berisi informasi mengenai *Ecobrick*, manfaat *Ecobrick* dan juga cara membuat *Ecobrick* yang ditunjukkan oleh angka 2 serta dilengkapi dengan video *Ecobrick* dan video mengenai Aplikasi, video tersebut langsung terhubung dengan youtube dan bisa langsung dilihat langsung melalui *website* yang ditunjukkan oleh angka 1.



1

2

**Gambar 4. 7** Tampilan Tentang *Ecobrick*



2

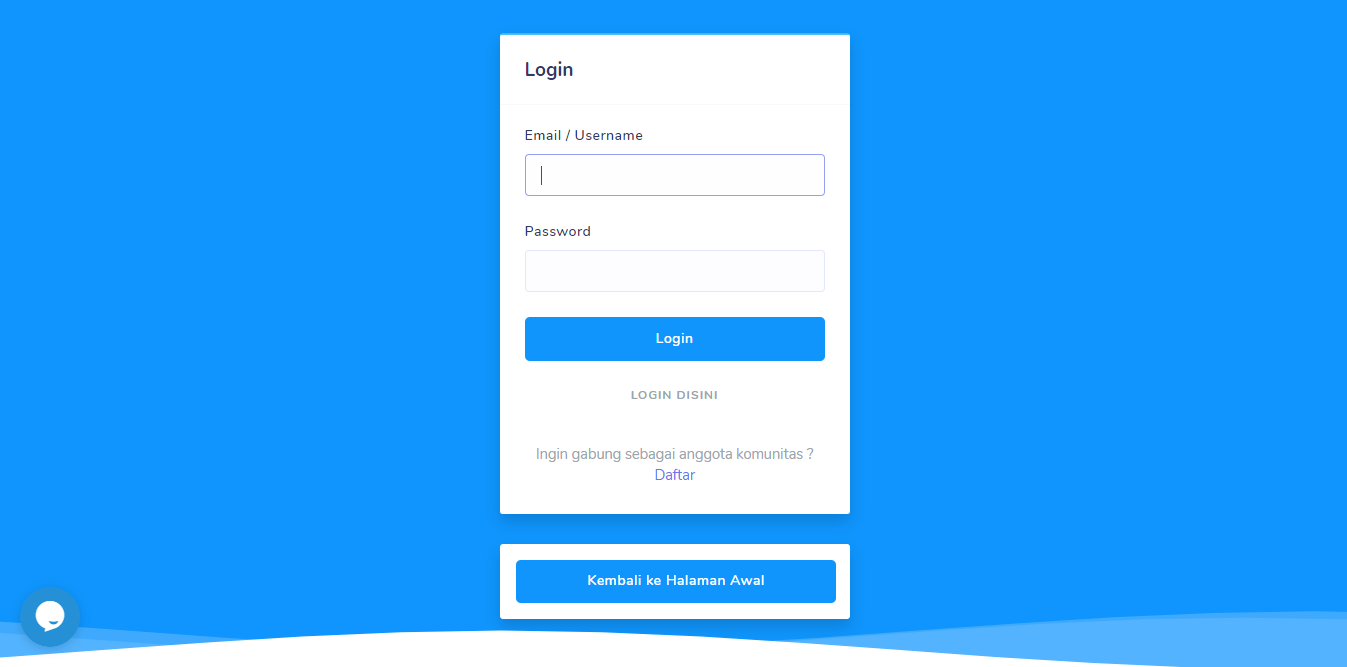
1

**Gambar 4. 8** Tampilan Gambar *Ecobrick*

Pada Gambar 4. 8 merupakan tampilan gambar *Ecobrick* yang meyediakan tampilan berupa daftar gambar *Ecobrick* yang sudah diaplikasikan ditunjukkan oleh angka 1 dan juga terdapat tombol kasih saran yang nantinya terdapat *form* untuk menambahkan saran yang dapat diaplikasikan sewaktu-waktu ditunjukkan oleh angka 2.

* + 1. **Halaman *Login***

Halaman *login* merupakan halaman proses validasi hak askes setiap pengguna yang masuk, pengguna yang meliputi bagian internal, bagian perencanaan, bagian pengadaan dapat mengoperasikan sistem aplikasi sesuai dengan hak akses masing – masing. Proses validasi *user* dilakukan dengan cara melakukan pemeriksaan *username* dan *password* dari *user* yang telah ada di *database* yang dapat di lihat pada Gambar 4. 9.



1

2

3

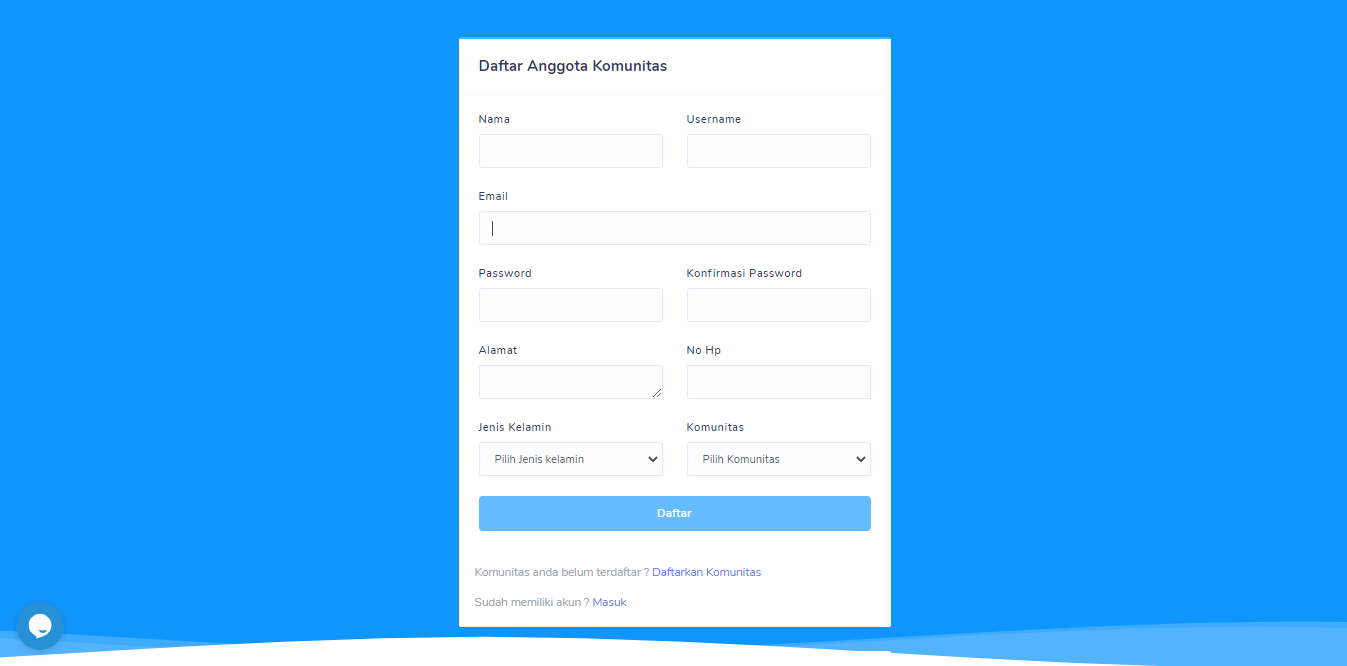
**Gambar 4. 9** Tampilan *Login*

Untuk penjelasan halaman *login* dari Gambar 4.9 sebagai berikut :

1. *Form* untuk mengisikan *username* dan *password* agar dapat masuk ke halaman beranda sesuai *role* yang dimasukkan.
2. Link untuk ke halaman daftar sebagai anggota komunitas.
3. Tombol untuk kembali ke halaman awal atau halaman utama *website*.

# Halaman Daftar Anggota Komunitas

Gambar 4. 10 merupakan halaman daftar sebagai anggota komunitas agar mendapatkan hak akses untuk masuk ke dalam halaman anggota komunitas dengan mengisi semua *form* yang ada sesuai data dirinya masing-masing. *Form*nya antara lain yaitu nama, *user*name, email, password, alamat, komunitas yang ditunjukkan oleh angka 1 dan link untuk ke halaman daftar komunitas dan *login*.

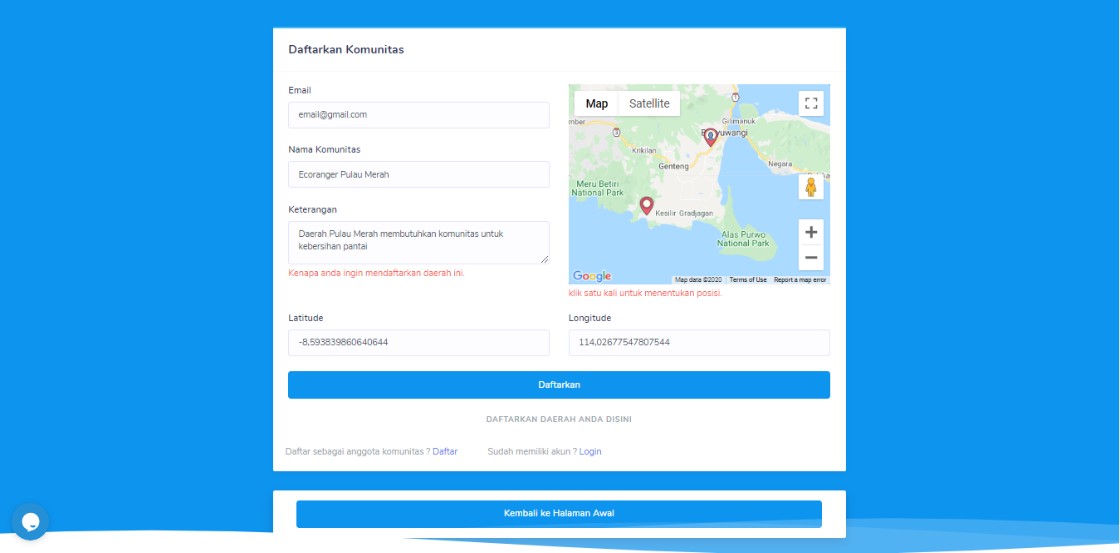


1

2

**Gambar 4. 10** Halaman Daftar Anggota Komunitas

# Halaman Daftar Komunitas



1

2

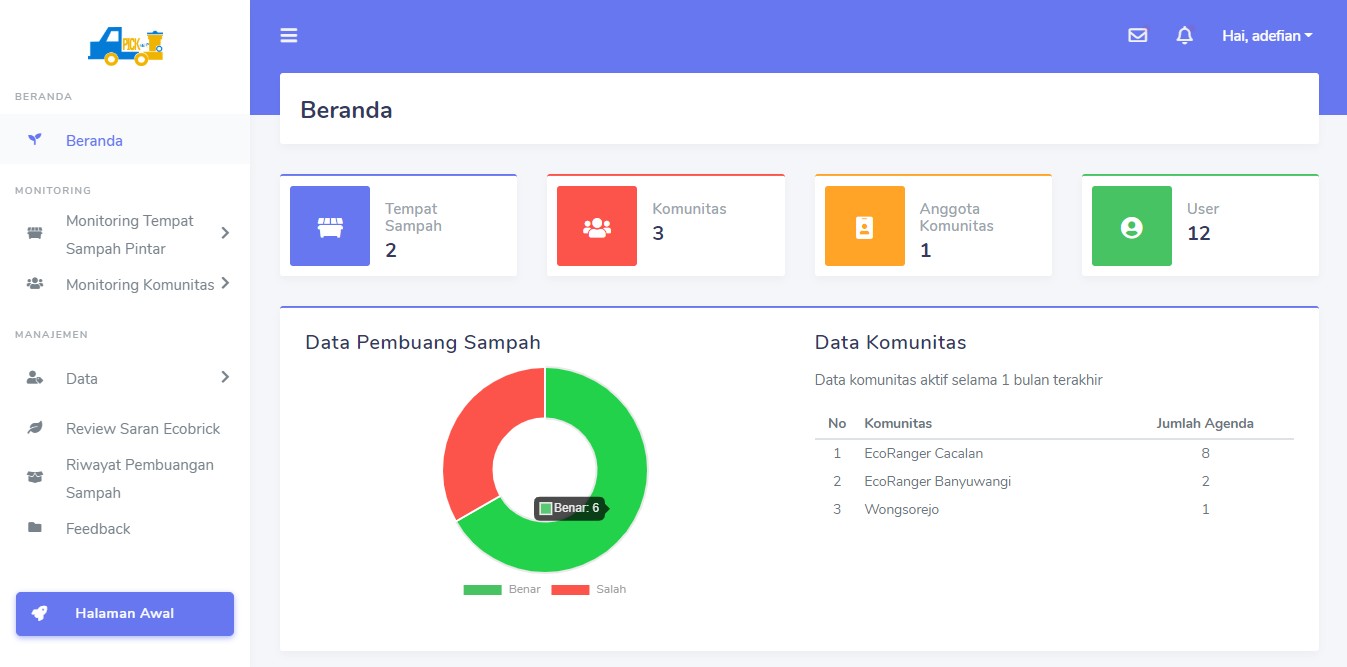
3

**Gambar 4. 11** Tampilan Daftar Komunitas

Gambar 4. 11 merupakan Halaman daftar komunitas untuk daerah yang ingin mendaftarkan daerahnya dengan mengisi semua *form* yang tersedia dan juga memilih lokasi dimana komunitas itu berada dengan menekan titik lokasi pada peta yang tersedia.

Untuk penjelasan halaman daftar komunitas dari Gambar 4. 11 sebagai berikut :

* + - 1. *Form* untuk mengisikan data dari komunitas yang dibutuhkan oleh *website*.
      2. Tampilan peta untuk menentukan lokasi komunitas agar lebih mudah untuk melihat lokasi sacara akurat.
      3. *Form* dari latitude dan longitude yang otomatis terisi ketika sudah menentukan posisi komunitas.
    1. **Halaman Beranda Pimpinan *EcoRanger***



2

5

4

3

1

**Gambar 4. 12** Halaman *Beranda* Pimpinan *EcoRanger*

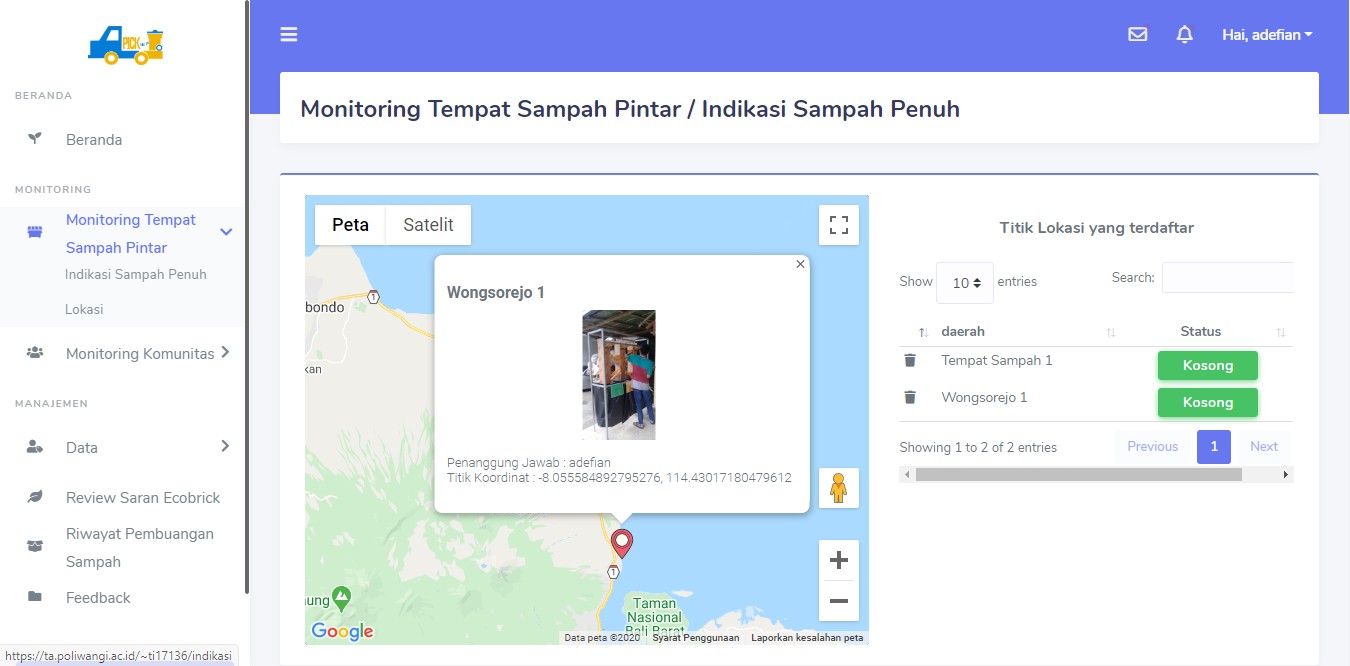
Halaman Beranda Pimpinan *EcoRanger* merupakan halaman yang hanya dapat diakses oleh *role* pimpinan *EcoRanger*, pada halaman ini menampilkan jumlah sebagian data yang dikelola, data pembuangan sampah dan data komunitas yang aktif selama satu bulan terakhir. Pada bagian beranda ini terdapat sidebar untuk menunjukkan halaman apa saja yang dapat diakses oleh *role* pimpinan *EcoRanger* dapat dilihat pada Gambar 4. 12.

