W

;/--n

**TUGAS AKHIR – KI1502**

**RANCANG BANGUN APLIKASI *ALERT SYSTEM* POTENSI BANJIR UNTUK RUMAH POMPA**

**MARDIANA SEKARSARI**

**NRP 5113100051**

**Dosen Pembimbing**

**Rizky Januar Akbar, S.Kom., M.Eng.**

**Adhatus Solichah Ahmadiyah, S.Kom., M.Sc.**

**JURUSAN TEKNIK INFORMATIKA**

**Fakultas Teknologi Informasi**

**Institut Teknologi Sepuluh Nopember**

**Surabaya 2017**

*[Halaman ini sengaja dikosongkan]*

**

**TUGAS AKHIR – KI1502**

**RANCANG BANGUN APLIKASI *ALERT SYSTEM* POTENSI BANJIR UNTUK RUMAH POMPA**

**MARDIANA SEKARSARI**

**NRP 5113100051**

**Dosen Pembimbing**

**Rizky Januar Akbar, S.Kom., M.Eng.**

**Adhatus Solichah Ahmadiyah, S.Kom., M.Sc.**

**JURUSAN TEKNIK INFORMATIKA**

**Fakultas Teknologi Informasi**

**Institut Teknologi Sepuluh Nopember**

**Surabaya 2017**

*[Halaman ini sengaja dikosongkan]*

****

**UNDERGRADUATE THESES – KI1502**

**DESIGN AND IMPLEMENTATION OF FLOOD POTENTIAL ALERT SYSTEM FOR RUMAH POMPA**

**MARDIANA SEKARSARI**

**NRP 5113100051**

**Supervisors**

**Rizky Januar Akbar, S.Kom., M.Eng.**

**Adhatus Solichah Ahmadiyah, S.Kom., M.Sc.**

**DEPARTMENT OF INFORMATICS**

**Faculty of Information Technology**

**Institut Teknologi Sepuluh Nopember**

**Surabaya 2017**

*[Halaman ini sengaja dikosongkan]*

# 

# LEMBAR PENGESAHAN

[Halaman ini sengaja dikosongkan]

**RANCANG BANGUN APLIKASI *ALERT SYSTEM* POTENSI BANJIR UNTUK RUMAH POMPA**

**Nama Mahasiswa : Mardiana Sekarsari**

**NRP : 5113 100 051**

**Jurusan : Teknik Informatika FTIf-ITS**

**Dosen Pembimbing 1 : Rizky Januar Akbar, S.Kom., M.Eng.**

**Dosen Pembimbing 2 : Adhatus Solichah Ahmadiyah, S.Kom., M.Sc.**

# Abstrak

***Kata kunci:***

# KATA PENGANTAR



Alhamdulillahirabbil’alamin, segala puji bagi Allah SWT, yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul ***“*RANCANG BANGUN APLIKASI *ALERT SYSTEM* POTENSI BANJIR UNTUK RUMAH POMPA*”*.**

Pengerjaan tugas akhir ini merupakan suatu kesempatan yang sangat baik bagi penulis. Dengan pengerjaan tugas akhir ini, penulis bisa belajar lebih banyak untuk memperdalam dan meningkatkan apa yang telah didapatkan penulis selama menempuh perkuliahan di Teknik Informatika ITS.

Selesainya tugas akhir ini tidak lepas dari bantuan dan dukungan beberapa pihak, sehingga pada kesempatan ini penulis mengucapkan syukur dan terima kasih kepada:

1. Allah SWT dan Nabi Muhammad SAW.
2. Ibunda, Ayahanda, dan ketiga Kakak, yang selalu mendoakan penulis dan mendukung setiap pilihan yang penulis ambil.
3. Bapak Rizky Januar Akbar, S.Kom., M.Eng. selaku pembimbing I yang selalu memberikan motivasi dan membimbing penulis selama pengerjaan tugas akhir.
4. Ibu Adhatus Solichah Ahmadiyah, S.Kom., M.Sc. selaku pembimbing II yang selama ini telah membantu dan membimbing penulis selama pengerjaan tugas akhir.
5. Bapak Ridho Rahman Hariadi, S.Kom.,M.Sc. yang telah membantu penulis dalam membuat sensor level air
6. Bapak Dr.Eng Darlis Herumurti, S.Kom.,M.Kom. selaku Kepala Jurusan Teknik Informatika ITS, Bapak Dr. Radityo Anggoro, S.Kom.,M.Sc. selaku koordinator TA, dan segenap dosen Teknik Informatika yang telah banyak memberikan ilmu kepada penulis.
7. Raras, Aby, dan Bilfash yang selalu memberikan dukungan dan semangat kepada penulis.
8. Dwi, Dewi, dan Tyas yang telah menjadi tempat berkeluh kesah dan selalu memberikan semangat kepada penulis.
9. Teman-teman Administrator Laboratorium RPL yang telah menjadi keluarga selama penulis menimba illmu di Teknik Informatika ITS.
10. Teman-teman Staf Departemen Dalam Negeri HMTC Berkarya 2014/2015 dan teman-teman Pengurus Harian KMI Ramah Bertauhid 2015/2016.
11. Teman-teman angkatan 2013 yang telah membantu, berbagi ilmu, menjaga kebersamaan, dan memberi motivasi kepada penulis.
12. Serta semua pihak yang yang telah turut membantu penulis dalam menyelesaikan tugas akhir ini.

Penulis menyadari bahwa tugas akhir ini masih memiliki banyak kekurangan, sehingga dengan kerendahan hati, penulis mengharapkan kritik dan saran dari pembaca untuk perbaikan ke depannya.

Surabaya, Juni 2017

Mardiana Sekarsari

# **DAFTAR ISI**

[LEMBAR PENGESAHAN vi](#_Toc480229204)

[Abstrak viii](#_Toc480229205)

[KATA PENGANTAR x](#_Toc480229206)

[DAFTAR ISI xii](#_Toc480229207)

[DAFTAR GAMBAR xv](#_Toc480229208)

[DAFTAR TABEL xxi](#_Toc480229209)

[DAFTAR KODE SUMBER xxv](#_Toc480229210)

[1 BAB I PENDAHULUAN 27](#_Toc480229211)

[1.1. Latar Belakang 27](#_Toc480229212)

[1.2. Rumusan Masalah 28](#_Toc480229213)

[1.3. Batasan Masalah 28](#_Toc480229214)

[1.4. Tujuan 29](#_Toc480229215)

[1.5. Manfaat 29](#_Toc480229216)

[1.6. Metodologi Pembuatan Tugas Akhir 29](#_Toc480229217)

[1.7. Sistematika Penulisan Laporan Tugas Akhir 32](#_Toc480229218)

[2 BAB II TINJAUAN PUSTAKA 35](#_Toc480229221)

[2.1. Banjir 35](#_Toc480229222)

[2.2. Sistem Peringatan (*Alert System*) 36](#_Toc480229223)

[2.3. Rumah Pompa 36](#_Toc480229224)

[2.4. Android 37](#_Toc480229225)

[2.5. REST Web Service 38](#_Toc480229226)

[2.6. PostgreSQL 40](#_Toc480229227)

[2.7. Volley 40](#_Toc480229228)

[2.8. Sensor Ketinggian Air 40](#_Toc480229229)

[2.9. *Weather Underground API* 40](#_Toc480229230)

[3 BAB III ANALISIS DAN PERANCANGAN SISTEM 41](#_Toc480229231)

[3.1. Analisis 41](#_Toc480229232)

[3.1.1. Analisis Permasalahan 41](#_Toc480229233)

[3.1.2. Deskripsi Umum Sistem 42](#_Toc480229234)

[3.1.3. Spesifikasi Kebutuhan Perangkat Lunak 43](#_Toc480229235)

[3.2. Perancangan 67](#_Toc480229236)

[3.2.1 Lingkungan Perancangan Perangkat Lunak 67](#_Toc480229237)

[3.2.2 Perancangan Arsitektur Sistem 68](#_Toc480229238)

[3.2.3 Perancangan Diagram Kelas 68](#_Toc480229239)

[3.2.4 Perancangan Struktur Data 68](#_Toc480229240)

[3.2.5 Perancangan Antarmuka Pengguna 81](#_Toc480229241)

[4. BAB IV IMPLEMENTASI 106](#_Toc480229242)

[4.1 Lingkungan Implementasi Perangkat Lunak 106](#_Toc480229243)

[4.2 Implementasi Antarmuka Pengguna 106](#_Toc480229244)

[5 BAB V PENGUJIAN DAN EVALUASI 107](#_Toc480229245)

[5.1 Lingkungan Pengujian 107](#_Toc480229246)

[5.2 Pengujian Fungsionalitas 107](#_Toc480229247)

[6 BAB VI KESIMPULAN DAN SARAN 108](#_Toc480229248)

[6.1 Kesimpulan 108](#_Toc480229249)

[6.2 Saran 108](#_Toc480229250)

[DAFTAR PUSTAKA 110](#_Toc480229251)

[LAMPIRAN 112](#_Toc480229252)

[BIODATA PENULIS 113](#_Toc480229253)

# 

# DAFTAR GAMBAR

[Gambar 3.1 Arsitektur Sistem 39](#_Toc480230737)

[Gambar 3.2 Diagram Kasus Penggunaan 45](#_Toc480230738)

[Gambar 3.3 Diagram Kelas Daerah 68](#_Toc480230739)

[Gambar 3.4 Diagram Kelas Bencana 69](#_Toc480230740)

[Gambar 3.5 Diagram Kelas Dokter 70](#_Toc480230741)

[Gambar 3.6 Diagram Kelas Jarak 71](#_Toc480230742)

[Gambar 3.7 Diagram Kelas Keahlian 72](#_Toc480230743)

[Gambar 3.8 Diagram Kelas Rumah Sakit 73](#_Toc480230744)

[Gambar 3.9 Diagram Kelas Penugasan 74](#_Toc480230745)

[Gambar 3.10 Diagram Kelas Daftar Penugasan 75](#_Toc480230746)

[Gambar 3.11 Conceptual Data Modeling 76](#_Toc480230747)

[Gambar 3.12 *Physical Data Modeling* 77](#_Toc480230748)

[Gambar 3.13 Rancangan Halaman Antarmuka Login Pengguna 78](#_Toc480230749)

[Gambar 3.14 Rancangan Halaman Antarmuka Melihat Data *User* 80](#_Toc480230750)

[Gambar 3.15 Rancangan Halaman Antarmuka Menambah Data *User* 81](#_Toc480230751)

[Gambar 3.16 Rancangan Halaman Antarmuka Mengubah Data *User* 83](#_Toc480230752)

[Gambar 3.17 Rancangan Antarmuka Menghapus Data *User* dan Rumah Pompa 84](#_Toc480230753)

[Gambar 3.18 Rancangan Halaman Antarmuka Melihat Data Rumah Pompa 86](#_Toc480230754)

[Gambar 3.19 Rancangan Halaman Antarmuka Menambah Data Rumah Pompa 87](#_Toc480230755)

[Gambar 3.20 Rancangan Halaman Antarmuka Mengubah Data Rumah Pompa 89](#_Toc480230756)

[Gambar 3.21 Rancangan Halaman Antarmuka Melihat Data Profil 91](#_Toc480230757)

[Gambar 3.22 Rancangan Halaman Antarmuka Mengubah Data Profil 93](#_Toc480230758)

[Gambar 3.23 Rancangan Halaman Antarmuka Melihat Data Ketinggian Air dan Cuaca (Admin) 95](#_Toc480230759)

[Gambar 3.24 Rancangan Halaman Antarmuka Melihat Data Ketinggian Air dan Cuaca 95](#_Toc480230760)

[Gambar 3.25 Rancangan Halaman Mendapat Peringatan Potensi Banjir 97](#_Toc480230761)

[Gambar 3.26 Rancangan Halaman Antarmuka Mencari Data User 98](#_Toc480230762)

[Gambar 3.27 Rancangan Halaman Antarmuka Mencari Data Rumah Pompa 100](#_Toc480230763)

*[Halaman ini sengaja dikosongkan]*

# DAFTAR TABEL

[Tabel 2.1 Metode HTTP dan Penggunaannya dalam REST 34](#_Toc480230764)

[Tabel 3.1 Kebutuhan Fungsional 39](#_Toc480230765)

[Tabel 3.2 Kualitas Perangkat Lunak 40](#_Toc480230766)

[Tabel 3.3 Aktor pada Sistem 41](#_Toc480230767)

[Tabel 3.4 Kasus Penggunaan 42](#_Toc480230768)

[Tabel 3.5 Spesifikasi Kasus Penggunaan UC-0001 44](#_Toc480230769)

[Tabel 3.6 Spesifikasi Kasus Penggunaan UC-0002 45](#_Toc480230770)

[Tabel 3.7 Spesifikasi Kasus Penggunaan UC-0003 46](#_Toc480230771)

[Tabel 3.8 Spesifikasi Kasus Penggunaan UC-0004 47](#_Toc480230772)

[Tabel 3.9 Spesifikasi Kasus Penggunaan UC-0005 48](#_Toc480230773)

[Tabel 3.10 Spesifikasi Kasus Penggunaan UC-0006 50](#_Toc480230774)

[Tabel 3.11 Spesifikasi Kasus Penggunaan UC-0007 51](#_Toc480230775)

[Tabel 3.12 Spesifikasi Kasus Penggunaan UC-0008 53](#_Toc480230776)

[Tabel 3.13 Spesifikasi Kasus Penggunaan UC-0009 54](#_Toc480230777)

[Tabel 3.14 Spesifikasi Kasus Penggunaan UC-0010 55](#_Toc480230778)

[Tabel 3.15 Spesifikasi Kasus Penggunaan UC-0011 57](#_Toc480230779)

[Tabel 3.16 Spesifikasi Kasus Penggunaan UC-0012 58](#_Toc480230780)

[Tabel 3.17 Spesifikasi Kasus Penggunaan UC-0013 59](#_Toc480230781)

[Tabel 3.18 Spesifikasi Kasus Penggunaan UC-0014 60](#_Toc480230782)

[Tabel 3.19 Lingkungan Perancangan Perangkat Lunak 62](#_Toc480230783)

[Tabel 3.20 Atribut Antarmuka Login *User* 77](#_Toc480230784)

[Tabel 3.21 Atribut Antarmuka Melihat Data *User* 78](#_Toc480230785)

[Tabel 3.22 Atribut Antarmuka Menambah Data *User* 79](#_Toc480230786)

[Tabel 3.23 Atribut Antarmuka Mengubah Data User 81](#_Toc480230787)

[Tabel 3.24 Atribut Antarmuka Menghapus Data User dan Rumah Pompa 83](#_Toc480230788)

[Tabel 3.25 Atribut Antarmuka Melihat Data Rumah Pompa 84](#_Toc480230789)

[Tabel 3.26 Atribut Antarmuka Menambah Data Rumah Pompa 86](#_Toc480230790)

[Tabel 3.27 Atribut Antarmuka Mengubah Data Rumah Pompa 87](#_Toc480230791)

[Tabel 3.28 Atribut Antarmuka Melihat Data Profil 89](#_Toc480230792)

[Tabel 3.29 Atribut Antarmuka Mengubah Data Profil 91](#_Toc480230793)

[Tabel 3.30 Atribut Antarmuka Melihat Data Ketinggian Air dan Cuaca 93](#_Toc480230794)

[Tabel 3.31 Atribut Antarmuka Mendapat Peringatan Banjir 95](#_Toc480230795)

[Tabel 3.32 Atribut Antarmuka Mencari Data User 96](#_Toc480230796)

[Tabel 3.33 Atribut Antarmuka Mencari Data Rumah Pompa 98](#_Toc480230797)

# DAFTAR KODE SUMBER

*[Halaman ini sengaja dikosongkan]*

# BAB I PENDAHULUAN

Bab ini menjelaskan garis besar tugas akhir yang meliputi latar belakang, tujuan, rumusan, batasan permasalahan, metodologi pembuatan tugas akhir, dan sistematika penulisan.

