



ITS
Institut
Teknologi
Sepuluh Nopember

KERJA PRAKTIK - KI141330

Pembuatan Sistem *Monitoring* Ketinggian Air Kota Surabaya Berbasis *Web*

**Dinas Komunikasi dan Informatika (Dinkominfo)
Kota Surabaya
Jl. Jimerto No. 25-27 Surabaya**

Periode: 6 Januari - 7 Februari 2020

Oleh:

| | |
|---------------------------|---------------|
| Mohammad Rizaldi Huzein P | 0511174000024 |
| Yasinta Yusniawati | 0511174000054 |

Pembimbing Jurusan
Hadziq Fabroyir, S.Kom., Ph.D.
Pembimbing Lapangan
Oky Yanuar Kusuma Atmaja, S.Kom., M.MT

**DEPARTEMEN INFORMATIKA
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020**

[Halaman ini sengaja dikosongkan]



KERJA PRAKTIK - KI141330

Pembuatan Sistem *Monitoring* Ketinggian Air Kota Surabaya Berbasis *Web*

Dinas Komunikasi dan Informatika (Dinkominfo) Kota Surabaya

Jl. Jimerto No. 25-27 Surabaya

Oleh:

Mohammad Rizaldi Huzein P 0511174000024

Yasinta Yusniawati 0511174000054

Pembimbing Jurusan

Hadziq Fabroyir, S.Kom., Ph.D.

Pembimbing Lapangan

Oky Yanuar Kusuma Atmaja, S.Kom., M.MT

DEPARTEMEN INFORMATIKA

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya 2020

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN

KERJA PRAKTIK

Pembuatan Sistem Monitoring Ketinggian Air Kota Surabaya Berbasis Web

Oleh :

Mohammad Rizaldi Huzein P
Yasinta Yusniawati

0511174000024
0511174000054

Mengetahui,

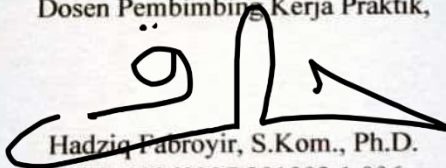
Pembimbing Kerja Praktik,



Okky Yudianto Kusuma Atmaja, S.Kom., M.MT.
NIP. 198701192010011011

Menyetujui

Dosen Pembimbing Kerja Praktik,



Hadziq Fabroyir, S.Kom., Ph.D.
NIP. 19860227 201903 1 006

[Halaman ini sengaja dikosongkan]

***Pembuatan Sistem Monitoring
Ketinggian Air Kota Surabaya Berbasis Web***

Nama Mahasiswa : Mohammad Rizaldi Huzein P
NRP : 0511174000024
Nama Mahasiswa : Yasinta Yusniawati
NRP : 0511174000054
Departemen : Informatika FTEIC-ITS
Pembimbing Jurusan : Hadziq Fabroyir, S.Kom., Ph.D

ABSTRAK

Dinas Komunikasi dan Informatika (Dinkominfo) Kota Surabaya adalah dinas yang mempunyai tugas melaksanakan kewenangan daerah di bidang pengelolaan Teknologi Informasi dan Komunikasi serta melaksanakan tugas pembantuan yang diberikan oleh Pemerintah Kota Surabaya. Sebagai lembaga pemerintahan Dinkominfo mempunyai tugas dan fungsi besar dalam pembangunan Teknologi Informasi dan Komunikasi (TIK) di Kota Surabaya. Salah satu tugas dari Dinkominfo yaitu membuat *website monitoring* ketinggian air untuk meminimalisir dampak banjir. *Website* ini digunakan untuk memudahkan proses pengawasan terutama untuk mengetahui ketinggian air, melihat batas ketinggian air, daya tampung BBM, BBM yang digunakan, dan mengirimkan notifikasi kepada petugas rumah pompa. Notifikasi ini untuk memberitahukan pembukaan pipa pembuangan air pada rumah pompa. *Website monitoring* ketinggian air dibuat dengan melihat kebutuhan akan pentingnya pengawasan ketinggian air dalam meminimalisir dampak banjir di Kota Surabaya.

Kata kunci: Dinkominfo Surabaya, Website Monitoring ketinggian air, Rumah Pompa

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena atas berkat limpahan rahmat dan lindungan-Nya penulis dapat melaksanakan salah satu kewajiban sebagai mahasiswa Teknik Informatika ITS yaitu Kerja Praktik (KP).

Penulis menyadari masih terdapat banyak kekurangan baik dalam pelaksanaan kerja praktik maupun penyusunan buku laporan ini. Penulis mengharapkan kritik dan saran yang membangun untuk kesempurnaan penulisan buku laporan ini

Melalui laporan ini penulis juga ingin menyampaikan rasa terima kasih kepada orang-orang yang telah membantu dalam pelaksanaan kerja praktik hingga penyusunan laporan kerja praktik baik secara langsung maupun tidak langsung. Orang-orang tersebut antara lain adalah:

1. Kedua orang tua penulis.
2. Bapak Hadziq Fabroyir, S.Kom., Ph.D., selaku dosen pembimbing Kerja Praktik yang telah membimbing penulis selama Kerja Praktik berlangsung.
3. Bapak Ary Mazharuddin Shiddiqi, S.Kom., M.Comp.Sc., Ph.D selaku koordinator Kerja Praktik.
4. Bapak Oky Yanuar Kusuma Atmaja, S.Kom., M.MT. selaku pembimbing lapangan kami di Dinas Komunikasi dan Informatika (Kominfo) Kota Surabaya.
5. Seluruh staf di Dinas Kominfo serta staf yang membantu kelancaran Kerja Praktik.
6. Teman-teman penulis yang senantiasa memberikan semangat dan dukungan untuk penulis.

Surabaya, November 2020

[Halaman ini sengaja dikosongkan]

DAFTAR ISI

| | |
|---|-------------|
| LEMBAR PENGESAHAN..... | V |
| ABSTRAK..... | VII |
| KATA PENGANTAR | IX |
| DAFTAR ISI..... | XI |
| DAFTAR GAMBAR | XV |
| DAFTAR TABEL..... | XVII |
| BAB I PENDAHULUAN..... | 1 |
| 1.1. LATAR BELAKANG..... | 1 |
| 1.2. TUJUAN | 2 |
| 1.3. MANFAAT | 2 |
| 1.4. RUMUSAN PERMASALAHAN | 3 |
| 1.5. LOKASI DAN WAKTU KERJA PRAKTIK | 3 |
| 1.6. METODOLOGI KERJA PRAKTIK | 3 |
| 1.7. SISTEMATIKA LAPORAN | 5 |
| BAB II PROFIL INSTANSI..... | 7 |
| 2.1. PROFIL DINAS KOMINFO KOTA SURABAYA | 7 |
| 2.2. VISI DAN MISI PERUSAHAAN | 7 |
| 2.3. BIDANG TEKNOLOGI DAN INFORMATIKA | 9 |
| BAB III TINJAUAN PUSTAKA..... | 11 |
| 3.1. APLIKASI WEB | 11 |
| 3.2. HTML..... | 11 |
| 3.3. CSS | 11 |
| 3.4. BOOTSTRAP..... | 12 |
| 3.5. JAVASCRIPT | 12 |
| 3.6. JQUERY | 12 |
| 3.7. PHP | 13 |
| 3.8. LARAVEL..... | 13 |

| | |
|--|-----------|
| 3.9. WEB SERVER..... | 13 |
| 3.10. MYSQL..... | 14 |
| 3.11. VISUAL STUDIO CODE | 14 |
| 3.12. GIT | 14 |
| BAB IV ANALISIS DAN DESAIN..... | 15 |
| 4.1 ANALISIS SISTEM..... | 15 |
| 4.4.1 DEFINISI UMUM FITUR | 15 |
| 4.4.2 ANALISIS KEBUTUHAN FUNGSIONAL..... | 15 |
| 4.2 DIAGRAM KASUS PENGGUNAAN | 16 |
| 4.3 SPESIFIKASI KASUS PENGGUNAAN..... | 17 |
| 4.3.1. Melakukan login..... | 17 |
| 4.3.2. Melakukan register..... | 19 |
| 4.3.3. Melihat data user | 21 |
| 4.3.4. Memonitoring ketinggian air di semua rumah pompa..... | 22 |
| 4.3.5. Menambahkan data lokasi rumah pompa..... | 24 |
| 4.3.6. Melihat detail data rumah pompa | 26 |
| 4.3.7. Mengirimkan notifikasi | 28 |
| 4.3.8. Mengedit profil user | 30 |
| 4.3.9. Menghapus akun user | 32 |
| 4.3.10. Memonitoring ketinggian air di lokasi yang dijaga | 34 |
| 4.3.11. Menambah pompa pada rumah pompa | 36 |
| 4.4 DIAGRAM AKTIVITAS..... | 38 |
| 4.4.1 Melakukan login | 38 |
| 4.4.2 Melakukan register..... | 39 |
| 4.4.3 Melihat data user | 40 |
| 4.4.4 Memonitor ketinggian air di semua rumah pompa | 41 |
| 4.4.5 Menambahkan data lokasi rumah pompa | 42 |
| 4.4.6 Melihat detail data rumah pompa..... | 43 |
| 4.4.7 Mengirimkan notifikasi | 44 |
| 4.4.8 Mengedit profil user | 45 |
| 4.4.9 Menghapus akun user | 46 |
| 4.4.10 Memonitor ketinggian air di lokasi yang dijaga | 47 |

| | | |
|--------------|--|-----------|
| 4.4.11 | Menambah pompa pada rumah pompa | 48 |
| 4.5 | DESAIN BASIS DATA | 49 |
| 4.5.1 | Desain Konseptual Data | 49 |
| 4.5.2 | Desain Fisik Data | 50 |
| 4.5.3 | Struktur Tabel User | 51 |
| 4.5.4 | Struktur Tabel Lokasi | 52 |
| 4.5.5 | Struktur Tabel Pompa | 53 |
| 4.5.6 | Struktur Tabel Ketinggian | 54 |
| BAB V | IMPLEMENTASI SISTEM | 55 |
| 5.1 | IMPLEMENTASI LAPISAN MODEL..... | 55 |
| 5.1.1 | Model Ketinggian | 55 |
| 5.1.2 | Model Lokasi | 56 |
| 5.1.3 | Model Pompa | 56 |
| 5.1.4 | Model User | 57 |
| 5.2 | IMPLEMENTASI LAPISAN KONTROL..... | 59 |
| 5.1.5 | HomeController | 59 |
| 5.1.6 | KetinggianController | 60 |
| 5.1.7 | LokasiController | 62 |
| 5.1.8 | MonitoringController | 64 |
| 5.1.9 | NotifikasiController | 66 |
| 5.1.10 | PompaController | 67 |
| 5.1.11 | SendWarning Command | 68 |
| 5.1.12 | UserController | 71 |
| 5.3 | IMPLEMENTASI ANTARMUKA PENGGUNA | 73 |
| 5.2.1 | Halaman login | 73 |
| 5.2.2 | Halaman Dashboard Administrator | 74 |
| 5.2.3 | Halaman Tambah Lokasi Rumah Pompa | 75 |
| 5.2.4 | Halaman Data Rumah Pompa | 76 |
| 5.2.5 | Halaman Daftar User | 77 |
| 5.2.6 | Halaman Menambahkan User | 78 |
| 5.2.7 | Halaman Edit Profil | 79 |
| 5.2.8 | Halaman Dashboard Petugas | 80 |

| | |
|--|------------|
| BAB VI PENGUJIAN DAN EVALUASI | 81 |
| 6.1. SKENARIO PENGUJIAN | 81 |
| 6.1.1. Melakukan login..... | 81 |
| 6.1.2. Melakukan register..... | 81 |
| 6.1.3. Melihat daftar user yang terdaftar | 82 |
| 6.1.4. Memonitor ketinggian air di semua rumah pompa .. | 82 |
| 6.1.5. Menambahkan data lokasi rumah pompa..... | 83 |
| 6.1.6. Melihat detail data rumah pompa | 83 |
| 6.1.7. Mengirimkan notifikasi | 84 |
| 6.1.8. Mengedit profil user | 84 |
| 6.1.9. Menghapus akun user | 85 |
| 6.1.10. Memonitor ketinggian air di lokasi yang dijaga | 85 |
| 6.1.11. Menambah pompa pada rumah pompa | 86 |
| 6.2. EVALUASI PENGUJIAN | 87 |
| 6.2.1. Melakukan login..... | 89 |
| 6.2.2. Melakukan register..... | 91 |
| 6.2.3. Melihat data user | 93 |
| 6.2.4. Memonitor ketinggian air di semua rumah pompa .. | 94 |
| 6.2.5. Menambahkan data lokasi rumah pompa..... | 95 |
| 6.2.6. Melihat detail data rumah pompa | 97 |
| 6.2.7. Mengirimkan notifikasi | 98 |
| 6.2.8. Mengedit profil user | 99 |
| 6.2.9. Menghapus akun user | 101 |
| 6.2.10. Memonitor ketinggian air di lokasi yang dijaga ... | 102 |
| 6.2.11. Menambah pompa pada rumah pompa | 104 |
| BAB VII KESIMPULAN DAN SARAN | 107 |
| DAFTAR PUSTAKA | 108 |

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 4.1 Diagram Use Case Sistem Monitoring Ketinggian Air | 17 |
| Gambar 4.2 Diagram Aktivitas Melakukan Login | 38 |
| Gambar 4.3 Diagram Aktivitas Melakukan Register | 39 |
| Gambar 4.4 Diagram Aktivitas Melihat Data User | 40 |
| Gambar 4.5 Diagram Aktivitas Memonitor Ketinggian Air di Semua Rumah Pompa..... | 41 |
| Gambar 4.6 Diagram Aktivitas Menambahkan Data Lokasi Rumah Pompa | 42 |
| Gambar 4.7 Diagram Aktivitas Melihat Detail Data Rumah Pompa | 43 |
| Gambar 4.8 Diagram Aktivitas Mengirimkan Notifikasi | 44 |
| Gambar 4.9 Diagram Aktivitas Mengedit Profil User | 45 |
| Gambar 4.10 Diagram Aktivitas Menghapus Akun User..... | 46 |
| Gambar 4.11 Diagram Aktivitas Memonitor Ketinggian Air di Lokasi yang Dijaga | 47 |
| Gambar 4.12 Diagram Aktivitas Menambah Pompa pada Rumah Pompa | 48 |
| Gambar 4.13 Conceptual Data Model | 49 |
| Gambar 4.14 Physical Data Model..... | 50 |
| | |
| Gambar 5.1 Tampilan Halaman Login | 73 |
| Gambar 5.3 Tampilan Halaman Tambah Lokasi Rumah Pompa | 75 |
| Gambar 5.4 Tampilan Data Rumah Pompa..... | 76 |
| Gambar 5.5 Tampilan Halaman Daftar User | 77 |
| Gambar 5.6 Tampilan Halaman Menambahkan User | 78 |
| Gambar 5.7 Tampilan Halaman Edit Profil | 79 |
| Gambar 5.8 Tampilan Halaman Dashboard Petugas | 80 |
| | |
| Gambar 6.1 Tampilan Login | 89 |
| Gambar 6.2 Tampilan Berhasil Login Sebagai User | 90 |
| Gambar 6.3 Tampilan Berhasil Login Sebagai Admin | 90 |
| Gambar 6.4 Tampilan Form Register | 91 |
| Gambar 6.5 Tampilan Daftar User Setelah Register Berhasil | 92 |
| Gambar 6.6 Tampilan Daftar User | 93 |

| | |
|---|-----|
| Gambar 6.7 Tampilan Monitoring Ketinggian Air Semua Rumah Pompa | 94 |
| Gambar 6.8 Tampilan Form Tambah Rumah Pompa..... | 95 |
| Gambar 6.9 Tampilan Daftar Rumah Pompa Baru | 96 |
| Gambar 6.10 Tampilan Menu Monitoring Baru..... | 96 |
| Gambar 6.11 Tampilan Daftar Rumah Pompa | 97 |
| Gambar 6.12 Tampilan Detail Rumah Pompa..... | 97 |
| Gambar 6.13 Tampilan Notifikasi Bot Telegram MONIKA..... | 98 |
| Gambar 6.14 Tampilan Daftar User | 99 |
| Gambar 6.15 Tampilan Form Edit User | 100 |
| Gambar 6.16 Tampilan Daftar User Setelah Berhasil | 100 |
| Gambar 6.17 Tampilan Daftar User dan Tombol Hapus..... | 101 |
| Gambar 6.18 Tampilan Hasil Hapus Akun User..... | 101 |
| Gambar 6.19 Tampilan Grafik Seluruh Rumah Pompa pada Halaman Monitoring..... | 102 |
| Gambar 6.20 Tampilan Detail Monitor Rumah Pompa..... | 103 |
| Gambar 6.21 Tampilan Daftar Rumah Pompa | 104 |
| Gambar 6.22 Tampilan Daftar Pompa Pada Rumah Pompa | 105 |
| Gambar 6.23 Tampilan Form Tambah Pompa | 105 |
| Gambar 6.24 Tampilan Daftar Pompa Pada Rumah Pompa | 105 |

DAFTAR TABEL

| | |
|--|----|
| Tabel 4.1 Kebutuhan Fungsional..... | 16 |
| Tabel 4.2 Tabel usecase melakukan login | 18 |
| Tabel 4.3 Tabel usecase melakukan register | 19 |
| Tabel 4.4 Tabel usecase melihat data user | 21 |
| Tabel 4.5 Tabel usecase memonitor ketinggian air di semua rumah pompa..... | 22 |
| Tabel 4.6 Tabel usecase menambahkan data lokasi rumah pompa .. | 24 |
| Tabel 4.7 Tabel usecase melihat detail data rumah pompa | 26 |
| Tabel 4.8 Tabel usecase mengirimkan notifikasi | 28 |
| Tabel 4.9 Tabel usecase mengedit profil user | 30 |
| Tabel 4.10 Tabel usecase menghapus akun user | 32 |
| Tabel 4.11 Tabel usecase memonitoring ketinggian air di lokasi yang di jaga | 34 |
| Tabel 4.12 Tabel usecase menambah pompa pada rumah pompa... | 36 |
| Tabel 4.13 Struktur Tabel User | 51 |
| Tabel 4.14 Struktur Tabel Lokasi | 52 |
| Tabel 4.15 Struktur Tabel Pompa..... | 53 |
| Tabel 4.16 Struktur Tabel Ketinggian | 54 |
| Tabel 6.1 Tabel Evaluasi Pengujian Aplikasi Sesuai Kebutuhan..... | 87 |

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi informasi semakin membantu manusia dalam menjalankan aktifitas sehari-hari. Saat ini, hampir seluruh kegiatan manusia tidak terlepas dari peran teknologi informasi. Seiring berkembangnya teknologi informasi, kebutuhan akan pembuatan aplikasi terus meningkat, salah satunya di Dinas Komunikasi dan Informatika (Kominfo) Kota Surabaya.