Untuk penjelasan halaman beranda pimpinan *EcoRanger* dari Gambar 4. 12 sebagai berikut :

* + - 1. Sidebar untuk menunjukkan halaman apa saja yang dapat di akses *role*

pimpinan *EcoRanger*.

* + - 1. Menampilkan nama akun yang telah *login*, simbol pesan dan notifikasi.
      2. Menunjukkan beberapa jumlah data sesuai dengan nama data yang tertera.
      3. Menampilkan grafik berbentuk lingkaran untuk menunjukkan data benar atau salah ketika pengguna memasukkan sampah pada tempat sampah pintar yang sudah terhubung dengan server.
      4. Menampilkan data komunitas yang aktif mengadakan agenda selama 1 bulan terakhir.
    1. ***Monitoring* Tempat Sampah**

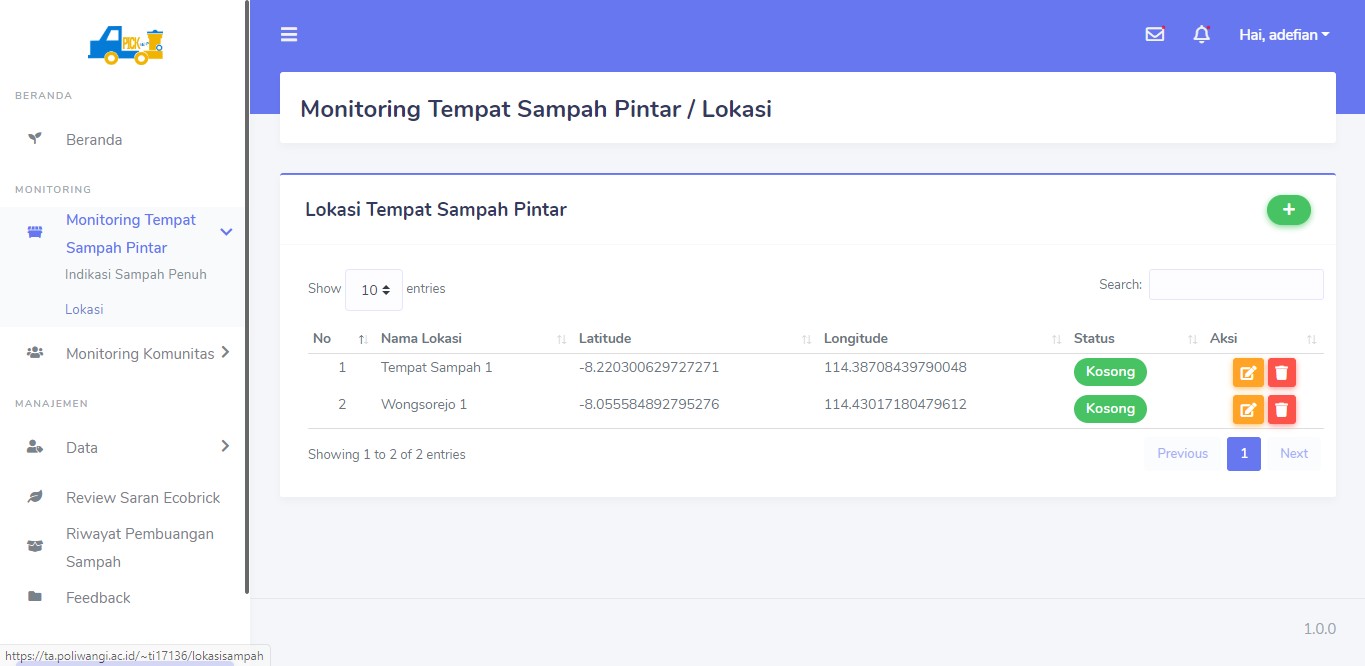


1

2

**Gambar 4. 13** Halaman Indikasi Sampah Penuh

Pada Gambar 4. 13 merupakan halaman indikasi sampah jika salah satu sampah dalam keadaan kosong, penuh atau dalam keadaan pengambilan oleh petugas lapangan ditungukkan oleh angka 1. Dalam halaman ini menampilkan peta yang menandakan posisi tempat sampah ditunjukkan oleh angka 2.



2

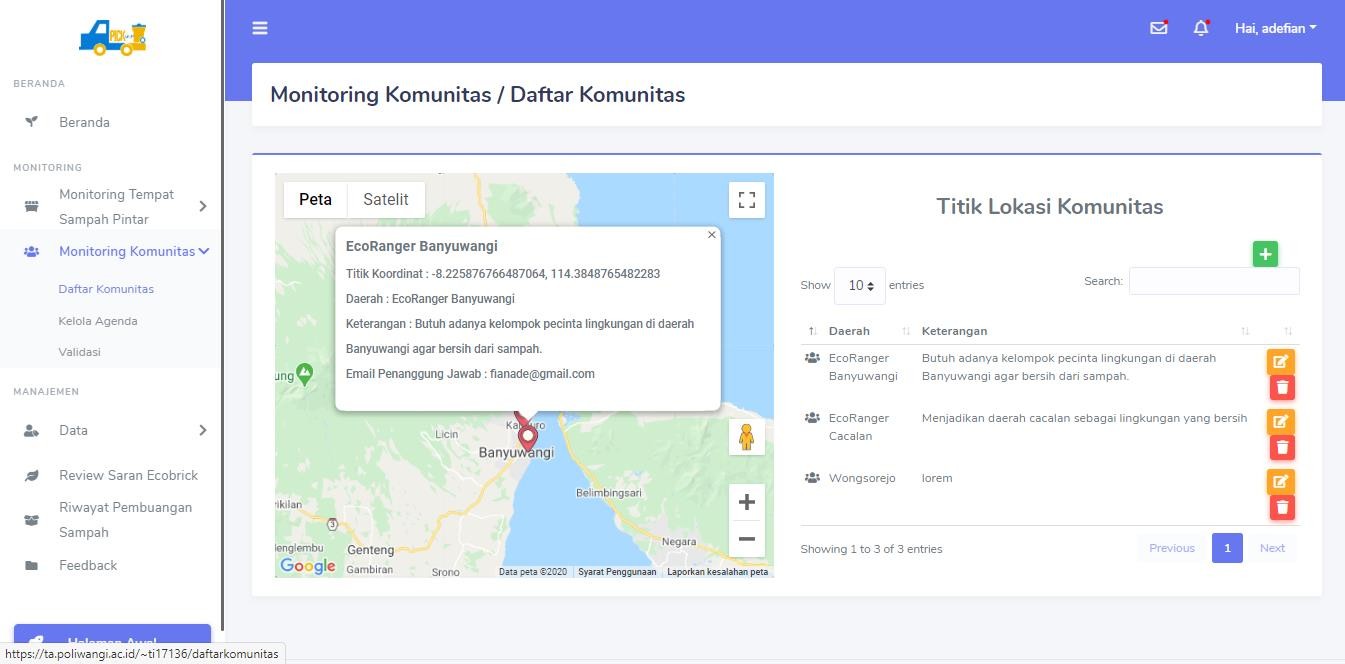
3

1

**Gambar 4. 14** Halaman Lokasi Sampah

Pada Gambar 4. 14 merupakan halaman lokasi sampah yang menampilkan data tempat sampah ditunjukkan oleh angka 1 dan digunakan juga untuk menambahkan lokasi tempat sampah ditunjukkan oleh angka 2, mengubah lokasi tempat sampah dan menghapus tempat sampah jika perlu adanya tindakan ditunjukkan oleh angka 3.

* + 1. ***Monitoring* Komunitas**



3

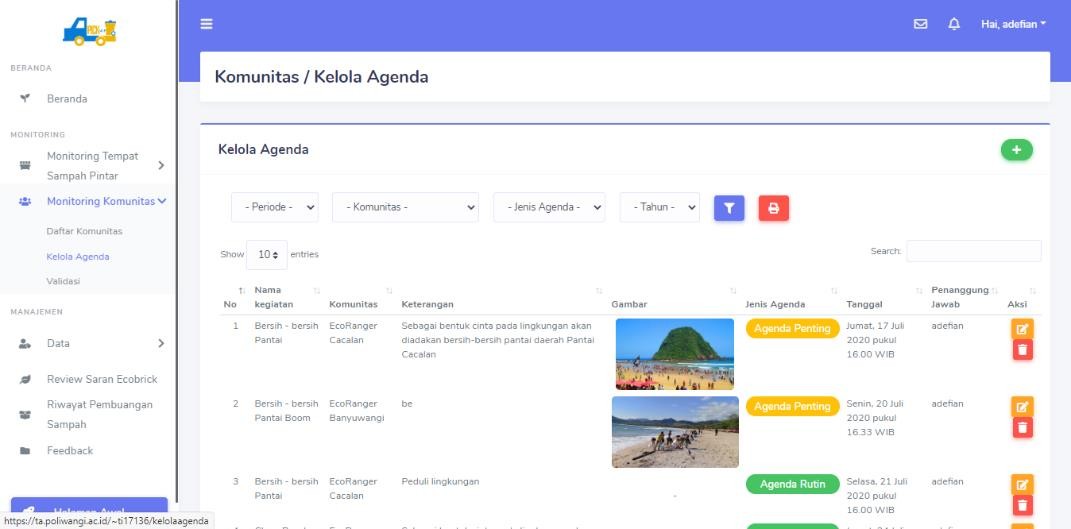
4

1

2

**Gambar 4. 15** Halaman Daftar Komunitas

Pada Gambar 4. 15 merupakan halaman daftar komunitas yang sudah tervalidasi dan bisa menambahkan agenda yang dilakukan oleh komunitas ditunjukkan oleh angka 1, juga terdapat peta untuk mengetahui titik lokasi komunitas serta rincian komunitas sesuai markernya masing-masing ditunjukkan oleh angka 2. Pada halaman ini juga dapat menambahkan komunitas ditunjukkan oleh angka 3, mengubah lokasi komunitas dan menghapus komunitas jika perlu adanya tindakan yang ditunjukkan oleh angka 4.

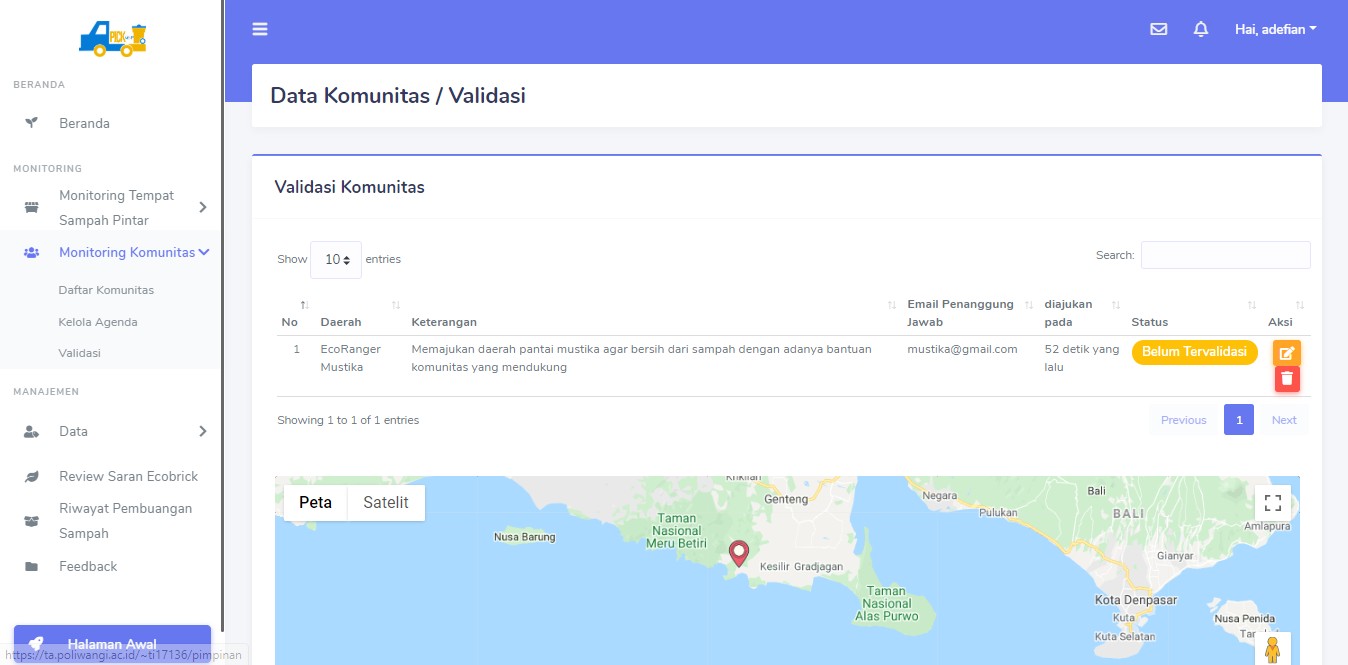


2

1

**Gambar 4. 16** Halaman Kelola Agenda Komunitas

Pada Gambar 4. 16 merupakan halaman kelola agenda komunitas yang berisi daftar agenda yang akan dilaksanakan disertai nama kegiatan, komunitas, keterangan, gambar, jenis agenda, penanggung jawab dan juga tanggal dilaksanakannya kegiatan ditunjukkan oleh angka 1. Dalam halaman ini dapat menambahkan, mengubah, menghapus agenda, juga terdapat fitur filter dan cetak pdf ditunjukkan oleh angka 2.