* 1. Latar Belakang

Banjir merupakan bencana alam yang biasanya terjadi di beberapa daerah di Indonesia, terutama di daerah perkotaan yang memiliki laju pertumbuhan penduduk yang tinggi. Banjir seolah menjadi permasalahan yang tidak bisa dihindarkan terjadi ketika musim hujan tiba. Hal ini tentu saja membawa dampak negatif bagi masyarakat, seperti terbatasnya mobilitas dalam beraktivitas, rusaknya jalan dan prasarana akibat sering tergenang banjir, dan mengalami kerugian materi.

Surabaya merupakan salah satu daerah yang rawan terjadi bencana banjir dengan total kerugian akibat banjir di Surabaya diperkirakan sebesar Rp 85 milyar (US$ 12 juta) per tahun yang menimpa rumah tangga, usaha, dan hal-hal yang terkait dengan perjalanan transportasi [1]. Salah satu cara yang dilakukan oleh Pemerintah Kota Surabaya untuk mencegah terjadinya banjir adalah dengan membuat rumah pompa yang tersebar di beberapa titik. Rumah pompa ini berfungsi untuk menstabilkan debit air sungai. Air sungai yang diterima rumah pompa akan dialirkan menuju sungai besar atau laut dengan menggunakan pompa air. Apabila hujan turun, maka pompa air ini akan diaktifkan oleh petugas rumah pompa. Tetapi, apabila kondisi cuaca tidak hujan, maka petugas harus mengecek ketinggian air terlebih dahulu sebelum mengaktifkan pompa air. Para petugas harus secara manual mengecek kondisi cuaca dan ketinggian air. Untuk pengecekan ketinggian air, para petugas harus melakukannya secara berkala. Hal tersebut dinilai kurang efektif dan efisien.

Untuk mengatasi permasalahan tersebut, diusulkan sebuah solusi berupa pembuatan aplikasi yang berfungsi sebagai pemberi peringatan (alert) apabila ada potensi banjir, dimana aplikasi ini utamanya diperuntukkan bagi petugas rumah pompa. Sehingga, petugas rumah pompa tidak perlu lagi mengecek kondisi cuaca dan ketinggian air secara manual untuk mengaktifkan pompa. Selain itu, aplikasi akan dibangun pada platform mobile (perangkat bergerak) untuk mendukung kebergunaan dan memudahkan pemantauan dari mana saja. Untuk mendapatkan nilai ketinggian air, akan digunakan sensor level air yang akan dipasang pada saluran masuknya air pada rumah pompa. Sedangkan untuk mengetahui kondisi cuaca, aplikasi ini akan menggunakan Weather Provider API.

* 1. Rumusan Masalah

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana merancang dan membangun aplikasi *alert system* potensi banjir?
2. Bagaimana mendapatkan kondisi ketinggian air dari sensor level air?
3. Bagaimana mendapatkan kondisi cuaca dengan menggunakan *Weather Underground API*?
4. Bagaimana menampilkan notifikasi atau peringatan potensi banjir?
   1. Batasan Masalah

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan, yaitu sebagai berikut:

1. Aplikasi dapat dijalankan apabila aplikasi tersambung dengan internet.
2. Sistem perangkat lunak dibangun dengan menggunakan bahasa Java dan aplikasi Android Studio.
3. Kondisi ketinggian air didapatkan dari sensor level air.
4. Kondisi cuaca didapatkan dari *Weather Provider* API, yaitu *Weather Underground API*.
5. Penentuan pemberian peringatan potensi banjir didasarkan pada hasil wawancara terhadap beberapa petugas rumah pompa.
6. Uji coba aplikasi ini akan dilakukan pada lima rumah pompa di Surabaya
   1. Tujuan

Tugas akhir ini mempunyai beberapa tujuan, yaitu sebagai berikut:

1. Merancang dan membangun aplikasi *alert system* potensi banjir.
2. Mendapatkan dan menampilkan kondisi ketinggian air dari sensor level air.
3. Mendapatkan dan menampilkan kondisi cuaca dengan menggunakan *Weather Underground API*.
4. Menampilkan notifikasi atau peringatan potensi banjir.
   1. Manfaat

Manfaat yang didapat dari dibuatnya tugas akhir ini, antara lain:

1. Memudahkan petugas rumah pompa dan Dinas Pekerjaan Umum Bina Marga dan Pematusan dalam memantau kondisi cuaca dan kondisi ketinggian air tanpa harus mengecek secara manual.
2. Memberikan peringatan pada petugas rumah pompa apabila terdapat potensi banjir.
   1. Metodologi Pembuatan Tugas Akhir

Tahapan-tahapan yang dilakukan dalam pengerjaan tugas akhir ini adalah sebagai berikut:

1. Penyusunan proposal tugas akhir.

Proposal tugas akhir ini berisi tentang deskripsi pendahuluan dari tugas akhir yang akan dibuat. Pendahuluan ini terdiri atas hal yang menjadi latar belakang diajukannya usulan tugas akhir, rumusan masalah yang diangkat, batasan masalah untuk tugas akhir, tujuan dari pembuatan tugas akhir, dan manfaat dari hasil pembuatan tugas akhir. Selain itu dijabarkan pula tinjauan pustaka yang digunakan sebagai referensi pendukung pembuatan tugas akhir. Sub bab metodologi berisi penjelasan mengenai tahapan penyusunan tugas akhir mulai dari penyusunan proposal hingga penyusunan buku tugas akhir. Terdapat pula sub bab jadwal kegiatan yang menjelaskan jadwal pengerjaan tugas akhir.

1. Studi literatur

Pada studi literatur ini, akan dipelajari sejumlah referensi yang diperlukan dalam pembuatan aplikasi yaitu mengenai Android sebagai perangkat untuk pengembangan aplikasi, *RESTful web service* yang digunakan sebagai media komunikasi data, PostgreSQL untuk pengelolaan database, *Weather Underground API* sebagai layanan yang menyediakan data cuaca, dan sensor level air yang digunakan untuk mendapatkan data ketinggian air.

1. Analisis dan desain perangkat lunak

Tahap ini meliputi perumusan kebutuhan fungsional, kebutuhan non-fungsional, kasus penggunaan, diagram aktivitas, diagram kelas, diagram sekuens, rancangan antarmuka pengguna untuk akun petugas, pengawas, dan admin, serta pembuatan rancangan basis data.

1. Implementasi perangkat lunak

Aplikasi ini diimplementasikan dengan menggunakan kakas bantu :

1. Sistem operasi Android dengan spesifikasi minimal Android 4.0 (*Ice Cream Sandwich*).
2. Bahasa pemrograman yang digunakan adalah Java.
3. IDE yang digunakan adalah Android Studio.
4. Database yang digunakan adalah PostgreSQL.
5. Kerangka kerja *web service* CodeIgniter.
6. *Hosting web* yang menjadi acuan dalam perancangan aplikasi ini dari dewaweb.com
7. Postman, kakas bantu untuk menguji integrasi *web service* dengan sistem yang sudah dibangun.
8. Jetbrains PhpStorm sebagai *text editor* dalam pengerjaan *web service.*
9. Pengujian dan evaluasi

Pengujian dan evaluasi aplikasi perangkat lunak hasil dari tugas akhir ini diujicobakan pada beberapa rumah pompa di Surabaya. Percobaan akan didokumentasikan melalui pengambilan video dengan menggunakan smartphone Android.

1. Penyusunan buku tugas akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain:

1. Pendahuluan
2. Latar Belakang
3. Rumusan Masalah
4. Batasan Masalah
5. Tujuan
6. Manfaat
7. Metodologi Pembuatan Tugas Akhir
8. Sistematika Penulisan Laporan Tugas Akhir
9. Tinjauan Pustaka
10. Analisis dan Perancangan Sistem
11. Pengujian dan Evaluasi
12. Kesimpulan dan Saran
13. Daftar Pustaka
    1. Sistematika Penulisan Laporan Tugas Akhir

Buku tugas akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan tugas akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku tugas akhir terdiri atas beberapa bagian seperti berikut ini.

**Bab I Pendahuluan**

Bab yang berisi mengenai latar belakang, rumusan masalah, batasan masalah, tujuan, dan manfaat dari pembuatan tugas akhir. Selain itu metodologi yang digunakan dan sistematika penulisan laporan akhir juga merupakan bagian dari bab ini.

**Bab II Tinjauan Pustaka**

Bab ini berisi penjelasan secara detail mengenai dasar-dasar penunjang dan teori-teori yang digunakan untuk mendukung pembuatan tugas akhir ini.

**Bab III Analisis dan Perancangan Sistem**

Bab ini berisi tentang analisis permasalahan, deskripsi umum sistem, spesifikasi kebutuhan perangkat lunak, lingkungan perancangan, perancangan arsitektur sitem, diagram kelas, dan struktur data.

**Bab IV Implementasi**

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Penjelasan berupa kode sumber yang digunakan untuk proses implementasi.

**Bab V Pengujian dan Evaluasi**

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian kinerja dari sistem yang telah dibuat.

**Bab VI Kesimpulan dan Saran**

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian kinerja dari sistem yang telah dibuat.



*[Halaman ini sengaja dikosongkan]*

# BAB II TINJAUAN PUSTAKA

Bab ini menjelaskan teori-teori yang berkaitan dengan pembangunan aplikasi *Alert System* Potensi Banjir untuk Rumah Pompa yang diajukan untuk tugas akhir ini. Penjelasan ini bertujuan untuk memberikan gambaran secara umum terhadap perangkat lunak yang dibuat dan berguna sebagai penunjang dalam pengembangan perangkat lunak.

1. Banjir

Banjir adalah peristiwa atau keadaan dimana terendamnya suatu daerah atau daratan karena volume air yang meningkat [2]. Banjir terjadi apabila air dari saluran yang ada meluap dan menggenangi wilayah sekitarnya. Banjir merupakan ancaman bencana alam yang paling sering terjadi dan paling banyak merugikan, baik dari segi kemanusiaan maupun ekonomi. Sembilan puluh persen dari kejadian bencana (tidak termasuk bencana kekeringan) berhubungan dengan banjir [3]. Beberapa karakteristik yang berkaitan dengan banjir, di antaranya adalah [4]:

1. Banjir dapat datang secara tiba-tiba dengan intensitas besar namun dapat langsung mengalir.
2. Banjir datang secara perlahan namun intensitas hujannya sedikit.
3. Pola banjirnya musiman.
4. Banjir datang secara perlahan namun dapat menjadi genangan yang lama di daerah depresi.
5. Akibat yang ditimbulkan adalah terjadinya genangan, erosi, dan sedimentasi. Sedangkan akibat lainnya adalah terisolasinya daerah pemukiman dan diperlukan evakuasi penduduk.

Kerugian akibat banjir pada umumnya sulit diidentifikasi secara jelas, dimana terdiri dari kerugian banjir akibat banjir langsung dan tak langsung. Kerugian akibat banjir langsung, merupakan kerugian fisik akibat banjir yang terjadi, antara lain robohnya gedung sekolah, industri, rusaknya sarana transportasi, hilangnya nyawa, hilangnya harta benda, kerusakan di pemukiman, kerusakan daerah pertanian dan peternakan, kerusakan sistem irigasi, sistem air bersih, sistem drainase, sistem kelistrikan, sistem pengendali banjir termasuk bangunannya, kerusakan sungai, dan sebagainya. Sedangkan kerugian akibat banjir tak langsung berupa kerugian kesulitan yang timbul secara tak langsung diakibatkan oleh banjir, seperti komunikasi, pendidikan, kesehatan, kegiatan bisnis terganggu dan sebagainya [5].

1. Sistem Peringatan (*Alert System*)

Sistem Peringatan adalah serangkaian sistem untuk memberitahukan akan timbulnya kejadian alam, dapat berupa bencana maupun tanda-tanda alam lainnya. Bagi masyarakat Indonesia, sistem peringatan dini dalam menghadapi bencana sangatlah penting, mengingat secara geologis dan klimatologis wilayah Indonesia termasuk daerah rawan bencana alam. Dengan ini diharapkan akan dapat dikembangkan upaya-upaya yang tepat untuk mencegah atau paling tidak mengurangi terjadinya dampak bencana alam bagi masyarakat. Keterlambatan dalam menangani bencana dapat menimbulkan kerugian yang semakin besar bagi masyarakat [6].

Informasi yang berhubungan dengan bahaya yang akan terjadi biasanya berupa tanda/sinyal tertentu yang disebut alarm (*alert*). Alarm bertujuan agar pihak-pihak yang terkait dengan bahaya tersebut mampu mempersiapkan diri untuk mengatasi, atau paling tidak terhindar dari bencana tersebut [7].

1. Rumah Pompa

Pembangunan rumah pompa merupakan salah satu program pengendalian banjir guna mengatasi permasalahan banjir. Rumah Pompa merupakan tempat yang digunakan oleh pompa air untuk memindahkan atau menaikkan debit air serta mengatur besarnya air yang dapat dikeluarkan oleh pompa tersebut [8]. Setiap rumah pompa di Surabaya memiliki daerah layanan masing-masing. Pertama, air yang berasal dari daerah layanan rumah akan masuk melalui saluran menuju tempat penampungan air. Pada saluran ini terdapat indikator level ketinggian air yang berfungsi untuk mengetahui ketinggian air saat ini. Pada saluran ini, juga terdapat screen yang berfungsi untuk menyaring sampah yang terbawa air.

Setelah melewati screen, air kemudian menuju ke bak penampungan. Pada bak penampungan ini, air ditampung agar debit yang diambil oleh pompa dapat kontinu sehingga tidak merusak pompa. Pada bak penampung ini terdapat 2 ruang. Ruang pertama adalah ruang menuju pintu air dan ruang kedua adalah ruang menuju ke pompa. Kedua ruangan ini saling berhubungan melalui lubang besar dengan screen tambahan untuk menyaring kembali air yang akan dipompa. Fungsi ruang menuju pompa ini adalah untuk mengekualisasikan debit yang akan disalurkan oleh pompa sehingga tidak merusak pompa. Sedangkan fungsi dari screen kedua adalah untuk menyaring kembali air yang masuk menuju ruang pompa agar dipastikan tidak ada kotora yang masuk yang nantinya akan mengganggu kinerja pompa [9].