Dinas Komunikasi dan Informatika (Kominfo) Kota Surabaya adalah dinas yang mempunyai tugas melaksanakan kewenangan daerah di bidang pengelolaan Teknologi Informasi dan Komunikasi serta melaksanakan tugas pembantuan yang diberikan oleh Pemerintah Kota Surabaya. Kominfo Kota Surabaya terbagi menjadi 4 bidang, yaitu Sekretariat, Bidang Informasi dan Komunikasi Publik, Bidang Teknologi dan Informatika (TI), dan Bidang Layanan Pemerintah Berbasis Elektronik. [1]

Salah satu tugas Bidang Teknologi dan Informatika (TI) adalah membuat *website monitoring* ketinggian air. Untuk saat ini, *website monitoring* ketinggian air memiliki tampilan yang sederhana dan masih bersifat internal, yaitu hanya dapat dilihat oleh petugas yang berada di Kominfo saja. Sedangkan untuk membuka pipa pembuangan air pada rumah pompa, petugas yang berada di Kominfo harus menelfon petugas lapangan terlebih dahulu.

Untuk mengatasi masalah diatas, penulis mengembangkan sebuah Sistem *Monitoring Ketinggian Air* Berbasis Web untuk melakukan *monitoring* ketinggian air dan mempermudah komunikasi antara petugas Kominfo dan petugas penjaga rumah pompa. *Website* ini dikembangkan dengan tampilan yang lebih menarik dari sebelumnya dan di intergasikan dengan API Telegram untuk memudahkan komunikasi antar petugas.

1.2.Tujuan

Tujuan kerja praktik ini adalah untuk menyelesaikan kewajiban kuliah Kerja Praktik di Institut Teknologi Sepuluh Nopember dengan beban dua SKS. Selain itu, untuk memenuhi kebutuhan yang diperlukan oleh Kominfo Kota Surabaya dengan membuat Sistem *Monitoring Ketinggian Air*.

Tujuan dari pembuatan sistem *monitoring* ketinggian dalam pelaksanaan Kerja Praktik ini yaitu:

1. Membuat fitur yang memungkinkan pengguna melihat informasi ketinggian air di rumah pompa.
2. Membuat desain *user interface* yang mudah dipahami oleh *user*.
3. Optimasi sistem dari cara manual menjadi digital.

1.3.Manfaat

Manfaat dari pembuatan Sistem *Monitoring Ketinggian Air* dalam pelaksanaan Kerja Praktik ini adalah

1. Memudahkan petugas dalam mengawasi atau memantau ketinggian air.

2. Memudahkan proses pemantauan penggunaan BBM pada setiap rumah pompa beserta daya tampungnya.
3. Mempermudah komunikasi antara petugas Kominfo dan petugas penjaga rumah pompa.

1.4. Rumusan Permasalahan

Berikut rumusan masalah pada Kerja Praktik pembuatan Sistem *Monitoring* Ketinggian Air:

1. Bagaimana perancangan dan implementasi Sistem *Monitoring* Ketinggian Air?
2. Bagaimana pengelolaan manajemen *user* (*register* dan *login*)?
3. Bagaimana penambahan data lokasi rumah pompa?
4. Bagaimana proses pengiriman notifikasi kepada para petugas penjaga rumah pompa?

1.5. Lokasi dan Waktu Kerja Praktik

Kerja praktik kali ini dilaksanakan pada waktu dan tempat sebagai berikut:

Lokasi : Dinas Komunikasi dan Informatika
Kota Surabaya

Alamat: Jl. Jimerto No. 25-27 Ketabang Kecamatan
Genteng Kota Surabaya

Waktu : 6 Januari 2020 – 7 Februari 2020

1.6. Metodologi Kerja Praktik

1. Perumusan Masalah

Penjelasan mengenai masalah diberikan oleh pembimbing lapangan Kerja Praktik, yaitu sistem *monitoring* ketinggian air. Penjelasan tersebut

menghasilkan catatan-catatan penting mengenai gambaran sistem berbasis *web* yang akan dibuat. Berdasarkan catatan-catatan tersebut, kami memutuskan untuk membuat *web* dengan *framework* Laravel. Pada kenyataannya, sebelum sistem ini dibuat, sudah ada sistem *monitoring* ketinggian air yang digunakan Pemerintah Kota Surabaya. Oleh karena itu, kami menyesuaikan pengembangan berdasarkan sistem yang sudah ada.

2. Studi Literatur

Setelah menentukan *framework* yang digunakan, kami melakukan studi literatur mengenai cara implementasi sistem sesuai dengan kebutuhan. Proses analisa kebutuhan ini dengan melakukan wawancara kepada petugas lapangan dan petugas rumah pompa serta melakukan survei ke lokasi rumah pompa.

3. Analisis dan Perancangan Sistem

Tahap ini mendefinisikan alur dari implementasi. Langkah- langkah yang dikerjakan juga didefinisikan ditahap ini, seperti:

- a. Menganalisis kebutuhan sistem *monitoring* ketinggian air
- b. Menganalisis basis data sistem *monitoring* ketinggian air dari data hasil wawancara dan survei lapangan
- c. Memodelkan basis data sistem *monitoring* ketinggian air
- d. Menganalisis keterkaitan sistem *monitoring* ketinggian air yang akan dibuat dengan sistem yang sudah digunakan di

Dinas Kominfo Kota Surabaya.

4. Implementasi Sistem

Tahap implementasi sistem didasarkan pada perancangan dan analisis sebelumnya. Dalam mengimplementasikan sistem *monitoring* ketinggian air, digunakan *framework* Laravel. Pada tahap ini ada 2 pekerjaan utama, yaitu membuat *front-end* dan *back-end*. Pengerjaan *web* dilakukan di komputer local, lalu di *deploy* ke server yang disediakan oleh Dinas Kominfo Kota Surabaya.

5. Pengujian dan Evaluasi

Pengujian dilakukan dengan menguji fitur-fitur yang telah dibuat. Kesesuaian sistem dengan kebutuhan akan menentukan keberhasilan dalam pengujian. Hal ini akan menentukan apakah hasil evaluasi terhadap sistem sudah sesuai dengan tujuan dan kebutuhan sistem yang sudah di analisis.

1.7. Sistematika Laporan

Laporan kerja praktik ini terdiri dari 7 bab dengan rincian sebagai berikut:

1. Bab I: Pendahuluan

Bab ini berisi tentang latar belakang masalah, tujuan, manfaat, rumusan masalah, lokasi dan waktu kerja praktik, metodologi, dan sistematika laporan Kerja Praktik.

2. Bab II: Profil Instansi

Bab ini menjelaskan tentang profil singkat Dinas Komunikasi dan Informatika (Kominfo) Kota Surabaya.

3. Bab III: Tinjauan Pustaka

Bab ini menjelaskan mengenai tinjauan pustaka dan literatur yang digunakan dalam penyelesaian Kerja Praktik.

4. Bab IV: Analisis dan Perancangan Sistem

Bab ini membahas tentang proses analisa kebutuhan berdasarkan kondisi yang sesungguhnya dan perancangannya yang meliputi desain aplikasi yang akan dikembangkan.

5. Bab V: Desain Model dan Implementasi Sistem

Bab ini menjelaskan tahap-tahap yang dilakukan untuk proses implementasi sistem.

6. Bab VI: Pengujian dan Evaluasi

Dalam bab ini dibahas tentang skenario pengujian, dan evaluasi pengujian setelah model selesai dibangun.

7. Bab VII: Kesimpulan dan Saran

Bab ini memaparkan kesimpulan yang dapat diambil dan juga saran selama pengerjaan Kerja Praktik

BAB II

PROFIL INSTANSI

2.1. Profil Dinas Kominfo Kota Surabaya

Dinas Komunikasi dan Informatika adalah Dinas yang mempunyai tugas melaksanakan kewenangan daerah di bidang pengelolaan Teknologi Informasi dan Komunikasi serta melaksanakan tugas pembantuan yang diberikan oleh Pemerintah dan atau Pemerintah Provinsi dimana dalam setiap kegiatannya selalu berhubungan dengan Pembangunan dan Pengembangan Sistem Informasi, Pengembangan dan Pemeliharaan Jaringan Komputer Antar Bidang, Pengelolaan Produksi Informasi dan Publikasi, Pengelolaan dan Pengembangan Komunikasi Publik.

Kegiatan-kegiatan tersebut terbagi menjadi empat bidang yang dibawah oleh kepala bidang dari setiap bidangnya dan satu sekretaris. Sebagai Lembaga pemerintahan yang mempunyai tanggung jawab besar dan bergerak di dalam lingkungan Pemerintah Kota Surabaya Dinas Kominfo mempunyai tugas pokok dan fungsi yang besar dalam membangun Teknologi Informasi dan Komunikasi (TIK) di Kota Surabaya. [1]

2.2. Visi dan Misi Perusahaan

Visi dan Misi Kementerian Komunikasi dan Informatika mengacu pada Visi dan Misi Presiden Joko Widodo dan Wakil Presiden KH. Ma'ruf Amin. Sebagai kelanjutan, percepatan, pengembangan, serta pemajuan dari visi di tahun 2014-2019, selama 5 tahun ke depan.

Visi:

Terwujudnya Indonesia Maju yang Berdaulat, Mandiri, dan Berkepribadian Berlandaskan Gotong Royong

Misi:

- 1 Peningkatan kualitas manusia Indonesia.
- 2 Struktur ekonomi yang produktif, mandiri, dan berdaya saing.
- 3 Pembangunan yang merata dan berkeadilan.
- 4 Mencapai lingkungan hidup yang berkelanjutan.
- 5 Kemajuan budaya yang mencerminkan kepribadian bangsa.
- 6 Penegakan sistem hukum yang bebas korupsi, bermartabat, dan terpercaya.
- 7 Perlindungan bagi segenap bangsa dan memberikan rasa aman pada seluruh warga.
- 8 Pengelolaan pemerintahan yang bersih, efektif, dan terpercaya.
- 9 Sinergi pemerintah daerah dalam kerangka Negara Kesatuan

Sembilan misi ini merupakan percepatan, pengembangan, dan pemajuan Nawa Cita I dengan tetap konsisten menerapkan Trisakti sebagai pijakan strategis operasional dengan senantiasa mengutamakan pembangunan manusia (berpusat pada manusia) [2].

2.3. Bidang Teknologi dan Informatika

Merupakan salah satu bidang yang ada di Dinas Kominfo Kota Surabaya yang bertanggung jawab dalam pembangunan dan pengembangan sistem informasi serta pengembangan dan pemeliharaan jaringan komputer antar bidang. Bidang ini terdiri dari tiga bagian yaitu Infrastruktur TIK, Keamanan Informasi dan Persandian, dan Manajemen Data dan Integrasi Sistem.

[Halaman ini sengaja dikosongkan]

BAB III

TINJAUAN PUSTAKA

3.1. Aplikasi Web

Aplikasi web merupakan aplikasi yang diakses menggunakan *web browser* melalui jaringan internet atau intranet. Aplikasi *web* juga merupakan suatu perangkat lunak komputer yang dikodekan dalam bahasa pemrograman yang mendukung perangkat lunak berbasis web. [3]

3.2. HTML

HTML (*Hypertext Markup Language*) adalah bahasa *markup* yang mendefinisikan struktur konten sebuah web. HTML terdiri atas serangkaian elemen, yang dapat digunakan untuk menyisipkan atau membungkus konten di dalamnya sehingga mempunyai tampilan tertentu sesuai fungsi elemennya. Contohnya, konten dapat disusun dalam bentuk sebuah paragraf, penomoran berbentuk titik, atau menampilkan gambar dan tabel. [4]

3.3. CSS

CSS (*Cascading Style Sheet*) berfungsi untuk mengatur tampilan elemen yang tertulis dalam bahasa markup. Fungsi lain dari CSS yakni untuk memisahkan konten dari tampilan visual dalam sebuah *website*. CSS memberi kemudahan dengan tidak perlu mendeskripsikan tampilan masing-masing elemen secara berulang. [5]

3.4. Bootstrap

Bootstrap merupakan sebuah *library framework* CSS yang telah dibuat khusus untuk mengembangkan *front end* sebuah *website*. Bootstrap merupakan *framework* CSS, HTML, Javascript. Bootstrap digunakan untuk mengembangkan *website* agar lebih responsif. Untuk menggunakan bootstrap, hanya perlu memanggil setiap kelas yang digunakan. [6]

3.5. JavaScript

JavaScript adalah bahasa pemrograman yang digunakan untuk menambahkan fitur interaktif pada *website*, seperti ketika ingin membuat *game*, melakukan perubahan ketika mengklik tombol, efek dinamik, animasi, dan masih banyak lagi. JavaScript sendiri adalah bahasa yang cukup kompleks namun sangat fleksibel, dan banyak *Developer (Programmer)* telah menyediakan *tool* yang berdiri di atas *core* JavaScript agar anda dapat menggunakan fungsi - fungsi ekstra. [7]

3.6. JQuery

JQuery adalah library JavaScript multiplatform yang dirancang untuk memudahkan penyusunan *client-side script* pada file HTML. Sintaks jQuery dirancang sedemikian rupa untuk memudahkan pengembang *website* dalam menavigasi dokumen, menyeleksi elemen-elemen DOM, menerapkan animasi, mengaplikasikan *events*, serta membangun aplikasi AJAX. [8]

3.7. PHP

PHP merupakan singkatan rekursif *Hypertext Preprocessor*. Definisi dari PHP adalah bahasa pemrograman *server-side* yang dirancang untuk pengembangan web. PHP dikatakan *server-side* karena program yang diberikan akan dijalankan/diproses pada komputer yang bertindak sebagai server. [9]

3.8. Laravel

Laravel adalah sebuah framework PHP yang dirilis dibawah lisensi MIT, dibangun dengan konsep MVC (model view controller). Laravel adalah pengembangan *website* berbasis MVP yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan menghemat waktu. [10]

3.9. Web Server

Web server adalah istilah yang kerap digunakan untuk menyebut sebuah perangkat keras dan perangkat lunak yang berperan dalam menyediakan layanan akses atas berkas-berkas pada suatu situs web. Akses data tersebut kemudian didistribusikan kepada pengguna lewat protokol HTTP maupun HTTPS. Secara umum web server bisa disimpulkan sebagai pemberi data kepada web *client (browser)* agar bisa menampilkan konten dari halaman *website* tertentu. [11]

3.10. MySQL

MySQL adalah DBMS yang *open source* dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi MySQL adalah *database server* yang gratis dengan lisensi GNU *General Public License* (GPL) sehingga dapat Anda pakai untuk keperluan pribadi atau komersil tanpa harus membayar lisensi yang ada. [12]

3.11. Visual Studio Code

Visual Studio Code (VS Code) adalah sebuah teks editor multiplatform yang komplit dan handal buatan Microsoft. Selain tersedia untuk Windows, Visual Studio Code (VS Code) juga tersedia untuk versi Linux dan Mac. Teks editor mendukung banyak bahasa pemrograman seperti JavaScript, Typescript, dan Node.js, serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang di Visual Studio Code seperti C++, C#, Python, Go, Java, dll. [13]

3.12. Git

Git adalah salah satu sistem pengontrol versi (*Version Control System*) pada proyek perangkat lunak yang diciptakan oleh Linus Torvalds. Pengontrol versi bertugas mencatat setiap perubahan pada file proyek yang dikerjakan oleh banyak orang maupun sendiri. Git dikenal juga dengan *distributed revision control* (VCS terdistribusi), artinya penyimpanan database Git tidak hanya berada dalam satu tempat saja. [14]

BAB IV

ANALISIS DAN DESAIN

4.1 Analisis Sistem

4.4.1 Definisi Umum Fitur

Secara umum, sistem *monitoring* ketinggian air merupakan sebuah sistem berbasis *web* yang dapat digunakan oleh Dinas Kominfo Kota Surabaya untuk melakukan *monitoring* ketinggian air pada sebuah rumah pompa dan mempermudah komunikasi antara petugas Kominfo dengan petugas penjaga rumah pompa. Selain itu, sistem ini dapat digunakan untuk memudahkan proses pemantauan penggunaan BBM pada setiap rumah pompa beserta daya tampungnya.

Sistem *monitoring* ketinggian air memiliki 2 tipe pengguna, yaitu administrator dan petugas. Administrator dapat menambahkan *user*, melihat data *user*, memonitor ketinggian air pada rumah pompa, menambahkan data lokasi rumah pompa, melihat detail data rumah pompa, mengirimkan notifikasi, dan dapat mengelola akun *user*. Petugas rumah pompa dapat melihat ketinggian air pada lokasi yang dijaga melalui *website* dan dapat melihat detail data rumah pompa seperti batas status normal, waspada, bahaya, daya tampung BBM dan jumlah BBM yang telah digunakan.

4.4.2 Analisis Kebutuhan Fungsional

Penjelasan mengenai fitur-fitur yang harus ada dalam sistem *monitoring* ketinggian air

dijelaskan oleh pembimbing lapangan Kerja Praktik. Dari hasil penjelasan tersebut menghasilkan kebutuhan fungsional. Kebutuhan fungsional yang diperlukan pada sistem *monitoring* ketinggian air dapat dilihat pada Tabel 4.1 dibawah ini.