1

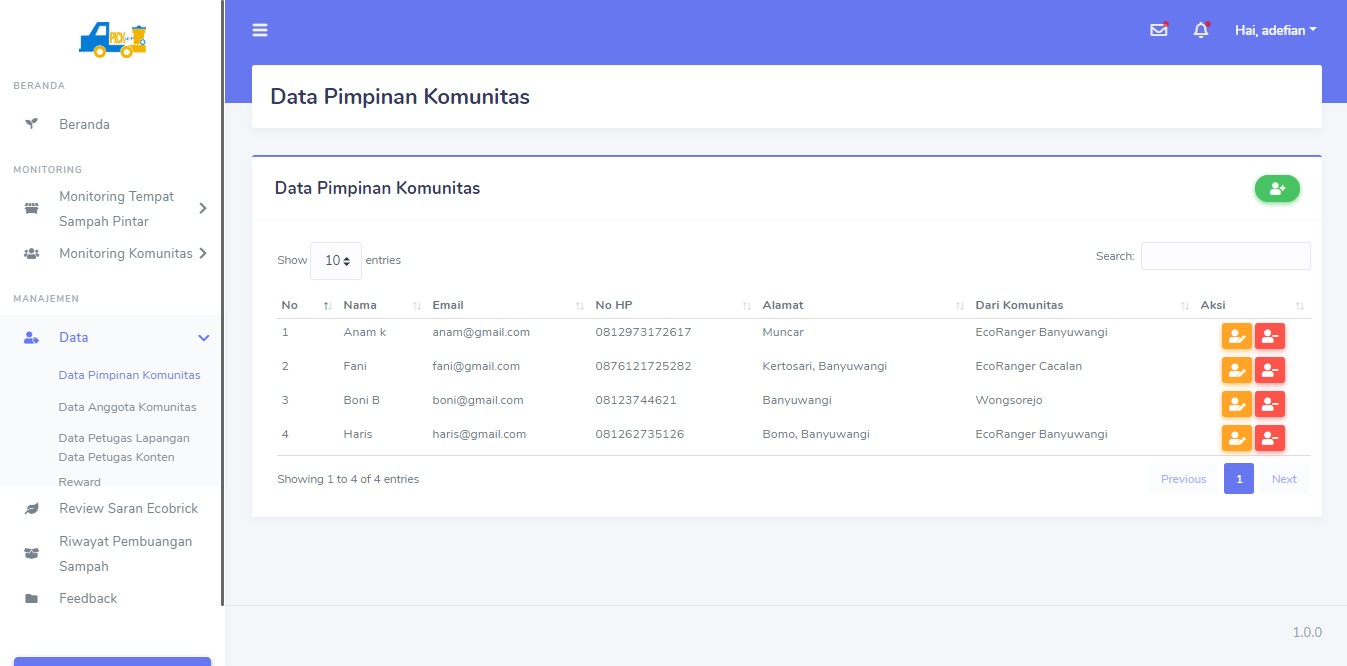
2

**Gambar 4. 17** Halaman Validasi Komunitas

Pada Gambar 4. 17 terdapat halaman validasi komunitas yang sudah mendaftar melalui *form* daftar komunitas yang selanjutnya akan muncul dalam daftar validasi ditunjukkan oleh angka 1, jika komunitas yang mendaftar sesuai maka dari *role* pimpinan *EcoRanger* dapat melakukan validasi dengan menekan tombol validasi yang ditunjukkan oleh angka 2.

* + 1. **Halaman Data *User***

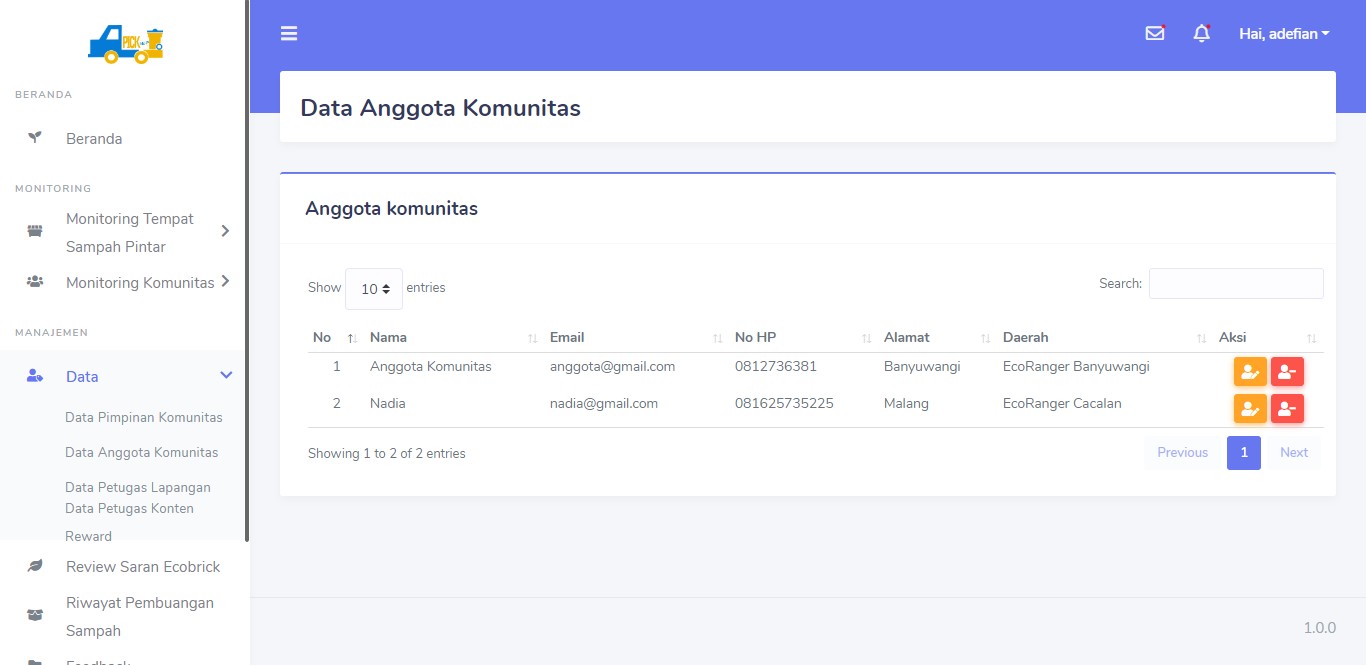
Halaman data merupakan halaman yang mengelola sebagian dari *role* pimpinan komunitas, anggota komunitas, petugas lapangan dan petugas konten *reward* dari menambahkan, mengubah dan menghapus data agar masing-masing *role* mempunyai *user*name, email dan password ditunjukkan oleh angka 1. Pada halaman ini juga dapat melakukan *monitoring* data-data dari *role* pimpinan komunitas yang terdapat pada Gambar 4. 18 serta data anggota komunitas, data petugas lapangan dan data petugas konten *reward*. Tetapi yang mempunyai hak akses penuh untuk halaman data hanya dari *role* pimpinan *EcoRanger*, untuk masing-masing *role* dapat melihat data sesuai akun yang *login*.



1

**Gambar 4. 18** Halaman Data Pimpinan Komunitas

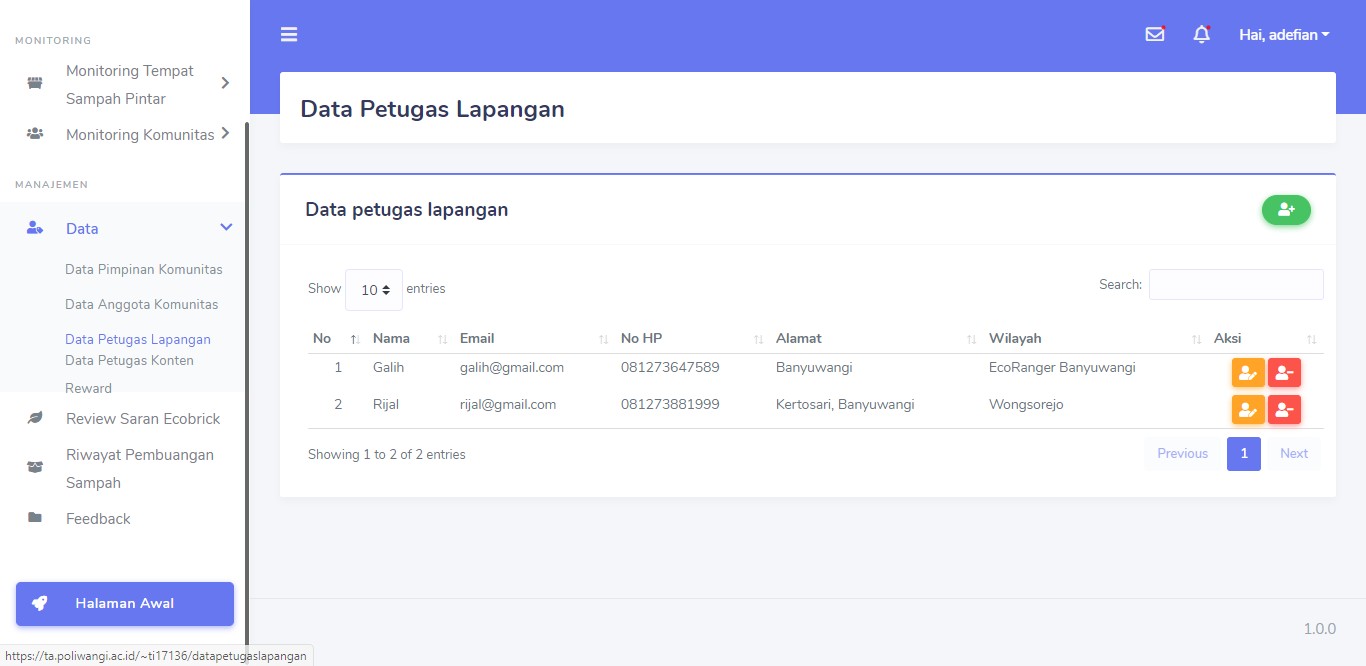
Pada Gambar 4. 19 merupakan halaman data anggota komunitas yang sudah terdaftar dalam aplikasi dan dapat menggunakan akun tersebut untuk *login* dalam aplikasi *website*. Semua data anggota komunitas yang sudah terdaftar pada aplikasi ini ditunjukkan pada angka 1 beserta dengan daerah yang dipilih pada saat mendaftar. Pada halaman data anggota komunitas ini hanya bisa edit dan hapus data tanpa bisa menambahkan data.



1

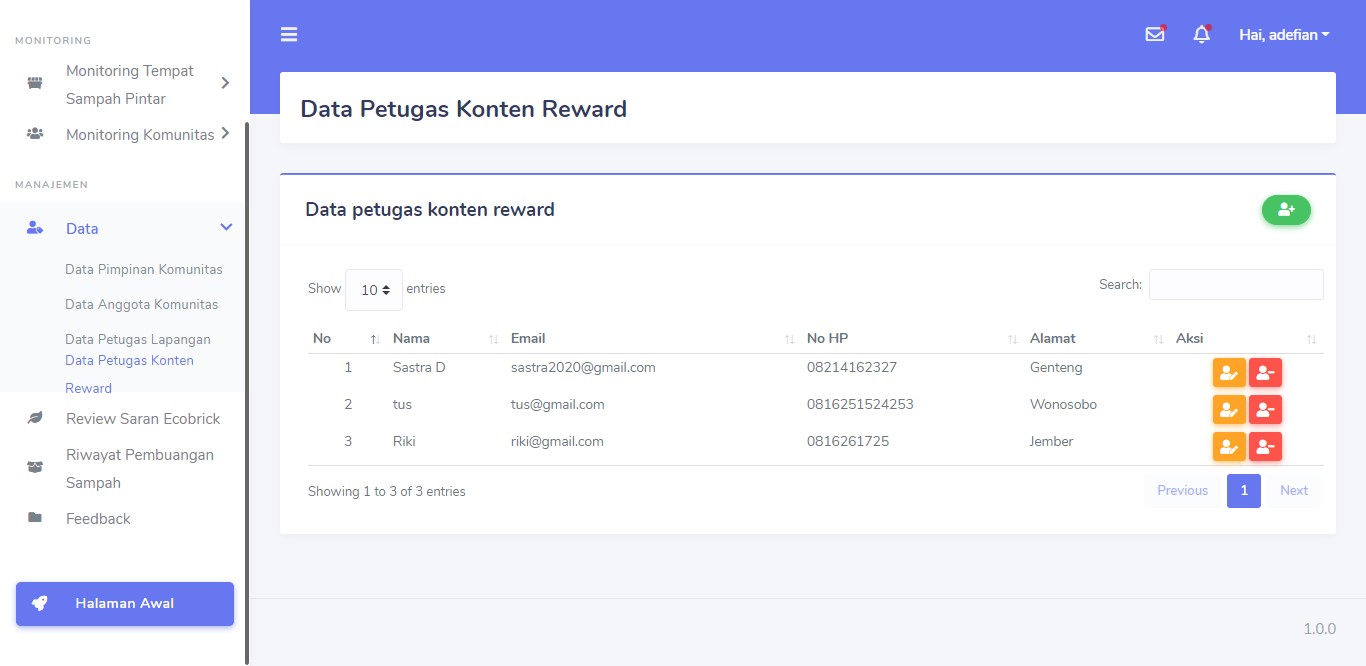
**Gambar 4. 19** Halaman Data Anggota Komunitas

Pada Gambar 4. 20 merupakan halaman data petugas lapangan yang sudah terdaftar dalam aplikasi dan dapat menggunakan akun tersebut untuk *login* dalam aplikasi *website* dan android. Semua data petugas lapangan yang sudah terdaftar pada aplikasi ini ditunjukkan pada angka 1 beserta dengan daerah yang dipilih pada saat ditambahkan oleh pimpinan *EcoRanger*. Pada halaman data petugas lapangan ini memiliki akses penuh yaitu menambahkan data, edit dan juga hapus.



**Gambar 4. 20** Halaman Data Petugas Lapangan

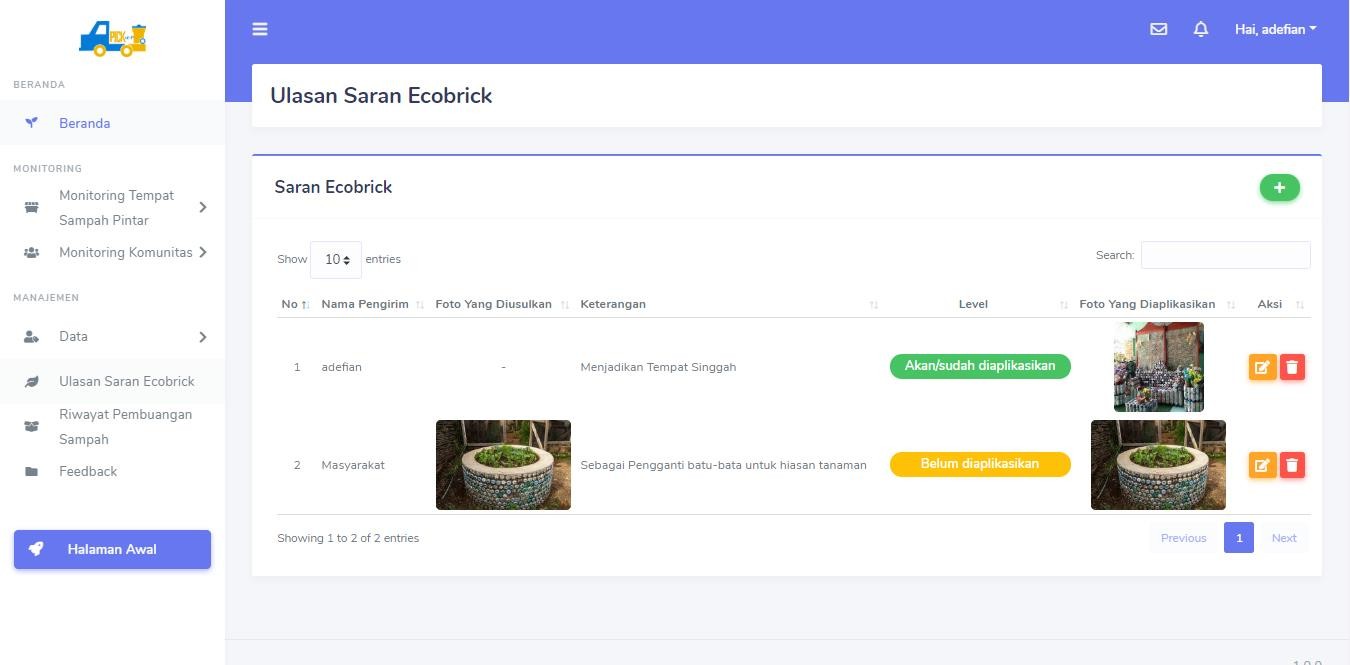
Pada Gambar 4. 21 merupakan halaman data petugas lapangan yang sudah terdaftar dalam aplikasi dan dapat menggunakan akun tersebut untuk *login* dalam aplikasi android. Semua data petugas lapangan yang sudah terdaftar pada aplikasi ini ditunjukkan pada angka 1 beserta dengan daerah yang dipilih pada saat ditambahkan oleh pimpinan *EcoRanger*. Pada halaman data petugas konten ini memiliki akses penuh yaitu menambahkan data, edit dan juga hapus.



**Gambar 4. 21** Halaman Data Petugas Konten *Reward*

* + 1. **Ulasan Saran *Ecobrick***

Pada Gambar 4. 22 merupakan halaman dari ulasan saran *ecobrick* menampilkan daftar dari saran *ecobrick* yang ditambahkan oleh masyarakat melalui halaman utama *ecobrick* dengan menekan tombol kasih saran yang berada pada atas daftar gambar *ecobrick* yang akan menampilkan tampilan *pop-up* berisi *form* yang ditampilkan Gambar 4. 23.