Setiap rumah pompa memiliki 2 macam pompa, yaitu pompa air dan pompa *Sludge.* Pompa air berfungsi untuk mengalirkan air yang sudah ditampung menuju ke tujuan pembuangan, sedangkan pompa *Sludge* merupakan pompa lumpur yang biasanya memiliki kapasitas yang lebih kecil daripada pompa air.

Pada setiap rumah pompa, terdapat petugas yang siap siaga 24 jam untuk mengatur pompa. Petugas akan langsung mengaktifkan pompa ketika hujan turun. Namun, jika kondisi cuaca tidak hujan, petugas harus selalu mengecek ketinggian air secara berkala. Pengecekan kondisi cuaca dan ketinggian air dilakukan secara manual oleh petugas.

1. Android

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat bergerak layar sentuh seperti telepon pintar dan komputer tablet. Android awalnya dikembangkan oleh Android, Inc., dengan dukungan finansial dari Google, yang kemudian membelinya pada tahun 2005. Sistem operasi ini dirilis secara resmi pada tahun 2007, bersamaan dengan didirikannya Open Handset Alliance, konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler [10]. Sebanyak 59,91 persen smartphone yang beredar di Indonesia menggunakan sistem operasi Android [11]. Karena alasan itulah, penulis mengembangkan aplikasi ini dalam sistem Android.

1. REST Web Service

REST (REpresentational State Transfer) merupakan standar arsitektur komunikasi berbasis web yang sering diterapkan dalam pengembangan layanan berbasis web. Umumnya menggunakan HTTP (Hypertext Transfer Protocol) sebagai protocol untuk komunikasi data. REST pertama kali diperkenalkan oleh Roy Fielding pada tahun 2000. Pada arsitektur REST, REST server menyediakan resources (sumber daya/data) dan REST client mengakses dan menampilkan resource tersebut untuk penggunaan selanjutnya. Setiap resource diidentifikasi oleh URIs (Universal Resource Identifiers) atau global ID. Resource tersebut direpresentasikan dalam bentuk format teks, JSON atau XML. Pada umumnya formatnya menggunakan JSON dan XML [12].

Penggunaan metode-metode HTTP dalam REST adalah sebagai berikut [13] :

Tabel 2.1 Metode HTTP dan Penggunaannya dalam REST

|  |  |
| --- | --- |
| **Metode** | **Deskripsi** |
| GET | Mendapatkan (*read*) sebuah sumber daya (*resource*) yang diidentifikasi dengan URI (*Uniform Resource Identifier*) |
| POST | Mengirimkan sumber daya (*resource*) ke server. Digunakan untuk membuat (*create*) sumber daya baru |
| PUT | Mengirimkan sumber daya (*resource*) ke server. Digunakan untuk memasukkan (*insert*) atau memperbarui (*update*) sumber daya yang tersimpan. |
| DELETE | Menghapus (*delete*) sumber daya (*resource*) yang diidentifikasi dengan URI |
| HEAD | Mendapatkan metadata (*response header*) dari sumber daya (*resource*) yang diidentifikasi dengan URI. |

Komponen dari HTTP Response adalah:

* Status/Response Code, mengindikasikan status *server* terhadap *resource* yang diminta. Misal: 404, artinya *resource* tidak ditemukan dan 200 response OK.
* HTTP Version, menunjukkan versi dari HTTP yang digunakan, contoh HTTP v1.1.
* Response Header, berisi metadata untuk HTTP Response. Contoh, type server, panjang content, tipe content, waktu response, dll.
* Response Body, konten dari data yang diberikan.

1. PostgreSQL

PostgreSQL dikembangkan oleh University of California di Berkeley Computer Science Department. Dengan sifatnya yang *open source* menjadikan pula database ini dapat dikembangkan sesuai dengan kebutuhan. Sebagai ORDBMS (*Object Relational Database Management System*) yang ada saat ini, PostgreSQL memiliki berbagai macam kemapuan yang dimiliki oleh database komersil umum lainnya, seperti dukungan akan perintah-perintah SQL, dimana dengan menggunakan perintah-perintah SQL memungkinkan database administrator lebih mudah berinteraksi dengan database PostgreSQL, baik dalam manipulasi data seperti: *insert*, *update*, ataupun *delete*. Dengan kemampuan untuk dapat menvariasikan perintah *select* dengan berbagai macam klausa yang ada, menjadikan perintah *select* pada database ini jauh lebih fleksibel dalam hal menquery data dari tabel-tabel yang ada [14].

1. Volley

Volley adalah *library* HTTP yang mempermudah dan mempercepat *networking* pada aplikasi Android [15]. Volley yang diterapkan pada aplikasi ini terdiri dari:

* Mengirim *request* yang merupakan penggunaan Volley dengan membuat RequestQueue dan menyampaikannya objek Request [16].
* Membuat Request standar dengan menggunakan StringRequest yang menentukan URL dan menerima *string*, serta JsonRequest (JsonObjectRequest dan JsonArrayRequest) yang menentukan URL dan memperoleh objek JSON atau *array* sebagai respon [17].

1. Sensor Ketinggian Air

Sensor ketinggian air mendeteksi ketinggian pada zat yang mengalir, misalnya air. Zat yang diukur dapat berada pada sebuah wadah atau mengalir pada sungai atau danau. Terdapat dua macam pengukuran ketinggian air, yaitu pengkuruan secara *continuous* dan *point values*. Sensor ketinggian air secara *continuous* akan mengukur ketinggian air pada kisaran tertentu dan menunjukkan angka ketinggian air. Sedangkan sensor ketinggian air *point values* hanya mengindikasikan apakah air berada di atas atau di bawah batas ketinggian [18].

Sensor yang akan digunakan pada tugas akhir ini menggunakan sensor ultrasonik. Sensor ketinggian air mengukur jarak antara sensor dan permukaan air dengan cara menembakkan gelombang ultrasonik menuju permukaan air. Setelah gelombang menyentuh permukaan air, maka target akan memantulkan kembali gelombang tersebut. Gelombang pantulan dari target akan ditangkap oleh sensor, kemudian sensor menghitung selisih antara waktu pengiriman gelombang dan waktu gelombang pantul diterima.

Pada sensor ini terdapat aplikasi dalam bahasa python yang berfungsi untuk mendapatkan ketinggian air. Setiap kali aplikasi ini dijalankan, maka akan menghasilkan nilai ketinggian air dalam sentimeter (cm).

1. *Weather Underground API*

*Weather Underground API* adalah layanan online yang menyediakan API untuk data cuaca, termasuk data saat cuaca, ramalan, dan data historis untuk para pengembang layanan web dan aplikasi *mobile*. *API* ini menyediakan data yang dapat diandalkan, ramalan cuaca yang akurat, dan mencakup 80 bahasa.

Sebelum menggunakan *API* ini, diperlukan *API key*. API key bisa didapatkan pada menu *Key Settings* pada menu *Weather API for Developer*. *Request API* dibuat melalui HTTP dan akan mengembalikan respon dalam JSON atau XML [19]. Ada beberapa cara yang dapat dilakukan untuk mendapatkan data cuaca pada sebuah daerah, antara lain:

1. Mendapatkan cuaca berdasarkan nama kota

Informasi cuaca bisa didapatkan dengan menyebutkan nama kota saja, atau nama kota beserta nama negaranya. API akan memberikan daftar yang sesuai dengan nama yang dicari.

1. Mendapatkan cuaca berdasarkan kode bandara

Informasi cuaca bisa didapatkan dengan menyebutkan kode bandara.

1. Mendapatkan cuaca berdasarkan koordinat geografis

Informasi cuaca bisa didapatkan dengan menyebutkan latitude dan longitude koordinat dari lokasi yang dipilih.

1. Mendapatkan cuaca berdasarkan kode pos

Informasi cuaca bisa didapatkan dengan menyebutkan kode pos dan nama negara.

Terdapat beberapa fitur pada API ini, antara lain:

1. *Conditions*

Pada fitur ini, respon berupa kondisi cuaca saat ini, meliputi deskripsi cuaca, suhu udara, kelembapan, kondisi angin, dan lain-lain.

1. *Forecast*

Pada fitur ini mengembalikan respon ringkasan perkiraan cuaca selama 3 hari ke depan.

1. *Forecast 10 day*

Pada fitur ini mengembalikan respon ringkasan perkiraan cuaca selama 10 hari ke depan.

1. *Hourly*

Pafa fitur ini mengembalikan respon perkiraan cuaca tiap jam selama 24 jam ke depan

1. dan lain sebagainya

|  |
| --- |
| {  "response": {  "version": "0.1",  "termsofService": "http://www.wunderground.com/weather/api/d/terms.html",  "features": {  "hourly": 1  }  },  "hourly\_forecast": [{  "FCTTIME": {  "hour": "18",  "hour\_padded": "18",  "min": "00",  "min\_unpadded": "0",  "sec": "0",  "year": "2017",  "mon": "4",  "mon\_padded": "04",  "mon\_abbrev": "Apr",  "mday": "19",  "mday\_padded": "19",  "yday": "108",  "isdst": "0",  "epoch": "1492599600",  "pretty": "6:00 PM WIB on April 19, 2017",  "civil": "6:00 PM",  "ampm": "PM",  "tz": "",  "age": "",  "UTCDATE": ""  },  "temp": {  "english": "82",  "metric": "28"  },  "dewpoint": {  "english": "76",  "metric": "24"  },  "condition": "Thunderstorm",  "icon": "tstorms", |

|  |
| --- |
| "icon\_url": "http://icons.wxug.com/i/c/k/nt\_tstorms.gif",  "fctcode": "15",  "sky": "75",  "wspd": {  "english": "6",  "metric": "10"  },  "wdir": {  "dir": "ENE",  "degrees": "70"  },  "wx": "Thunderstorms",  "uvi": "0",  "humidity": "82",  "windchill": {  "english": "-9999",  "metric": "-9999"  },  "heatindex": {  "english": "90",  "metric": "32"  },  "feelslike": {  "english": "90",  "metric": "32"  },  "qpf": {  "english": "0.08",  "metric": "2"  },  "snow": {  "english": "0.0",  "metric": "0"  },  "pop": "61",  "mslp": {  "english": "29.84",  "metric": "1011"  }  }, |

Gambar 2.1 Respon Weather Underground API

Tabel 2.2 Deskripsi Respon Weather Underground API

|  |  |
| --- | --- |
| Respon | Keterangan |
| tempm | Suhu udara (Celcius) |
| tempi | Suhu udara (Fahrenheit) |
| dewptm | Titik embun (Celcius) |
| dewpti | Titik embundalam (Fahrenheit) |
| hum | Kelembapan (%) |
| wspdm | Kecepatan angin (km/jam) |
| wspdi | Kecepatan angin (mil/jam) |
| wgustm | Hembusan angin (km/jam) |
| wgusti | Hembusan angin (mil/jam) |
| wdird | Arah angin (derajat) |
| wdire | Deskripsi arah angin |
| vism | Jarak penglihatan (km) |
| visi | Jarak penglihatan (mil) |
| pressurem | Tekanan (mBar) |
| pressurei | Tekanan (inHg) |
| windchillm | Kesejukan angin (Celcius) |
| windchilli | Kesejukan angin (Fahrenheit) |
| heatindexm | Index panas (celcius) |
| heatindexi | Index panas (Fahrenheit) |
| precipm | Volume hujan (mm) |
| precipi | Volume hujan (inci) |
| pop | Kemungkinan terjadinya hujan |
| conds | Deskripsi kondisi cuaca (Hujan, Berawan, dan lain-lain) |

# 

# BAB III ANALISIS DAN PERANCANGAN SISTEM

Bab ini membahas analisis kebutuhan dan rancangan yang akan digunakan untuk membangun perangkat lunak yang diajukan sebagai tugas akhir.

1. Analisis

Tahap analisis dibagi menjadi beberapa bagian, antara lain analisis permasalahan, deskripsi umum sistem, dan spesifikasi kebutuhan perangkat lunak.

1. Analisis Permasalahan

Surabaya merupakan salah satu daerah yang rawan terjadi banjir. Untuk mengatasi permasalahan banjir tersebut, Pemerintah Kota Surabaya mendirikan rumah pompa yang tersebar di beberapa daerah. Rumah pompa berfungsi untuk menstabilkan debit air sungai dengan cara mengalirkan air yang diterima oleh rumah pompa dari beberapa daerah layanan menuju ke sungai besar atau ke laut. Setiap rumah pompa memiliki 2 macam pompa, yaitu pompa air dan pompa *Sludge*. Pompa air inilah yang akan mengalirkan air yang sudah ditampung menuju ke tujuan pembuangan, sedangkan pompa *Sludge* merupakan pompa lumpur yang biasanya memiliki kapasitas yang lebih kecil daripada pompa air.

Pompa air akan dinyalakan apabila terjadi potensi banjir. Pada rumah pompa, terjadinya potensi banjir dipengaruhi oleh dua hal, yaitu ketinggian air pada saluran masuk rumah pompa dan kondisi cuaca. Potensi banjir terjadi apabila cuaca hujan atau berpotensi hujan. Namun, apabila tidak terjadi hujan atau tidak berpotensi hujan, maka akan dilihat dari ketinggian air pada saluran masuknya air pada rumah pompa. Potensi banjir terjadi apabila ketinggian air melebihi *threshold* atau ambang batas ketinggian air yang telah ditetapkan oleh petugas pada masing-masing rumah pompa. Agar tidak terjadi banjir, maka petugas pompa akan selalu memantau kondisi cuaca dan ketinggian air. Namun, proses pemantauannya masih dilakukan secara manual. Hal tersebut dinilai kurang efektif dan efisien.

Untuk mengatasi permasalahan tersebut, penulis memanfaatkan *smartphone* untuk mengimplementasikan teknologi yang memudahkan petugas pompa untuk memantau kondisi cuaca dan ketinggian air pada rumah pompa. Hingga nanti aplikasi ini akan memberikan peringatan apabila terjadi potensi banjir. Aplikasi ini dibangun pada platform *mobile* (perangkat bergerak) untuk mendukung kebergunaan dan memudahkan pemantauan kondisi cuaca dan ketinggian air dari mana saja.