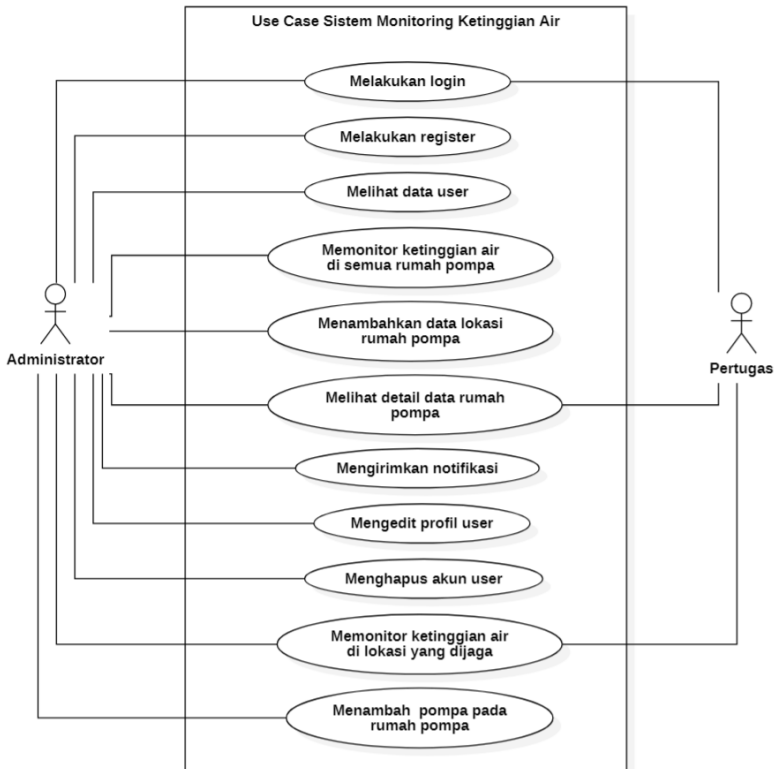
Tabel 4.1 Kebutuhan Fungsional

| Kode Kebutuhan | Deskripsi Kebutuhan |
|----------------|--|
| FR-001 | Melakukan <i>login</i> |
| FR-002 | Melakukan <i>register</i> |
| FR-003 | Melihat data <i>user</i> |
| FR-004 | Memonitor ketinggian air di semua rumah pompa |
| FR-005 | Menambahkan data lokasi rumah pompa |
| FR-006 | Melihat detail data rumah pompa |
| FR-007 | Mengirimkan notifikasi |
| FR-008 | Mengedit profil <i>user</i> |
| FR-009 | Menghapus akun <i>user</i> |
| FR-010 | Memonitor ketinggian air di lokasi yang dijaga |
| FR-011 | Menambah pompa pada rumah pompa |

4.2 Diagram Kasus Penggunaan

Daftar kebutuhan fungsional dapat direpresentasikan menjadi diagram kasus penggunaan (*Use Case Diagram*) sehingga memudahkan untuk

dipahami. *Use Case* Diagram yang telah dibuat dapat dilihat pada Gambar 4.1.



Gambar 4.1 Diagram Use Case Sistem Monitoring Ketinggian Air

4.3 Spesifikasi Kasus Penggunaan

4.3.1. Melakukan *login*

Salah satu kebutuhan yang ada pada *website monitoring* ketinggian air adalah melakukan login. Spesifikasi kasus penggunaan untuk kebutuhan melakukan login dapat dilihat pada tabel 4.2.

Tabel 4.2 Tabel usecase melakukan login

| | |
|-----------------------------|--|
| Nama | Melakukan <i>login</i> |
| Kode | UC001 |
| Deskripsi | Sebelum masuk ke sistem, pengguna baik <i>user</i> maupun admin melakukan <i>login</i> terlebih dulu. |
| Tipe | Fungsional |
| Pemicu | Pengguna melakukan <i>login</i> |
| Aktor | Pengguna |
| Kondisi Awal | Pengguna belum masuk ke sistem |
| Kondisi Akhir | Pengguna sudah login dengan akun masing-masing. |
| Alur Kejadian Secara Normal | <ol style="list-style-type: none"> 1. Aktor masuk halaman <i>login</i>. 2. Aktor mengisikan email dan password. 4. Sistem memberi <i>session</i> dengan hak akses sebagai administrator / petugas. 5. Sistem menampilkan halaman <i>dashboard</i> sistem <i>monitoring</i> ketinggian air. |
| Alur Kejadian Alternatif | - |
| Pengecualian | - |

4.3.2. Melakukan *register*

Salah satu kebutuhan yang ada pada *website monitoring* ketinggian air adalah melakukan *register*. Spesifikasi kasus penggunaan untuk kebutuhan melakukan *register* dapat dilihat pada tabel 4.3.

Tabel 4.3 Tabel usecase melakukan register

| | |
|-----------------------------|--|
| Nama | Melakukan <i>register</i> |
| Kode | UC002 |
| Deskripsi | Sebelum masuk ke sistem, pengguna baik petugas maupun administrator melakukan <i>register</i> untuk membuat akun. Administator dapat memberikan hak akses sebagai admin maupun hak akses sebagai petugas kepada pengguna. |
| Tipe | Fungsional |
| Pemicu | Administrator menekan tombol tambah akun setelah berhasil login. |
| Aktor | Administrator |
| Kondisi Awal | Pengguna belum mempunyai akun |
| Kondisi Akhir | Pengguna mempunyai akun untuk masuk ke sistem. |
| Alur Kejadian Secara Normal | <ol style="list-style-type: none">1. Aktor <i>login</i> ke sistem.2. Aktor menekan tombol tambah akun.3. Aktor mengisi data diri pengguna dan meberikan hak akses sebagai seorang petugas / sebagai seorang administrator. |

| | |
|--------------------------|---|
| | <p>4. Sistem menerima <i>request</i> dan menyimpan data yang telah diisikan ke database.</p> <p>5. Sistem mengirimkan respon dengan menampilkan halaman daftar <i>user</i>.</p> |
| Alur Kejadian Alternatif | - |
| Pengecualian | - |

4.3.3. Melihat data *user*

Salah satu kebutuhan yang ada pada *website monitoring* ketinggian air adalah melakukan melihat data *user*. Spesifikasi kasus penggunaan untuk kebutuhan melihat data *user* dapat dilihat pada tabel 4.4.

Tabel 4.4 Tabel usecase melihat data *user*

| | |
|-----------------------------|--|
| Nama | Melihat data <i>user</i> |
| Kode | UC003 |
| Deskripsi | Seorang administrator dapat melihat semua data <i>user</i> yang telah teregistrasi. |
| Tipe | Fungsional |
| Pemicu | Administrator menekan menu pengaturan dan memilih tab <i>user</i> setelah melakukan <i>login</i> . |
| Aktor | Administrator |
| Kondisi Awal | Data <i>user</i> belum muncul pada <i>website</i> . |
| Kondisi Akhir | Sistem menampilkan semua data <i>user</i> yang telah telah teregistrasi. |
| Alur Kejadian Secara Normal | <ol style="list-style-type: none">1. Aktor masuk halaman <i>login</i>.2. Aktor menekan menu pengaturan dan memilih tab <i>user</i>.3. Sistem menerima <i>request</i> dan mengirimkan respon dengan menampilkan semua data <i>user</i> yang telah teregistrasi. |
| Alur Kejadian Alternatif | - |
| Pengecualian | - |

4.3.4. Memonitoring ketinggian air di semua rumah pompa

Salah satu kebutuhan yang ada pada *website monitoring* ketinggian air adalah memonitor ketinggian air di semua rumah pompa. Spesifikasi kasus penggunaan untuk kebutuhan memonitor ketinggian air di semua rumah pompa dapat dilihat pada tabel 4.5.

Tabel 4.5 Tabel usecase memonitor ketinggian air di semua rumah pompa

| | |
|-----------------------------|---|
| Nama | Memonitor ketinggian air di semua rumah pompa |
| Kode | UC004 |
| Deskripsi | Pengguna yang memiliki hak akses sebagai seorang administrator dapat memonitor ketinggian air di semua rumah pompa. |
| Tipe | Fungsional |
| Pemicu | Pengguna menekan menu <i>monitoring</i> . |
| Aktor | Pengguna |
| Kondisi Awal | Data <i>monitoring</i> belum muncul pada <i>website</i> . |
| Kondisi Akhir | Sistem menampilkan semua data rumah pompa. |
| Alur Kejadian Secara Normal | 1. Aktor masuk halaman <i>login</i> . 2. Aktor menekan menu <i>monitoring</i> . 3. Sistem menerima <i>request</i> dan mengirimkan respon dengan menampilkan semua data rumah pompa. |

| | |
|--------------------------|---|
| Alur Kejadian Alternatif | Setelah pengguna berhasil <i>login</i> ke sistem, secara otomatis sistem menampilkan data rumah pompa yang tersedia pada <i>website</i> . |
| Pengecualian | - |

4.3.5. Menambahkan data lokasi rumah pompa

Salah satu kebutuhan yang ada pada *website monitoring* ketinggian air adalah menambahkan data lokasi rumah pompa. Spesifikasi kasus penggunaan untuk kebutuhan menambahkan data lokasi rumah pompa dapat dilihat pada tabel 4.6.

Tabel 4.6 Tabel usecase menambahkan data lokasi rumah pompa

| | |
|-----------------------------|--|
| Nama | Menambahkan data lokasi rumah pompa |
| Kode | UC005 |
| Deskripsi | Seorang administrator dapat menambahkan data lokasi rumah pompa. |
| Tipe | Fungsional |
| Pemicu | Administaror menekan menu tambah lokasi. |
| Aktor | Administrator |
| Kondisi Awal | Data lokasi rumah pompa belum ditambahkan. |
| Kondisi Akhir | Data lokasi rumah pompa berhasil ditambahkan. |
| Alur Kejadian Secara Normal | <ol style="list-style-type: none">1. Aktor masuk halaman <i>login</i>.2. Aktor menekan menu tambah lokasi.3. Sistem menampilkan <i>dialog box</i> tambah lokasi.4. Aktor mengisi data rumah pompa.5. Sistem menerima <i>request</i> dan menyimpan data yang telah diisikan ke <i>database</i>. |

| | |
|--------------------------|---|
| | 6. Sistem mengirimkan respon dengan menampilkan <i>pop up</i> data lokasi rumah pompa berhasil ditambahkan. |
| Alur Kejadian Alternatif | - |
| Pengecualian | - |

4.3.6. Melihat detail data rumah pompa

Salah satu kebutuhan yang ada pada *website monitoring* ketinggian air adalah melihat detail data rumah pompa. Spesifikasi kasus penggunaan untuk kebutuhan melihat detail data rumah pompa dapat dilihat pada tabel 4.7.

Tabel 4.7 Tabel usecase melihat detail data rumah pompa

| | |
|-----------------------------|--|
| Nama | Melihat detail data rumah pompa |
| Kode | UC006 |
| Deskripsi | Seorang administrator dapat melihat semua detail data rumah pompa yang ada pada website. |
| Tipe | Fungsional |
| Pemicu | Administaror menekan menu ‘Data rumah pompa’. |
| Aktor | Pengguna |
| Kondisi Awal | Detail data rumah pompa belum muncul pada <i>website</i> . |
| Kondisi Akhir | Sistem menampilkan detail data rumah pompa. |
| Alur Kejadian Secara Normal | <ol style="list-style-type: none">1. Aktor masuk halaman <i>login</i> dan melakukan login sebagai pengguna.2. Aktor menekan menu ‘Data rumah pompa’.3. Sistem menerima <i>request</i> dan mengirimkan respon dengan menampilkan detail data rumah pompa. |
| Alur Kejadian Alternatif | |

| | |
|--------------|---|
| Pengecualian | - |
|--------------|---|

4.3.7. Mengirimkan notifikasi

Salah satu kebutuhan yang ada pada *website monitoring* ketinggian air adalah mengirimkan notifikasi. Spesifikasi kasus penggunaan untuk kebutuhan mengirimkan notifikasi dapat dilihat pada tabel 4.8.

Tabel 4.8 Tabel usecase mengirimkan notifikasi

| | |
|-----------------------------|--|
| Nama | Mengirimkan Notifikasi |
| Kode | UC007 |
| Deskripsi | Sistem dapat mengirimkan notifikasi kepada petugas penjaga rumah pompa melalui telegram ketika ketinggian air berada pada status waspada. |
| Tipe | Fungsional |
| Pemicu | Ketinggian air di suatu rumah pompa berada pada status waspada. |
| Aktor | Petugas |
| Kondisi Awal | Notifikasi belum dikirim ke petugas penjaga rumah pompa. |
| Kondisi Akhir | Petugas rumah pompa mendapatkan notifikasi dari administrator. |
| Alur Kejadian Secara Normal | <ol style="list-style-type: none">1. Ketinggian rumah pompa mencapai status waspada.2. Sistem mengirimkan notifikasi ketinggian air kepada petugas melalui telegram.3. Petugas menerima notifikasi pesan pada aplikasi telegram. |

| | |
|--------------------------|---|
| Alur Kejadian Alternatif | - |
| Pengecualian | - |

4.3.8. Mengedit profil *user*

Salah satu kebutuhan yang ada pada *website monitoring* ketinggian air adalah mengedit profil *user*. Spesifikasi kasus penggunaan untuk kebutuhan mengedit profil *user* dapat dilihat pada tabel 4.9.

Tabel 4.9 Tabel usecase mengedit profil *user*

| | |
|-----------------------------|--|
| Nama | Mengedit profil <i>user</i> |
| Kode | UC008 |
| Deskripsi | Seorang administrator dapat mengedit profil <i>user</i> yang telah teregistrasi pada sistem. |
| Tipe | Fungsional |
| Pemicu | Jika ada perubahan data pada akun <i>user</i> . |
| Aktor | Administrator |
| Kondisi Awal | Akun <i>user</i> belum berubah. |
| Kondisi Akhir | Akun <i>user</i> telah berubah sesuai dengan data baru yang diinputkan. |
| Alur Kejadian Secara Normal | <ol style="list-style-type: none">1. Aktor masuk halaman <i>login</i> dan melakukan <i>login</i> sebagai administrator.2. Aktor menekan menu ‘Pengaturan’ tab menu ‘<i>User</i>’ dan menekan opsi ‘Edit’.3. Aktor melakukan perubahan sesuai dengan data baru yang diinputkan.4. Sistem menerima <i>request</i> dan mengirimkan respon dengan menampilkan data hasil perubahan. |

| | |
|--------------------------|---|
| Alur Kejadian Alternatif | - |
| Pengecualian | - |

4.3.9. Menghapus akun *user*

Salah satu kebutuhan yang ada pada *website monitoring* ketinggian air adalah menghapus akun *user*. Spesifikasi kasus penggunaan untuk kebutuhan menghapus akun *user* dapat dilihat pada tabel 4.10.

Tabel 4.10 Tabel usecase menghapus akun *user*

| | |
|-----------------------------|--|
| Nama | Menghapus akun <i>user</i> . |
| Kode | UC009 |
| Deskripsi | Seorang administrator dapat menghapus profil <i>user</i> yang telah teregistrasi pada sistem. |
| Tipe | Fungsional |
| Pemicu | Administrator menekan menu 'Pengaturan' tab menu ' <i>User</i> ' dan menekan opsi 'Hapus'. |
| Aktor | Administrator |
| Kondisi Awal | Data <i>user</i> belum terhapus dalam sistem. |
| Kondisi Akhir | Data <i>user</i> telah telah terhapus dalam sistem. |
| Alur Kejadian Secara Normal | <ol style="list-style-type: none">1. Aktor masuk halaman <i>login</i> dan melakukan <i>login</i> sebagai administrator.2. Aktor menekan menu 'Pengaturan' tab menu '<i>User</i>' dan menekan opsi 'Hapus'.3. Sistem menerima <i>request</i> dan mengirimkan respon dengan menampilkan daftar <i>user</i> dalam sistem. |

| | |
|--------------------------|---|
| Alur Kejadian Alternatif | - |
| Pengecualian | - |

4.3.10. Memonitoring ketinggian air di lokasi yang dijaga

Salah satu kebutuhan yang ada pada *website monitoring* ketinggian air adalah memonitor ketinggian air di lokasi yang dijaga. Spesifikasi kasus penggunaan untuk kebutuhan memonitor ketinggian air di lokasi yang dijaga dapat dilihat pada tabel 4.11.

Tabel 4.11 Tabel usecase memonitoring ketinggian air di lokasi yang di jaga

| | |
|-----------------------------|---|
| Nama | Memonitor ketinggian air di lokasi yang dijaga. |
| Kode | UC0010 |
| Deskripsi | Pengguna yang memiliki hak akses sebagai petugas dapat memonitor ketinggian air di lokasi rumah pompa yang dijaga. |
| Tipe | Fungsional |
| Pemicu | Petugas menekan menu <i>monitoring</i> . |
| Aktor | Petugas |
| Kondisi Awal | Data <i>monitoring</i> belum muncul pada <i>website</i> . |
| Kondisi Akhir | Sistem menampilkan data rumah pompa yang dijaga oleh petugas yang bersangkutan. |
| Alur Kejadian Secara Normal | 1. Aktor masuk halaman <i>login</i> dan melakukan <i>login</i> sebagai petugas. 2. Aktor menekan menu <i>monitoring</i> . 3. Sistem menerima <i>request</i> dan mengirimkan respon dengan |

| | |
|--------------------------|---|
| | menampilkan data rumah pompa pada lokasi yang digaja oleh petugas rumah pompa yang bersangkutan. |
| Alur Kejadian Alternatif | Setelah petugas berhasil <i>login</i> ke sistem, secara otomatis sistem menampilkan data rumah pompa yang pada lokasi yang digaja oleh petugas rumah pompa yang bersangkutan. |
| Pengecualian | - |

4.3.11. Menambah pompa pada rumah pompa

Salah satu kebutuhan yang ada pada *website monitoring* ketinggian air adalah menambah pompa pada rumah pompa. Spesifikasi kasus penggunaan untuk kebutuhan menambah pompa pada rumah pompa dapat dilihat pada tabel 4.12.