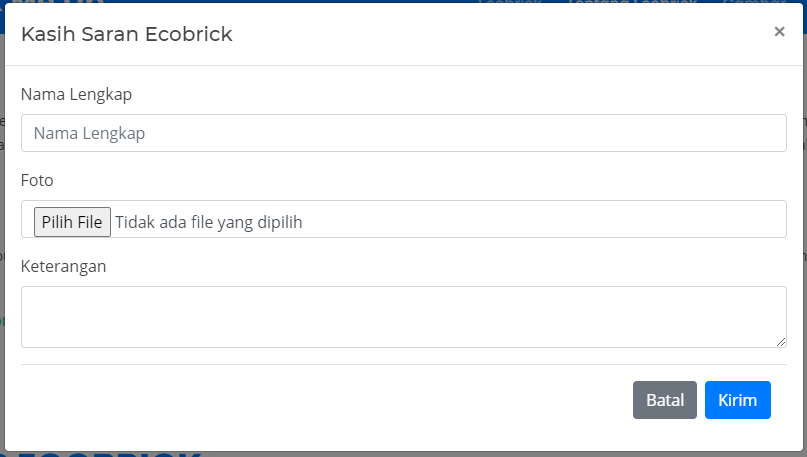
4

1

2 3

**Gambar 4. 22** Halaman Ulasan Saran *Ecobrick*

Berikut penjelasan dari halaman ulasan saran *Ecobrick* :

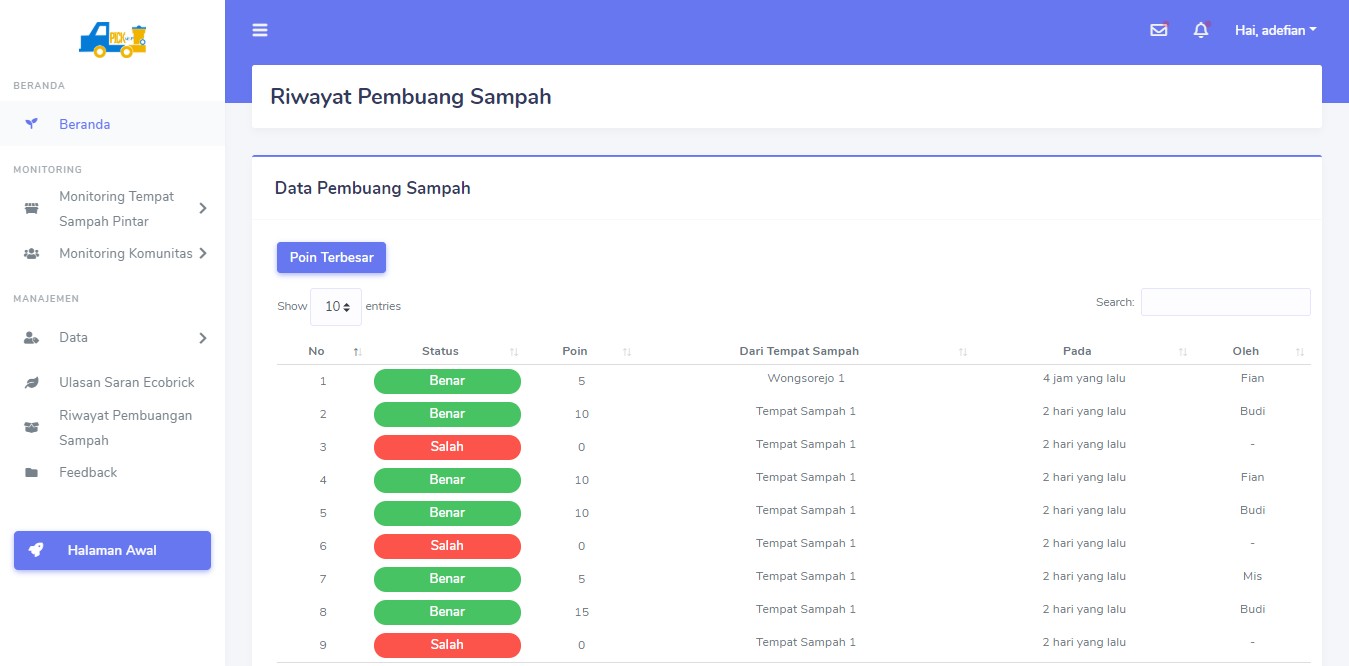
* + - 1. Daftar saran *Ecobrick* yang telah ditambahkan oleh masyarakat.
      2. Tampilan gambar pada *form* yang diusulkan tetapi untuk ditampilkan pada gambar di halaman utama *Ecobrick* hanya dari foto yang diaplikasikan.
      3. Tampilan gambar yang telah atau akan diaplikasikan.
      4. Tombol untuk menambahkan *Ecobrick* yang telah atau akan diaplikasikan.

**Gambar 4. 23** Tampilan Kasih Saran *Ecobrick*

# Riwayat Pembuangan Sampah

Pada Gambar 4. 24 merupakan halaman riwayat pembuang sampah menampilkan daftar pembuang sampah yang berisi status, poin, dari tempat sampah mana dan oleh siapa ditunjukkan oleh angka 1. Untuk mengetahui siapa saja yang membuang sampah pada tempat sampah pintar bisa terlihat jika status saat membuang sampah benar dan masyarakat megenerate kode *reward*. Terdapat

tombol untuk melihat poin terbesar yang didapatkan oleh masyarakat ditunjukkan oleh angka 2.



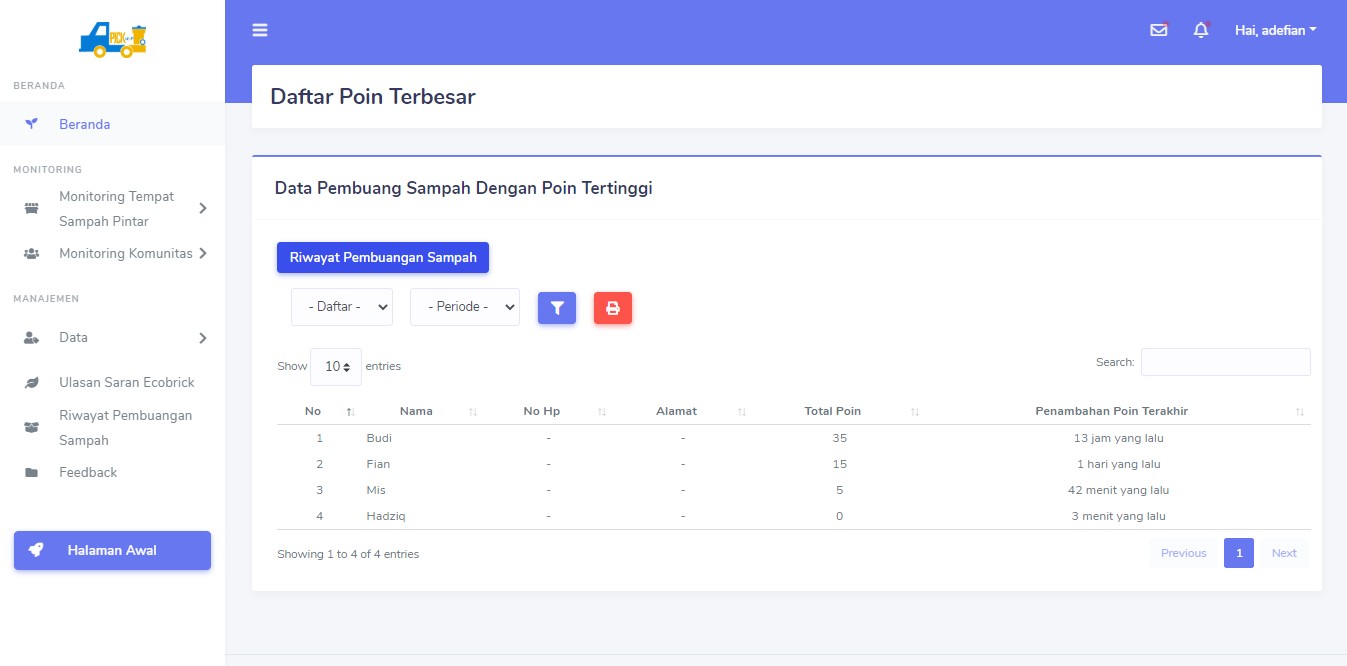
2

1

**Gambar 4. 24** Halaman Riwayat Pembuang Sampah

Untuk penjelasan halaman daftar poin terbesar dari Gambar 4.25 sebagai berikut :

1. Daftar poin yang terurut dari yang terbesar berada pada posisi paling atas.
2. Fitur filter untuk mencari 10 besar, 20 besar atau dalam periode tertentu serta cetak pdf untuk melindungi keutuhan dokumen.
3. Tombol untuk kembali ke halaman riwayat pembuang sampah sebagai aktifitas *monitoring* pembuang sampah.



3

2

1

**Gambar 4. 25** Halaman Daftar Poin Terbesar

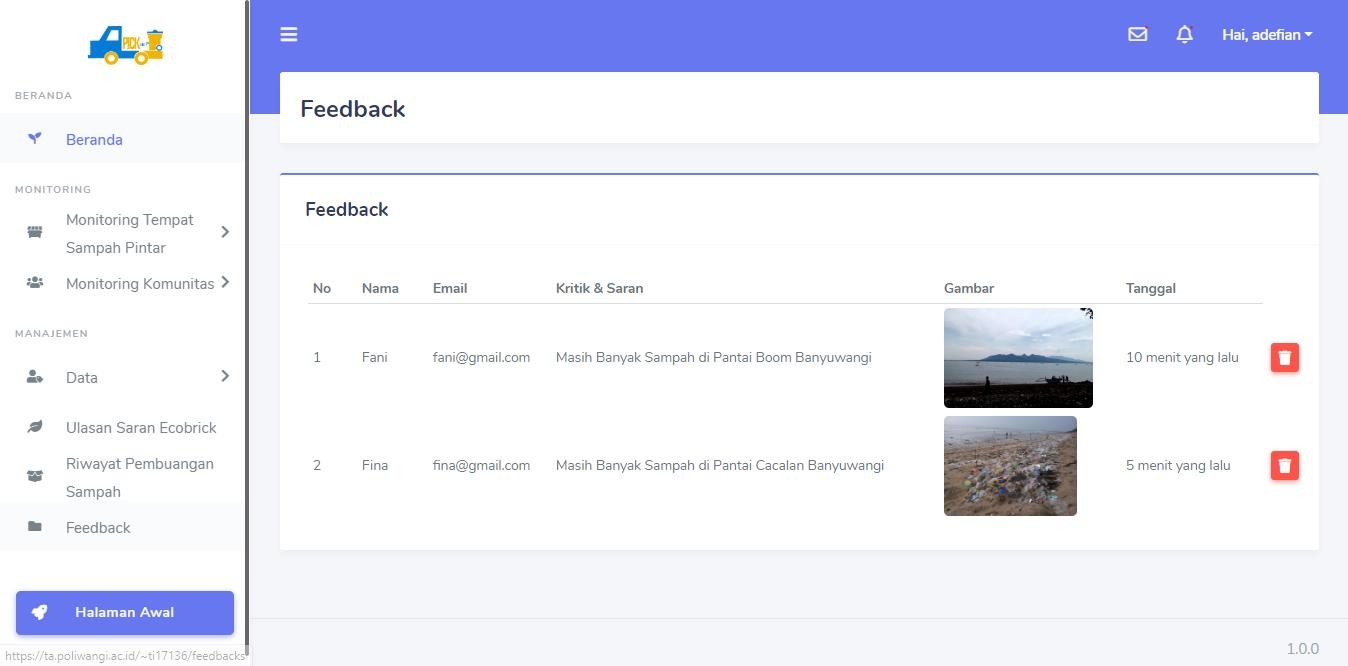
## Feedback

Pada Gambar 4. 26 Halaman *Feedback* yang menampilkan daftar dari

*feedback* yang didapat dari masyarakat pada halaman utama *website* berupa nama,

email, kritik saran, tanggal dan gambar ditunjukkan oleh angka 1. Pada halaman ini

*role* pimpinan *EcoRanger* hanya diberi akses untuk menghapus data.



1

**Gambar 4. 26** Halaman *Feedback*

# Pembahasan

Aplikasi Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah di Pulau Merah Banyuwangi. Merupakan aplikasi yang berguna untuk membantu pengguna dalam pencarian informasi komunitas peduli lingkungan di kabupaten Banyuwangi. Pencarian informasi berupa jadwal agenda setiap komunitas, ecobrick sebagai produk yang dihasilkan dari sampah plastik yang dapat berguna sebagai fasilitas penunjang wisata pantai dan untuk melestarikan lingkungan dari sampah plastik serta menyediakan peta yang mengetahui posisi komunitas dan tempat sampah pintar yang disediakan oleh pihak komunitas.

Terdapat lima pengguna dalam aplikasi ini yaitu pimpinan *EcoRanger*, pimpinan komunitas, anggota komunitas, petugas lapangan dan *public user* (masyarakat). Untuk penjelasan lebih lanjut ditunjukkan pada Tabel 4.1

**Tabel 4. 1** Tabel User

|  |  |  |
| --- | --- | --- |
| No | Pengguna | Keterangan |
| 1. | Pimpinan *EcoRanger* | Pengguna yang memiliki hak akses tertinggi dan berperan penting dalam aplikasi seperti melakukan *monitoring* tempat sampah dan komunitas serta melakukan penambahan atau  perubahan pada semua data dalam sistem |

|  |  |  |
| --- | --- | --- |
| 2. | Pimpinan Komunitas | Hanya memiliki akses untuk melakukan *monitoring* komunitasnya masing-masing termasuk menambahkan dan mengubah agenda komunitas serta mengulas saran *Ecobrick* dari  masyarakat |
| 3. | Anggota Komunitas | Hanya memiliki akses untuk melakukan  *monitoring* komunitasnya masing-masing serta mengulas saran *Ecobrick* dari masyarakat |
| 4. | Petugas Lapangan | Hanya memiliki akses untuk melakukan *monitoring* tempat sampah juga dapat melihat posisi komunitas yang ada dalam sistem dan dapat melihat riwayat pembuangan sampah yang dilakukan oleh masyarakat pada tempat  sampah pintar |
| 5. | *Public User*  (Masyarakat) | Dapat mengetahui segala informasi yang ada  pada halaman awal *website* |

Diharapkan aplikasi ini dapat membantu masyarakat untuk melestarikan lingkungan wisata pantai serta komunitas peduli lingkungan wisata pantai di kabupaten Banyuwangi termasuk komunitas *EcoRanger* Banyuwangi.

# Melakukan Validasi data komunitas

Untuk menghasilkan komunitas yang tervalidasi agar tidak sembarangan komunitas masuk dalam sistem ini maka dibuatlah proses validasi data komunitas pada halaman validasi dengan *source code* sebagai berikut :

*public* function index()

{

$data = Komunitas::where('level', 0)->get();

Return view('admins.layouts\_sidebar.*monitoring*\_komunitas. validasi', compact('data'));

}

**Gambar 4. 27** *Source code function index* WebValidasiController

Pada gambar 4. 27 dapat dilihat bahwa *source code* di atas digunakan untuk menampilkan semua data yang diperoleh dari *model* Komunitas dengan kondisi

kolom level berisi angka 0 sehingga isi dari kolom level selain angka 0 tidak ditampilkan. *Model* Komunitas disini adalah fungsi yang berhubungan dengan tabel komunitas. Data komunitas yang ditampilkan sama dengan *variabel* data dengan tampilan yang berada pada file validasi.blade.php.