1. Analisis Faktor yang Mempengaruhi Potensi Banjir

Pompa air pada rumah pompa akan dinyalakan ketika terjadi potensi banjir. Terdapat dua faktor utama yang mempengaruhi terjadinya potensi banjir, yaitu kondisi cuaca dan ketinggian air pada saluran masuk rumah pompa. Berdasarkan hasil wawancara yang dilakukan pada beberapa petugas rumah pompa, kondisi cuaca merupakan faktor paling penting untuk menentukan diaktifkannya pompa air. Ketika kondisi cuaca hujan, pompa air akan dinyalakan tanpa melihat kondisi ketinggian air yang terdapat pada saluran masuk rumah pompa. Pada sistem ini, kondisi cuaca didapatkan dengan menggunakan teknologi *weather API*, yaitu *weather underground API*. Pada *weather underground API*, digunakan fitur *hourly.* Fitur *hourly* akan memberikan respon data ramalan cuaca tiap jam hingga 24 jam mendatang setelah meminta *request* kepada *API*. Fitur ini dipilih, karena menyediakan respon tiap jam, sehingga data cuaca yang didapatkan lebih detail. Respon dapat dilihat pada Gambar 2.1. Berdasarkan respon tersebut, data yang digunakan untuk mendapatkan data cuaca adalah pop (*probability of precipitation*). Pop merupakan presentase kemungkinan terjadinya hujan di sebuah lokasi pada area perkiraan selama jangka waktu tertentu. Keterangan nilai pop dapat dilihat pada Tabel 3.1

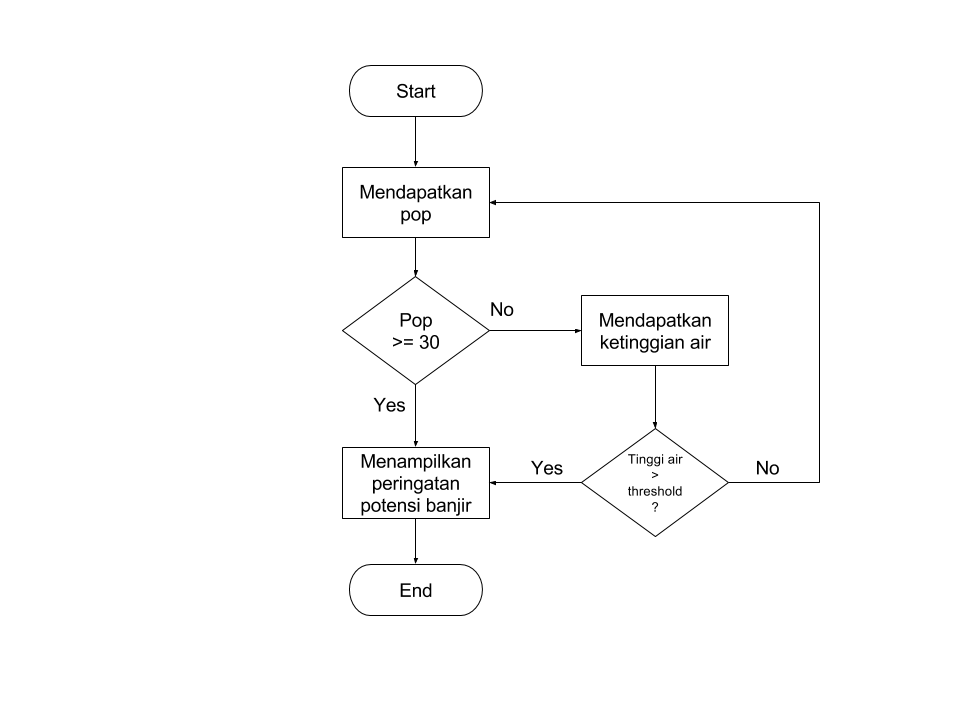
Tabel 3.1 Keterangan *Probability of Precipitation*

|  |  |
| --- | --- |
| Pop *(Probability of precipitation)* | Keterangan |
| 0 % | Tidak terjadi hujan |
| 10 - 20% | Berpeluang kecil, cakupan area kecil |
| 30 - 50% | Berpeluang terjadi hujan, area tersebar |
| 60 - 70% | Berpeluang besar terjadi hujan, terjadi di banyak daerah |
| 80 - 100% | Berpeluang besar terjadi hujan, area tersebar luar |

Karena nilai pop mulai dari 30% sudah berpeluang terjadi hujan, maka terjadinya potensi banjir berdasarkan cuaca akan terjadi apabila pop ≥ 30.

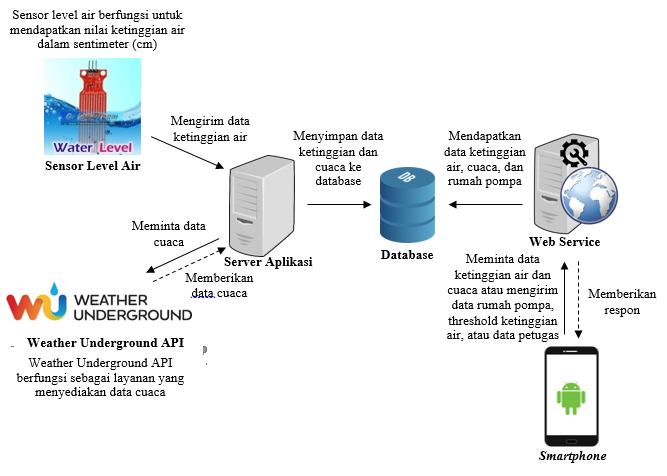
Faktor kedua yang mempengaruhi terjadinya potensi banjir adalah ketinggian air. Ketinggian air pada rumah pompa akan dipantau menggunakan sensor level air. Sensor level air yang digunakan bekerja dengan cara memancarkan gelombang ultrasonik ke permukaan air. Setelah gelombang menyentuh permukaan air, maka target akan memantulkan kembali gelombang tersebut. Gelombang pantulan dari target akan ditangkap oleh sensor, kemudian sensor menghitung selisih antara waktu pengiriman gelombang dan waktu gelombang pantul diterima. Hasil yang didapatkan adalah jarak antara sensor dengan permukaan air dalam sentimeter (cm). Untuk mendapatkan ketinggian air pada rumah pompa, perlu diketahui ketinggian diletakannya sensor. Kemudian, hasil yang didapat dikurangi dengan hasil yang dibaca oleh sensor. Potensi banjir terjadi apabila ketinggian air melebihi *threshold* ketinggian air pada setiap rumah pompa.

Berdasarkan kedua faktor diatas, disimpulkan flowchart untuk menampilkan peringatan potensi banjir seperti pada Gambar 3.1.



Gambar 3.1 Flowchart Menampilkan Peringatan Potensi Banjir

1. Deskripsi Umum Sistem



Gambar 3.2 Arsitektur Sistem

Arsitektur sistem aplikasi *mobile* ini digambarkan seperti pada Gambar 3.1. Untuk mendapatkan data ketinggian air, akan digunakan bantuan dari sensor level air. Sensor level air ini berfungsi untuk memonitoring level ketinggian air dengan memberikan informasi mengenai ketinggian air pada saluran masuknya air dari daerah layanan rumah pompa. Sensor level air akan menghasilkan ketinggian air dalam sentimeter (cm). Selain itu, terdapat *Weather Underground API* yang akan digunakan untuk mendapatkan data cuaca. Data cuaca yang digunakan adalah deskripsi cuaca dan *probability of precipitation* (pop) atau kemungkinan terjadinya hujan dalam bentuk presentase. Kemudian, sensor level air akan mengirimkan data ketinggian air yang didapat untuk disimpan pada database melalui server aplikasi. Sedangkan untuk data cuaca, server aplikasi akan meminta data cuaca pada *Weather Underground API* dan respon data yang didapatkan dari API akan disimpan pada database.

Selanjutnya, *web service* akan mengambil data ketinggian air dan data cuaca pada database. Data-data yang telah diperoleh akan diolah oleh *web service* untuk ditentukan apakah terdapat potensi banjir atau tidak. Potensi banjir akan muncul jika cuaca hujan. Namun, jika kondisi cuaca tidak hujan, maka potensi banjir akan muncul apabila ketinggian air pada rumah pompa melebihi *threshold* ketinggian air pada setiap rumah pompa. *Threshold* ketinggian air adalah batas ketinggian air dimana pompa harus mulai diaktifkan. *Threshold* ketinggian air pada setiap rumah pompa bisa berbeda-beda, hal ini berdasarkan pada survei yang telah dilakukan. Sehingga, keluaran yang ditampilkan pada aplikasi adalah data cuaca dan ketinggian air serta peringatan potensi banjir.

1. Spesifikasi Kebutuhan Perangkat Lunak

Sesuai dengan cakupan perangkat lunak yang telah dijelaskan pada bagian deskripsi umum sistem, dibutuhkan adanya spesifikasi perangkat lunak agar dapat memberikan solusi dari permasalahan yang diberikan dan dapat bekerja dengan baik dalam mengakomodasi kebutuhan. Diharapkan dengan adanya spesifikasi ini dapat menyesuaikan kebutuhan-kebutuhan pengguna. Spesifikasi kebutuhan perangkat lunak adalah penjelasan mengenai kebutuhan sistem yang diinginkan pelanggan atau klien dalam bentuk tulisan. Spesifikasi kebutuhan perangkat lunak pada tugas akhir ini terdiri dari kebutuhan fungsional, kebutuhan non-fungsional, aktor, dan kasus penggunaan.

1. Kebutuhan Fungsional

Tabel 3.2 Kebutuhan Fungsional

|  |  |  |
| --- | --- | --- |
| No | Kebutuhan Fungsional | Deskripsi |
| 1 | Menangani kelola data user | Menampilkan data user, melakukan proses penambahan, pengeditan, serta penghapusan data user |
| 2 | Menangani kelola data rumah pompa | Menampilkan data rumah pompa, melakukan proses penambahan, pengeditan, serta penghapusan data rumah pompa |
| 3 | Menangani kelola data profil | Menampilkan data profil akun, melakukan proses penambahan, pengeditan, serta penghapusan data profil akun |
| 4 | Memberikan informasi data cuaca dan ketinggian air | Menampilkan deskripsi data cuaca dan ketinggian air dalam sentimeter (cm) pada rumah pompa |
| 5 | Memberikan peringatan potensi banjir | Menampilkan peringatan potensi banjir pada petugas yang bertugas pada rumah pompa yang berpotensi banjir |
| 6 | Menangani pencarian rumah pompa | Melakukan proses pencarian data rumah pompa berdasarkan nama rumah pompa yang diinputkan |
| 7 | Menangani pencarian user | Melakukan proses pencarian data user berdasarkan nama user yang diinputkan |

1. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional yang harus dipenuhi oleh sistem sebagai berikut:

1. Kebutuhan Performa

Perangkat lunak akan berjalan dengan performa terbaik jika dijalankan di atas spesifikasi minimal.

1. Kebutuhan Perlindungan Keamanaan

*Username* dan *password* dibutuhkan untuk mengamankan data pengguna yang berhak mengakses sistem sebagai syarat memasuki sistem dan melakukan semua fungsionalitas pada sistem.

1. Kualitas perangkat lunak dapat dilihat pada Tabel 3.2.

Tabel 3.3 Kualitas Perangkat Lunak

|  |  |  |
| --- | --- | --- |
| No | Parameter | Deskripsi |
| 1 | Ketersediaan | Aplikasi harus dapat berjalan pada sistem operasi yang sesuai dengan platform perangkat bergerak sesuai dengan platform yang telah disebutkan. Aplikasi dapat berjalan tanpa dibatasi waktu |
| 2 | Tingkat kualitas | Aplikasi dibangun dengan antarmuka pengguna yang konsisten, mudah dipahami dan mudah dioperasikan |
| 3 | *Portability* | Aplikasi mudah untuk dioperasikan pada *smartphone* dengan *platform* Android |
| 4 | Bahasa | Bahasa yang digunakan pada antarmuka merupakan bahasa Indonesia. |

1. Aktor

Pengertian pengguna adalah pihak-pihak, baik manusia maupun sistem atau perangkat lain yang terlibat dan berinteraksi secara langsung dengan sistem. Aktor pada sistem ini akan dijelaskan pada Tabel 3.3.

Tabel 3.4 Aktor pada Sistem

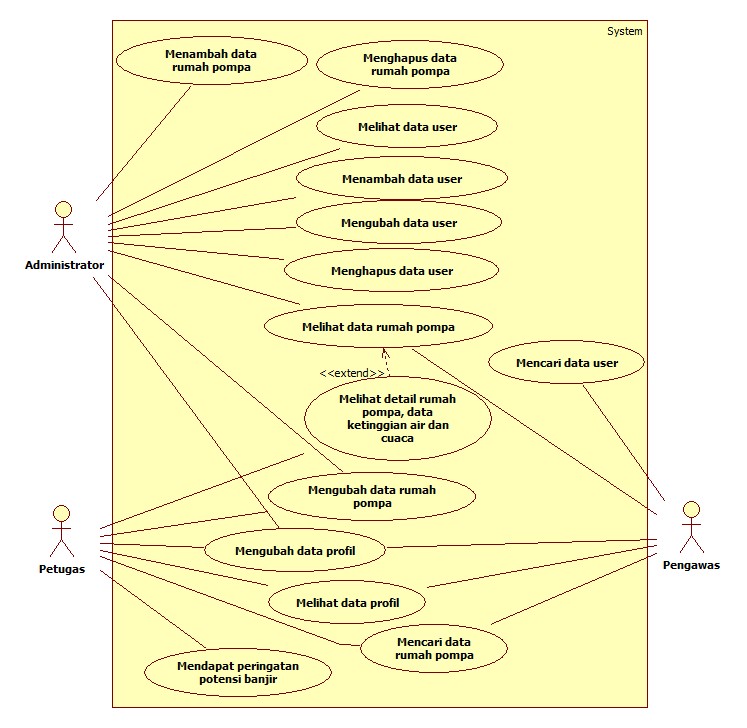
|  |  |  |
| --- | --- | --- |
| **Aktor** | **Tugas** | **Hak Akses ke aplikasi** |
| Petugas  (Petugas rumah pompa) | Memonitoring kondisi ketinggian air dan cuaca pada rumah pompa yang dibawahi dan mendapatkan peringatan potensi banjir pada rumah pompa yang dibawahi. | * Membuat dan mengubah data akun * Melihat kondisi ketinggian air dan cuaca pada rumah pompa yang dibawahi * Mengatur threshold ketinggian air dan tinggi kedalaman inlet * Mendapat peringatan potensi banjir untuk rumah pompa yang dibawahi saja |
| Pengawas  (Pegawai [Dinas Pekerjaan Umum Bina Marga dan Pematusan](http://www.surabaya.go.id/berita/8075-dinas-pekerjaan-umum-bina-marga-dan-pematusan)) | Memonitoring kondisi ketinggian air dan cuaca semua rumah pompa | * Membuat dan mengubah data akun * Melihat semua data rumah pompa * Melihat kondisi ketinggian air dan cuaca pada semua rumah pompa |
| Admin | Mengelola data | Bisa mengelola (menambah, mengubah dan menghapus) data user dan data rumah pompa |

1. Kasus Penggunaan

Berdasarkan analisis spesifikasi kebutuhan fungsional dan analisis aktor dari sistem, dibuat kasus penggunaan sistem. Kasus penggunaan digambarkan dalam tabel penjelasan kasus penggunaan dan diagram kasus penggunaan. Tabel penjelasan kasus penggunaan dapat dilihat pada Tabel 3.4 dan diagram kasus penggunaan dapat dilihat Gambar 3.2.