Tabel 4.12 Tabel usecase menambah pompa pada rumah pompa

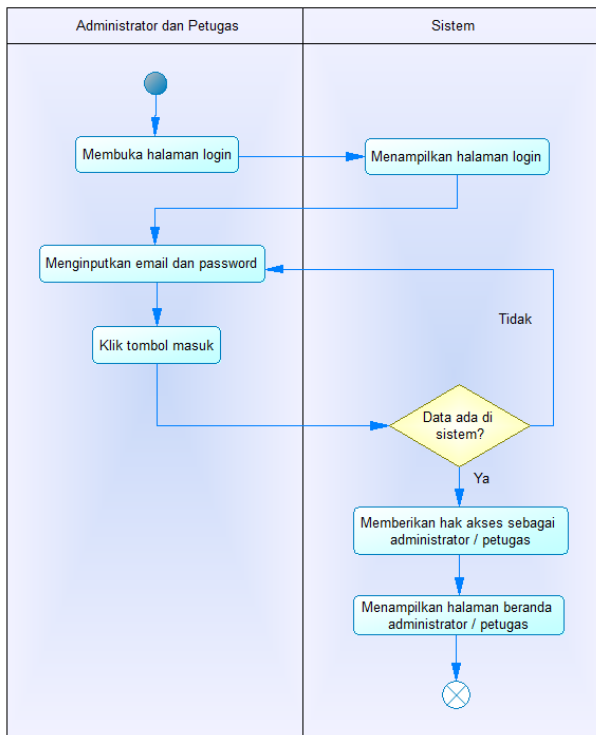
| | |
|-----------------------------|--|
| Nama | Menambah pompa pada rumah pompa. |
| Kode | UC0011 |
| Deskripsi | Pengguna yang memiliki hak akses sebagai petugas dapat menambah pompa pada rumah pompa. |
| Tipe | Fungsional |
| Pemicu | Administrator menambahkan data pompa pada rumah pompa. |
| Aktor | Administrator |
| Kondisi Awal | Data pompa pada rumah pompa belum ditambahkan. |
| Kondisi Akhir | Sistem menampilkan data pompa yang berhasil ditambahkan. |
| Alur Kejadian Secara Normal | <ol style="list-style-type: none">1. Aktor masuk halaman login dan melakukan <i>login</i> sebagai administrator.2. Aktor menekan menu 'Daftar rumah pompa'.3. Sistem menampilkan daftar rumah pompa yang terdaftar pada sistem.4. Aktor memilih rumah pompa mana yang akan ditambahkan pompa baru |

| | |
|--------------------------|--|
| | <p>dengan menekan nama lokasi rumah pompa.</p> <p>5. Sistem menampilkan daftar pompa pada rumah pompa yang dipilih.</p> <p>6. Aktor menekan tombol ‘tambah pompa’ dan mengisi <i>form</i> yang ditamamkan sistem.</p> <p>7. Sistem menampilkan data pompa yang berhasil ditambahkan.</p> |
| Alur Kejadian Alternatif | - |
| Pengecualian | - |

4.4 Diagram Aktivitas

4.4.1 Melakukan *login*

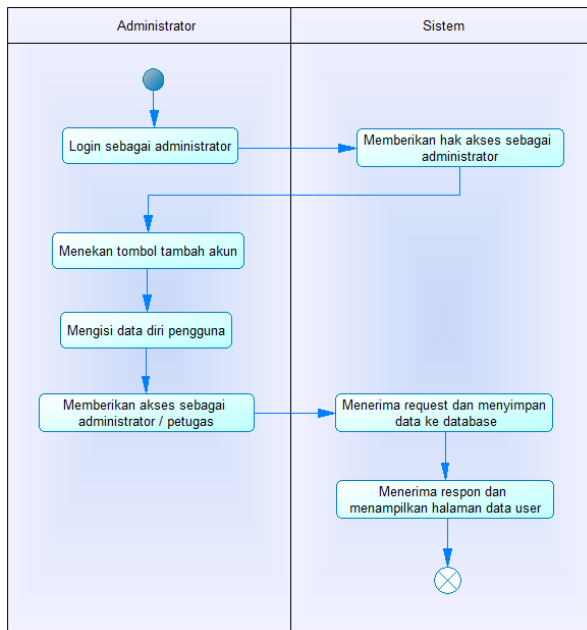
Alur kejadian normal dan alternatif dari sebuah kebutuhan fungsional dapat disajikan dalam diagram aktivitas. Diagram aktivitas dari kebutuhan melakukan login dapat dilihat pada gambar 4.2.



Gambar 4.2 Diagram Aktivitas Melakukan Login

4.4.2 Melakukan *register*

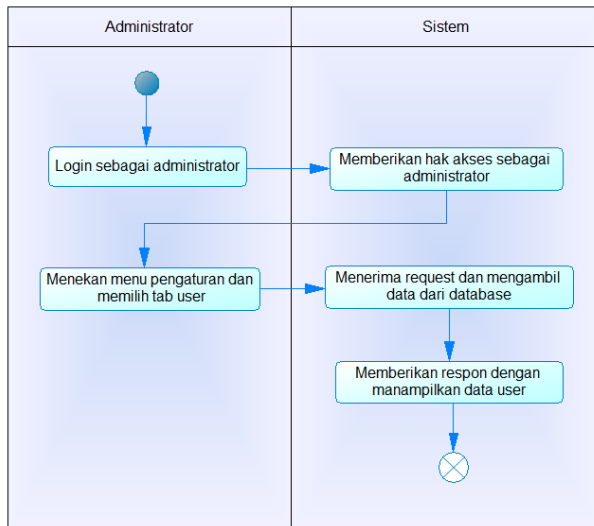
Alur kejadian normal dan alternatif dari sebuah kebutuhan fungsional dapat disajikan dalam diagram aktivitas. Diagram aktivitas dari kebutuhan melakukan *register* dapat dilihat pada gambar 4.3.



Gambar 4.3 Diagram Aktivitas Melakukan Register

4.4.3 Melihat data *user*

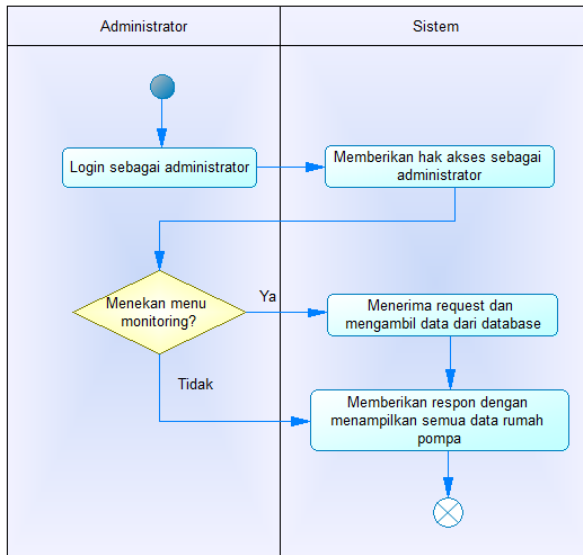
Alur kejadian normal dan alternatif dari sebuah kebutuhan fungsional dapat disajikan dalam diagram aktivitas. Diagram aktivitas dari kebutuhan melihat data *user* dapat dilihat pada gambar 4.4.



Gambar 4.4 Diagram Aktivitas Melihat Data User

4.4.4 Memonitor ketinggian air di semua rumah pompa

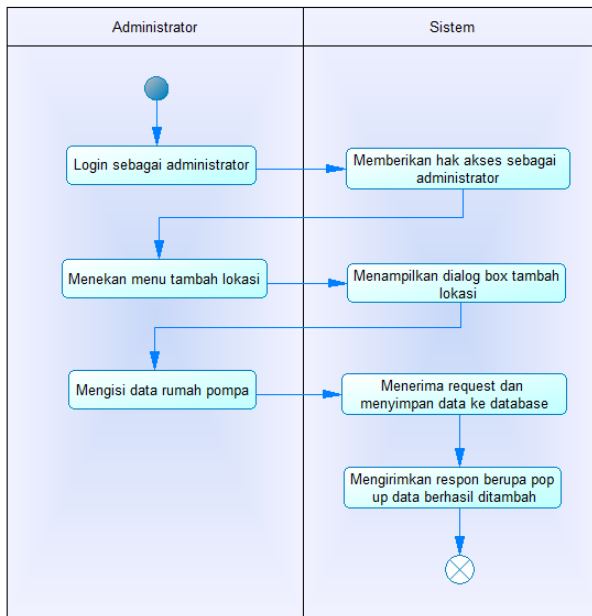
Alur kejadian normal dan alternatif dari sebuah kebutuhan fungsional dapat disajikan dalam diagram aktivitas. Diagram aktivitas dari kebutuhan memonitor ketinggian air di semua rumah pompa dapat dilihat pada gambar 4.5.



Gambar 4.5 Diagram Aktivitas Memonitor Ketinggian Air di Semua Rumah Pompa

4.4.5 Menambahkan data lokasi rumah pompa

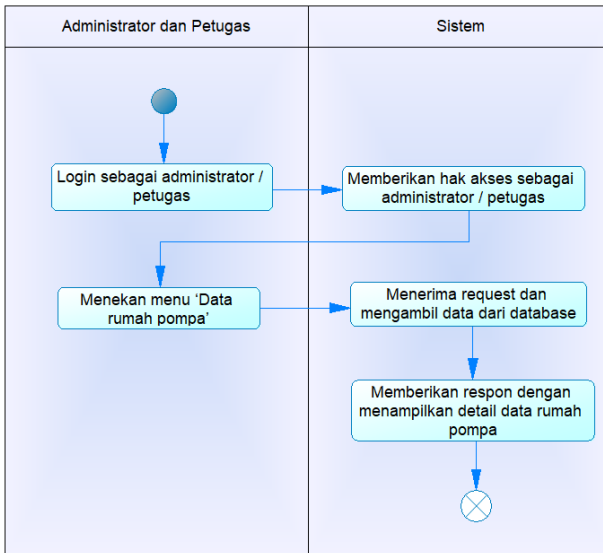
Alur kejadian normal dan alternatif dari sebuah kebutuhan fungsional dapat disajikan dalam diagram aktivitas. Diagram aktivitas dari kebutuhan menambahkan data lokasi rumah pompa dapat dilihat pada gambar 4.6.



Gambar 4.6 Diagram Aktivitas Menambahkan Data Lokasi Rumah Pompa

4.4.6 Melihat detail data rumah pompa

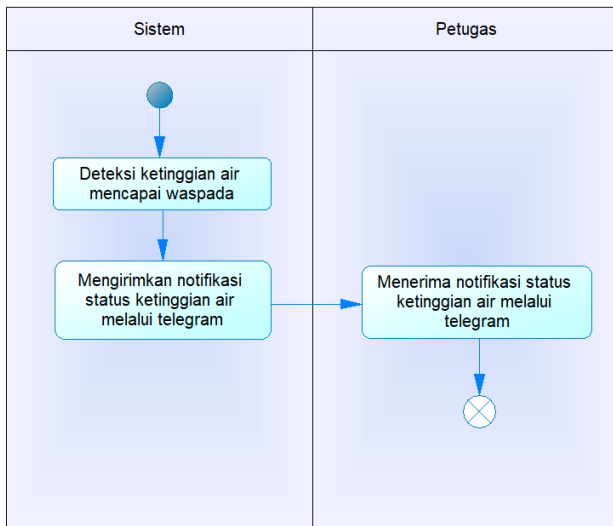
Alur kejadian normal dan alternatif dari sebuah kebutuhan fungsional dapat disajikan dalam diagram aktivitas. Diagram aktivitas dari kebutuhan melihat detail data rumah pompa dapat dilihat pada gambar 4.7.



Gambar 4.7 Diagram Aktivitas Melihat Detail Data Rumah Pompa

4.4.7 Mengirimkan notifikasi

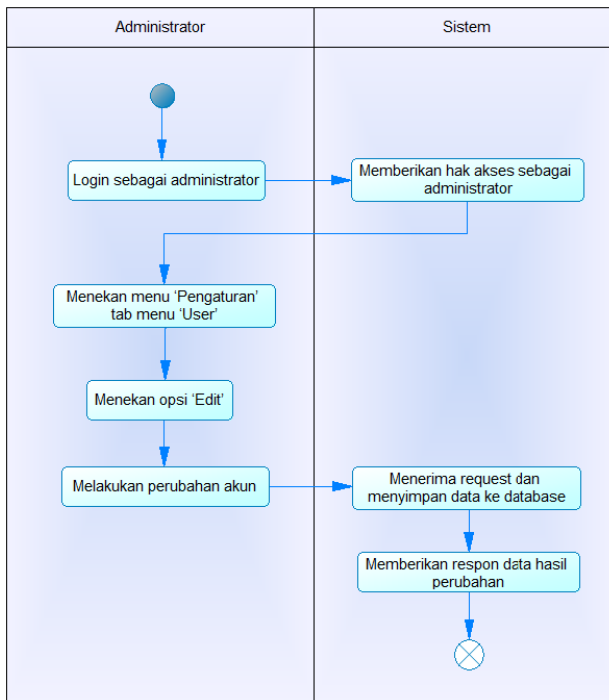
Alur kejadian normal dan alternatif dari sebuah kebutuhan fungsional dapat disajikan dalam diagram aktivitas. Diagram aktivitas dari kebutuhan mengirimkan notifikasi dapat dilihat pada gambar 4.8.



Gambar 4.8 Diagram Aktivitas Mengirimkan Notifikasi

4.4.8 Mengedit profil *user*

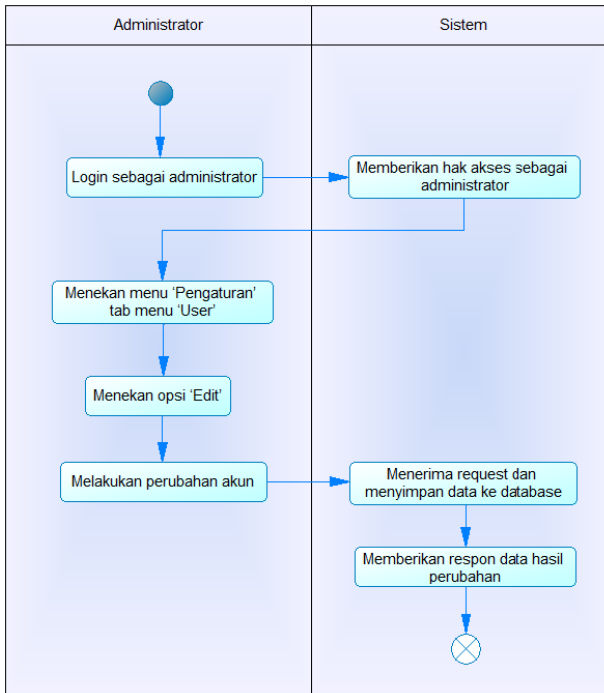
Alur kejadian normal dan alternatif dari sebuah kebutuhan fungsional dapat disajikan dalam diagram aktivitas. Diagram aktivitas dari kebutuhan mengedit profil *user* dapat dilihat pada gambar 4.9.



Gambar 4.9 Diagram Aktivitas Mengedit Profil User

4.4.9 Menghapus akun *user*

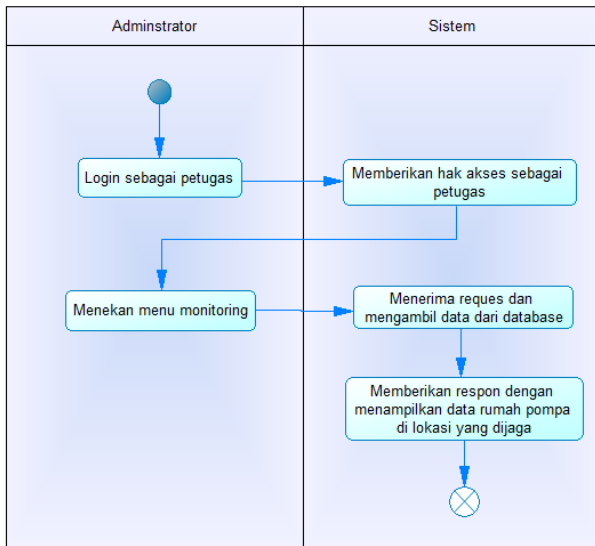
Alur kejadian normal dan alternatif dari sebuah kebutuhan fungsional dapat disajikan dalam diagram aktivitas. Diagram aktivitas dari kebutuhan menghapus akun *user* dapat dilihat pada gambar 4.10.



Gambar 4.10 Diagram Aktivitas Menghapus Akun User

4.4.10 Memonitor ketinggian air di lokasi yang dijaga

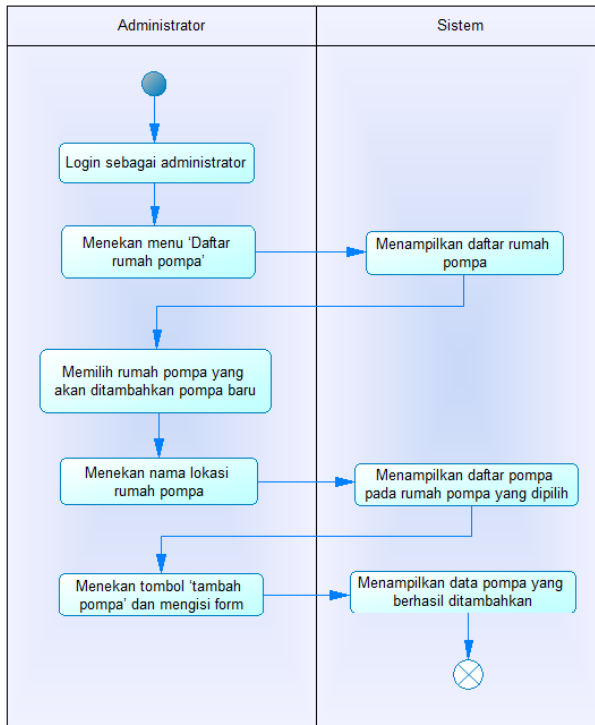
Alur kejadian normal dan alternatif dari sebuah kebutuhan fungsional dapat disajikan dalam diagram aktivitas. Diagram aktivitas dari kebutuhan memonitor ketinggian air di lokasi yang dijaga dapat dilihat pada gambar 4.11.



Gambar 4.11 Diagram Aktivitas Memonitor Ketinggian Air di Lokasi yang Dijaga

4.4.11 Menambah pompa pada rumah pompa

Alur kejadian normal dan alternatif dari sebuah kebutuhan fungsional dapat disajikan dalam diagram aktivitas. Diagram aktivitas dari kebutuhan menambah pompa pada rumah pompa dapat dilihat pada gambar 4.12.