*public* function update(Request $request, $id)

{

$komunitas = Komunitas::findOrFail($id);

$input = ([

'level' => $request->level

]);

$komunitas->update($input);

alert()->success('Berhasil validasi data', 'telah tervalidasi'); return redirect('daftarkomunitas');

}

**Gambar 4. 28** *Source code function update* WebValidasiController

Pada gambar 4.28 dapat dilihat bahwa *source code* diatas digunakan untuk validasi data komunitas dengan merubah isi kolom level sesuai data yang diinginkan, dalam hal ini jika komunitas akan divalidasi maka isi dari kolom level harus angka 1. Jika sudah melakukan validasi maka akan ditampilkan *pop-up* ‘Berhasil validasi data’ dan selanjutnya akan dialihkan ke halaman daftar komunitas yang sudah tervalidasi dengan mengetahui data lengkap dari komunitas serta titik lokasi komunitas.

* + 1. **Komunikasi data antara *website* dan *hardware* dengan *MQTT***

Agar dari perangkat *hardware* dapat mengirim data ke *database server* maka dibutuhkan protokol komunikasi yang dapat digunakan dengan baik, dalam sistem ini menggunakan *Node.JS* untuk *subscribe* atau menerima data dari *hardware* dengan *source code* sebagai berikut :

1.

2.

3.

4.

5.

6.

7.

8.

9.

10.

11.

const mqtt = require('mqtt') ;

const client = mqtt.connect('mqtt://test.mosquitto.org'); const axios = require('axios');

client.on('connect', () => {

client.subscribe('datasensor/tempatsampah',{qos:1});

})

client.on('message',function(topic, message){ console.log('this message :',message.toString());

var sendDATA= message.toString();

12.

1. const header = {
2. 'Content-Type': 'application/x-www-form-urlencoded',
3. 'Content-Length': Buffer.byteLength(sendDATA)

16. }

1. axios.post('https://ta.poliwangi.ac.id/~ti17136/api/ambildata',
2. sendDATA, header)
3. .then(res => {
4. console.log(res.data) 21. })
5. .catch(err => {
6. console.log(err.message) 24. })

25. })

**Gambar 4. 29** *Source Code Subscriber.js*

Pada gambar 4. 29 dapat dilihat bahwa *source code* diatas digunakan untuk *subscribe* dari hardware menggunakan broker *MQTT* dari test.mosquitto.org ditunjukkan pada baris ke-3 dengan *topic* ‘datasensor/tempatsampah’ pada baris ke-

6. Dari *topic* tersebut *hardware* bisa mengirimkan data atau *publish* data dengan format *topic* yang sama. Kemudian pesan disimpan dalam variabel sendDATA ditunjukkan pada baris ke-11 yang selanjutnya pesan tersebut dikirim ke *route* web *service* ditunjukkan pada baris ke-17. Pada baris ke-19 sampai baris ke-24 digunakan untuk melihat *respons* dari web *service* apakah mengalami kendala saat pengujian.



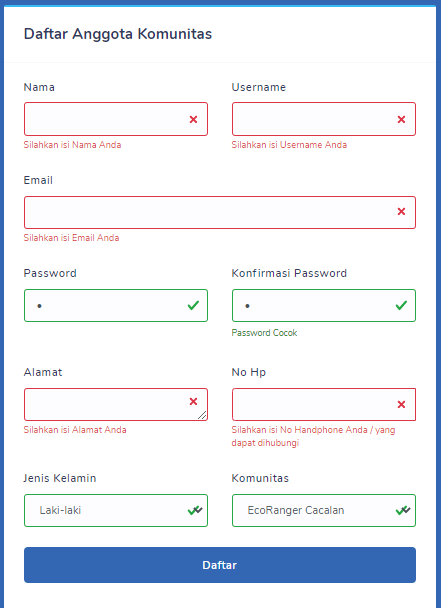
**Gambar 4. 30** Tampilan *Subscriber.js* saat dijalankan

Pada gambar 4.30 menampilkan saat *subscribe* data *MQTT* dari tempat sampah pintar dengan pesan yang dikirim sesuai dengan yang dibutuhkan oleh Aplikasi *website* ini. Data yang dibutuhkan berupa id tempat sampah, status tempat sampah, data sampah yang didapat saat masyarakat membuang sampah dalam keadaan benar atau salah dan kode *reward*. Data hanya bisa diterima jika *subscribe.js* dijalankan, untuk menjalankan *subscriber.js* pada sistem ini menggunakan netlify sebagai *sever-side Javascript*.

# Pengujian Sistem

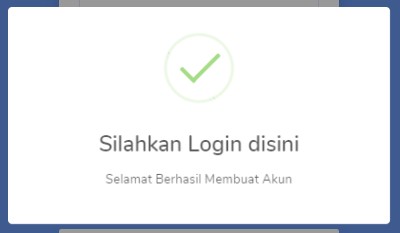
Pengujian (testing) merupakan suatu tahapan yang dianggap penting dalam siklus pembangunan perangkat lunak. Pengujian ini dilakukan untuk menjamin kualitas dan untuk mengetahui kelemahan dari perangkat lunak yang telah dibuat. Metode yang digunakan dalam pengujian ini yaitu metode *black box* testing. *Black box* testing adalah metode pengujian yang dilakukan hanya mengamati hasil eksekusi melalui aplikasi dengan memasukkan data untuk melihat respon sistem serta menguji fungsi dari aplikasi yang tidak bekerja dengan benar.

# Pengujian Menu Daftar Anggota Komunitas



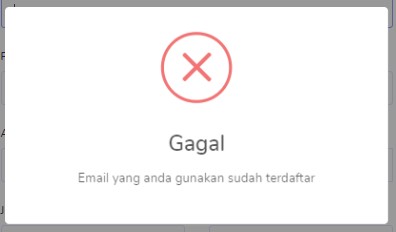
**Gambar 4. 31** Tampilan Jika *Form* Daftar Aggota Komunitas Kosong

Jika *Form* Daftar Anggota Komunitas Kosong atau dalam keadaan *form* tidak diisi maka akan muncul peringatan *form* tidak boleh kosong yang ditunjukkan pada Gambar 4.31.



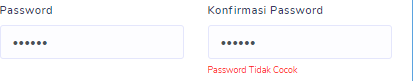
**Gambar 4. 32** Tampilan Daftar Anggota Komunitas Berhasil

Pada Gambar 4.32 menunjukkan pesan daftar anggota komunitas berhasil dan sistem akan langsung mengarahkan ke halaman *login* agar pengguna bisa langsung *login* dan masuk ke halaman beranda.



**Gambar 4. 33** Tampilan Daftar Anggota Komunitas Gagal

Pada Gambar 4. 33 menunjukkan pesan gagal saat *email* yang dimasukkan telah terdaftar di dalam sistem maka akan muncul pesan gagal mendaftar sebagai anggota komunitas dan harus menggunakan *email* yang berbeda.



**Gambar 4. 34** Tampilan Jika *Password* tidak sama

Pada Gambar 4. 34 pesan *password* tidak cocok karena saat memasukkan konfirmasi *password* berbeda dengan *password* yang telah dimasukkan sebelumnya.

* + 1. **Pengujian Halaman *Login***



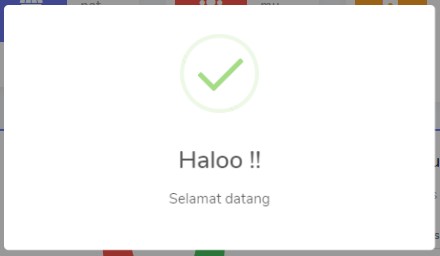
**Gambar 4. 35** Tampilan Jika *Form* Login Kosong

Sebelum pengguna akan melakukan aktivitas pada halaman beranda pada setiap *role-*nya masing-masing, tentunya harus *login* terlebih dahulu. Pengguna dapat *login* dengan *username* dan *password* yang sudah terdaftar dalam sistem. Pendaftaran anggota komunitas dilakukan oleh masyarakat melalui halaman daftar anggota komunitas. Pada Gambar 4. 35 menampilkan tampilan halaman login dengan kondisi pengguna tidak mengisi *form username* dan *password*.



**Gambar 4. 36** Tampilan *Login* Gagal

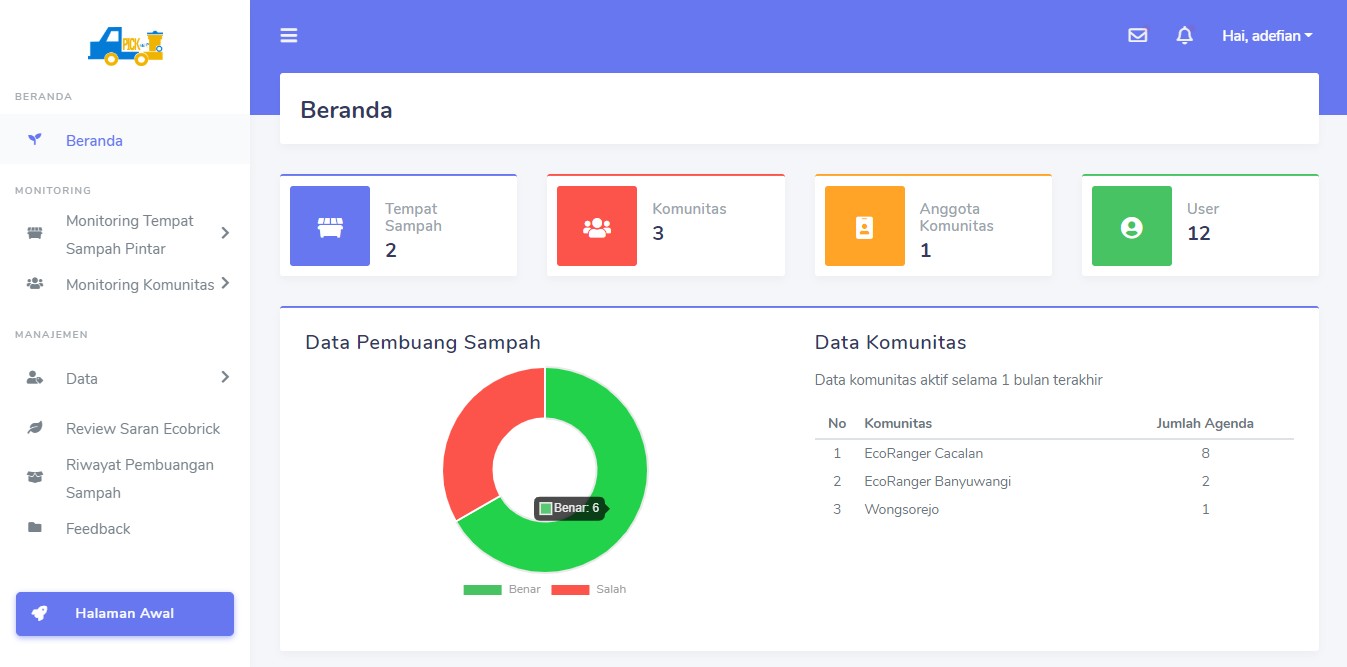
Apabila pengguna *login* dengan *username* dan *password* yang belum terdaftar maka akan muncul tampilan seperti pada Gambar 4. 36.



**Gambar 4. 37** Tampilan *Login* Berhasil

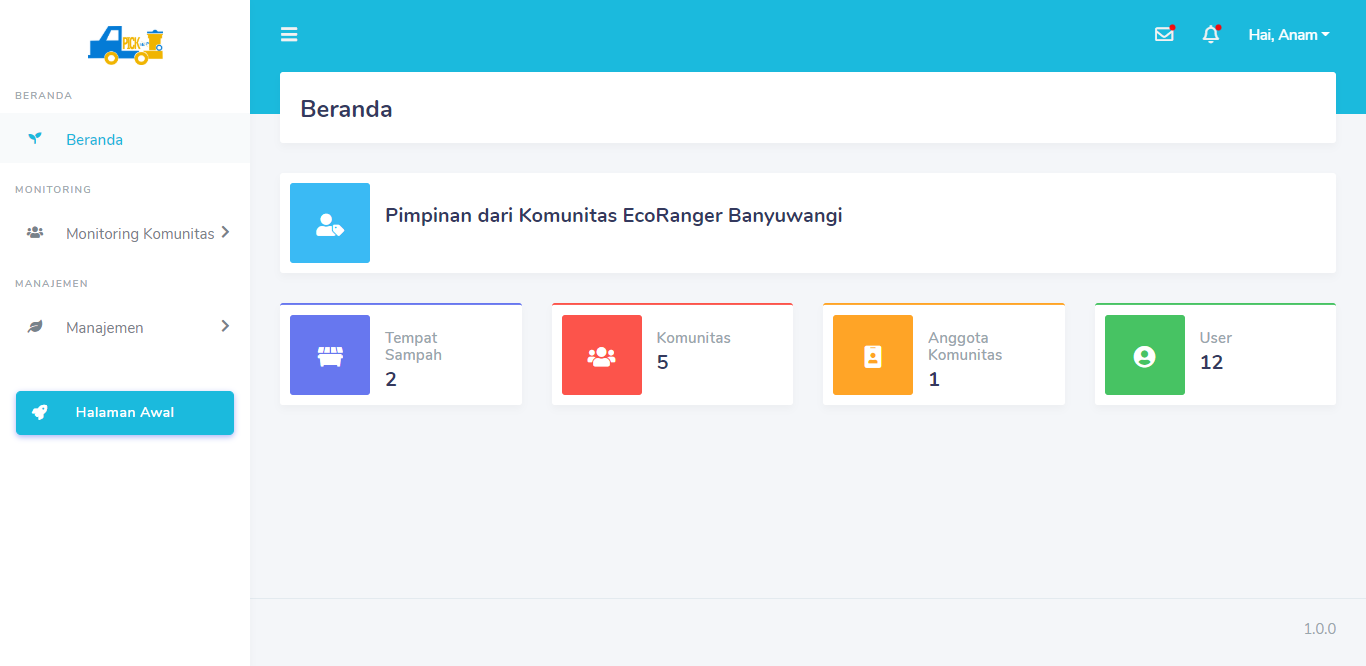
Apabila pengguna *login* dengan *username* dan *password* yang sudah terdaftar maka akan muncul tampilan seperti pada Gambar 4. 32.

# Pengujian Halaman Beranda *User*



**Gambar 4. 38** Halaman Beranda Pimpinan *EcoRanger*

Pada Gambar 4. 38 merupakan tampilan beranda pimpinan *EcoRanger* setelah pengguna melakukan *login* menggunakan *role* Pimpinan *EcoRanger* dengan *username* dan *password* yang sudah terdaftar pada *database*.



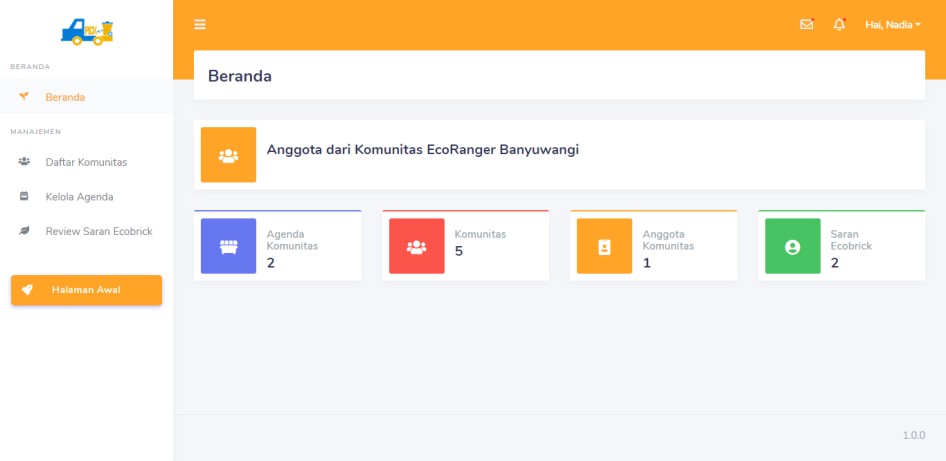
2

1

**Gambar 4. 39** Halaman Beranda Pimpinan Komunitas

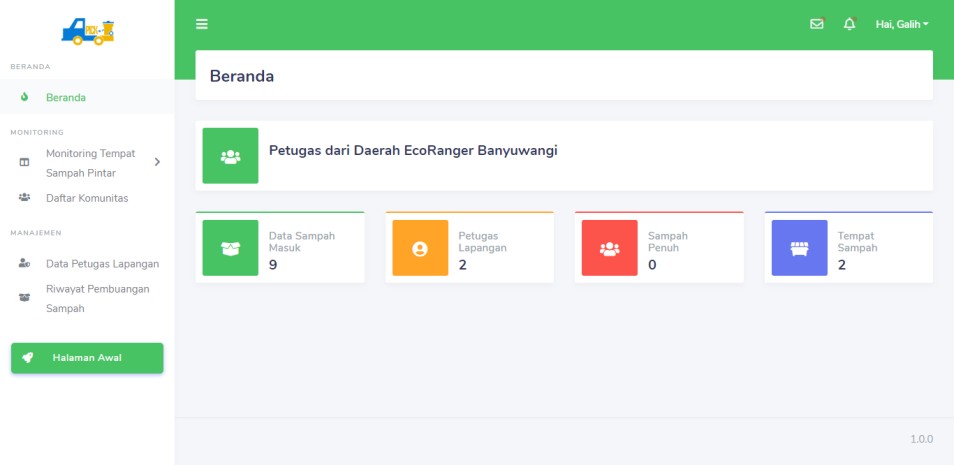
Pada Gambar 4. 39 merupakan halaman beranda pimpinan komunitas setelah pengguna melakukan *login* menggunakan *role* Pimpinan Komunitas dengan *username* dan *password* yang sudah terdaftar pada *database*. Berikut penjelasan dari Gambar 4. 39 :

* + - 1. Menampilkan *role* dari akun yang telah login beserta daerah komunitas yang telah dipilih.
      2. Menunjukkan beberapa jumlah data sesuai dengan nama data yang tertera.