Tabel 3.5 Kasus Penggunaan

|  |  |  |
| --- | --- | --- |
| **Kode Kasus Penggunaan** | **Nama** | **Aktor** |
| UC-0001 | Melihat data *user* | Admin |
| UC-0002 | Menambah data *user* | Admin |
| UC-0003 | Mengubah data *user* | Admin |
| UC-0004 | Menghapus data *user* | Admin |
| UC-0005 | Melihat data rumah pompa | Admin, pengawas |
| UC-0006 | Menambah data rumah pompa | Admin |
| UC-0007 | Mengubah data rumah pompa | Admin, petugas |
| UC-0008 | Menghapus data rumah pompa | Admin |
| UC-0009 | Melihat data profil | Admin, petugas, pengawas |
| UC-0010 | Mengubah data profil | Admin, petugas, pengawas |
| UC-0011 | Melihat detail rumah pompa, data ketinggian air dan cuaca | Admin, petugas, pengawas |
| UC-0012 | Mendapat peringatan potensi banjir | Petugas |
| UC-0013 | Mencari data user | Admin |
| UC-0014 | Mencari data rumah pompa | Admin, Pengawas |



Gambar 3.3 Diagram Kasus Penggunaan

1. Melihat Data *User* (UC-0001)

Kasus penggunaan nomor UC-0001 ini diakses ketika admin akan melihat data *user* yang terdaftar pada sistem. Hasil akhir dari kasus penggunaan ini adalah akan ditampilkan data user dalam bentuk list. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.5, Gambar 0.1, dan Gambar 0.30.

Tabel 3.6 Spesifikasi Kasus Penggunaan UC-0001

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0001 | |
| **Nama Use Case** | Melihat data *user* | |
| **Aktor** | Admin | |
| **Deskripsi** | Admin dapat melihat data *user* | |
| **Relasi** | - | |
| **Kondisi Awal** | Data user sudah tersimpan pada database | |
| **Kondisi Akhir** | Sistem menampilkan data *user* dalam bentuk list | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1. Memilih menu untuk melihat data *user* |  |
|  | 2. Menampilkan semua data *user* |
| 3. Memilih salah satu data user untuk melihat detail informasi data *user* |  |
|  | 4. Menampilkan detail informasi mengenai data *user* yang dipilih |
| **Alur kejadian alternatif** | **Aktor** | **Sistem** |
|  |  |



Gambar 3.4 Diagram Aktivitas Melihat Data *User*

1. Menambah Data *User* (UC-0002)

Kasus penggunaan nomor UC-0002 ini diakses ketika admin akan menambahkan data *user* ke database. Untuk menambah *user*, sistem akan menampilkan form input *username*, nama, tipe *user*, alamat, nomor telepon, dan *password*. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.6, Gambar 0.2, dan Gambar 0.31.

Tabel 3.7 Spesifikasi Kasus Penggunaan UC-0002

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0002 | |
| **Nama Use Case** | Menambah data *user* | |
| **Aktor** | Admin | |
| **Deskripsi** | Admin dapat menambah data *user* pada sistem | |
| **Relasi** | - | |
| **Kondisi Awal** | - | |
| **Kondisi Akhir** | Data *user* yang ditambahkan tersimpan dalam database | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1. Memilih pilihan untuk menambahkan data *user* |  |
|  | 2. Menampilkan form tambah data *user* |
| 3. Mengisi data *user* yang akan ditambahkan pada form dan memilih untuk menyimpan data *user* |  |
|  | 4. Menyimpan data *user* yang baru ditambah |
| **Alur kejadian alternatif** | **Aktor** | **Sistem** |
|  |  |



Gambar 3.5 Diagram Aktivitas Menambah Data *User*

1. Mengubah Data *User* (UC-0003)

Kasus penggunaan nomor UC-0003 ini diakses ketika admin hendak mengubah suatu data *user*. Pada kasus penggunaan ini, sistem menampilkan form untuk pengisian perubahan data pada *user* yang ingin diubah. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.7, Gambar 0.3, dan Gambar 0.32.

Tabel 3.8 Spesifikasi Kasus Penggunaan UC-0003

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0003 | |
| **Nama Use Case** | Mengubah data *user* | |
| **Aktor** | Admin | |
| **Deskripsi** | Admin dapat mengubah data *user* yang dipilih | |
| **Relasi** | - | |
| **Kondisi Awal** | * Data *user* yang akan diubah sudah tersimpan dalam database * Sistem menampilkan semua data *user* | |
| **Kondisi Akhir** | Sistem menyimpan data *user* yang diubah | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1. Memilih data *user* yang akan diubah |  |
|  | 2. Menampilkan detail informasi data *user* yang akan diubah |
| 3.Memilih menu untuk mengubah data *user* |  |
|  | 4.Menampilkan form ubah data *user* |
| 5.Mengubah data *user* pada form yang disediakan dan memilih untuk menyimpan perubahan |  |
|  | 6.Menyimpan data *user* yang sudah diubah |
| **Alur kejadian alternatif** | **Aktor** | **Sistem** |
|  |  |



Gambar 3.6 Diagram Aktivitas Menambah Data *User*

1. Menghapus Data *User* (UC-0004)

Kasus penggunaan nomor UC-0004 ini dilakukan admin ketika akan menghapus suatu data *user*. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.8, Gambar 0.4, dan Gambar 0.33.

Tabel 3.9 Spesifikasi Kasus Penggunaan UC-0004

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0004 | |
| **Nama Use Case** | Menghapus data *user* | |
| **Aktor** | Admin | |
| **Deskripsi** | Admin dapat menghapus data *user* yang dipilih | |
| **Relasi** | - | |
| **Kondisi Awal** | * Data user yang akan dihapus sudah tersimpan dalam database * Sistem menampilkan semua data *user* | |
| **Kondisi Akhir** | Data *user* yang dihapus tidak akan ditampilkan lagi | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1. Memilih data *user* yang akan dihapus |  |
|  | 2. Menampilkan detail informasi data *user* yang akan dihapus |
| 3.Memilih untuk menghapus data *user* |  |
|  | 4. Menghapus data *user* yang dipilih |
| **Alur kejadian alternatif** | **Aktor** | **Sistem** |
|  |  |



Gambar 3.7 Diagram Aktivitas Menghapus Data User

1. Melihat Data Rumah Pompa (UC-0005)

Kasus penggunaan nomor UC-0005 ini diakses ketika admin dan pengawas akan melihat data rumah pompa yang ada di Surabaya. Data rumah pompa akan ditampilkan dalam bentuk *list* dan peta. Admin hanya akan melihat data rumah pompa dalam bentuk list, sedangkan pengawas dapat melihat rumah pompa dalam bentuk *list* dan peta. Pada rumah pompa dalam bentuk *list*, pengawas dapat melakukan *filter* pada data rumah pompa berdasarkan status potensi banjir. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.9, Gambar 0.5, dan Gambar 0.34.

Tabel 3.10 Spesifikasi Kasus Penggunaan UC-0005

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0005 | |
| **Nama Use Case** | Melihat data rumah pompa | |
| **Aktor** | Admin, Pengawas | |
| **Deskripsi** | Admin dan Pengawas dapat melihat data rumah pompa | |
| **Relasi** | - | |
| **Kondisi Awal** | Data rumah pompa telah tersimpan dalam database | |
| **Kondisi Akhir** | Sistem sudah menampilkan data rumah pompa | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1. Memilih menu untuk melihat data rumah pompa |  |
|  | 2. Menampilkan semua data rumah pompa |
| **Alur kejadian alternatif** | **Aktor** | **Sistem** |
|  |  |



Gambar 3.8 Diagram Aktivitas Melihat Data Rumah Pompa

1. Menambah Data Rumah Pompa (UC-0006)

Kasus penggunaan nomor UC-0006 ini diakses ketika admin akan menambah data rumah pompa pada sistem. Untuk menambah rumah pompa, sistem akan menampilkan form input nama rumah pompa, alamat, nomor telepon, latitude, longitude, *threshold* (ambang batas) ketinggian air, dan ketinggian saluran masuk air. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.10, Gambar 0.6, dan Gambar 0.35.

Tabel 3.11 Spesifikasi Kasus Penggunaan UC-0006

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0006 | |
| **Nama Use Case** | Menambah data rumah pompa | |
| **Aktor** | Admin | |
| **Deskripsi** | Admin dapat menambah data rumah pompa pada sistem | |
| **Relasi** | - | |
| **Kondisi Awal** | - | |
| **Kondisi Akhir** | Sistem sudah menyimpan data rumah pompa yang ditambah oleh admin | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1. Memilih menu untuk menambah data rumah pompa |  |
|  | 2. Menampilkan form tambah data rumah pompa |
| 3. Mengisi data rumah pompa yang akan ditambahkan dan memilih untuk menyimpan data rumah pompa |  |
|  | 4.Menyimpan data rumah pompa yang baru ditambah |
| **Alur kejadian alternatif** | **Aktor** | **Sistem** |
|  |  |



Gambar 3.9 Diagram Aktivitas Menambah Data Rumah Pompa

1. Mengubah Data Rumah Pompa (UC-0007)

Kasus penggunaan nomor UC-0007 ini diakses ketika admin akan mengubah suatu data rumah pompa. Pada kasus penggunaan ini, sistem menampilkan form untuk pengisian perubahan data pada rumah pompa yang ingin diubah. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.11, Gambar 0.7, dan Gambar 0.36.

Tabel 3.12 Spesifikasi Kasus Penggunaan UC-0007

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0007 | |
| **Nama Use Case** | Mengubah data rumah pompa | |
| **Aktor** | Admin, Petugas | |
| **Deskripsi** | Admin dapat mengubah data rumah pompa yang ingin diubah | |
| **Relasi** | - | |
| **Kondisi Awal** | Data rumah pompa yang akan diubah sudah tersimpan dalam database | |
| **Kondisi Akhir** | Sistem menyimpan data rumah pompa yang sudah diubah | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1. Memilih menu untuk melihat data rumah pompa yang akan diubah |  |
|  | 2. Menampilkan detail informasi data rumah pompa yang akan diubah |
| 3. Memilih menu untuk mengubah data rumah pompa |  |
|  | 4.Menampilkan form ubah data rumah pompa |
| 5.Mengubah data rumah pompa pada form yang disediakan dan memilih untuk menyimpan perubahan |  |
|  | 6. Menyimpan data rumah pompa yang sudah diubah |
| **Alur kejadian alternatif** | **Aktor** | **Sistem** |
|  |  |



Gambar 3.10 Diagram Aktivitas Mengubah Data Rumah Pompa

1. Menghapus Data Rumah Pompa (UC-0008)

Kasus penggunaan nomor UC-0008 ini diakses ketika ketika admin akan menghapus suatu data rumah pompa. Ketika suatu rumah pompa sudah terhapus, maka rumah pompa tersebut tidak akan ditampilkan lagi pada dafatr data rumah pompa. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.12, Gambar 0.8, dan Gambar 0.37.

Tabel 3.13 Spesifikasi Kasus Penggunaan UC-0008

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0008 | |
| **Nama Use Case** | Menghapus data rumah pompa | |
| **Aktor** | Admin | |
| **Deskripsi** | Admin dapat menghapus data rumah pompa yang diperlukan | |
| **Relasi** | - | |
| **Kondisi Awal** | * Data user yang akan dihapus sudah tersimpan dalam database * Sistem menampilkan semua data rumah pompa | |
| **Kondisi Akhir** | Data rumah pompa yang dihapus tidak akan ditampilkan lagi | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1. Memilih data rumah pompa yang akan dihapus |  |
|  | 2. Menampilkan detail informasi data rumah pompa yang akan dihapus |
| 3. Memilih untuk menghapus data rumah pompa |  |
|  | 4. Menghapus data rumah pompa yang dipilih |
| **Alur kejadian alternatif** | **Aktor** | **Sistem** |



Gambar 3.11 Diagram Aktivitas Menghapus Data Rumah Pompa

1. Melihat Data Profil (UC-0009)

Kasus penggunaan nomor UC-0009 ini diakses ketika admin, petugas, dan pengawas telah *login* ke akunnya sesuai dengan hak aksesnya. Lalu, akan ditampilkan detail informasi mengenai profil *user*. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.13, Gambar 0.9, dan Gambar 0.38.

Tabel 3.14 Spesifikasi Kasus Penggunaan UC-0009

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0009 | |
| **Nama Use Case** | Melihat data profil | |
| **Aktor** | Admin, pengawas, petugas | |
| **Deskripsi** | Admin, pengawas, dan petugas dapat melihat data profil | |
| **Relasi** | - | |
| **Kondisi Awal** | Admin, pengawas, dan petugas telah *login* ke akunnya | |
| **Kondisi Akhir** | Sistem menampilkan data profil *user* | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1. Memilih menu untuk menampilkan data profil |  |
|  | 2. Menampilkan detail informasi data profil |
| **Alur kejadian alternatif** | **Aktor** | **Sistem** |
|  |  |



Gambar 3.12 Diagram Aktivitas Melihat Data Profil

1. Mengubah Data Profil (UC-0010)

Kasus penggunaan nomor UC-0010 ini diakses ketika admin, pengawas, atau petugas akan mengubah data profilnya. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.14, Gambar 0.10, dan Gambar 0.39.

Tabel 3.15 Spesifikasi Kasus Penggunaan UC-0010

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0010 | |
| **Nama Use Case** | Mengubah data profil | |
| **Aktor** | Admin, pengawas, dan petugas | |
| **Deskripsi** | Admin dapat mengubah data profilnya. | |
| **Relasi** | - | |
| **Kondisi Awal** | * Data admin, pengawas, atau petugas yang akan mengubah profilnya telah tersimpan pada database. * Sistem menampilkan detail informasi profil admin, petugas, atau pengawas. | |
| **Kondisi Akhir** | Sistem menyimpan data admin, pengawas, atau petugas yang telah diubah | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1.a. Memilih menu untuk mengubah profil |  |
|  | 2. Menampilkan form ubah data profil |
| 3. Mengubah data profil pada form yang disediakan dan memilih untuk menyimpan perubahan |  |
|  | 4. Menyimpan data profil yang sudah diubah |
| **Alur kejadian alternatif** | **Aktor** | **Sistem** |
| 1.b. Memilih menu untuk mengubah *password* |  |
|  | 1.b.1. Menampilkan form ubah *password* |
| 7.c. Mengisi form dengan *password* yang akan diubah dan memilih untuk menyimpan perubahan |  |
|  | 7.c.1.Menyimpan data *password* yang sudah diubah |



Gambar 3.13 Diagram Aktivitas Mengubah Data Profil

1. Melihat Detail Rumah Pompa, Data Ketinggian Air dan Cuaca (UC-0011)

Kasus penggunaan nomor UC-0011 ini diakses ketika pengawas dan petugas akan melihat detail rumah pompa beserta data ketinggian air dan cuaca dari sebuah rumah pompa. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.14, Gambar 0.11, dan Gambar 0.40.