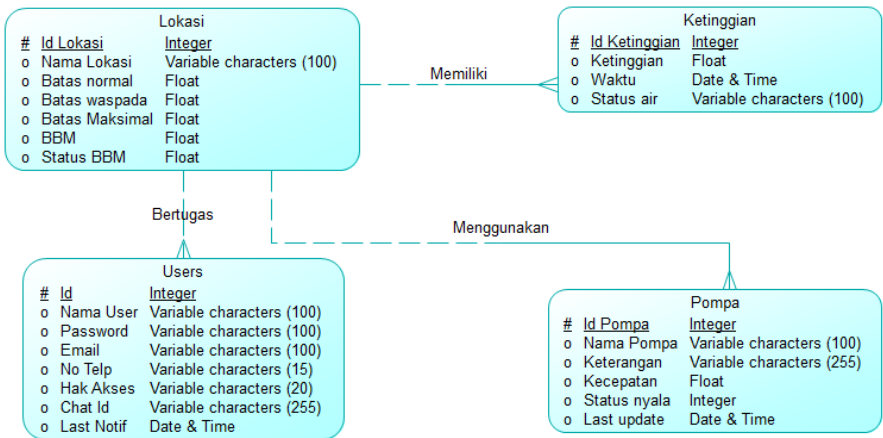


Gambar 4.12 Diagram Aktivitas Menambah Pompa pada Rumah Pompa

4.5 Desain Basis Data

4.5.1 Desain Konseptual Data

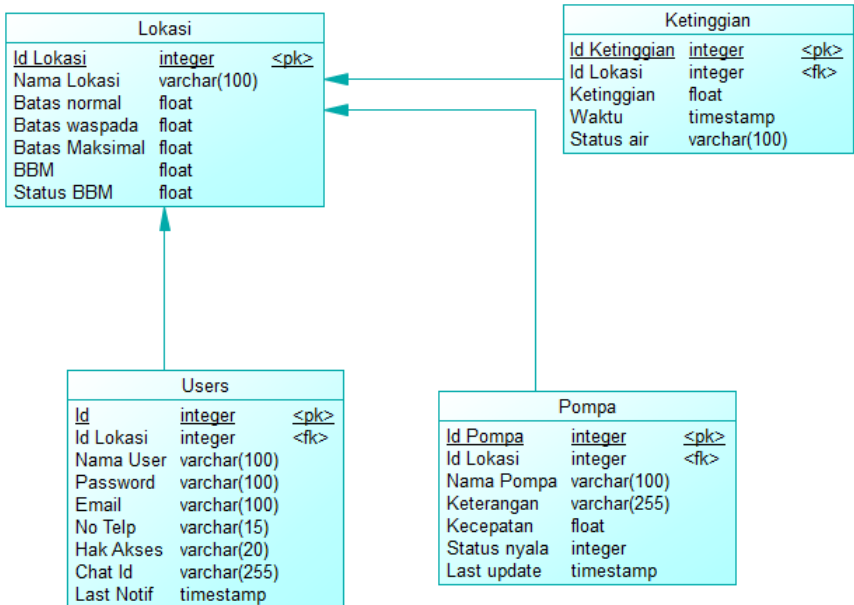
Dalam membangun *website monitoring* ketinggian air diperlukan sebuah *database* untuk menyimpan data-data yang diperlukan. Desain konseptual dari *database* yang dirancang dapat dilihat pada gambar 4.13.



Gambar 4.13 Conceptual Data Model

4.5.2 Desain Fisik Data

Berdasarkan desain konseptual data, dibuat desain fisik yang kemudian akan diimplementasikan menggunakan SQL. Desain fisik dari *database* yang diimplementasikan dapat dilihat pada gambar 4.14.



Gambar 4.14 Physical Data Model

4.5.3 Struktur Tabel *User*

Tabel *user* digunakan untuk menyimpan data diri user. Struktur tabel *user* dapat dilihat pada tabel 4.13.

Tabel 4.13 Struktur Tabel User

| No | Nama Field | Tipe Data | Panjang | Keterangan |
|-----|------------|-----------|---------|--|
| 1. | Id | Integer | - | <i>Primary key</i> dan <i>auto increment</i> . |
| 2. | Nama | Varchar | 100 | Nama dari <i>user</i> . |
| 3. | Email | Varchar | 100 | Email dari <i>user</i> . |
| 8. | Password | Varchar | 100 | Password dari <i>user</i> . |
| 9. | No_telp | Varchar | 15 | Nomor telepon dari <i>user</i> . |
| 10. | Hak_akses | Varchar | 20 | Hak Akses dari masing-masing <i>user</i> . |
| 11. | Id_lokasi | Integer | - | Foreign Key dari tabel Lokasi. |
| 12. | Chat_id | Varchar | 255 | ID dari chat <i>user</i> telegram |
| 13. | Last_notif | Datetime | - | Waktu dan tanggal terakhir kali diberikan notifikasi |

4.5.4 Struktur Tabel Lokasi

Tabel lokasi digunakan untuk menyimpan detail informasi lokasi rumah pompa yang terdaftar pada sistem. Struktur tabel lokasi dapat dilihat pada tabel 4.14.

Tabel 4.14 Struktur Tabel Lokasi

| No | Nama Field | Tipe Data | Panjang | Keterangan |
|----|----------------|-----------|---------|---|
| 1. | Id_lokasi | Int | - | <i>Primary key</i> dan <i>auto increment</i> . |
| 2. | Nama | Varchar | 100 | Nama dari lokasi rumah pompa. |
| 3. | Batas_normal | Float | - | Batas normal ketinggian air rumah pompa. |
| 4. | Batas_waspada | Float | - | Batas waspada ketinggian air rumah pompa. |
| 5. | Batas_maksimal | Float | - | Batas maksimal ketinggian air rumah pompa |
| 6. | BBM | Float | - | Banyak bbm yang tersedia pada pompa rumah pompa |

| | | | | |
|----|------------|---------|---|-----------------------------------|
| 7. | Status_bbm | Integer | - | Status BBM pompa pada rumah pompa |
|----|------------|---------|---|-----------------------------------|

4.5.5 Struktur Tabel Pompa

Tabel pompa digunakan untuk menyimpan detail informasi pompa yang telah dibuat oleh *user*. Struktur tabel pompa dapat dilihat pada tabel 4.15.

Tabel 4.15 Struktur Tabel Pompa

| No | Nama Field | Tipe Data | Panjang | Keterangan |
|----|--------------|-----------|---------|--|
| 1. | Id_pompa | Integer | - | <i>Primary key</i> dan <i>auto increment</i> . |
| 2. | Nama | Varchar | 255 | Nama dari pompa. |
| 3. | Keterangan | Varchar | 255 | Keterangan dari pompa. |
| 4. | Kecepatan | Float | - | Kecepatan dari pompa. |
| 5. | Status_nyala | Integer | - | Status nyala dari pompa. |
| 6. | Id_lokasi | Integer | - | <i>Foreign key</i> dari tabel lokasi. |
| 7. | Last_update | Int | - | Nomor surat. |

4.5.6 Struktur Tabel Ketinggian

Tabel ketinggian digunakan untuk menyimpan detail informasi ketinggian air rumah pompa yang dikirim oleh *script* yang dijalankan pada *detector* di rumah pompa. Struktur tabel ketinggian dapat dilihat pada tabel 4.16.

Tabel 4.16 Struktur Tabel Ketinggian

| No | Nama Field | Tipe Data | Panjang | Keterangan |
|----|---------------|-----------|---------|--|
| 1. | Id_ketinggian | Integer | - | <i>Primary key</i> dan <i>auto increment</i> . |
| 2. | Ketinggian | Float | - | Ketinggian air pada rumah pompa. |
| 3. | Waktu | Datetime | - | Waktu direkamnya ketinggian air. |
| 4. | Status_air | Varchar | 100 | Status dari ketinggian air rumah pompa. |
| 5. | Id_lokasi | Integer | - | <i>Foreign key</i> dari tabel lokasi. |

BAB V IMPLEMENTASI SISTEM

Bab ini membahas tentang implementasi dari sistem *monitoring* ketinggian air. Sistem ini dibuat menggunakan metode MVC (*Model View Controller*).

5.1 Implementasi Lapisan Model

Implementasi lapisan model bertugas untuk mengatur, menyiapkan, memanipulasi dan mengorganisasikan data (dari *database*) sesuai dengan instruksi dari *controller*. Pada lapisan model ini terdiri dari model ketinggian, lokasi, pompa dan *user*.

5.1.1 Model Ketinggian

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Ketinggian extends Model
{
    //
    protected $table = 'ketinggian';
    protected $primaryKey = 'id_ketinggian';
    public $timestamp = 'false';
}
```

5.1.2 Model Lokasi

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Lokasi extends Model
{
    protected $table = 'lokasi';

    protected $primaryKey = 'id_lokasi';

    protected $fillable = [
        'nama', 'batas_normal', 'batas_maksimal', '
batas_waspada', 'bbm', 'status_bbm',
    ];

    public $timestamps = false;
}
```

5.1.3 Model Pompa

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Pompa extends Model
{
```

```

//
protected $table = 'pompa';

protected $primaryKey = 'id_pompa';

protected $fillable = [
    'nama', 'keterangan', 'kecepatan', 'status_
nyala', 'id_lokasi', 'last_update'
];

public $timestamps = false;
}

```

5.1.4 Model User

```

<?php

namespace App;

use Illuminate\Contracts\Auth\MustVerifyEmail;
use Illuminate\Foundation\Auth\User as Authenticatable;
use Illuminate\Notifications\Notifiable;

class User extends Authenticatable
{
    use Notifiable;

    /**
     * The attributes that are mass assignable.

```

```

*
* @var array
*/

protected $table = 'users';

protected $fillable = [
    'nama', 'email', 'password', 'no_telp', 'ha
k_akses', 'id_lokasi', 'chat_id', 'last_notif'
];

/**
 * The attributes that should be hidden for arr
ays.
 *
 * @var array
 */
protected $hidden = [
    'password', 'remember_token',
];

public $timestamps = false;
}

```

5.2 Implementasi Lapisan Kontrol

Implementasi lapisan kontrol mengatur apa yang harus dilakukan model, dan *view* mana yang harus ditampilkan berdasarkan permintaan dari pengguna.

5.1.5 HomeController

```
<?php

namespace App\Http\Controllers;
use App\User;
use Illuminate\Support\Facades\Hash;
use Illuminate\Http\Request;
use Auth;
use App\Lokasi;

class HomeController extends Controller
{

    public function login (Request $r){
        if (Auth::attempt(['email' => $r->input('email'), 'password' => $r->password]))
        {
            return redirect('');
        }
        return redirect('login');
    }

    public function register(Request $r)
    {
        User::create([
            'nama' => $r->nama,
```

```

        'email' => $r->email,
        'password' => Hash::make($r->password),
        'no_telp' => $r->no_telp,
        'hak_akses' => $r->hak_akses,
        'id_lokasi' => $r->id_lokasi,
    ]);

    return redirect('/daftar_user');
}

public function dash()
{
    $lokasis = Lokasi::where('id_lokasi','>','1
')->get();
    return view('dashboard.dashboard',['lokasis
' => $lokasis]);
}
}

```

5.1.6 KetinggianController

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Lokasi;
use App\Ketinggian;

class KetinggianController extends Controller
{
    //

```

```

public function get_tinggi($id)
{
    $data = Ketinggian::where('id_lokasi',$id)-
>orderBy('waktu', 'DESC')->first();
    $now = \Carbon\Carbon::now();
    $lokasi = Lokasi::find($id);
    if(is_null($data)){
        $mati = 1;
        $tanggal = \Carbon\Carbon::parse($now)-
>format('d-m-Y');
        $waktu = \Carbon\Carbon::parse($now-
>addSeconds(10))->toTimeString();
        return response()->json([
            'tanggal' => $tanggal,
            'waktu' => $waktu,
            'lokasi' => $id,
            'ketinggian' => 0,
            'status' => 'mati',
            'mati' => $mati,
            'bbm' => $lokasi->bbm,
            'status_bbm' => $lokasi->status_bbm
        ]);
    }
    $date = new \Carbon\Carbon($data->waktu);
    if($date <= $now && $date >= $now-
>subSeconds(10)){
        $mati = 0;
        $tanggal = \Carbon\Carbon::parse($data-
>waktu)->format('d-m-Y');
        $waktu = \Carbon\Carbon::parse($data-
>waktu)->toTimeString();

```

```

    }else{
        $mati = 1;
        $tanggal = \Carbon\Carbon::parse($now)-
>format('d-m-Y');
        $waktu = \Carbon\Carbon::parse($now-
>addSeconds(10))->toTimeString();
    }
    return response()->json([
        'tanggal' => $tanggal,
        'waktu' => $waktu,
        'lokasi' => $data->id_lokasi,
        'ketinggian' => $data->ketinggian,
        'status' => $data->status_air,
        'mati' => $mati,
        'bbm' => $lokasi->bbm,
        'status_bbm' => $lokasi->status_bbm
    ]);
}
}
}

```

5.1.7 LokasiController

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Lokasi;

class LokasiController extends Controller
{
    public function store(Request $r)

```



```

{
    Lokasi::create([
        'nama' => $r->nama,
        'batas_normal' => $r->batas_normal,
        'batas_maksimal' => $r->batas_maksimal,
        'batas_waspada' => $r->batas_waspada,
        'bbm' => $r->bbm,
        'status_bbm' => 0,
        'last_update' => \Carbon\Carbon::now()
    ]);

    return redirect('');
}

public function show()
{
    $lokasi = Lokasi::all();
    return view('monitoring.data', ['lokasis' =>
$lokasis]);
}

public function xml($id)
{
    $lokasi = Lokasi::find($id);
    return response()->json([
        'nama' => $lokasi->nama,
        'batas_normal' => $lokasi-
>batas_normal,
        'batas_waspada' => $lokasi-
>batas_waspada,

```

```

        'batas_maksimal' => $lokasi->
>batas_maksimal,
        'bbm' => $lokasi->bmm,
        'status_bbm' => $lokasi->status_bmm
    ]);
}
}

```

5.1.8 MonitoringController

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Lokasi;
use App\Ketinggian;
use App\Pompa;

class MonitoringController extends Controller
{
    public function monitor($id)
    {
        $lokasis = Lokasi::find($id);
        $pompas = Pompa::where('id_lokasi',$id)-
>get();
        $ketinggian = Ketinggian::where('id_lokasi'
,$id)->orderBy('waktu','DESC')->take(5)->get();
        return view('monitoring.monitoring',['lokasi' => $lokasis,'ketinggian' => $ketinggian, 'pompas'=>$pompas]);
    }
}

```

```

    }

    public function showAll()
    {
        $lokasis = Lokasi::where('id_lokasi','!','=','
1')->get();
        return view('monitoring.showAll', ['lokasis'
=> $lokasis]);
    }

    public function detail($id)
    {
        $time = \Carbon\Carbon::now();
        $lokasis = Lokasi::find($id);
        $ketinggian = Ketinggian::where('id_lokasi'
,$id)->where('waktu','<=',$time-
>toDateTimeString())->where('waktu','>=',$time-
>subHour()->toDateTimeString())-
>orderBy('waktu','DESC')->get();
        return view('monitoring.detail',
[
            'lokasi' => $lokasis,
            'ketinggian' => $ketinggian,
            'mulai' => $time->toDateTimeString(),
            'hingga' => $time->toDateTimeString()
]);
    }
    public function xml_histori(Request $r)
    {
        $mulai = $r->mulai;
        $hingga = $r->hingga;
    }

```

```

        $lokasi = Lokasi::find($r->id);
        $ketinggian = Ketinggian::where('id_lokasi'
, $r->id)->where('waktu', '<=', $hingga)-
>where('waktu', '>=', $mulai)-
>orderBy('waktu', 'DESC')->get();
        // $ketinggian = $ketinggian->reverse();
        return response()->json(
            [
                'ketinggians' => $ketinggian
            ]
        );
    }
}

```

5.1.9 NotifikasiController

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

class NotifikasiController extends Controller
{
    public function kirim()
    {
        return view('notifikasi.notifikasi');
    }
}

```

5.1.10 PompaController

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Lokasi;
use App\Pompa;

class PompaController extends Controller
{
    //
    public function show($id)
    {
        $pompas = Pompa::where('id_lokasi',$id)-
>get();
        $lokasi = Lokasi::find($id);
        $lokasis = Lokasi::where('id_lokasi','!=','
1')->get();
        return view('monitoring.pompa',['pompas' =>
        $pompas, 'lokasi' => $lokasi,'lokasis' => $lokasis
]);
    }

    public function add(Request $r)
    {
        Pompa::create([
            'nama' => $r->nama,
            'keterangan' => $r->keterangan,
            'kecepatan' => $r->kecepatan,
            'id_lokasi' => $r->id_lokasi,
```

```

        'status_nyala' => 0,
        'last_update' => \Carbon\Carbon::now()
    ]);

    return redirect('pompa/' . $r->id_lokasi);
}

public function xml_pompa($id)
{
    $xml = Pompa::find($id);
    $lokasi = Lokasi::find($xml->id_lokasi);
    return response()->json([
        'data' => $xml,
        'lokasi' => $lokasi
    ]);
}
}

```

5.1.11 SendWarning Command

```

<?php

namespace App\Console\Commands;
use Illuminate\Console\Command;
use Telegram\Bot\Laravel\Facades\Telegram;
use App\User;
use App\Lokasi;
use App\Pompa;
use App\Ketinggian;

class SendWarning extends Command
{

```

```

protected $signature = 'send:warning';

protected $description = 'Mengirim peringatan
tinggi air';

public function __construct()
{
    parent::__construct();
}

public function handle()
{
    $x = 0;
    $last_notif = null;
    while(TRUE){
        $lokasis = Lokasi::where('id_lokasi', '
>', '1')->get();
        foreach($lokasis as $lokasi){
            $id_check = $lokasi->id_lokasi;
            $tinggi = Ketinggian::where('id_lo
kasi', $id_check)->orderBy('waktu', 'DESC')-
>first();

            $date = \Carbon\Carbon::now();
            if($tinggi == null) continue;
            if(\Carbon\Carbon::createFromForma
t('Y-m-d H:i:s', $tinggi-
>waktu) >= \Carbon\Carbon::now()->subMinute()){
                if($tinggi-
>ketinggian > $lokasi->batas_normal){
                    if($tinggi-
>ketinggian <= $lokasi->batas_waspada){

```


5.1.12 UserController

```
<?php

namespace App\Http\Controllers;
use Illuminate\Support\Facades\Hash;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;
use App\User;
use App\Lokasi;

class UserController extends Controller
{
    public function show()
    {
        $lokasi = Lokasi::all();
        $users = User::all();
        return view('user.user', ['users' => $users,
'lok' => $lokasi]);
    }

    public function delete(Request $r)
    {
        DB::table('users')->where('id', $r->hapus)-
>delete();
        return redirect('daftar_user');
    }

    public function edit($id)
    {
        $lokasi = Lokasi::all();
        $user = User::find($id);
    }
}
```

```

        return view('user.edit', ['user' => $user,
'lok' => $lokasi]);
    }

    public function simpanEdit($id, Request $r)
    {
        $pengguna = User::find($id);
        if ($r->input('nama') != NULL) $pengguna-
>nama = $r->nama;
        if ($r->input('email') != NULL) $pengguna-
>email = $r->email;
        if ($r-
>input('no_telp') != NULL) $pengguna->no_telp = $r-
>no_telp;
        if ($r-
>input('password') != NULL) $pengguna-
>password = Hash::make($r->password);
        if ($r-
>input('hak_akses') != NULL) $pengguna-
>hak_akses = $r->hak_akses;
        if ($r-
>input('id_lokasi') != NULL) $pengguna-
>id_lokasi = $r->id_lokasi;
        $pengguna->save();

        return redirect('daftar_user/edit/'.$id);
    }
}

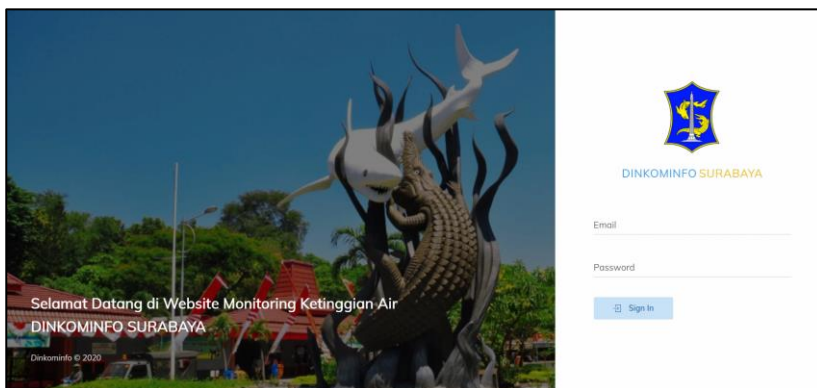
```

5.3 Implementasi Antarmuka Pengguna

Bagian ini merupakan bagian tampilan antar muka yang ada pada *website monitoring* ketinggian air.