**Gambar 4. 40** Halaman Beranda Anggota Komunitas

Pada Gambar 4. 40 merupakan halaman beranda anggota komunitas setelah pengguna melakukan login menggunakan role anggota komunitas dengan *username* dan *password* yang sudah terdaftar pada *database*.

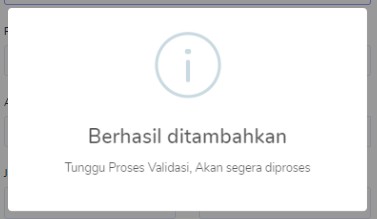


**Gambar 4. 41** Halaman Beranda Petugas Lapangan

pada Gambar 4. 41 merupakan halaman beranda petugas lapangan setelah pengguna melakukan *login* menggunakan *role* Petugas Lapangan dengan *username* dan *password* yang sudah terdaftar pada *database*.

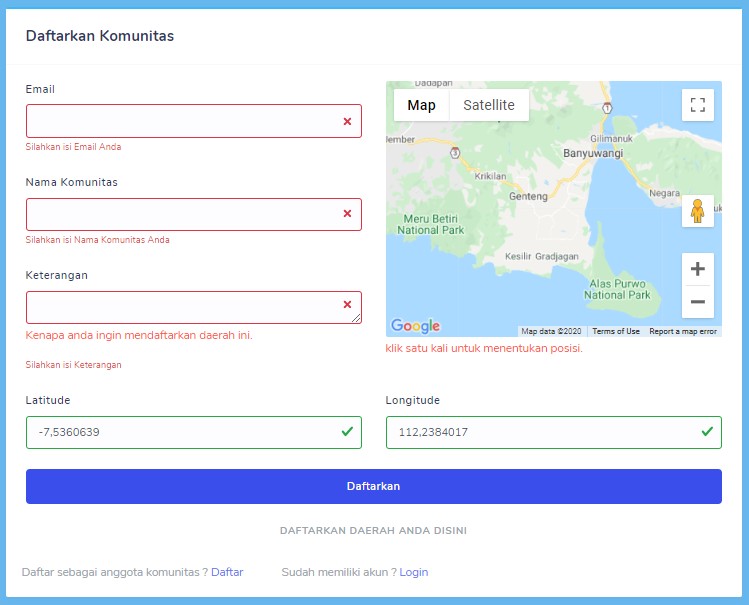
# Pengujian Menu Daftar Komunitas

Saat pengguna mendaftarkan komunitas pada halaman daftar komunitas dan data sudah dimasukkan pada semua *form* termasuk latitude dan longitude untuk menentukan lokasi komunitas yang akan didaftarkan maka akan muncul pesan *pop- up* seperti pada Gambar 4. 42.



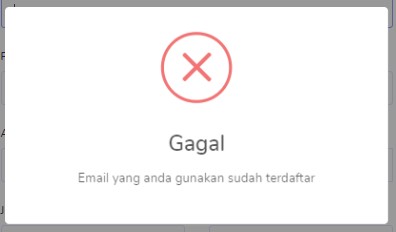
**Gambar 4. 42** Tampilan Berhasil Mendaftarkan Komunitas

Apabila pada saat pengguna mengisi *form* daftar komunitas dan terdapat *form* yang belum diisi maka sistem akan menampilkan pesan pemberitahuan form harus diisi. Pada Gambar 4. 43 merupakan tampilan *form* daftar komunitas ketika terdapat *form* yang belum diisi.



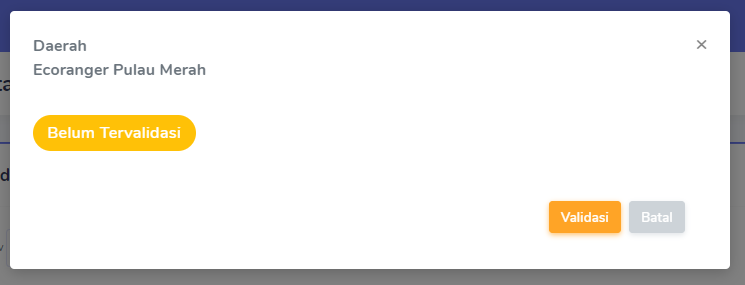
**Gambar 4. 43** Tampilan Jika Daftar Komunitas Kosong

Pada Gambar 4. 44 menunjukkan pesan gagal saat *email* yang dimasukkan telah terdaftar di dalam sistem maka akan muncul pesan gagal mendaftar komunitas dan harus menggunakan *email* yang berbeda.



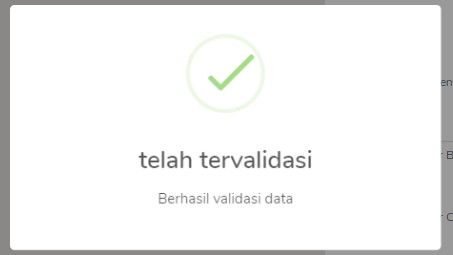
**Gambar 4. 44** Tampilan Daftar Komunitas Gagal

# Pengujian Menu Validasi Komunitas



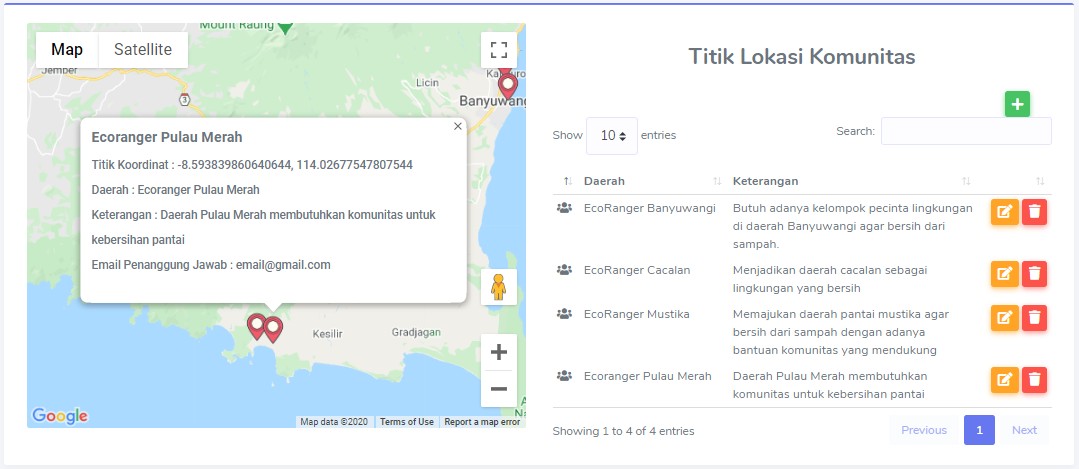
**Gambar 4. 45** Tampilan Validasi Komunitas

Pada Gambar 4. 45 merupakan tampilan *pop-up* pada halaman validasi ketika menekan tombol edit data pada halaman validasi, setelah menekan tombol validasi maka akan muncul pesan seperti pada Gambar 4. 41.



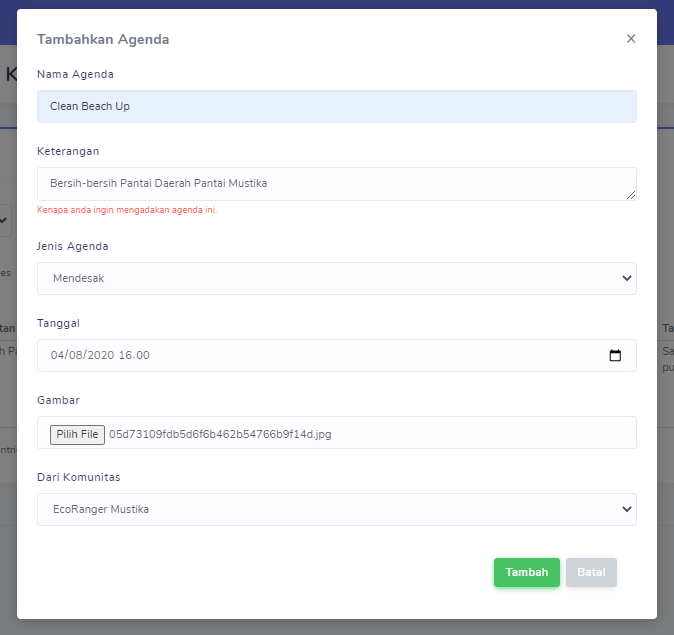
**Gambar 4. 46** Tampilan Komunitas Telah Tervalidasi

Jika komunitas sudah dilakukan validasi data oleh pimpinan *ecoranger*, maka sistem akan menampilkan pesan komunitas telah tervalidasi dan langsung diarahkan ke halaman daftar komunitas dengan menampilkan nama komunitas, keterangan dan juga peta dengan titik lokasi dimana komunitas berada. Pada Gambar 4. 47 merupakan tampilan daftar komunitas yang telah tervalidasi.



**Gambar 4. 47** Tampilan Daftar Komunitas Telah Tervalidasi

# Pengujian Halaman Kelola Agenda Komunitas



1

2

3

4

5

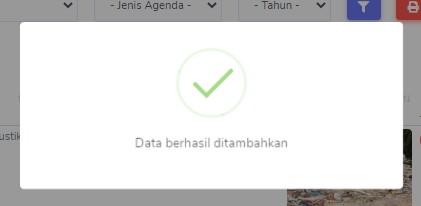
6

7

**Gambar 4. 48** Tampilan Tambahkan Agenda Komunitas

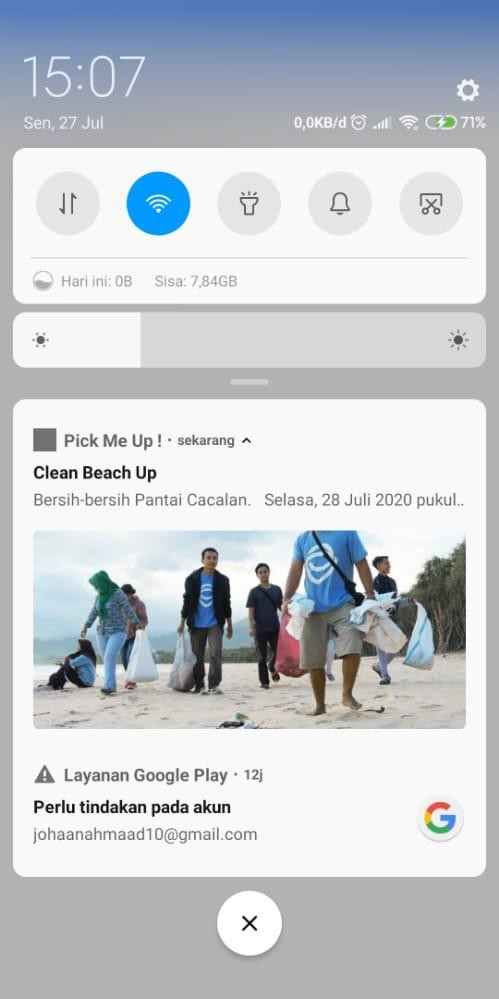
Untuk penjelasan tampilan tambahkan agenda dari Gambar 4. 48 dan Gambar 4. 49 sebagai berikut :

* + - 1. Kolom nama agenda untuk memberi identitas nama pada agenda.
      2. Kolom keterangan untuk memberi identitas keterangan pada agenda kenapa akan diadakanya kegiatan tersebut.
      3. Kolom jenis agenda digunakan untuk membedakan kegiatan menurut jenisnya.
      4. Kolom tanggal digunakan untuk menentukan kapan diadakanya kegiatan tersebut.
      5. Kolom gambar digunakan sebagai tambahan jika kegiatan tersebut memberikan gambar pada *form*.
      6. Kolom pilihan komunitas yang akan mengadakan kegiatan tersebut.
      7. Tombol tambah untuk menambahkan agenda dan tombol batal untuk membatalkan agenda.
      8. Tampilan *pop-up* saat data agenda berhasil ditambahkan.

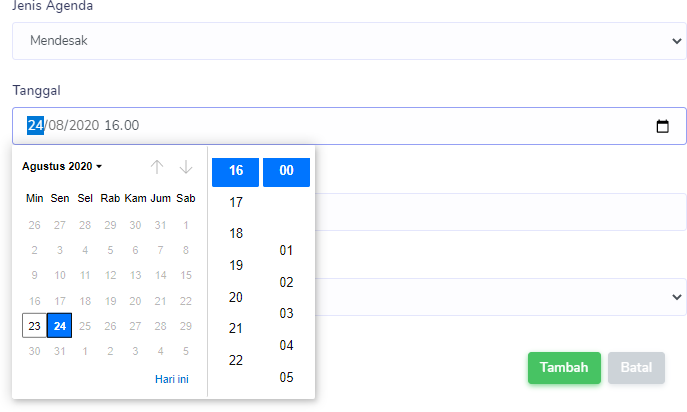


8

**Gambar 4. 49** Tampilan Data Agenda Berhasil Ditambahkan



**Gambar 4. 50** Tampilan Notifikasi Agenda Komunitas Pada Aplikasi Android Apabila agenda komunitas dalam sistem berhasil ditambahkan maka akan

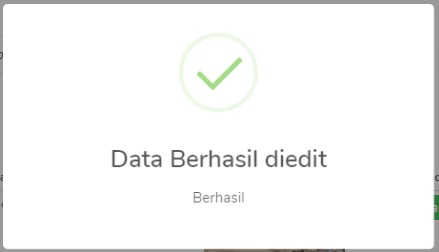
muncul pesan data berhasil ditambahkan dan juga akan menampilkan notifikasi pesan pada aplikasi android yang sudah teritegrasi dengan sistem ini, pesan yang dikirimkan berupa nama agenda komunitas serta keterangan diadakannya agenda dan gambar. Pada Gambar 4. 50 merupakan tampilan notifikasi pesan agenda komunitas pada aplikasi android.

**Gambar 4. 51** Tampilan Jika Jenis Agenda Mendesak

Pada Gambar 4. 51 merupakan tampilan *form* untuk menambahkan agenda komunitas yang jika *form* jenis agenda terpilih mendesak, maka *form* tanggal hanya bisa ditambahkan dari jam sekarang sampai satu hari kedepan. Jika *form* jenis agenda yang terpilih rutin maka pada *form* tanggal akan muncul pesan pilih hari dan tanggal untuk memulai agenda rutin selama 4 minggu ke depan ditunjukkan pada Gambar 4.52.