Tabel 3.16 Spesifikasi Kasus Penggunaan UC-0011

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0011 | |
| **Nama Use Case** | Melihat detail rumah pompa, data ketinggian air dan cuaca | |
| **Aktor** | Admin, Pengawas, Petugas | |
| **Deskripsi** | Admin, pengawas, dan petugas dapat melihat detail rumah pompa beserta data ketingian air dan cuaca dari suatu rumah pompa | |
| **Relasi** | - | |
| **Kondisi Awal** | Data rumah pompa, ketinggian air dan cuaca telah tersimpan pada database | |
| **Kondisi Akhir** | Sistem menampilkan detail informasi rumah pompa, data ketinggian air dan cuaca dari rumah pompa yang dipilih | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1. Memilih untuk melihat detail rumah pompa, ketinggian air, dan cuaca |  |
|  | 2. Menampilkan data rumah pompa beserta data ketinggian air dan cuaca |
| **Alur kejadian alternatif** | **Aktor** | **Sistem** |
|  |  |  |



Gambar 3.14 Diagram Aktivitas Melihat Detail Rumah Pompa, Data Ketinggian Air, dan Cuaca

1. Mendapat Peringatan Potensi Banjir (UC-0012)

Kasus penggunaan nomor UC-0012 ini terjadi apabila terjadi apabila terdapat potensi pada suatu rumah pompa. Peringatan potensi banjir akan dikirim kepada petugas yang bekerja pada rumah pompa yang berpotensi banjir. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.15, Gambar 0.12, dan Gambar 0.41.

Tabel 3.17 Spesifikasi Kasus Penggunaan UC-0012

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0012 | |
| **Nama Use Case** | Mendapat peringatan potensi banjir | |
| **Aktor** | Petugas | |
| **Deskripsi** | Petugas akan mendapat peringatan potensi banjir apabila rumah pompa tempat ia bekerja berpotensi banjir | |
| **Relasi** | - | |
| **Kondisi Awal** | Petugas berhasil *login* pada akunnya. | |
| **Kondisi Akhir** | Ditampilkannya peringatan potensi banjir | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1. Melakukan *login* pada akunnya |  |
|  | 2. Mengirim notifikasi potensi banjir |
| 3.Mendapatkan peringatan potensi banjir |  |
| **Alur kejadian alternative** | **Aktor** | **Sistem** |
|  |  |



Gambar 3.15 Diagram Aktivitas Mendapat Peringatan Potensi Banjir

1. Mencari Data *User* (UC-0013)

Kasus penggunaan nomor UC-0013 ini diakses ketika admin akan melakukan pencarian pada data *user*. Pencarian data *user* dilakukan dengan memasukkan nama *user* yang dicari pada form. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.17, Gambar 0.13, dan Gambar 0.42.

Tabel 3.18 Spesifikasi Kasus Penggunaan UC-0013

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0013 | |
| **Nama Use Case** | Mencari data user | |
| **Aktor** | Admin | |
| **Deskripsi** | Admin dapat mencari data *user* dengan memasukkan nama *user* | |
| **Relasi** | - | |
| **Kondisi Awal** | Data *user* telah tersimpan pada database. | |
| **Kondisi Akhir** | Sistem menampilkan data *user* yang dicari | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1. Memilih menu untuk melihat data *user* |  |
|  | 2.Menampilkan semua data *user* |
| 3. Memasukkan nama *user* yang akan dicari pada form |  |
|  | 4. Menampilkan data *user* yang dicari |
| **Alur kejadian alternative** | **Aktor** | **Sistem** |
|  |  |
|  |  |  |



Gambar 3.16 Diagram Aktivitas Mencari Data *User*

1. Mencari Data Rumah Pompa (UC-0014)

Kasus penggunaan nomor UC-0014 ini diakses ketika admin akan melakukan pencarian pada data rumah pompa. Pencarian data rumah pompa dilakukan dengan memasukkan nama rumah pompa yang dicari pada form. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.18, Gambar 0.14, dan Gambar 0.43.

Tabel 3.19 Spesifikasi Kasus Penggunaan UC-0014

|  |  |  |
| --- | --- | --- |
| **Kode Use Case** | UC-0014 | |
| **Nama Use Case** | Mencari data rumah pompa | |
| **Aktor** | Admin, Pengawas | |
| **Deskripsi** | Admin dapat mencari data rumah pompa dengan memasukkan nama rumah pompa | |
| **Relasi** | - | |
| **Kondisi Awal** | * Data rumah pompa telah tersimpan pada database. * Sistem menampilkan semua data rumah pompa dan form untuk mencari data rumah pompa. | |
| **Kondisi Akhir** | Sistem menampilkan data rumah pompa yang dicari | |
| **Alur kejadian normal** | **Aktor** | **Sistem** |
| 1. Memilih menu untuk melihat data rumah pompa |  |
|  | 2. Menampilkan semua data rumah pompa |
| 3. Memasukkan nama rumah pompa yang akan dicari pada form |  |
|  | 4. Menampilkan data rumah pompa yang dicari |
| **Alur kejadian alternatif** | **Aktor** | **Sistem** |
|  |  |



Gambar 3.17 Diagram Aktivitas Mencari Data Rumah Pompa

1. Perancangan

Perancangan dalam subbab ini membahas perancangan dari aplikasi tugas akhir. Subbab ini terdiri dari lingkungan perancangan perangkat lunak, perancangan arsitektur sistem, perancangan diagram kelas, perancangan basis data, dan perancangan antarmuka pengguna.

1. Lingkungan Perancangan Perangkat Lunak

Spesifikasi perangkat keras serta perangkat lunak yang digunakan dalam tahap perancangan perangkat lunak tugas akhir ini seperti dijelaskan pada Tabel 3.19.

Tabel 3.20 Lingkungan Perancangan Perangkat Lunak

|  |  |  |
| --- | --- | --- |
| Perangkat Keras | Komputer | Asus A46CB |
| Prosesor | Intel® Core™ i5-3317U CPU @ 1.70GHz (1.70GHz) |
| Memori Primer | 4 GB |
| Memori Sekunder | 500 GB |
| Perangkat Lunak | Sistem Operasi | *Windows* 8 Pro 64-bit |
| Perangkat Lunak | Android Studio v2.2.2, Justinmind Prototyper 7.1.1, Sybase PowerDesigner 16.5, Microsoft Word 2013, Jetbrains PhpStorm 2016.2.2 |

1. Perancangan Arsitektur Sistem
2. Perancangan Diagram Kelas

Perancangan Diagram Kelas dapat dilihat pada Gambar 3.3, Gambar 3.4, Gambar 3.5, Gambar 3.6, Gambar 3.7, Gambar 3.8, Gambar 3.9, dan Gambar 3.10.

1. Perancangan Struktur Data

Dalam membuat suatu aplikasi perangkat bergerak, diperlukan analisis kebutuhan berupa perancangan basis data. Basis data yang digunakan adalah PostgreSQL. PostgreSQL dipilih menjadi basis data aplikasi ini karena sifat RDBMS yang *open source*, mudah digunakan, dan memiliki fitur yang lebih lengkap.

Rancangan basis data ditampilkan dalam bentuk *Conceptual Data Model* (selanjutnya disebut CDM) dan *Physical Data Model* (selanjutnya disebut PDM). Penjelasan lebih lengkap berupa CDM dan PDM terdapat pada Gambar 3.11 dan Gambar 3.12.

1. Tabel *User*

Tabel user adalah tabel yang digunakan untuk menyimpan data-data para pengguna yang memiliki hak akses pada sistem. Tabel ini merupakan tabel utama yang mempunyai atribut-atribut, yaitu *username*, nama *user*, alamat *user*, no telp *user*, *password* dan token. Token berfungsi untuk menyimpan id dari *device* yang digunakan oleh pengguna untuk *login* pada sistem. Hal ini bertujuan dalam hal pengiriman *push notification* potensi banjir untuk pengguna.

1. Tabel Rumah Pompa

Tabel rumah pompa adalah tabel yang digunakan untuk menyimpan data-data rumah pompa. Tabel ini merupakan tabel utama yang mempunyai atribut-atribut, yaitu id rumah pompa, nama, jalan, no telp rumah pompa, *threshold* tinggi air, *latitude*, *longitude*, ketinggian sungai, *alert*. *Threshold* tinggi air berfungsi untuk menyimpan data ambang batas ketinggian air dari suatu rumah pompa.

*Alert* merupakan status potensi banjir dari suatu rumah pompa. Apabila *alert* bernilai *true*, maka terjadi potensi banjir. Sedangkan jika *alert* bernilai *false*, maka tidak terjadi potensi banjir.

1. Tabel Data

Tabel data adalah tabel yang digunakan untuk menyimpan data ketinggian air dan cuaca pada suatu rumah pompa. Tabel ini merupakan tabel utama yang mempunyai atribut-atribut, yaituid data, id rumah pompa, ketinggian air, cuaca, waktu, *chance of rain*. *Chance of rain* merupakan atribut untuk menyimpan data presentase kemungkinan terjadinya banjir pada suatu rumah pompa.

1. Tabel *Role*

Tabel *role* adalah tabel yang digunakan untuk menyimpan data macam-macam hak akses untuk masuk ke dalam sistem. Tabel ini merupakan tabel utama yang mempunyai atribut-atribut yaitu id *role* dan nama *role*. Role terdiri dari admin, petugas, dan pengawas.

1. Tabel Roleuser

Tabel roleuser adalah tabel yang digunakan untuk mencatat *role* untuk setiap *user*. Tabel ini merupakan tabel utama yang mempunyai atribut-atribut, yaitu *username*, id *role*, dan *isactive*. *Isactive* berfungsi sebagai tanda aktif atau tidak aktifnya *username* dengan *role* yang ada pada data.

1. Tabel User\_rumahpompa

Tabel user\_rumahpompa adalah tabel yang digunakan untuk mencatat rumah pompa yang dibawahi oleh setiap *user*. Tabel ini merupakan tabel utama yang mempunyai atribut-atribut, yaituid rumah pompa, *username*, dan *isactive*. *Isactive* berfungsi sebagai tanda aktif atau tidak aktifnya *username* dengan rumah pompa yang ada pada data.



Gambar 3.18 Diagram Kelas Daerah



Gambar 3.19 Diagram Kelas Bencana



Gambar 3.20 Diagram Kelas Dokter



Gambar 3.21 Diagram Kelas Jarak



Gambar 3.22 Diagram Kelas Keahlian



Gambar 3.23 Diagram Kelas Rumah Sakit

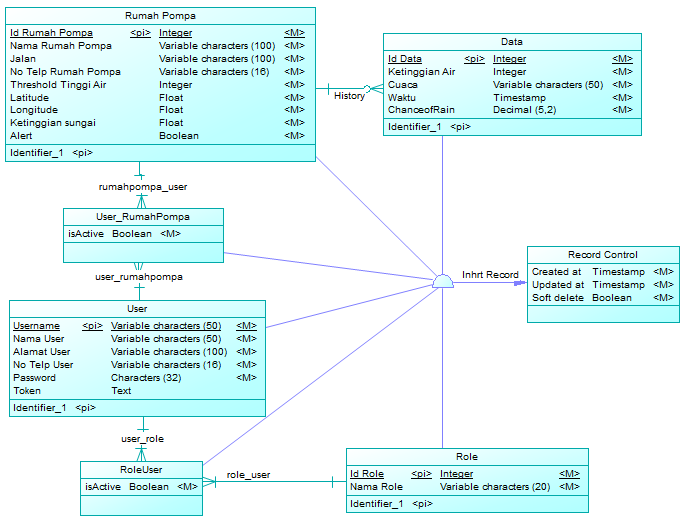


Gambar 3.24 Diagram Kelas Penugasan



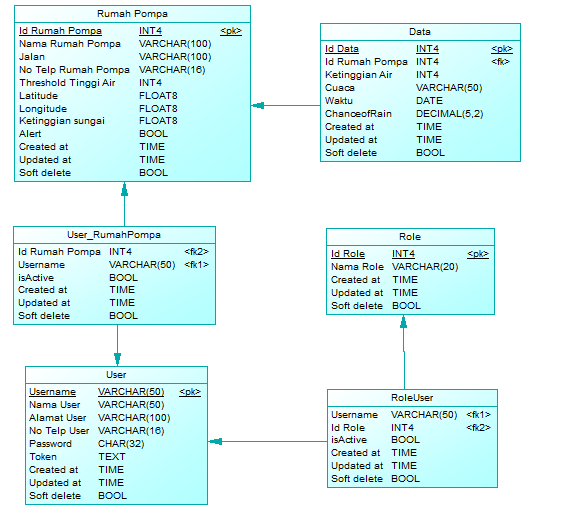
Gambar 3.25 Diagram Kelas Daftar Penugasan

1. *Conceptual Data Modeling* (CDM)



Gambar 3.26 Conceptual Data Modeling

1. *Physical Data Modeling* (PDM)



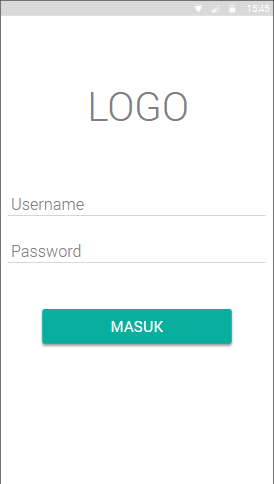
Gambar 3.27 *Physical Data Modeling*

1. Perancangan Antarmuka Pengguna

Perancangan antarmuka pengguna merupakan hal yang penting dalam melakukan perancangan perangkat lunak. Antarmuka pengguna yang berhubungan langsung dengan aktor harus memiliki kemudahan-kemudahan dan tampilan yang menarik bagi penggunanya. Aplikasi ini memiliki dua hak akses, yaitu admin dan dokter dengan halaman *login* pengguna yang sama. Pada hak akses admin, aplikasi ini memiliki beberapa antarmuka, yaitu halaman penugasan, penugasan hasil, penugasan ulang, dan riwayat progres penugasan. Pada hak akses dokter, terdapat halaman daftar penugasan, riwayat progres penugasan, dan konfirmasi penugasan selesai.

1. Rancangan Halaman Antarmuka Login Pengguna

Halaman ini digunakan oleh *user* untuk masuk ke sistem. *User* harus mengisi isian *username* dan *password* untuk dapat masuk ke dalam sistem. Rancangan halaman dapat dilihat pada Gambar 3.13. Atribut antarmuka dapat dilihat pada Tabel 3.21.



Gambar 3.28 Rancangan Halaman Antarmuka Login Pengguna

Tabel 3.21 Atribut Antarmuka Login *User*

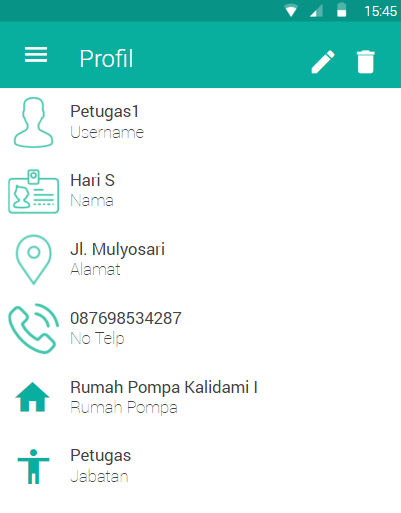
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *edt\_username* | *Edit text* | Input untuk memasukkan *username* *user* | *String* |
| 2 | *edt\_password* | *Edit text* | Input untuk memasukkan *password* *user* | *String* |
| 3 | *btn\_login* | *Button* | Tombol untuk menuju ke halaman utama *user* | *Button click* |

1. Rancangan Halaman Antarmuka Melihat Data User

Halaman ini akan ditampilkan ketika admin akan melihat data user. Rancangan halaman dapat dilihat pada Gambar 3.14. Atribut antarmuka dapat dilihat pada Tabel 3.22.