5.2.1 Halaman login

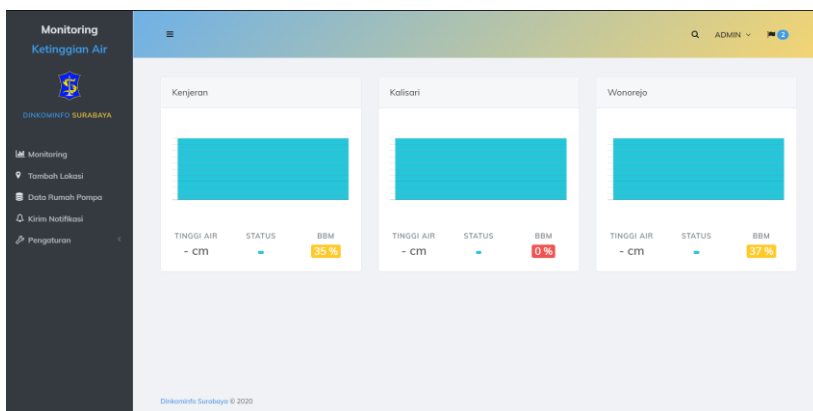
Halaman ini merupakan halaman login sebelum pengguna masuk ke sistem, baik itu administrator maupun petugas. Pengguna dapat *login* ke sistem dengan mengisi email dan password yang telah didaftarkan sebelumnya oleh administrator. Implementasi dari halaman login dapat dilihat pada gambar 5.1.



Gambar 5.1 Tampilan Halaman Login

5.2.2 Halaman *Dashboard Administrator*

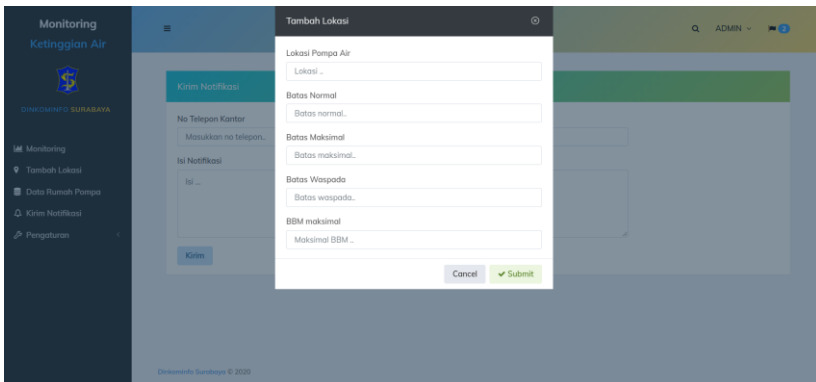
Halaman ini menampilkan semua data rumah pompa yang terdaftar dalam sistem. Pengguna yang mempunyai hak akses sebagai administrator dapat melihat semua data rumah pompa yang ada pada *website monitoring* ketinggian air. Implementasi dari halaman *dashboard* administrator dapat dilihat pada gambar 5.2.



Gambar 5.2 Tampilan Halaman Dashboard Administrator

5.2.3 Halaman Tambah Lokasi Rumah Pompa

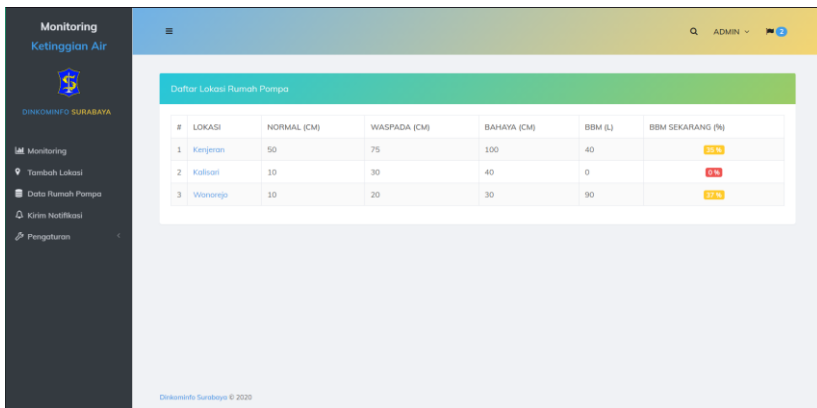
Halaman ini merupakan halaman untuk menambahkan data rumah pompa. Pengguna yang mempunyai hak akses sebagai administrator dapat menambahkan lokasi rumah pompa dengan mengisi data rumah pompa. Implementasi dari halaman tambah lokasi rumah pompa dapat dilihat pada gambar 5.3.



Gambar 5.3 Tampilan Halaman Tambah Lokasi Rumah Pompa

5.2.4 Halaman Data Rumah Pompa

Halman ini merupakan halaman data rumah pompa yang menampilkan informasi mengenai batas ketinggian disuatu rumah pompa meliputi batas normal, waspada, dan bahaya. Serta menampilkan informasi daya tampung BBM dan penggunaan BBM. Implementasi dari halaman data rumah pompa dapat dilihat pada gambar 5.4.

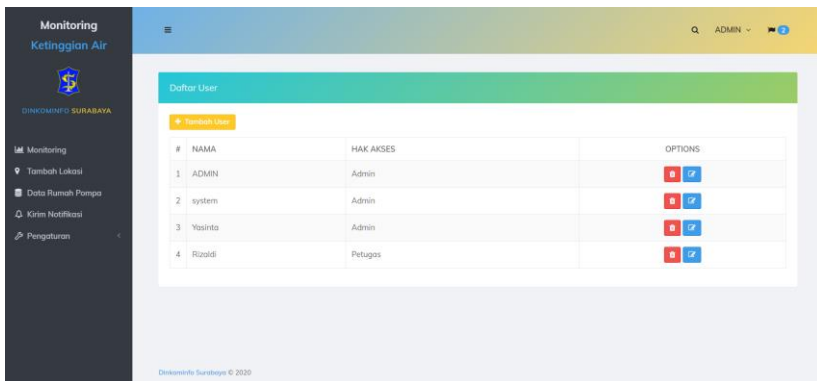


| # | LOKASI | NORMAL (CM) | WASPADA (CM) | BAHAYA (CM) | BBM (L) | BBM SEKARANG (%) |
|---|----------|-------------|--------------|-------------|---------|------------------|
| 1 | Kinjeran | 50 | 75 | 100 | 40 | 15.1% |
| 2 | Kalsari | 10 | 30 | 40 | 0 | 0% |
| 3 | Wanarejo | 10 | 20 | 30 | 90 | 17.1% |

Gambar 5.4 Tampilan Data Rumah Pompa

5.2.5 Halaman Daftar User

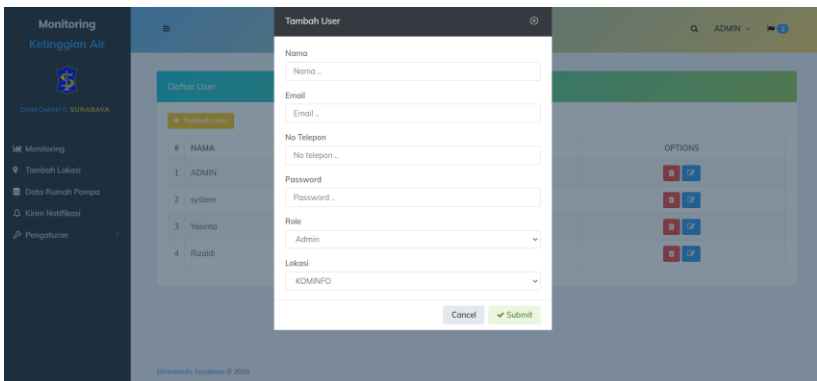
Halaman ini menampilkan semua data pengguna yang terdaftar dalam sistem. Pengguna yang mempunyai hak akses sebagai administrator dapat melihat semua data pengguna yang ada pada *website monitoring* ketinggian air. Implementasi dari halaman daftar *user* dapat dilihat pada gambar 5.6.



Gambar 5.5 Tampilan Halaman Daftar User

5.2.6 Halaman Menambahkan User

Halaman ini merupakan halaman untuk menambahkan *user*. *User* yang mempunyai hak akses sebagai administrator dapat menambahkan data *user* dengan mengisi data diri *user* pada form yang telah disediakan. Implementasi dari halaman tambah lokasi rumah pompa dapat dilihat pada gambar 5.7.



Gambar 5.6 Tampilan Halaman Menambahkan User

5.2.7 Halaman Edit Profil

Halaman ini merupakan halaman untuk mengedit profil *user*. *User* yang mempunyai hak akses sebagai administrator dapat mengedit profil *user* dan melakukan perubahan sesuai dengan data baru yang diinputkan. Implementasi dari halaman edit profil dapat dilihat pada gambar 5.8.

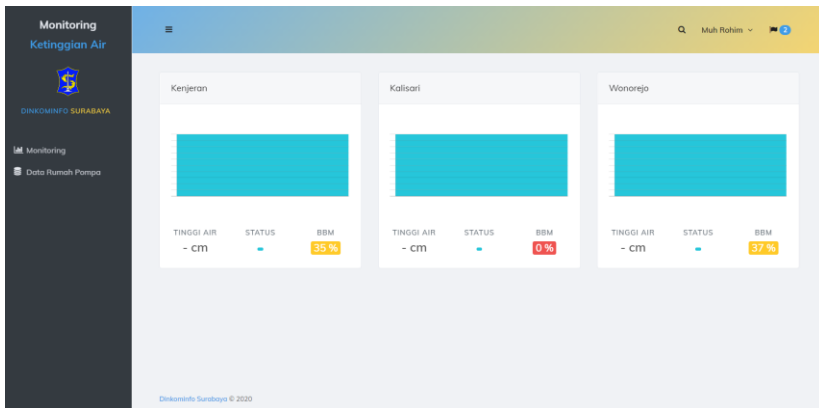


The screenshot displays the 'Edit Profil' interface. On the left is a dark sidebar with the title 'Monitoring Ketinggian Air' and the logo of 'DINCOMINFO SURABAYA'. The sidebar menu includes 'Monitoring', 'Tambah Lokasi', 'Data Rumah Pompa', 'Kirim Notifikasi', and 'Pengaturan'. The main content area has a light blue header with a search icon and 'ADMIN' with a dropdown arrow. Below the header is a green bar with the title 'Edit Profil'. The form contains the following fields: 'Nama' (Yasinta), 'Email' (yasinta@gmail.com), 'No Telepon' (081779313648), 'Password' (Password ...), 'Role' (Admin), and 'Lokasi' (KUMINFO). At the bottom of the form are two buttons: 'Cancel' and 'Simpan'.

Gambar 5.7 Tampilan Halaman Edit Profil

5.2.8 Halaman *Dashboard* Petugas

Halaman ini menampilkan data rumah pompa dilokasi yang dijaga oleh petugas yang bersangkutan. Implementasi dari halaman *dashboard* petugas dapat dilihat pada gambar 5.9.



Gambar 5.8 Tampilan Halaman Dashboard Petugas

BAB VI

PENGUJIAN DAN EVALUASI

Bab ini menjelaskan tahap uji coba terhadap *website monitoring* ketinggian air. Pengujian dilakukan untuk memastikan kualitas perangkat lunak yang dibangun dan kesesuaian hasil eksekusi perangkat lunak dengan analisis dan perancangan perangkat lunak.

6.1. Skenario Pengujian

6.1.1. Melakukan *login*

Skenario pengujian aplikasi adalah sebagai berikut:

1. Membuka monika.surabaya.go.id.
2. Memastikan kebutuhan-kebutuhan berikut sudah berfungsi sesuai prosedur yang disepakati yaitu terdapat form untuk menuliskan *username*, *password*, dan tombol masuk.
3. Memastikan data yang diisikan di form tersimpan pada *database* ketika tombol masuk di klik dan menyimpan session *user* tersebut.

6.1.2. Melakukan *register*

Skenario pengujian aplikasi adalah sebagai berikut:

1. Membuka monika.surabaya.go.id.
2. Masuk ke halaman *user*.
3. Memastikan kebutuhan-kebutuhan berikut sudah berfungsi sesuai prosedur yang disepakati yaitu terdapat form untuk

menuliskan *username*, *password*, dan tombol tambah *user*.

4. Memastikan data yang diisikan di form tersimpan pada *database* ketika tombol daftar di klik.

6.1.3. Melihat daftar *user* yang terdaftar

Skenario pengujian aplikasi adalah sebagai berikut:

1. Membuka monika.surabaya.go.id.
2. Masuk ke halaman Daftar *User*.
3. Memastikan kebutuhan-kebutuhan berikut sudah berfungsi sesuai prosedur yang disepakati yaitu terdapat tabel yang berisi informasi daftar *user*, yakni nama, hak akses, dan opsi edit atau hapus *user*.

6.1.4. Memonitor ketinggian air di semua rumah pompa

Skenario pengujian aplikasi adalah sebagai berikut:

1. Membuka monika.surabaya.go.id.
2. Masuk ke halaman *dashboard*.
3. Memastikan kebutuhan-kebutuhan berikut sudah berfungsi sesuai prosedur yang disepakati yaitu terdapat grafik ketinggian air masing-masing rumah pompa, status ketinggian air masing-masing rumah pompa, data angka ketinggian air rumah pompa, dan persentase BBM masing-masing rumah pompa.
4. Memastikan data masing-masing rumah pompa terupdate setiap saat.

6.1.5. Menambahkan data lokasi rumah pompa

Skenario pengujian aplikasi adalah sebagai berikut:

1. Membuka monika.surabaya.go.id.
2. Memilih menu tambah lokasi.
3. Memastikan kebutuhan-kebutuhan berikut sudah berfungsi sesuai prosedur yang disepakati yaitu terdapat tombol tambah rumah pompa, *form* untuk menuliskan lokasi rumah pompa, batas normal rumah pompa, batas waspada rumah pompa, batas maksimal rumah pompa, dan bbm maksimal rumah pompa.
4. Memastikan data yang diisikan di form tersimpan pada database ketika tombol daftar di klik.

6.1.6. Melihat detail data rumah pompa

Skenario pengujian aplikasi adalah sebagai berikut:

1. Membuka monika.surabaya.go.id.
2. Masuk ke halaman beranda .
3. Memilih data rumah pompa.
4. Memastikan kebutuhan-kebutuhan berikut sudah berfungsi sesuai prosedur yang disepakati yaitu terdapat tabel yang berisi daftar rumah pompa, batas normal, waspada, dan maksimal rumah pompa, bbm maksimal rumah pompa, dan persentase bbm saat ini.
5. Memilih salah satu rumah pompa.
6. Memastikan kebutuhan-kebutuhan berikut sudah berfungsi sesuai prosedur yang disepakati yaitu terdapat tabel yang berisi

daftar pompa, keterangan pompa, kecepatan pompa, dan status nyala pompa.

6.1.7. Mengirimkan notifikasi

Skenario pengujian aplikasi adalah sebagai berikut:

1. Membuka aplikasi telegram
2. Masuk kedalam grup Bot Telegram MONIKA
3. Memastikan kebutuhan-kebutuhan berikut sudah berfungsi sesuai prosedur yang disepakati yaitu terdapat notifikasi mengenai status ketinggian air yang memasuki batas waspada atau lebih yang berisi ketinggian air dan lokasi rumah pompa.

6.1.8. Mengedit profil *user*

Skenario pengujian aplikasi adalah sebagai berikut:

1. Membuka monika.surabaya.go.id
2. Masuk ke halaman pengaturan *user*
3. Memastikan kebutuhan-kebutuhan berikut sudah berfungsi sesuai prosedur yang disepakati yaitu terdapat daftar *user* yang terdaftar dalam system, tombol edit pada masing-masing *user*.
4. Memilih pilihan edit dan memastikan kebutuhan sudah berfungsi dan sesuai prosedur yang disepakati yaitu terdapat *form* edit dari *user* yang bersangkutan untuk menuliskan perubahan nama, email, password, role, lokasi yang ditempatkan.

5. Memastikan perubahan profil *user* tersimpan pada *database* ketika tombol simpan di klik.

6.1.9. Menghapus akun *user*

Skenario pengujian aplikasi adalah sebagai berikut:

1. Membuka monika.surabaya.go.id.
2. Masuk ke halaman pengaturan *user*.
3. Memastikan kebutuhan-kebutuhan berikut sudah berfungsi sesuai prosedur yang disepakati yaitu terdapat daftar *user* dan tombol hapus pada masing-masing *user*.
4. Memastikan penghapusan *user* tersimpan dalam *database* ketika tombol hapus di klik.