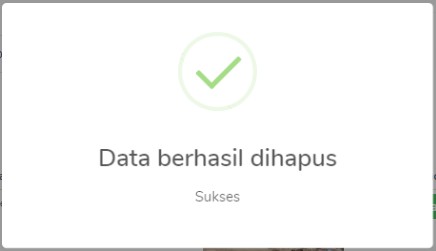
**Gambar 4. 52** Tampilan Jika Jenis Agenda Rutin

Jika berhasil merubah data agenda komunitas maka sistem akan menampilkan pesan pemberitahuan agenda berhasil diedit yang ditunjukkan Pada Gambar 4. 53.



**Gambar 4. 53** Tampilan Ketika Berhasil Merubah Agenda Komunitas.

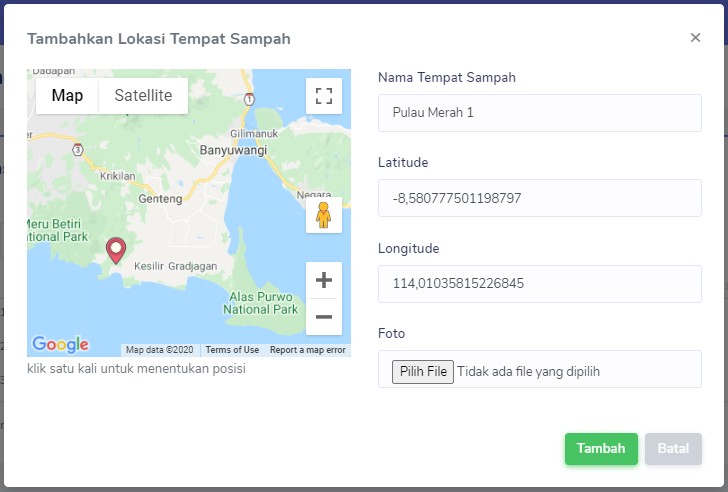
Jika berhasil menghapus data agenda komunitas maka sistem akan menampilkan pesan pemberitahuan agenda berhasil dihapus yang ditunjukkan Pada Gambar 4. 54.



**Gambar 4. 54** Tampilan Ketika Berhasil Menghapus Agenda Komunitas

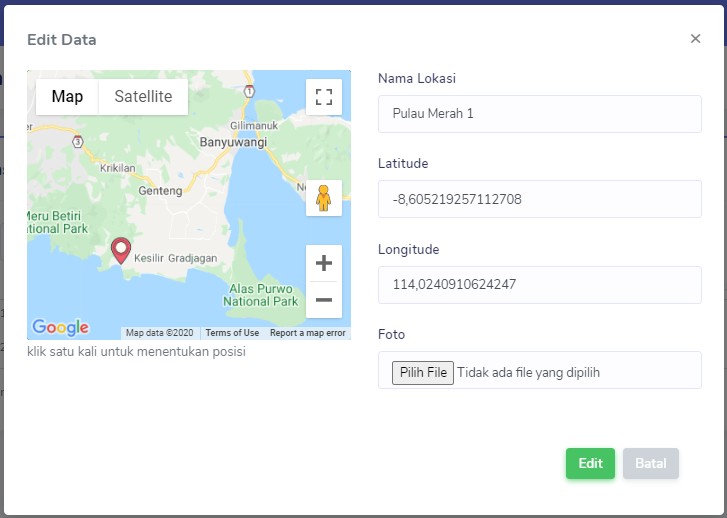
# Pengujian Halaman Kelola Tempat Sampah

Pengujian menu tambah lokasi tempat sampah dilakukan oleh *role* user pimpinan ecoranger dan petugas lapangan melalui *form* tambah lokasi tempat sampah yang berisi *form* nama tempat sampah, latitude dan longitude sebagai titik lokasi tempat sampah. Juga *form* foto sebagai foto kondisi tempat sampah. ditunjukkan pada Gambar 4. 55.

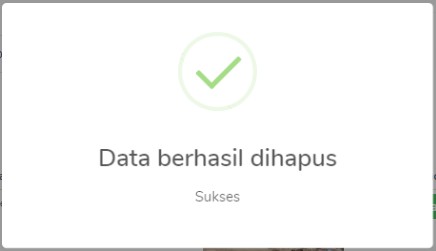


**Gambar 4. 55** Tampilan Tambahkan Lokasi Tempat Sampah

Pada Gambar 4. 56 menampilkan *form* edit data tempat sampah dengan mengisi *form* nama tempat sampah, latitude dan longitude sebagai titik lokasi tempat sampah. Juga *form* foto sebagai foto kondisi tempat sampah.

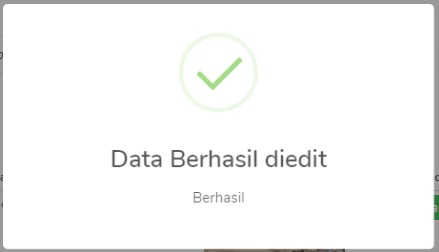


**Gambar 4. 56** Tampilan Edit Data Lokasi Tempat Sampah



**Gambar 4. 57** Tampilan Berhasil Menghapus Data Lokasi Tempat Sampah Jika data tempat sampah berhasil dihapus maka akan menampilkan pesan

pemberitahuan data tempat sampah berhasil dihapus. Pada Gambar 4.57 merupakan tampilan ketika data tempat sampah berhasil dihapus.



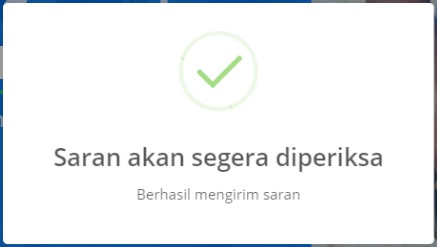
**Gambar 4. 58** Tampilan Berhasil Merubah Data Lokasi Tempat Sampah

Jika data tempat sampah berhasil diedit maka akan menampilkan pesan pemberitahuan data tempat sampah berhasil diedit. Pada Gambar 4.58 merupakan tampilan ketika data tempat sampah berhasil diedit.

# Pengujian Menu Tambah Saran Pengaplikasian *Ecobrick*

**Gambar 4. 59** Tampilan Jika *Form* Saran *Ecobrick* Kosong

Pengujian menu saran *Ecobrick* dilakukan oleh masyarakat (*public user*) melalui *form* saran *Ecobrick* yang berisi *form* nama lengkap, keterangan juga *form* foto sebagai foto saran. Pada Gambar 4. 55 merupakan tampilan *form* untuk menambahkan saran *Ecobrick*.



**Gambar 4. 60** Tampilan Berhasil Mengirim Saran *Ecobrick*

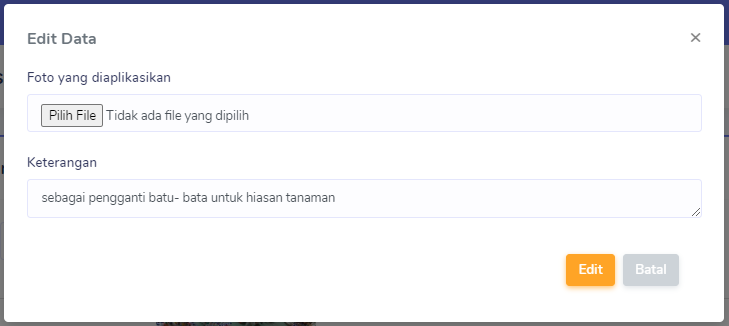
Jika saran *Ecobrick* berhasil ditambahkan maka sistem akan menampilkan pesan pemberitahuan saran akan segera diperiksa. Pada Gambar 4.60 merupakan tampilan ketika saran berhasil ditambahkan.

# Pengujian Halaman Ulasan Saran Ecobrick

**Gambar 4. 61** Tampilan Daftar Saran *Ecobrick*

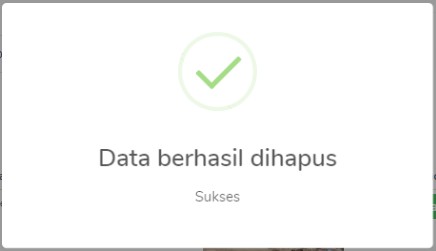
Menu daftar saran *Ecobrick* digunakan untuk menampilkan daftar saran yang telah ditambahkan oleh masyarakat (*public user*) dan untuk menampilkan status saran. Apabila status saran belum diaplikasikan maka sistem akan menampilkan tulisan warna kuning yang berarti saran tersebut belum diaplikasikan, jika status saran sudah / akan diaplikasikan maka sistem akan menampilkan tulisan warna hijau yang berarti gambar pada kolom foto yang diaplikasikan akan tampil

pada gambar *Ecobrick* pada halaman utama *Ecobrick.* Pada Gambar 4. 61 merupakan tampilan daftar saran *Ecobrick.*



**Gambar 4. 62** Tampilan *Form* Edit Data Saran *Ecobrick*

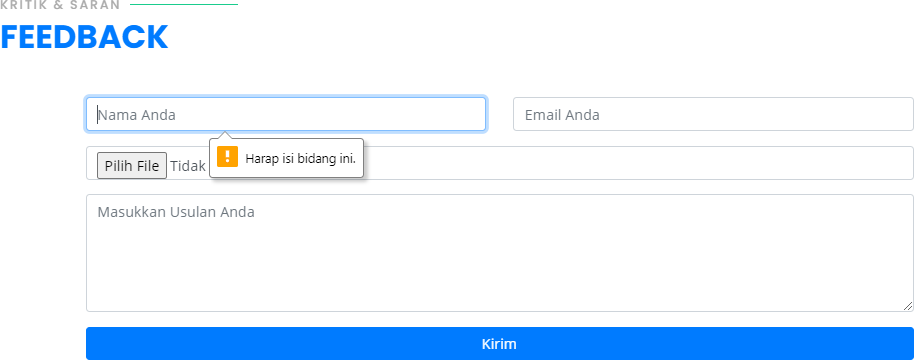
Pada Gambar 4. 62 merupakan tampilan *form* edit saran Ecobrick untuk menambahkan foto yang diaplikasikan dan merubah keterangan dari hasil produk *Ecobrick.*



**Gambar 4. 63** Tampilan Berhasil Menghapus Data Saran Ecobrick

Jika berhasil menghapus data saran *Ecobrick* maka sistem akan menampilkan pesan pemberitahuan saran *Ecobrick* berhasil dihapus yang ditunjukkan Pada Gambar 4. 63.

* + 1. **Pengujian Menu *Feedback***



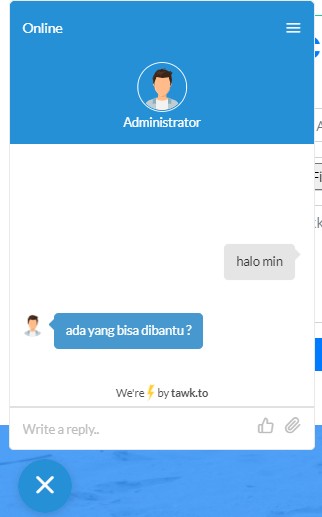
**Gambar 4. 64** Tampilan Jika *Form Feedback* Kosong

Pengujian *form feedback* dilakukan oleh masyarakat (*public user*) melalui *form feedback* yang berisi nama, email, *feedback* yang akan ditambahkan dan juga foto sebagai opsi tambahan jika *feedback* yang ditambahkan berupa gambar. Pada Gambar 4. 64 merupakan tampilan *form feedback* jika *form* tidak diisi maka akan muncul peringatan. Pada Gambar 4. 65 menunjukkan pesan berhasil menambahkan *feedback.*



**Gambar 4. 65** Tampilan Berhasil Menambahkan *Feedback*

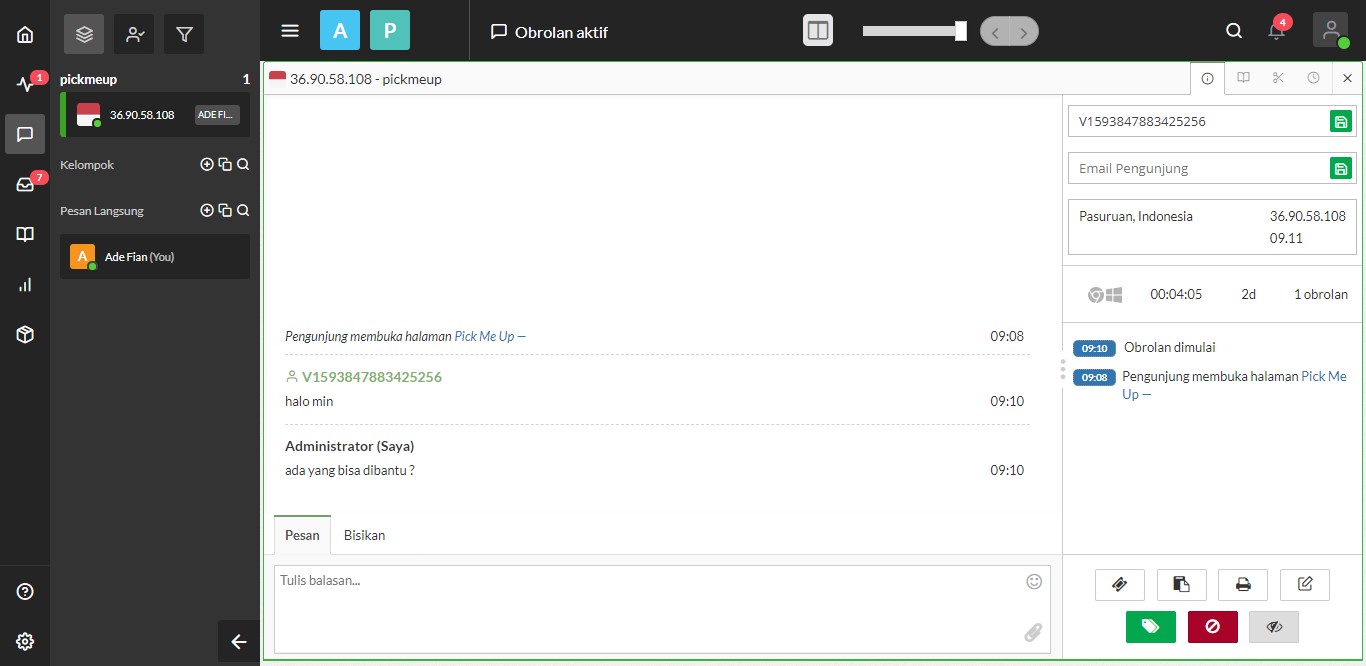
* + 1. **Pengujian Menu *Feedback* dengan *Live Chat Support***



**Gambar 4. 66** Tampilan Ketika Menambahkan *Feedback* dengan *Live Chat*

Pengujian menambahkan *feedback* dengan *live chat support by tawk.to* yang dilakukan oleh masyarakat (*public user*) melalui tombol *feedback* yang berada pada posisi pojok kiri bawah di *website*. Pada Gambar 4. 66 merupakan tampilan *interface feedback* dengan *live chat*. Pada Gambar 4. 67 menunjukkan tampilan

dashboard *tawk.to* yang digunakan oleh petugas untuk menjawab langsung

*feedback* dari masyarakat (*public user*).

**Gambar 4. 67** Tampilan Dashboard *Tawk.To*

* + 1. **Tabel Pengujian *Black Box***

Berikut penjelasan singkat mengenai pengujian menggunakan metode

*blackbox testing* yang akan dijelaskan pada Tabel 4.2.