Gambar 3.29 Rancangan Halaman Antarmuka Melihat Data *User*



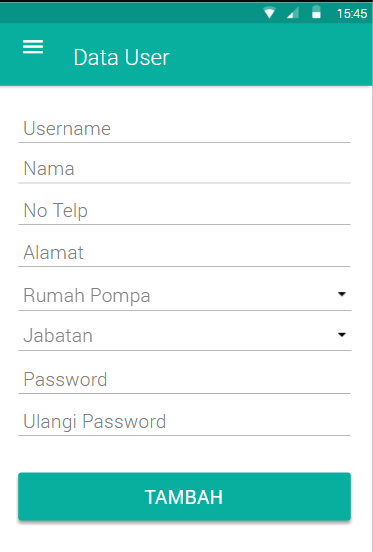
Gambar 3.30 Rancangan Halaman Antarmuka Melihat Detail Data *User*

Tabel 3.22 Atribut Antarmuka Melihat Data *User*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *spinner\_user* | *Drop down list* | Berisi pilihan *role* untuk menampilkan *user* berdasarkan *role* yang dipilih | *Button click* |
| 2 | *ac\_searchuser* | *Auto complete text view* | Input untuk memasukkan nama *user* yang dicari | *Text* |
| 3 | *lv\_user* | *List view* | Berisi list data *user* | *List* |
| 4 | *fab\_adduser* | *Floating action button* | Tombol untuk menuju halaman menambah data *user* | *Button click* |
| 5 | *tv\_username* | *Text view* | Berisi informasi username dari *user* | *String* |
| 6 | *tv\_namauser* | *Text view* | Berisi informasi nama *user* | *String* |
| 7 | *tv\_alamatuser* | *Text view* | Berisi informasi alamat *user* | *String* |
| 8 | *tv\_notelpuser* | *Text view* | Berisi informasi nomor telepon *user* | *String* |
| 9 | *tv\_rumah pompauser* | *Text view* | Berisi informasi rumah pompa yang dibawahi oleh *user* | *String* |
| 10 | *tv\_jabatanuser* | *Text view* | Berisi informasi role yang dimiliki *user* | *String* |
| 11 | *btn\_edituser* | *Button* | Tombol untuk menuju halaman mengubah data *user* | *Button click* |
| 12 | *btn\_deleteuser* | *Button* | Tombol untuk menghapus data *user* | *Button click* |

1. Rancangan Halaman Antarmuka Menambah Data *User*

Halaman ini akan ditampilkan setelah admin menekan *button* *fab\_adduser* pada halaman melihat data user. Halaman ini menampilkan form untuk menambah data *user*. Rancangan halaman dapat dilihat pada Gambar 3.15. Atribut antarmuka dapat dilihat pada Tabel 3.23.



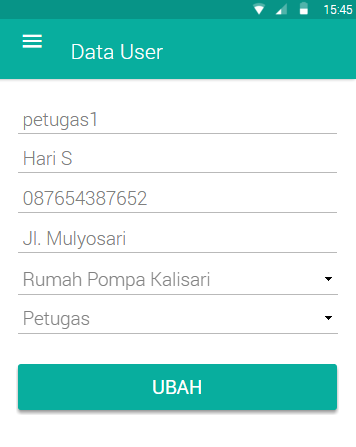
Gambar 3.31 Rancangan Halaman Antarmuka Menambah Data *User*

Tabel 3.23 Atribut Antarmuka Menambah Data *User*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *edt\_username* | *Edit text* | Input untuk memasukkan username | *String* |
| 2 | *edt \_nama* | *Edit text* | Input untuk memasukkan nama *user* | *String* |
| 3 | *edt\_alamat* | *Edit text* | Input untuk memasukkan alamat *user* | *String* |
| 4 | *edt\_notelp* | *Edit text* | Input untuk memasukkan nomor telepon *user* | *String* |
| 5 | *spinner\_ru mahpompa* | *Drop down list* | Berisi pilihan rumah pompa yang dibawahi oleh *user* | *Button click* |
| 6 | *spinner\_jaba tan* | *Drop down list* | Berisi pilihan *role* | *Button click* |
| 7 | *edt\_password* | *Edit text* | Input untuk memasukkan password *user* | *String* |
| 8 | *edt\_re password* | *Edit text* | Input untuk memasukkan ulang password. | *String* |
| 9 | *btn\_adduser* | *Button* | Tombol aksi untuk menyimpan data *user* yang baru ditambahkan | *Button click* |

1. Rancangan Halaman Antarmuka Mengubah Data *User*

Halaman ini akan ditampilkan ketika admin akan mengubah data user yang diinginkan. Halaman ini akan ditampilkan setelah admin menekan tombol *edit btn\_edituser* pada Gambar 3.16. Rancangan halaman dapat dilihat pada Gambar 3.19. Atribut antarmuka dapat dilihat pada Tabel 3.24.



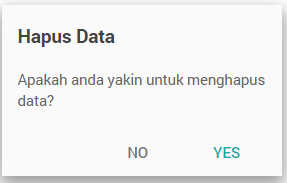
Gambar 3.32 Rancangan Halaman Antarmuka Mengubah Data *User*

Tabel 3.24 Atribut Antarmuka Mengubah Data User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *edt\_username* | *Edit text* | Input untuk memasukkan username | *String* |
| 2 | *edt \_nama* | *Edit text* | Input untuk memasukkan nama *user* | *String* |
| 3 | *edt\_alamat* | *Edit text* | Input untuk memasukkan alamat *user* | *String* |
| 4 | *edt\_notelp* | *Edit text* | Input untuk memasukkan nomor telepon *user* | *String* |
| 5 | *spinner\_ru mahpompa* | *Drop down list* | Berisi pilihan rumah pompa yang dibawahi oleh *user* | *Button click* |
| 6 | *spinner\_jaba tan* | *Drop down list* | Berisi pilihan *role* | *Button click* |
| 7 | *btn\_edituser* | *Button* | Tombol aksi untuk menyimpan data *user* yang diubah | *Button click* |

1. Rancangan Antarmuka Menghapus Data *User* dan Rumah Pompa

Antarmuka ini berupa dialog yang akan ditampilkan pada sistem ketika admin akan menghapus data *user* atau data rumah pompa. Antarmuka ini akan ditampilkan setelah admin menekan tombol hapus *btn\_deleteuser* pada Gambar 3.16 dan *btn\_deleterumahpompa* pada Gambar 3.40. Rancangan halaman dapat dilihat pada Gambar 3.17. Atribut antarmuka dapat dilihat pada Tabel 3.24.



Gambar 3.33 Rancangan Antarmuka Menghapus Data *User* dan Rumah Pompa

Tabel 3.25 Atribut Antarmuka Menghapus Data User dan Rumah Pompa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *dialog\_delete* | *Dialog* | Informasi nama daerah yang jadwal penugasannya diulang | *String* |
| 2 | *btn\_yes* | *Button* | Tombol aksi untuk menyetujui menghapus data *user* yang dipilih | *Button click* |
| 3 | *btn\_no* | *Button* | Tombol aksi untuk menolak menghapus data *user* yang dipilih | *Button click* |

1. Rancangan Halaman Antarmuka Melihat Data Rumah Pompa

Halaman ini akan ditampilkan ketika admin dan pengawas akan melihat data semua rumah pompa di Surabaya. Rancangan halaman untuk melihat semua rumah pompa dapat dilihat pada Gambar 3.18. Atribut antarmuka dapat dilihat pada Tabel 3.25.



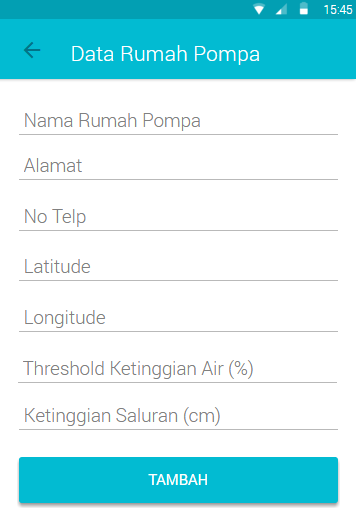
Gambar 3.34 Rancangan Halaman Antarmuka Melihat Data Rumah Pompa

Tabel 3.26 Atribut Antarmuka Melihat Data Rumah Pompa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *spinner\_ru mahpompa* | *Drop down list* | Berisi pilihan untuk menampilkan list rumah pompa berdasarkan status potensi banjir | *Button click* |
| 2 | *ac\_rumahpompa* | *Auto complete text view* | Input untuk memasukkan nama rumah pompa yang dicari | *String* |
| 3 | *lv\_rumah pompa* | *List view* | Berisi list untuk menampilkan data rumah pompa | *List* |
| 4 | *btn\_addrumahpompa* | *Button* | Tombol untuk menuju halaman menambah data rumah pompa | *Button click* |

1. Rancangan Halaman Antarmuka Menambah Data Rumah Pompa

Halaman ini akan ditampilkan ketika admin akan menambah data rumah pompa baru. Halaman ini akan muncul setelah admin menekan tombol *btn\_addrumahpompa* pada Gambar 3.18*.*  Rancangan halaman dapat dilihat pada Gambar 3.19 . Atribut antarmuka dapat dilihat pada Tabel 3.26.



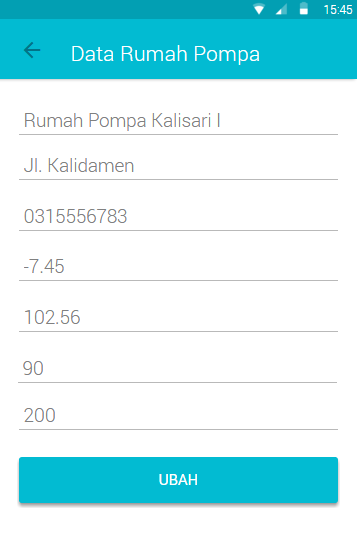
Gambar . Rancangan Halaman Antarmuka Menambah Data Rumah Pompa

Tabel 3.27 Atribut Antarmuka Menambah Data Rumah Pompa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *edt\_nama rumahpompa* | *Edit text* | Input untuk memasukkan nama rumah pompa | *String* |
| 2 | *edt\_alamat rumahpompa* | *Edit text* | Input untuk memasukkan alamat rumah pompa | *String* |
| 3 | *edt\_notelp rumahpompa* | *Edit text* | Input untuk memasukkan nomor telepon rumah pompa | *String* |
| 4 | *edt\_latitude* | *Edit text* | Input untuk memasukkan latitude rumah pompa | *Float* |
| 5 | *edt\_longitude* | *Edit text* | Input untuk memasukkan longitude rumah pompa | *Float* |
| 6 | *edt\_threshold tinggiair* | *Edit text* | Input untuk memasukkan ambang batas ketinggian air | *Integer* |
| 7 | *edt\_ketinggiansaluran* | *Edit text* | Input untuk memasukkan ketinggian saluran masuknya air pada rumah pompa | *Float* |
| 8 | *btn\_addrumahpompa* | *Button* | Tombol untuk menyimpan data rumah pompa yang baru ditambahkan | *Button click* |

1. Rancangan Halaman Antarmuka Mengubah Data Rumah Pompa

Halaman ini akan ditampilkan ketika admin akan mengubah data rumah pompa yang dipilih. Halaman ini akan ditampilkan setelah admin memilih tombol *btn\_editrumahpompa* pada Gambar 3.19. Rancangan halaman dapat dilihat pada Gambar 3.21. Atribut antarmuka dapat dilihat pada Tabel 3.41.



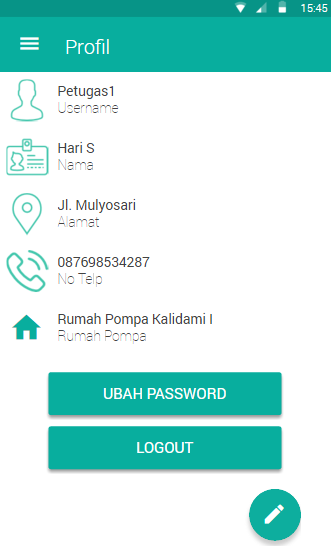
Gambar . Rancangan Halaman Antarmuka Mengubah Data Rumah Pompa

Tabel 3.28 Atribut Antarmuka Mengubah Data Rumah Pompa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *edt\_nama rumahpompa* | *Edit text* | Input untuk memasukkan nama rumah pompa | *String* |
| 2 | *edt\_alamat rumahpompa* | *Edit text* | Input untuk memasukkan alamat rumah pompa | *String* |
| 3 | *edt\_notelp rumahpompa* | *Edit text* | Input untuk memasukkan nomor telepon rumah pompa | *String* |
| 4 | *edt\_latitude* | *Edit text* | Input untuk memasukkan latitude rumah pompa | *Float* |
| 5 | *edt\_longitude* | *Edit text* | Input untuk memasukkan longitude rumah pompa | *Float* |
| 6 | *edt\_threshold tinggiair* | *Edit text* | Input untuk memasukkan ambang batas ketinggian air | *Integer* |
| 7 | *edt\_ketinggiansaluran* | *Edit text* | Input untuk memasukkan ketinggian saluran masuknya air pada rumah pompa | *Float* |
| 8 | *btn\_edit* | *Button* | Tombol untuk menyimpan data rumah pompa yang baru diubah | *Button click* |

1. Rancangan Halaman Antarmuka Melihat Data Profil

Halaman ini akan ditampilkan setelah *user login* pada akunnya. Rancangan halaman dapat dilihat pada Gambar 3.22. Atribut antarmuka dapat dilihat pada Tabel 3.28.