6.1.10. Memonitor ketinggian air di lokasi yang dijaga

Skenario pengujian aplikasi adalah sebagai berikut:

1. Membuka monika.surabaya.go.id.
2. Masuk ke halaman detail surat.
3. Memilih rumah pompa yang akan di lihat secara detail dengan menekan judul pada grafik ketinggian air.
4. Memastikan kebutuhan-kebutuhan berikut sudah berfungsi sesuai prosedur yang disepakati yaitu terdapat grafik ketinggian air terbaru, tabel yang berisi informasi riwayat ketinggian air, status pompa pada rumah pompa yang berisi kapasitas BBM, persentase BBM rumah pompa, status nyala masing-masing pompa.

6.1.11. Menambah pompa pada rumah pompa

Skenario pengujian aplikasi adalah sebagai berikut:

1. Membuka monika.surabaya.go.id.
2. Masuk ke halaman daftar rumah pompa.
3. Memilih rumah pompa yang akan ditambahkan pompanya.
4. Memastikan kebutuhan-kebutuhan berikut sudah berfungsi sesuai prosedur yang disepakati yaitu terdapat tombol *download* pada halaman detail surat, terdapat tabel yang berisi daftar pompa yang terdaftar pada rumah pompa yang berisi nama, keterangan, kecepatan, dan status, tombol tambah pompa.
5. Menekan tombol tambah rumah pompa.
6. Memastikan modal tambah rumah pompa tampil dan berisi *form* untuk menuliskan nama pompa, keterangan, dan kecepatan pompa.
7. Memastikan data yang diisikan pada *form* tersimpan pada *database* ketika tombol submit ditekan.

6.2. Evaluasi Pengujian

Pada subbab ini akan diberikan hasil evaluasi dari pengujian-pengujian yang telah dilakukan. Hasil evaluasi pengujian dapat dilihat pada Tabel 6.1.

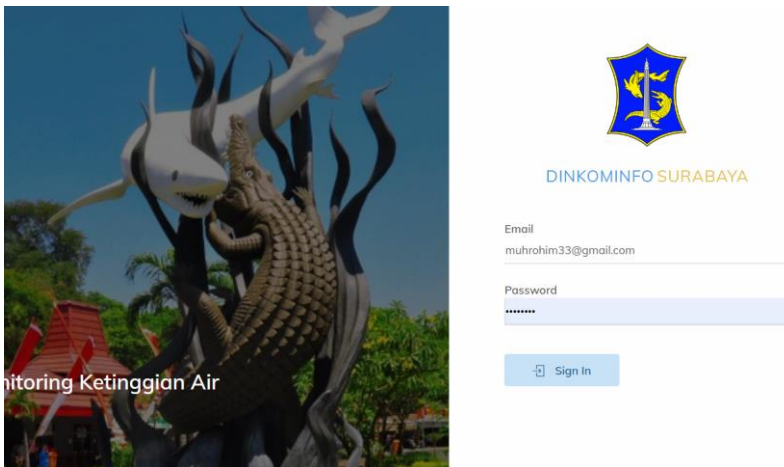
Tabel 6.1 Tabel Evaluasi Pengujian Aplikasi Sesuai Kebutuhan

| No. | Kebutuhan | Uji Coba | Status |
|-------|---|---|----------|
| UC001 | Melakukan <i>login</i> | <i>User</i> membuka halaman login. <i>User</i> mengisikan form <i>username password</i> . Klik tombol 'masuk'. | Berhasil |
| UC002 | Melakukan <i>register</i> | Admin membuka halaman pengaturan <i>user</i> . Admin memilih tombol tambah <i>user</i> pada daftar <i>user</i> . Admin mengisikan detail <i>user</i> . Admin menekan tombol submit. | Berhasil |
| UC003 | Melihat data <i>user</i> | Admin membuka halaman beranda. Admin memilih menu pengaturan, <i>user</i> . | Berhasil |
| UC004 | Memonitor ketinggian air di semua rumah pompa | <i>User</i> membuka homepage sistem informasi. <i>User</i> memilih menu <i>monitoring</i> . Sistem menampilkan grafik ketinggian air seluruh rumah pompa. | Berhasil |
| UC005 | Menambahkan data lokasi rumah pompa | Admin membuka homepage. Admin memilih menu tambah lokasi rumah pompa. Admin mengisikan detail rumah pompa pada form yang disediakan. Admin menekan tombol submit. | Berhasil |
| UC006 | Melihat detail data rumah pompa | <i>User</i> membuka halaman beranda. <i>User</i> menekann menu daftar rumah pompa. <i>User</i> | Berhasil |

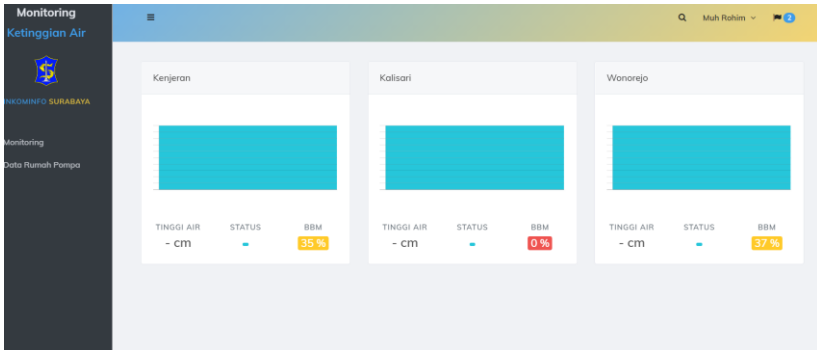
| | | | |
|-------|--|---|----------|
| | | menekan salah satu rumah pompa. | |
| UC007 | Mengirimkan notifikasi | <i>User</i> melihat notifikasi ketinggian air melalui bot telegram MONIKA. | Berhasil |
| UC008 | Mengedit profil <i>user</i> | Admin membuka homepage. Admin memilih menu pengaturan <i>user</i> . Admin memilih <i>user</i> yang akan diedit. Admin menekan tombol edit <i>user</i> . Admin merubah detail <i>user</i> yang tertera pada form. Admin menekan tombol submit. | Berhasil |
| UC009 | Menghapus akun <i>user</i> | Admin membuka homepage. Admin memilih menu pengaturan <i>user</i> . Admin memilih <i>user</i> yang akan dihapus. Admin menekan tombol hapus <i>user</i> . | Berhasil |
| UC010 | Memonitor ketinggian air di lokasi yang dijaga | <i>User</i> membuka homepage. <i>User</i> menekan judul grafik salah satu rumah pompa yang akan dilihat. | Berhasil |
| UC011 | Menambah pompa pada rumah pompa | Admin membuka homepage. Admin memilih menu daftar rumah pompa. Admin memilih rumah pompa yang akan ditambahkan pompa. Admin menekan tombol tambah pompa. Admin memasukkan detail pompa. Admin menekan tombol submit | Berhasil |

6.2.1. Melakukan *login*

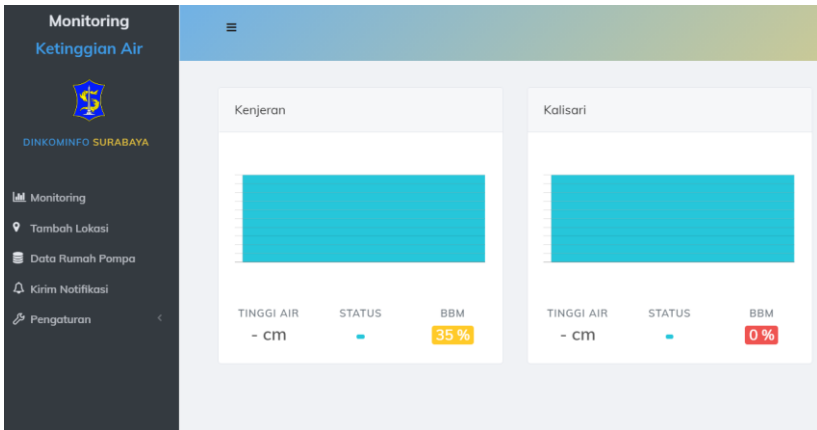
User membuka url *website monitoring* ketinggian air. *User* memasukkan email dan password kedalam *form username* dan *password*. *User* menekan tombol '*sign-in*'. Apabila *login* berhasil, *user* akan diarahkan ke halaman *monitoring* ketinggian air rumah pompa. Tampilan *login* dapat dilihat pada gambar 6.1. Apabila *user log-in* sebagai pengguna biasa, maka akan diarahkan ke tampilan halaman *monitoring* data dilihat pada gambar 6.2, apabila *user log-in* sebagai admin, maka akan ada tambahan menu seperti pada gambar 6.3.



Gambar 6.1 Tampilan Login



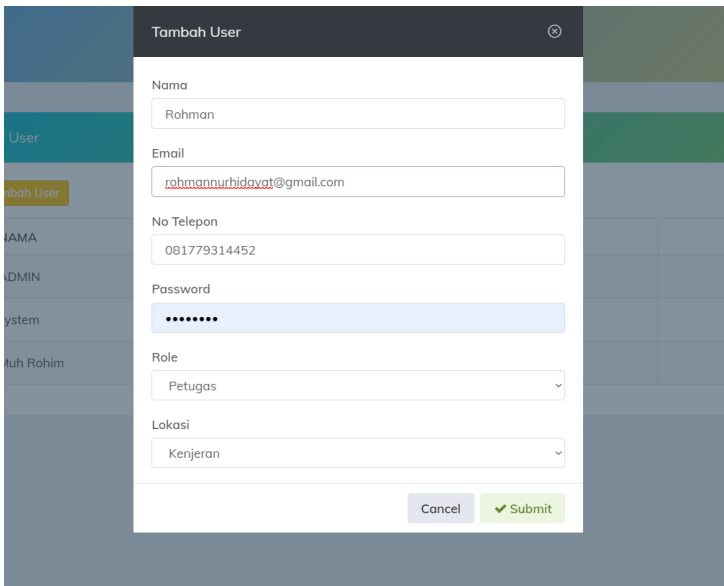
Gambar 6.2 Tampilan Berhasil Login Sebagai User



Gambar 6.3 Tampilan Berhasil Login Sebagai Admin

6.2.2. Melakukan *register*









Admin membuka *homepage*. Admin memilih menu Pegaturan > *User*. Admin menekan tombol ‘Tambah *User*’. Admin mengisi detail *user* yang akan didaftarkan, berupa nama, email, password, nomor telepon, *role*, dan lokasi rumah pompa yang ditempatkan. Apabila tambah *user* sukses maka *user* akan masuk ke daftar *user* yang tertera pada halaman daftar *user*. Tampilan *form register* akan di tampilkan seperti pada gambar 6.4, dan tampilan daftar *user* setelah berhasil *register* akan muncul seperti gambar 6.5.



The image shows a 'Tambah User' (Add User) form. The form fields are as follows:

| Field | Value |
|------------|----------------------------|
| Nama | Rohman |
| Email | rahmannurhidayat@gmail.com |
| No Telepon | 081779314452 |
| Password | |
| Role | Petugas |
| Lokasi | Kenjeran |

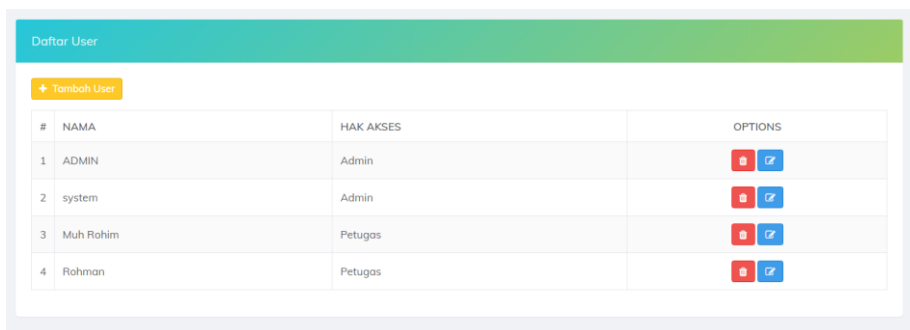
Gambar 6.4 Tampilan Form Register

| # | NAMA | HAK AKSES | OPTIONS |
|---|-----------|-----------|---|
| 1 | ADMIN | Admin |   |
| 2 | system | Admin |   |
| 3 | Muh Rohim | Petugas |   |
| 4 | Rohman | Petugas |   |









Gambar 6.5 Tampilan Daftar User Setelah Register Berhasil

6.2.3. Melihat data *user*

Admin membuka halaman beranda. Admin memilih menu Pengaturan sub menu *User*. Sistem akan menampilkan daftar *user* yang terdaftar pada sistem. Tampilan daftar *user* tersebut dapat dilihat pada gambar 6.6.



The screenshot shows a web interface titled "Daftar User". At the top left, there is a teal header with the text "Daftar User". Below the header is a yellow button with a plus sign and the text "Tambah User". The main content is a table with four columns: "#", "NAMA", "HAK AKSES", and "OPTIONS". The table contains four rows of user data. Each row in the "OPTIONS" column has a red square icon with a white minus sign and a blue square icon with a white plus sign.

| # | NAMA | HAK AKSES | OPTIONS |
|---|-----------|-----------|---|
| 1 | ADMIN | Admin |   |
| 2 | system | Admin |   |
| 3 | Muh Rohim | Petugas |   |
| 4 | Rohman | Petugas |   |

Gambar 6.6 Tampilan Daftar User

6.2.4. Memonitor ketinggian air di semua rumah pompa

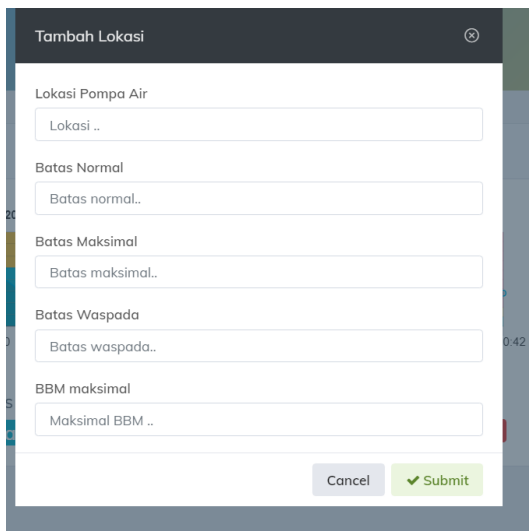
User membuka *homepage* sistem informasi. User memilih menu *monitoring*. Sistem akan menampilkan grafik ketinggian air masing-masing rumah pompa dengan grafik dan data yang di update setiap waktu. Tampilan dari menu *monitoring* tersebut ada pada gambar 6.7.



Gambar 6.7 Tampilan Monitoring Ketinggian Air Semua Rumah Pompa

6.2.5. Menambahkan data lokasi rumah pompa

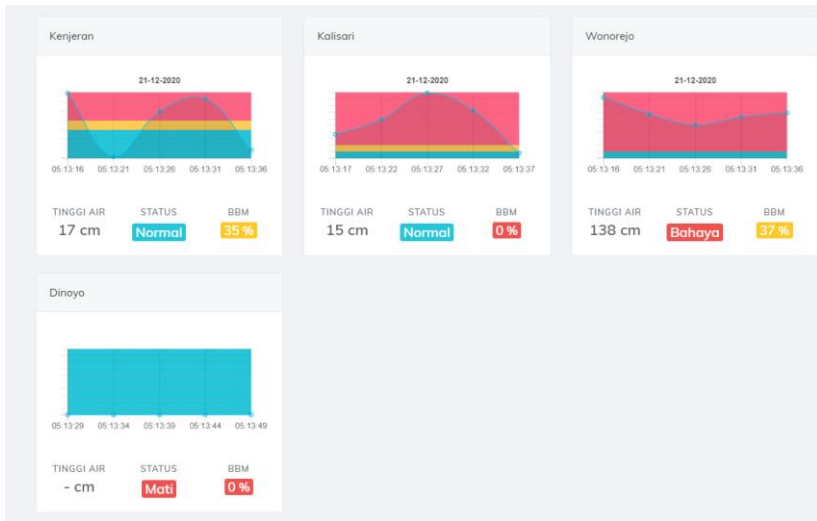
Admin membuka *homepage*. Admin memilih menu tambah lokasi pada *sidebar*. Sistem akan menampilkan modal dan *form* yang berisi detail lokasi rumah pompa. Admin menuliskan detail dari lokasi rumah pompa tersebut. Setelah admin menekan tombol '*submit*', maka lokasi rumah pompa akan terdaftar dalam sistem, dan dapat muncul pada halaman daftar rumah pompa, atau pada halaman *monitoring* rumah pompa. Tampilan *form* lokasi rumah pompa ada pada gambar 6.8, sedangkan hasil dari penambahan lokasi yang berhasil, pada daftar rumah pompa ditampilkan pada gambar 6.9, dan pada halaman *monitoring*, ditampilkan pada gambar 6.10



Gambar 6.8 Tampilan Form Tambah Rumah Pompa

| # | LOKASI | NORMAL (CM) | WASPADA (CM) | BAHAYA (CM) | BBM (L) | BBM SEKARANG (%) |
|---|----------|-------------|--------------|-------------|---------|------------------|
| 1 | Kenjeran | 50 | 75 | 100 | 40 | 35% |
| 2 | Kalisari | 10 | 30 | 40 | 0 | 0% |
| 3 | Wonorejo | 10 | 20 | 30 | 90 | 37% |
| 4 | Dinoyo | 60 | 90 | 120 | 300 | 0% |

Gambar 6.9 Tampilan Daftar Rumah Pompa Baru



Gambar 6.10 Tampilan Menu Monitoring Baru

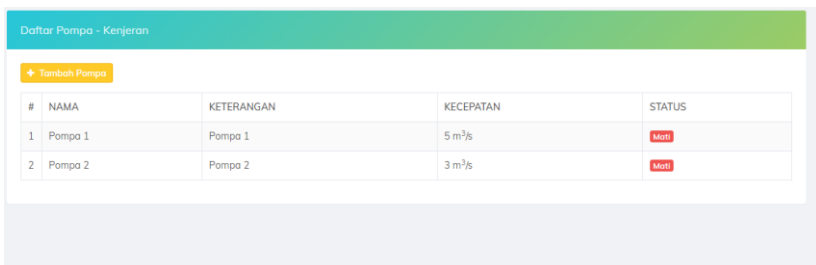
6.2.6. Melihat detail data rumah pompa

User membuka homepage. *User* memilih menu Daftar Rumah Pompa. Sistem akan menampilkan daftar rumah pompa yang terdaftar dalam sistem, ditampilkan pada gambar 6.11. *User* dapat mengakses detail rumah pompa dengan menekan nama rumah pompa yang tertera untuk melihat pompa apa saja yang sudah terdaftar pada rumah pompa. Tampilan detail rumah pompa dapat dilihat pada gambar 6.12.