**Tabel 4. 2** Tabel Pengujian *Blackbox*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | ***Features*** | **Skenario**  **Pengujian** | ***Test Case*** | **Hasil Pengujian** | **Status** | **Bukti**  **Pengujian** |
| 1. | Daftar Sebagai Anggota Komunitas | Mendaftar menggunakan akun yang belum terdaftar | Nama : Ade Username : adefian Email :  [ade@gmail.com](mailto:ade@gmail.com) Password : 123456  Konfirmasi *Password* : 123456  alamat : Banyuwangi No Hp : 08123456789  Jenis Kelamin : Laki - Laki komunitas: *EcoRanger*  Banyuwangi | Sistem langsung mengarahkan pada halaman *login* dengan muncul tampilan Silahkan *Login* disini | Valid | Gambar 4. 32 |
| Mendaftar menggunakan email yang sudah terdaftar | Email : pimpinan@ gmail.com | Sistem menampilkan pesan pemberitahuan email sudah  terdaftar | Valid | Gambar 4. 33 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | Jika *Form* daftar anggota komunitas kosong | *-* | Sistem menampilkan pesan pemberitahuan  *form* harus diisi | Valid | Gambar 4. 31 |
| Jika Konfirmasi *password* yang dimasukkan tidak sama dengan  *password* | *Password* : 123 Konfirmasi *Password* : 12345 | Sistem menampilkan pesan pemberitahuan *password* tidak sama | Valid | Gambar 4.34 |
| 2. | *Login* | Jika form *email* dan *password* kosong | *-* | Sistem menampilkan pesan pemberitahuan  *form* harus diisi | Valid | Gambar 4. 35 |
| *Login* menggunakan akun yang sudah terdaftar | *Username* : [pimpinan](mailto:velandaaden77@gmail.com) *Password* : rahasia | Pengguna akan masuk ke dalam dashboard dengan pesan  sukses *login* | Valid | Gambar 4. 37 |
| *Login* menggunakan *email* dan *password* tidak terdaftar | *Email* : [pimpinan](mailto:velandaaden77@gmail.com) Password : 123456789 | Sistem menampilkan pesan pemberitahuan *usernmae* dan  *password* salah | Valid | Gambar 4. 36 |
| 3. | Halaman Beranda *User* | *Login* menggunakan akun dengan *role* Pimpinan  *EcoRanger* | *Username* : pimpinan *Password* : rahasia | Sistem akan mengarahkan ke halaman beranda Pimpinan  *EcoRanger* | Valid | Gambar 4. 38 |
| *Login* menggunakan akun dengan  *role* Pimpinan Komunitas | *Username* : haris *Password* : rahasia | Sistem akan mengarahkan ke halaman beranda  Pimpinan Komunitas | Valid | Gambar 4. 39 |
| *Login* menggunakan akun dengan  *role* anggota komunitas | *Username* : adefian *Password* : 123456 | Sistem akan mengarahkan ke halaman beranda  anggota komunitas | Valid | Gambar 4. 40 |
| *Login* menggunakan akun dengan  *role* petugas lapangan | *Username* : petugas *Password* : password | Sistem akan mengarahkan ke halaman beranda petugas lapangan | Valid | Gambar 4. 41 |
| 4. | Daftar Komunitas | Daftar komunitas menggunakan *email* yang belum terdaftar | *Email :* email@gmail. com  Nama Komunitas : Ecoranger Pulau Merah Keterangan: | Sistem menampilkan pesan pemberitahuan Komunitas Berhasil ditambahkan,  Tunggu Proses Validasi | Valid | Gambar 4. 42 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  | Daerah Pulau Merah Bersih |  |  |  |
| Jika *form* daftar komunitas kosong | - | Sistem menampilkan pesan pemberitahuan  *form* harus diisi | Valid | Gambar 4. 43 |
| *Login* menggunakan *email* yang sudah terdaftar | *Email* : [ade@gmail.com](mailto:ade@gmail.com) | Sistem menampilkan pesan pemberitahuan email sudah  terdaftar | Valid | Gambar 4. 44 |
| 5. | Validasi Komunitas | Jika akan melakukan validasi data | Komunitas : *EcoRanger* Pulau Merah *Email* : email@gmail.  com | Sistem akan menampilkan tampilan untuk validasi data | Valid | Gambar 4. 45 |
| Jika komunitas akan dilakukan validasi | Komunitas : *EcoRanger* Pulau Merah *Email* : email@gmail.  com | Sistem akan menampilkan pesan pemberitahuan berhasil validasi  data | Valid | Gambar 4. 46 |
| Jika komunitas sudah divalidasi | Komunitas : *EcoRanger* Pulau Merah *email* penanggung jawab :  email@gmail. com | Sistem akan menampilkan komunitas pada daftar komunitas yang sudah tervalidasi dalam sistem | Valid | Gambar 4. 47 |
| 6. | Kelola Agenda Komunitas | Jika akan menambahkan agenda komunitas | Menekan tombol tambah | Sistem akan menampilkan tampilan *form* untuk  menambahkan agenda | Valid | Gambar 4. 48 |
| Menambahkan agenda komunitas baru | Nama agenda : Clean Beach Up tanggal : 04/08/2020  16:00 | Sistem akan menampilkan pesan pemberitahuan data berhasil  ditambahkan | Valid | Gambar 4. 49 |
| Menampilkan notifikasi ke aplikasi  android | Menambahkan agenda | Sistem akan menampilkan notifikasi ke  aplikasi android | Valid | Gambar 4. 50 |
| Menambahkan agenda komunitas dengan pilihan jenis agenda mendesak | Tanggal : 18/07/2020  16:00 | Pengguna hanya bisa menambahkan agenda komunitas dengan tanggal selama 1 hari  kedepan | Valid | Gambar 4. 51 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | Menambahkan agenda komunitas dengan pilihan jenis agenda rutin | Tanggal : 18/07/2020  16:00 | Sistem akan menampilkan pesan pemberitahuan agenda akan ditambahkan rutin selama 4  minggu kedepan | Valid | Gambar 4. 52 |
| Merubah data agenda komunitas | Nama agenda : Clean Beach Up tanggal : 18/07/2020 | Sistem akan menampilkan pesan pemberitahuan data berhasil  diedit | Valid | Gambar 4. 53 |
| Menghapus data agenda komunitas | - | Sistem akan menampilkan pesan pemberitahuan data berhasil  dihapus | Valid | Gambar 4. 54 |
| 7. | Kelola Tempat Sampah | Menambahkan lokasi tempat sampah | Nama : Tempat Sampah 1  letak : latitude, longitude | Sistem akan menampilkan *form* untuk menambahkan lokasi tempat  sampah | Valid | Gambar 4. 55 |
| Merubah data lokasi tempat sampah | Nama : Pulau Merah 1  letak : latitude, longitude | Sistem akan menampilkan *form* untuk merubah data lokasi tempat  sampah | Valid | Gambar 4. 56 |
| Menghapus data tempat sampah | - | Sistem akan menampilakan pesan pemberitahuan  data bershasil dihapus | Valid | Gambar 4. 57 |
| Merubah status tempat sampah | Status : kosong | Sistem akan menampilkan pesan pemberitahuan  data berhasil diubah | Valid | Gambar 4. 58 |
| 8. | Tambah Saran Pengaplika sian *Ecobrick* | Menambahkan saran Pengaplikasian *Ecobrick* baru | Nama Lengkap : Masyarakat Foto : 123.jpg keterangan : sebagai pengganti batu-  bata untuk hiasan tanaman | Sistem akan menampilkan pesan pemberitahuan Saran akan segera diperiksa,  berhasil mengirim saran | Valid | Gambar 4. 60 |
| Jika *form* saran Pengaplikasian *Ecobrick* kosong | *-* | Sistem akan menampilkan pesan pemberitahuan  *form* harus diisi | Valid | Gambar 4. 59 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 9. | Mengulas Saran Pengaplika sian *Ecobrick* | Melihat saran Pengaplikasian *Ecobrick* | Nama Lengkap : Masyarakat Foto : 123.jpg keterangan : sebagai pengganti batu- bata untuk  hiasan tanaman | Sistem akan menampilkan daftar semua saran pengaplikasian *Ecobrick* | Valid | Gambar 4. 61 |
| Mengaplikasika n saran | Foto yang diaplikasikan : baru.jpg | Sistem akan menampilakan *form* untuk menambahkan gambar sebagai foto saran yang akan/sudah  diaplikasikan | Valid | Gambar 4. 62 |
| Menghapus saran | - | Sistem akan menampilkan pesan pemberitahuan  data berhasil dihapus | Valid | Gambar 4. 63 |
| 10. | *Feedback* | Menambahkan *feedback* berupa kritik dan saran | Nama : Ade Fian  email : [ade@gmail.com](mailto:ade@gmail.com) foto:12.jpg masukkan usulan : lebih sering mengadakan  agenda komunitas | Sistem akan menampilkan pesan pemberitahuan terima kasih sudah memberikan masukan | Valid | Gambar 4. 65 |
| Jika *form feedback* kosong | - | Sistem akan menampilkan pesan pemberitahuan  *form* harus diisi | Valid | Gambar 4. 64 |
| 11. | *Feedback* dengan *live chat support* | Pengguna (Masyarakat) mengirimkan *feedback* kepada petugas | Pesan dari Masyarakat : halo min | Sistem akan mengirimkan pesan ke petugas lewat *interface* web aplikasi ke pihak ketiga  *twak.wo* | Valid | Gambar 4. 66 |
| Admin (Petugas lapangan) membalas *feedback*  kepada masyarakat | Membalas pesan dari masyarakat  : ada yang bisa dibantu ? | Sistem akan mengirimkan pesan ke masyarakat dari web aplikasi pihak ketiga  *tawk.to* | Valid | Gambar 4. 67 |

# BAB 5 PENUTUP

# Kesimpulan

Berdasarkan perancangan dan analisa yang dibuat pada Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah di Pulau Merah Berbasis Web, maka dapat diambil beberapa kesimpulan:

* + 1. Aplikasi Manajemen Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah di Pulau Merah dibuat dengan Bahasa Pemrograman *PHP* disisi server dan sistem ini menggunakan *framework laravel 7* dengan basisdata *MySQL, framework Bootstrap 4* sebagai tampilan *Interface Website* serta *Node.Js* untuk integrasi ke hardware (tempat sampah).
    2. Sitem ini dapat melakukan *monitoring* komunitas dengan mengetahui posisi komunitas dan mengetahui daftar agenda yang akan dilaksanakan oleh komunitas yang ada dalam sistem
    3. Sistem ini dapat melakukan *monitoring* tempat sampah dengan mengetahui posisi dan konfidi tempat sampah dalam keadaan penuh atau masih kosong
    4. Sitem ini dapat memberikan notifikasi serta informasi kondisi tempat sampah dalam keadaan penuh atau masih kosong
    5. Sistem ini dapat membantu komunitas dalam menginformasikan agenda yang akan dilaksanakan ke media Aplikasi *website* dan android
    6. Sistem ini dapat melakukan *monitoring* perilaku masyarakat dalam membuang sampah dengan tepat atau masih belum bisa membedakan jenis- jenis sampah. Dalam aplikasi *website* ini menyediakan grafik untuk melihat perilaku masyarakat dalam membuang sampah

# Saran

Saran yang diberikan untuk pengembangan Manajemen Komunitas Peduli Lingkungan dan Pengelolaan Sampah di Pulau Merah Berbasis Web antara lain :

* + 1. Sistem ini dapat dikembangkan dengan menambahkan fitur pengusulan agenda komunitas oleh anggota komunitas
    2. Sistem ini dapat dikembangkan kembali dengan penambahan fitur lupa *password* agar pengguna tidak mendaftar ulang untuk masuk dalam sistem

*-- Halaman ini sengaja dikosongkan --*

# DAFTAR PUSTAKA

Andriastuti, Bella Tri, Arifin Fitria, Laili. 2019. Jurnal Teknologi Lingkungan Lahan Basah. *Pontensi Ecobrick Dalam Mengurangi Sampah Plastik Rumah Tangga Di Kecamatan Pontianak Barat*, 56, 59.

Astina, A. 2016. *Live chat Alternative Gunakan Tawk.to 100% Gratis & Kaya Fitur*. [internet]. [diunduh 6 Agustus 2020] Tersedia pada: andiastiana.com: https://andiastina.com/reviews/tawk-to-alternative-live-chat-gratis/

Harisantyo, B., Nugraha, L. S., Prasetyawan, N., Nugraha, S. N., & Sulaiman. 2015. *Makalah Pemrograman Berbasis Objek* Diagram *Activity.* Depok: Universitas Gunadrama Fakultas Teknologi Industri Teknik Informatika.

Jaya, T. S. 2018. Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis (Studi Kasus: Kantor Digital Politeknik Negeri Lampung). *Jurnal* Informatika*: Jurnal Pengembangan IT (JPIT), 03*(02), 45-46.

Kosasi, S., & Yuliani, I. A. 2015. Penerapan Rapid Application Development Pada Sistem Penjualan Sepeda Online. *Jurnal Simetris, 6*(1), 28-29.

Luthfi, M. M. 2016. Pengertian dan Keunggulan Framework Laravel. [internet]. [diunduh 2 Januari 2020] Tersedia pada: idcloudhost.com: https://[www.google.com/amp/s/idcloudhost.com/pengertian-dan-](http://www.google.com/amp/s/idcloudhost.com/pengertian-dan-) keunggulan-framework-laravel/amp/

Mustaqbal, M., Firdaus, R. F., & Rahmadi, H. 2015. Pengujian Aplikasi Menggunakan *Black box* Testing Boundary Value Analysis . *Jurnal Ilmiah Teknologi* Informasi *Terapan, 1*(3), 34.

Nelles, M., Morscheck, G., Lemke, A., & Naas, A. E. 2017. Applied Bioengineering Innovations and Future Directions. In T. Yoshida, *Treatment of Solid Waste* (p. 447). Osaka: Wiley-VCH.

Nugroho, A. 2017. Jurnal Pemikiran Sosiologi Volume 4 No.1. *Komunitas Muda Urban Mengelola Sampah: Kajian Partisiparatoris Gerakan Peduli Sampah Nasional di Kota Yogyakarta*.

Pradana, A. K., & Izzah, A. 2017. Sistem *Monitoring* Keuangan Perusahaan Kreditur. *Jurnal* Informatika *& Multimedia, 9*(01), 23-28.

Qalbi, I. A. 2014. *Konsep Dasar Basis Data.* Samarinda: Teknik Informasi, Komunikasi dan Komputer Universitas Mulawarman .

Ramadhan, F., & Latifah, F. 2018. *Perancangan Komunitas Pecinta Kucing Dengan Metode Waterfall Berbasis* Web.

Rembulan, A. D. 2015. *Makalah Unified Modeling Language (UML).* Cirebon: Stikom Poltek.

Rochman, H. A., Primananda, R., & Nurwasito, H. 2017. Sistem Kendali Berbasis Mikrokont*role*r Menggunakan Protokol *MQTT* pada Smarthome. *Jurnal Pengembangan Teknologi* Informasi *dan Ilmu Komputer*, 477.

Rukmana, A., & Desiyani, I. D. 2017. *Metodologi Dan Metode Rapid Application Development (RAD).* Sumedang: Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Sumedang.

Safitri, P. A., Purba, W. S., & Zulkifli, M. 2018. *STATISTIK LINGKUNGAN HIDUP INDONESIA 2018.* jakarta: Badan Pusat Statistik/BPS–Statistics Indonesia.

Santi, M. E. 2018. *Ecobrick* Solusi Cerdas dan Praktis Untuk Pengelolaan Sampah Plastik. 3-5.

Sucipto. 2017. Perancangan Active *Database* System pada Sistem Informasi Pelayanan Harga Pasar. *Jurnal Intensif, 1*(1), 35-36.

Suryani, A. S. 2016. *Persepsi Masyarakat dan Analisis Willingness to Pay terhadap Kebijakan Kantong Plastik Berbayar*, 360.

Tua, B. 2018. *Node.js HandBook.* Medan.

Wijaya, C., & Rifa'i, M. 2016. *Dasar-Dasar Manajemen Mengoptimalkan Pengelolaan Organisasi Secara Efektif dan Efisien* (1nd ed.). Medan: Perdana *Publish*ing.

Wulandari, E. 2018. Sistem Informasi Geografis Pemetaan Titik Lokasi Pembuangan Sampah Kota Pekanbaru.

Yasin, K. 2019. Cara Menggunakan Bootstrap 4 Untuk Membuat *Website*. [internet]. [diunduh 17 Oktober 2019] Tersedia pada: Niaga Hoster: https://[www.niagahoster.co.id/blog/cara-menggunakan-bootstrap/](http://www.niagahoster.co.id/blog/cara-menggunakan-bootstrap/)

Yusmiarti, K. 2016. Perancangan Sistem Distribusi Produk Teh Hitam Berbasis Web Pada PTPN VII Gunung Dempo Pagar Alam. *Jurnal* Informatika*, 4*(2), 3-4.