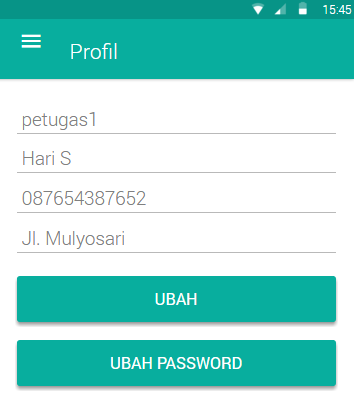
Gambar . Rancangan Halaman Antarmuka Melihat Data Profil

Tabel 3.29 Atribut Antarmuka Melihat Data Profil

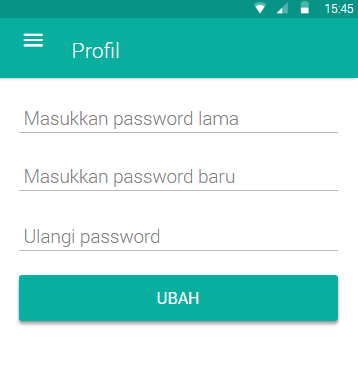
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *tv\_username* | *Text view* | Berisi informasi username dari *user* | *String* |
| 2 | *tv\_namauser* | *Text view* | Berisi informasi mengenai nama *user* | *String* |
| 3 | *tv\_alamatuser* | *Text view* | Berisi informasi mengenai alamat *user* | *String* |
| 4 | *tv\_notelpuser* | *Text view* | Berisi informasi mengenai nomor telepon *user* | *String* |
| 5 | *tv\_rumah*  *pompauser* | *Text view* | Berisi informasi mengenai rumah pompa yang dibawahi oleh *user* | *String* |
| 6 | *btn\_logout* | *Button* | Tombol untuk *logout* akun | *Button click* |
| 7 | *btn\_editprofil* | *Button* | Tombol untuk menuju ke halaman mengubah data profil | *Button click* |
| 8 | *btn\_editpassword* | *Button* | Tombol untuk menuju ke halaman mengubah *password* | *Button click* |

1. Rancangan Halaman Antarmuka Mengubah Data Profil

Pada rancangan ini, terdiri dari dua halaman, yaitu halaman untuk mengubah data profil dan halaman untuk mengubah *password*. Halaman mengubah data profil akan ditampilkan setelah *user* menekan tombol *btn\_editprofil* pada Gambar 3.22. Sedangkan, halaman untuk mengubah password akan ditampilkan setelah *user* menekan tombol *btn\_editpassword* pada Gambar 3.22. Rancangan halaman dapat dilihat pada Gambar 3.23. Atribut antarmuka dapat dilihat pada Tabel 3.29.



Gambar 3.38 Rancangan Halaman Antarmuka Mengubah Data Pofil



Gambar 3.39 Rancangan Halaman Antarmuka Mengubah Password

Tabel 3.30 Atribut Antarmuka Mengubah Data Profil

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *edt\_username* | *Edit text* | Input untuk memasukkan *username* | *String* |
| 2 | *edt\_namauser* | *Edit text* | Input untuk memasukkan nama *user* | *String* |
| 3 | *edt\_alamat user* | *Edit text* | Input untuk memasukkan alamat *user* | *String* |
| 4 | *edt\_notelpuser* | *Edit text* | Input untuk memasukkan nomor telepon *user* | *String* |
| 5 | *edt\_password lama* | *Edit text* | Input untuk memasukkan *password* lama *user* | *String* |
| 6 | *edt\_password baru* | *Edit text* | Input untuk memasukkan *password* baru *user* | *String* |
| 7 | *edt\_re password* | *Edit text* | Input untuk memasukkan ulang *password* baru | *String* |
| 8 | *btn\_submit password* | *Button* | Tombol untuk menyimpan data password yang diubah | *Button click* |
| 9 | *btn\_edit* | *Button* | Tombol untuk menyimpan data profil yang diubah | *Button click* |

1. Rancangan Halaman Antarmuka Melihat Detail Rumah Pompa, Data Ketinggian Air dan Cuaca

Pada admin dan pengawas, halaman ini akan ditampilkan setelah admin memilih salah satu rumah pompa pada halaman melihat data rumah pompa yang dapat dilihat pada Gambar 3.18. Pada petugas, halaman ini ditampilkan ketika petugas memilih menu untuk menampilkan rumah pompa yang dibawahinya. Halaman yang akan ditampilkan pada admin memiliki tombol *btn\_edit* dan *btn\_delete* seperti pada Gambar 3.24. Sedangkan, rancangan halaman untuk pengawas dan petugas dapat dilihat pada Gambar 3.25. Atribut antarmuka dapat dilihat pada Tabel 3.30



Gambar . Rancangan Halaman Antarmuka Melihat Data Ketinggian Air dan Cuaca (Admin)



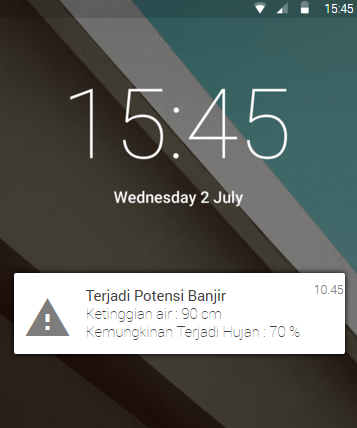
Gambar . Rancangan Halaman Antarmuka Melihat Data Ketinggian Air dan Cuaca

Tabel . Atribut Antarmuka Melihat Detail Data Rumah Pompa, Data Ketinggian Air dan Cuaca

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *tv\_namarumahpompa* | *Text view* | Berisi informasi nama rumah pompa | *String* |
| 2 | *tv\_alamat rumahpompa* | *Text view* | Berisi informasi alamat rumah pompa | *String* |
| 3 | *tv\_telprumah pompa* | *Text view* | Berisi informasi nomor telepon rumah pompa | *String* |
| 4 | *tv\_status* | *Text view* | Berisi informasi status potensi banjir pada suatu rumah pompa | *String* |
| 5 | *tv\_cuaca* | *Text view* | Berisi informasi deskripsi cuaca | *String* |
| 6 | *tv\_pop* | *Text view* | Berisi informasi kemungkinan terjadi banjir dalam presentase | *Float* |
| 7 | *tv\_tinggiair* | *Text view* | Berisi informasi ketinggian air pada rumah pompa | *Float* |
| 8 | *tv\_threshold tinggiair* | *Text view* | Berisi informasi ambang batas ketinggian air pada rumah pompa | *Integer* |
| 9 | *tv\_ketinggian saluranair* | *Text view* | Berisi informasi ketinggian saluran pada rumah | *Float* |
| 10 | *Btn\_editrumahpompa* | *Button* | Tombol untuk menuju ke halaman mengubah data rumah pompa | *Button click* |
| 11 | *Btn\_delete rumahpompa* | *Button* | Tombol untuk menuju ke halaman menghapus data rumah pompa | *Button click* |

1. Rancangan Halaman Antarmuka Mendapat Peringatan Potensi Banjir

Halaman ini berupa *push notification* yang akan ditampilkan ketika petugas mendapat peringatan potensi banjir. Rancangan halaman dapat dilihat pada Gambar 3.25. Atribut antarmuka dapat dilihat pada Tabel 3.29.



Gambar . Rancangan Halaman Mendapat Peringatan Potensi Banjir

Tabel . Atribut Antarmuka Mendapat Peringatan Banjir

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *notification* | *Push notification* | Berisi peringatan potensi banjir | *notification* |

1. Rancangan Halaman Antarmuka Mencari Data User

Halaman ini akan ditampilkan ketika admin akan mencari data *user* yang perlukan. Admin dapat memilih untuk menampilkan user berdasarkan *role* dengan cara memilih *role* pada *dropdown list* *spinner\_user.* Selain itu, admin dapat melakukan pencarian *user* berdasarkan nama *user* dengan cara memasukkan nama *user* yang dicari pada *auto complete textview* *ac\_searchuser*. Rancangan halaman dapat dilihat pada Gambar 3.26. Atribut antarmuka dapat dilihat pada Tabel 3.32.



Gambar 3.43 Rancangan Halaman Antarmuka Mencari Data User

Tabel . Atribut Antarmuka Mencari Data User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *spinner\_user* | *Drop down list* | Berisi pilihan *role* untuk menampilkan *user* berdasarkan *role* yang dipilih | *Button click* |
| 2 | *ac\_searchuser* | *Auto complete text view* | Input untuk memasukkan nama *user* yang dicari | *Text* |
| 3 | *lv\_user* | *List view* | Berisi list data *user* | *List* |
| 4 | *fab\_adduser* | *Floating action button* | Tombol untuk menuju halaman menambah data *user* | *Button click* |

1. Rancangan Halaman Antarmuka Mencari Data Rumah Pompa

Halaman ini akan ditampilkan ketika admin atau pengawas akan mencari data rumah pompa yang perlukan. Admin dan pengawas dapat memilih untuk menampilkan rumah pompa berdasarkan status potensi banjir dengan cara memilih pada *dropdown list* *spinner\_rumahpompa.* Selain itu, admin dan pengawas dapat melakukan pencarian rumah pompa berdasarkan nama rumah pompa dengan cara memasukkan nama rumah pompa yang dicari pada *auto complete textview* *ac\_rumahpompa*. Rancangan halaman dapat dilihat pada Gambar 3.26. Atribut antarmuka dapat dilihat pada Tabel 3.32.



Gambar . Rancangan Halaman Antarmuka Mencari Data Rumah Pompa

Tabel . Atribut Antarmuka Mencari Data Rumah Pompa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| 1 | *spinner\_ru mahpompa* | *Drop down list* | Berisi pilihan untuk menampilkan list rumah pompa berdasarkan status potensi banjir | *Button click* |
| 2 | *ac\_rumahpompa* | *Auto complete text view* | Input untuk memasukkan nama rumah pompa yang dicari | *String* |
| 3 | *lv\_rumah pompa* | *List view* | Berisi list untuk menampilkan data rumah pompa | *List* |
| 4 | *btn\_addrumahpompa* | *Button* | Tombol untuk menuju halaman menambah data rumah pompa | *Button click* |

*[Halaman ini sengaja dikosongkan]*

# BAB IV IMPLEMENTASI

Bab ini membahas implementasi yang dilakukan berdasarkan rancangan yang telah dijabarkan pada bab sebelumnya. Sebelum penjelasan implementasi akan ditunjukkan terlebih dahulu lingkungan untuk melakukan implementasi.

Pada bagian implementasi ini juga akan dijelaskan mengenai fungsi-fungsi yang digunakan dalam program tugas akhir ini dan disertai dengan kode sumber masing-masing fungsi utama.

* 1. Lingkungan Implementasi Perangkat Lunak
  2. Implementasi Antarmuka Pengguna

# BAB V PENGUJIAN DAN EVALUASI

Bab ini membahas tentang pengujian dan evaluasi pada perangkat lunak yang dibangun untuk tugas akhir ini. Pengujian dilakukan pada kasus penggunaan dari sistem perangkat lunak.

* 1. Lingkungan Pengujian
  2. Pengujian Fungsionalitas

# BAB VI KESIMPULAN DAN SARAN

Bab ini membahas mengenai kesimpulan yang dapat diambil dari hasil pengujian yang telah dilakukan sebagai jawaban dari rumusan masalah yang dikemukakan. Selain kesimpulan, juga terdapat saran yang ditujukan untuk pengembangan perangkat lunak di masa mendatang.

1. Kesimpulan
2. Saran

*[Halaman ini sengaja dikosongkan]*

# DAFTAR PUSTAKA

|  |  |
| --- | --- |
| [1] | Ary Bima Winardo, Arna Fariza, Yuliana Setiowati, "Investigasi Daerah Rawan Banjir di Kota Surabaya dengan Menggunakan Metode Fuzzy," *EEPIS Final Project,* p. 1, 2010. |
| [2] | Badan Nasional Penanggulangan Bencana, "Definisi dan Jenis Bencana," [Online]. Available: http://www.bnpb.go.id/pengetahuan-bencana/definisi-dan-jenis-bencana. [Accessed 1 Desember 2016]. |
| [3] | IDEP, Banjir! Cerita Tentang Peran Masyarakat Saat Terjadi Banjir, Yayasan IDEP, 2007. |
| [4] | A. Bahtiar A and P. H. Wibowo, "SIMULASI PEMILIHAN DESAIN DAN PERENCANAAN TEKNIS RETARDING POND DI KELURAHAN MINTARAGEN KOTA TEGAL (The Simulation of Design selection and Technic Specification of Retarding pond in Mintaragen Village Tegal City)," *Undergraduate thesis, F. TEKNIK UNDIP.,* 2010. |
| [5] | C. Erwin and P. Wiyan Aji, "PENGENDALIAN BANJIR SUNGAI CILIWUNG STA 165 – STA 705 DKI JAKARTA (Flood Management of Ciliwung River Sta 165 – Sta 705 DKI Jakarta)," *Undergraduate thesis, F. TEKNIK UNDIP.,* 2010. |
| [6] | Anonim, "Sistem Peringatan Dini (Early Warning System)," [Online]. Available: https://adekabang.wordpress.com/2010/11/12/sistem-peringatan-dini-early-warning-system/. [Accessed 4 December 2016]. |
| [7] | M. Zaenal Muttaqin, "Sistem Informasi Peringatan Dini Tsunami Berbasis Web (Simulasi Tsunami Bengkulu)," *FT Universitas Indonesia,* 2008. |
| [8] | Lutfiyah Rahmawati, Arna Fariza, Ira Prasetyaningrum, "Penentuan Lokasi Rumah Pompa Kota Surabaya Menggunakan Metode Analytical Hierarchy Process(AHP)," *EEPIS Final Project,* p. 1, 2010. |
| [9] | Anonim, "Laporan Kuliah Lapangan Rumah Pompa Kenjeran," [Online]. Available: http://documents.tips/documents/laporan-kuliah-lapangan-rumah-pompa-kenjeran-lama.html. [Accessed 19 November 2016]. |
| [10] | Wikipedia, "Android (sistem operasi)," [Online]. Available: https://id.wikipedia.org/wiki/Android\_(sistem\_operasi). [Accessed 20 November 2016]. |
| [11] | TechInAsia, "Android dan browser Opera dominasi pengguna mobile Indonesia selama 2014," [Online]. Available: https://id.techinasia.com/android-opera-dominasi-smartphone-indonesia-2014. [Accessed 20 November 2016]. |
| [12] | Feridi, "Mengenal RESTful Web Services," [Online]. Available: https://www.codepolitan.com/mengenal-restful-web-services. [Accessed 24 November 2016]. |
| [13] | M. Masse, REST API Design Rulebook, O'Reilly Media, 2011, p. 114. |
| [14] | S. Munawaroh, "Mengeksplorasi Database PostgreSQL dengan PgAdmin III," *Dinamik - Jurnal Teknologi Informasi,* vol. 10, no. 2, 2005. |
| [15] | A. Developers, "Transmitting Network Data Using Volley," [Online]. Available: https://developer.android.com/training/volley/index.html. |
| [16] | A. Developers, "Sending a Simple Request," [Online]. Available: https://developer.android.com/training/volley/simple.html. [Accessed 29 September 2016]. |
| [17] | A. Developers, "Making a Standard Request," [Online]. Available: https://developer.android.com/training/volley/request.html. [Accessed 29 September 2016]. |
| [18] | M. Faten Nadia, "Tank Water Level Sensor," 2010. |
| [19] | Weather Underground, "API | Weather Underground," [Online]. Available: https://www.wunderground.com/weather/api/d/docs?d=index. [Accessed 17 April 2017]. |
| [20] | OpenWeatherMap, "Current Weather Data," [Online]. Available: https://openweathermap.org/current. [Accessed 4 Desember 2016]. |

# 

# LAMPIRAN

BIODATA PENULIS