| # | LOKASI | NORMAL (CM) | WASPADA (CM) | BAHAYA (CM) | BBM (L) | BBM SEKARANG (%) |
|---|----------|-------------|--------------|-------------|---------|------------------|
| 1 | Kerjeran | 50 | 75 | 100 | 40 | 35% |
| 2 | Kalsari | 10 | 30 | 40 | 0 | 0% |
| 3 | Wonorejo | 10 | 20 | 30 | 90 | 37% |
| 4 | Dinoyo | 60 | 90 | 120 | 300 | 65% |

Gambar 6.11 Tampilan Daftar Rumah Pompa

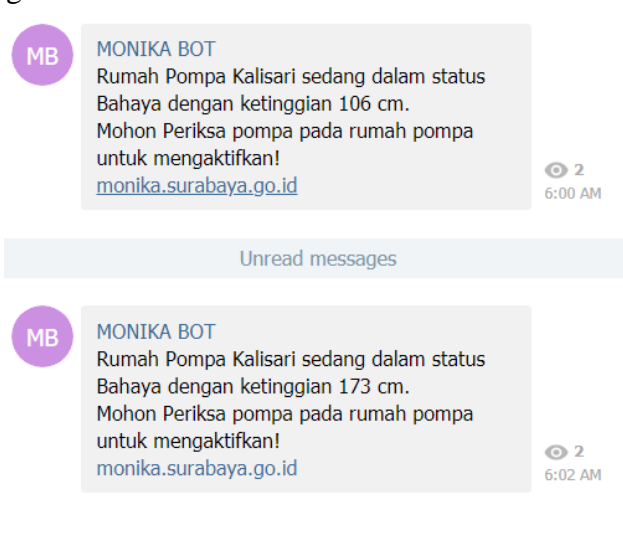


| # | NAMA | KETERANGAN | KECEPATAN | STATUS |
|---|---------|------------|---------------------|--------|
| 1 | Pompa 1 | Pompa 1 | 5 m ³ /s | Mati |
| 2 | Pompa 2 | Pompa 2 | 3 m ³ /s | Mati |

Gambar 6.12 Tampilan Detail Rumah Pompa

6.2.7. Mengirimkan notifikasi

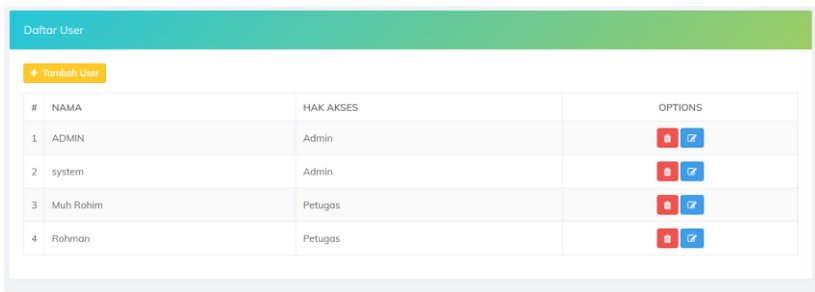
Ketika rumah pompa dalam keadaan waspada atau maksimal, maka Bot Telegram MONIKA akan mengirimkan notifikasi berupa pesan melalui telegram. Notifikasi melalui telegram tersebut dapat dilihat pada gambar 6.13.











Gambar 6.13 Tampilan Notifikasi Bot Telegram MONIKA

6.2.8. Mengedit profil *user*

Admin membuka *homepage*. Admin memilih menu Pengaturan tab menu *User*. Sistem akan menampilkan daftar *user* yang terdaftar pada sistem, seperti pada gambar 6.14. Admin dapat menekan tombol edit pada user yang akan di edit. Sistem akan menampilkan *form* pengeditan *user*. Admin dapat mengisi detail baru dari *user* yang akan diedit. Setelah selesai mengganti perubahan, maka admin dapat menyimpan dengan menekan tombol simpan. Form edit dapat dilihat pada gambar 6.15, dan perubahannya dapat dilihat pada gambar 6.16.



The screenshot shows a web interface titled "Daftar User". At the top left, there is a teal header with the text "Daftar User". Below the header is a yellow button with a plus sign and the text "Tambah User". The main content is a table with four columns: "#", "NAMA", "HAK AKSES", and "OPTIONS". The table contains four rows of user data. Each row in the "OPTIONS" column has a red square icon and a blue pencil icon.

| # | NAMA | HAK AKSES | OPTIONS |
|---|-----------|-----------|---|
| 1 | ADMIN | Admin |   |
| 2 | system | Admin |   |
| 3 | Muh Rohim | Petugas |   |
| 4 | Rohman | Petugas |   |

Gambar 6.14 Tampilan Daftar User

The image shows a web form titled "Edit Profil" with a teal header. The form contains the following fields and values:

- Nama:** Rohman Nurhidayat
- Email:** rohmannurhidayat@gmail.com
- No Telepon:** 081779314452
- Password:** Password ...
- Role:** Petugas
- Lokasi:** Kenjeran

At the bottom of the form are two buttons: "Cancel" (grey) and "Simpan" (green).





Gambar 6.15 Tampilan Form Edit User

| No | system | password |
|----|-------------------|----------|
| 3 | Muh Rohim | Petugas |
| 4 | Rohman Nurhidayat | Petugas |





Gambar 6.16 Tampilan Daftar User Setelah Berhasil

6.2.9. Menghapus akun *user*

Admin membuka *homepage*. Admin memilih menu Pengaturan tab menu *User*. Sistem akan menampilkan daftar *user* yang terdaftar pada sistem, seperti pada gambar 6.17. Admin dapat menekan tombol hapus pada *user* yang akan di hapus. Perubahannya dapat dilihat pada gambar 6.18.

| | | | |
|---|-----------|---------|---|
| 3 | Muh Rohim | Petugas |   |
| 4 | Rohman | Petugas |   |

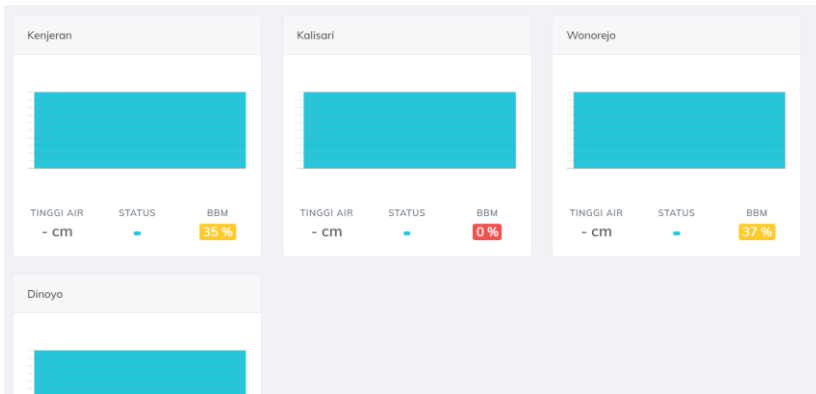
Gambar 6.17 Tampilan Daftar User dan Tombol Hapus

| | | | |
|---|-----------|---------|---|
| 2 | system | Admin |   |
| 3 | Muh Rohim | Petugas |   |

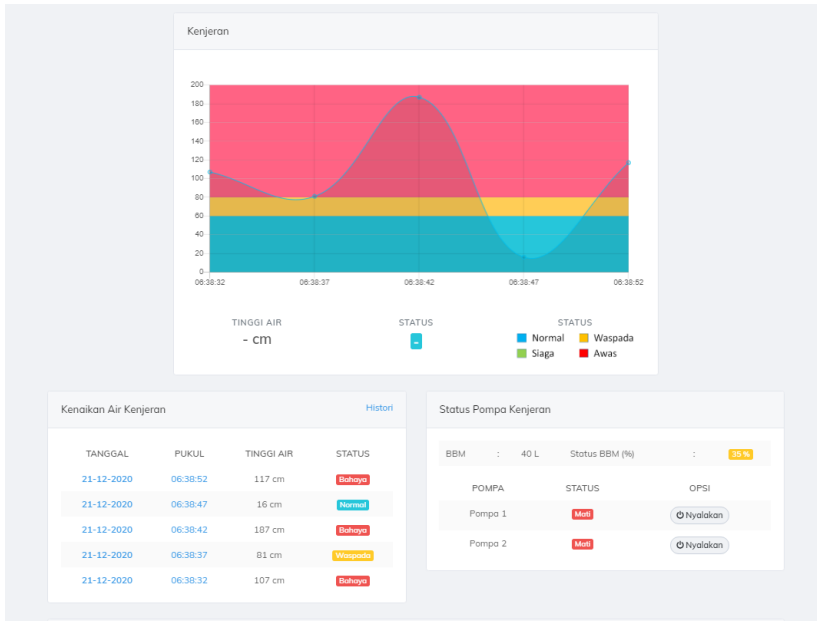
Gambar 6.18 Tampilan Hasil Hapus Akun User

6.2.10. Memonitor ketinggian air di lokasi yang dijaga

User membuka halaman *homepage*. *User* memilih salah satu rumah pompa yang akan di monitor dengan menekan judul grafik ketinggian air rumah pompa tersebut, ditampilkan pada gambar 6.19. Sistem akan menampilkan halaman detail monitor rumah pompa tersebut yang berisi grafik ketinggian air, pompa dan status BBM-nya, histori ketinggian selama 5 detik terakhir. Tampilan tersebut dapat dilihat pada gambar 6.20.



Gambar 6.19 Tampilan Grafik Seluruh Rumah Pompa pada Halaman Monitoring



Gambar 6.20 Tampilan Detail Monitor Rumah Pompa

6.2.11. Menambah pompa pada rumah pompa

Admin membuka *homepage*. Admin memilih menu Daftar rumah pompa. Sistem akan menampilkan daftar rumah pompa yang terdaftar pada sistem, ditampilkan pada gambar 6.21. Admin dapat memilih rumah pompa mana yang akan ditambahkan pompa baru dengan menekan nama lokasi rumah pompa tersebut. Sistem akan menampilkan daftar pompa pada rumah pompa tersebut, ditampilkan pada gambar 6.22. Admin dapat menekan tombol tambah pompa dan mengisi *form* yang ditampilkan oleh sistem tentang detail pompa, ditampilkan pada gambar 6.23. Tampilan pompa terbaru akan tertulis pada daftar pompa pada gambar 6.24.

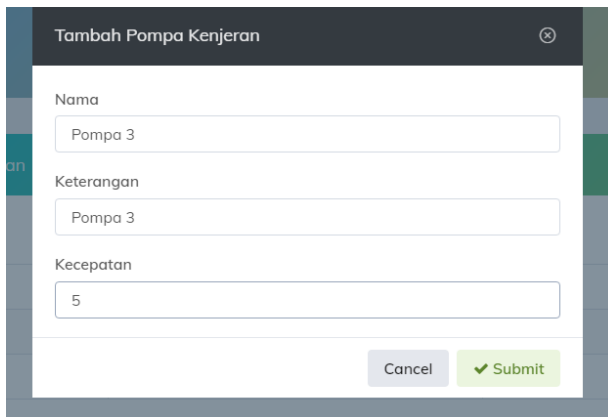


| # | LOKASI | NORMAL (CM) | WASPADA (CM) | BAHAYA (CM) | BBM (L) | BBM SEKARANG (%) |
|---|----------|-------------|--------------|-------------|---------|------------------|
| 1 | Kerjeran | 50 | 75 | 100 | 40 | 35% |
| 2 | Kalisari | 10 | 30 | 40 | 0 | 0% |
| 3 | Wonorejo | 10 | 20 | 30 | 90 | 37% |
| 4 | Dinoyo | 60 | 90 | 120 | 300 | 0% |

Gambar 6.21 Tampilan Daftar Rumah Pompa



Gambar 6.22 Tampilan Daftar Pompa Pada Rumah Pompa



Gambar 6.23 Tampilan Form Tambah Pompa

| # | NAMA | KETERANGAN | KECEPATAN |
|---|---------|------------|---------------------|
| 1 | Pompa 1 | Pompa 1 | 5 m ³ /s |
| 2 | Pompa 2 | Pompa 2 | 3 m ³ /s |
| 3 | Pompa 3 | Pompa 3 | 5 m ³ /s |

Gambar 6.24 Tampilan Daftar Pompa Pada Rumah Pompa

[Halaman ini sengaja dikosongkan]

BAB VII

KESIMPULAN DAN SARAN

7.1 Kesimpulan

Kesimpulan yang didapat setelah melakukan pembuatan *website monitoring* ketinggian air adalah sebagai berikut:

- a. Aplikasi yang dibangun telah sesuai dengan permintaan dan dapat dengan mudah dioperasikan oleh *user*.
- b. Dengan adanya *website monitoring* ketinggian air dapat memudahkan pihak Dinas Kominfo Surabaya dalam melakukan *monitoring* ketinggian air pada rumah pompa dengan adanya data yang *update* dan notifikasi telegram terhadap ketinggian air.

7.2 Saran

Saran yang penulis berikan untuk pembuatan *website monitoring* ketinggian air adalah dibutuhkannya *refactor* terhadap penulisan kode agar lebih bersih dan singkat. Yang kedua adalah diharapkan *monitoring* ini dapat digunakan dan dilihat oleh khalayak umum agar bisa diakses oleh siapa saja.

DAFTAR PUSTAKA

- [1] Dinkominfo Surabaya, 2017. Sejarah Dinkominfo. [Online] Available at: <https://dinkominfo.surabaya.go.id/index.php?pages=satis&id=6> [Diakses 12 September 2020].
- [2] Kementerian Komunikasi dan Informatika, 2020. Visi dan Misi. [Online] Available at: <https://www.kominfo.go.id/visi-dan-misi> [Diakses 11 November 2020].
- [3] Technophoria, 2014. Pengertian Tentang Aplikasi Berbasis Web.[online] Available at: <http://technophoriajogja.com/2014/01/28/pengertian-tentang-aplikasi-berbasis-web/> [Diakses 12 November 2020].
- [4] Mozilla Developer Network, 2019. HTML Dasar. [online] Available at: https://developer.mozilla.org/id/docs/Learn/Getting_started_with_the_web/HTML_basics [Diakses 12 November 2020].
- [5] Qwords, 2020. Apa itu CSS, inilah pengertian, Cara Kerja dan Kegunaannya. [Online] Available at: <https://qwords.com/blog/apa-itu-css/> [Diakses 12 November 2020].
- [6] Qwords, 2019. Apa itu Bootstrap? Inilah Pengertian dan Cara Menggunakannya. [Online] Available at: <https://qwords.com/blog/apa-itu-bootstrap/> [Diakses 12 November 2020].
- [7] Mozilla Developer Network, 2019. Dasar Javascript. [online] Available at: https://developer.mozilla.org/id/docs/Learn/Getting_started_with_the_web/JavaScript_basics [Diakses 12 November 2020].

- [8] Ekajogja, 2019. JQuery: Definisi, Fitur, Penggunaan JQuery Dalam Pengembangan Website. [online] Available at: <https://ekajogja.com/arsip/jquery-definisi-fitur-penggunaan-jquery-dalam-pengembangan-website/> [Diakses 12 November 2020].
- [9] Nesabamedia, 2019. Pengertian PHP Beserta Fungsi dan Sejarah PHP yang Wajib Anda Ketahui. [online] Available at: <https://www.nesabamedia.com/pengertian-php-dan-fungsinya/> [Diakses 12 November 2020].
- [10] IdCloudHost, 2016. Pengertian dan Keunggulan Framework Laravel. [Online] Available at: <https://idcloudhost.com/pengertian-dan-keunggulan-framework-laravel/> [Diakses 12 November 2020].
- [11] Qwords, 2019. Mengetahui Cara Kerja dan Jenis-jenis Web Server dalam Website. [online] Available at: <https://qwords.com/blog/pengertian-web-server/> [Diakses 12 November 2020].
- [12] Niagahoster, 2019. Pengertian MySQL, Fungsi, dan Cara Kerjanya. [Online] Available at: <https://www.niagahoster.co.id/blog/mysql-adalah/> [Diakses 12 November 2020].
- [13] Inixindo Jogja, 2020. 5 Ekstensi Visual Studio Code yang Membuat Kamu Lebih Produktif. [Online] Available at: <https://inixindojogja.co.id/5-ekstensi-visual-studio-code-yang-membuat-kamu-lebih-produktif/> [Diakses 12 November 2020].
- [14] Petani Code, 2020. Apa itu Git dan Kenapa Penting bagi Programmer?. [Online] Available at: <https://www.petanikode.com/git-untuk-pemula/> [Diakses 12 November 2020].

BIODATA PENULIS



Mohammad Rizaldi Huzein Prastomo, lahir pada tanggal 18 Agustus 1999 di Lumajang. Penulis merupakan mahasiswa yang sedang menempuh studi di Departemen Teknik Informatika Institut Teknologi Sepuluh Nopember (ITS). Penulis aktif dalam beberapa organisasi yaitu pada Himpunan Mahasiswa Teknik Computer-Informatika tahun 2019/2020 dalam departemen sosial masyarakat dan kegiatan *Dua Care Goes To School* yang merupakan kegiatan organisasi DuaCare sebagai organisasi alumni SMAN 2 Lumajang yang aktif dibidang peningkatan mutu murid SMA di Lumajang.



Yasinta Yusniawati, lahir pada tanggal 1 April 1999 di Purworejo. Penulis merupakan mahasiswa yang sedang menempuh studi di Departemen Teknik Informatika Institut Teknologi Sepuluh Nopember (ITS). Penulis aktif dalam beberapa organisasi dan kepanitiaan yaitu Himpunan Mahasiswa Teknik Computer-Informatika tahun 2019/2020 dalam departemen sosial masyarakat, Keluarga Mahasiswa Purworejo ITS (KAPORITS) 2020, Schematics 2018 dalam divisi *National Seminar of Technologi* (NST), dan Schematics 2019 divisi *National Seminar of Technologi* (NST).